



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Diseño e implementación de perfiles académicos online (E-profiles) de forma descentralizada implementados mediante el uso de elementos del proyecto Solid

Paz Naula, Diego Ignacio

Departamento de Ciencias de la Computación

Carrera de Ingeniería de Sistemas e Informática

Trabajo de titulación, previo a la obtención del título de Ingeniero en Sistemas e
Informática

Phd. Gualotuña Álvarez, Tatiana Marisol

28 de julio del 2022

REPORTE DE PORCENTAJE DE PLAGIO URKUND



Tesis-Paz-Diego.docx

Scanned on: 23:13 July 29, 2022 UTC



Overall Similarity Score



Results Found



Total Words in Text

Identical Words	536
Words with Minor Changes	71
Paraphrased Words	0
Omitted Words	1481

TATIANA MARISOL
GUALOTUNA
ALVAREZ

CERTIFICADO DE APROBACIÓN DE TUTOR



Departamento de Ciencias de la Computación

Carrera de Ingeniería de Sistemas e Informática

Certificación

Certifico que el trabajo de titulación: **"Diseño e implementación de perfiles académicos online (E-profiles) de forma descentralizada implementados mediante el uso de elementos del proyecto Solid"** fue realizado por el señor **Paz Naula, Diego Ignacio** el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizado en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Sangolquí, 03 de agosto de 2022

Firma:



.....
Ing. Gualotuña Álvarez, Tatiana Marisol

C. C. 1711498418

CERTIFICADO DE RESPONSABILIDAD DE AUTORÍA



Departamento de Ciencias de la Computación
Carrera de Ingeniería de Sistemas e Informática

Responsabilidad de Autoría

Yo, Paz Naula, Diego Ignacio, con cédula de ciudadanía n° 1723469712, declaro que el contenido, ideas y criterios del trabajo de titulación: **Diseño e implementación de perfiles académicos online (E-profiles) de forma descentralizada implementados mediante el uso de elementos del proyecto Solid** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Sangolquí, 03 de agosto de 2022

Firma

Paz Naula, Diego Ignacio

C.C.: 1723469712

CERTIFICADO DE AUTORIZACIÓN DE PUBLICACIÓN



Departamento de Ciencias de la Computación

Carrera de Ingeniería de Sistemas e Informática

Autorización de Publicación

Yo, **Paz Naula, Diego Ignacio**, con cédula de ciudadanía n° 1723469712, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **Diseño e implementación de perfiles académicos online (E-profiles) de forma descentralizada implementados mediante el uso de elementos del proyecto Solid** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi/nuestra responsabilidad.

Sangolquí, 03 de agosto de 2022

Firma

Paz Naula, Diego Ignacio

C.C.: 1723469712

Dedicatoria

A mi madre Ana Lucia,

Mi padre Richard,

Mi hermano Ismael,

Mis abuelos Enrique, Susana y Mercedes

que velaron por mi bienestar.

A Dios y

todos aquellos que aportaron

en mi crecimiento profesional y humano

“No puedes vivir de buenas intenciones solamente, para vivir en este mundo se necesitan acciones. Solo las acciones tienen el potencial de darle un sentido a la vida”

Dan Millman

Agradecimiento

El viaje para el desarrollo personal debe estar basado en dar lo mejor de uno mismo sin vivir del ayer que es historia o del mañana al ser un misterio, pero sí del ahora siendo uno de los más grandes regalos de Dios y es por lo que se llama presente. Al aplicar esto en la vida permite presentar la mejor versión de ti mismo, las personas que se dan cuenta te apoyan, acompañan y trabajan en equipo en las buenas y las malas marcan la diferencia entre los que somos y lo que podemos llegar a ser.

Por esta razón estoy agradecido con Dios el cual logró poner me a las personas adecuadas en los momentos indicados, las oportunidades que me brindo y que me ofrecerá. Confiando que contaré con lo necesario para lograr desempeñar de la mejor manera lo que tiene preparado para mí.

Quiero agradecer a mi familia por brindar me desde el primer día su apoyo, cuidado, enseñanzas éticas y morales. A mi padre que sin importar nada me ayuda con mis ocurrencias, mi madre que resguarda mi salud, mi hermano motivándome a mejorar como persona en el cual confié ciegamente valoro principalmente su opinión ante la toma de decisiones y con especial gratitud a mis abuelos que les debo todo lo que soy.

Doy las gracias a las personas que confiaron en mis habilidades, amigos de la carrera, del club de software ESPE, de la rama estudiantil IEEE, al compartir el tiempo con cada uno inspiró el presente trabajo al conocer a gente tan talentosa, dedicada e interesada en promover el conocimiento de forma libre me ha ayudado a ser una mejor persona. Sepan que tengo fe en cada uno de ustedes y si se proponen algo lo van a cumplir de la mejor manera.

Gratitud a los docentes, de los cuales aprendí que, sin esfuerzo, trabajo en equipo, dedicación, liderazgo y honestidad, la recompensa obtenida no significa nada sin las personas adecuadas para compartirlo generando una satisfacción incalculable tras haberlo logrado.

Para finalizar agradecer al PhD. Álvaro Suarez, catedrático de la Universidad de las Palmas de Gran Canaria quien guio con esmero y dedicación el trabajo de titulación de la mano con la PhD. Tatiana Gualotuña, quien supo ver algo especial en mí que el resto de los docentes no y al máster Jean Karlo Obando el cual confió en mí dentro y fuera del Club de Software, estoy muy seguro de que muchas más personas me ayudaron en el camino a los cuales les agradezco de todo corazón por su tiempo y apoyo depositados en mi persona. Gracias a ti y a todos los lectores del presente trabajo.

Índice de Contenidos

Índice de tablas	12
Índice de figuras	13
Resumen	17
Abstract	18
Capítulo I:	19
Introducción	19
Antecedentes	19
Problemática	21
Justificación	25
Objetivos	27
Objetivo general	27
Objetivos específicos	27
Alcance	28
Hipótesis	30
Capítulo II	31
Marco teórico y marco metodológico	31
Estado del Arte	31
Planteamiento de la revisión de literatura	31
Criterios de inclusión y exclusión	32
Conformación del grupo de control (GC) y extracción de palabras relevantes para la investigación	33
Construcción y afinación de la cadena de búsqueda	34
Selección de estudios	35
Resumen de los Estudios Primarios	37
Respuesta a las preguntas de investigación.	47
Resumen general y conclusión del estado del arte.	51

Metodología	51
Identificación del problema y motivación:.....	52
Definición de los objetivos de la solución	53
Diseño y desarrollo	53
Demostración	53
Evaluación.....	53
Comunicación.....	54
Marco Teórico	54
Categorización de variables.....	54
Fundación científica de la variable independiente.....	55
Diseño de sistemas descentralizados	55
Arquitectura de software	59
Aplicaciones Web Progresivas.....	61
Perfiles académicos en línea	63
Fundación científica de la variable independiente.....	65
Privacidad de los datos	65
Almacenamiento de la información	67
Actividades extra-curriculares	68
Reclutamiento de estudiantes	69
Capítulo III.....	71
Análisis y diseño de la arquitectura descentralizada para el desarrollo de e-profile	71
Introducción	71
Análisis	71
Actores del sistema para construcción de e-profile.....	71
Como se clasifican las habilidades socioemocionales (habilidades blandas).....	74
Planteamiento de historia de usuario	76
Diseño.....	79

Razón para usar un sistema descentralizado	80
SOLID server proyecto de almacenamiento descentralizado	82
Diseño de la arquitectura descentralizada para el registro de e-profile	84
Diseño del proceso para el registro de e-profile.....	96
Requisitos del sistema	97
Especificación de requisitos	97
Diseño del prototipo	97
Implementación y prueba del prototipo.....	98
Evaluación y comunicación para refinamiento.....	98
Capítulo IV	103
Desarrollo e implementación del prototipo de software	103
Introducción	103
Desarrollo	103
Desarrollo del sistema para estudiantes donde se registra los e-profile.....	103
Desarrollo necesario para la configuración de Firebase	104
Conexión con servidores Solid para consultar perfiles	105
Manejo de la información de estudiantes con e-profile.....	107
Despliegue del servidor Solid community	110
Puesta en marcha del cliente Pancka Ruray en host firebase	112
Capítulo V	114
Evaluación de prototipo de software	114
Proceso de evaluación del prototipo	114
Pasos para desarrollo del taller con estudiantes.....	115
Plan de pruebas utilizado para evaluar Pancka Ruray	127
Entorno y configuración de pruebas.....	128
Criterios de aprobación/rechazo.....	128
Resultados obtenidos en la encuesta de diseño y satisfacción de Pancka Ruray	129

Análisis de tráfico del cliente Pancka Ruray	131
Registro del resultado de componentes y análisis de los tiempos dentro del proceso del taller del uso de Pancka Ruray	133
Registro de usuarios en servidor Solid.....	133
Iniciar sesión en servidor Solid	135
Agregar información de presentación VCARD	137
Manejar permisos de acceso, público a VCARD.....	139
Agregar dos grupos de control para edición, agregar datos en POD	141
Crear, actualizar, modificar archivos dentro de un POD	143
Manejar permisos de acceso para edición, agregar datos en POD.....	145
Inicio de sesión en cliente Pancka Ruray.....	147
Agregar habilidad de estudiante	149
Editar información de e-profile	151
Certificación de habilidades	154
Resumen de los resultados obtenidos	156
Síntesis de las preguntas de investigación	157
Capítulo VI	160
Conclusiones, Recomendaciones y Trabajos Futuros	160
Conclusiones	160
Recomendaciones	161
Trabajos Futuros.....	162
Bibliografía	163
Anexos.....	171
Anexo 1: Manual de instalación para servidor Solid dentro de DigitalOcean.....	171
Anexo 2: Compra de dominio y certificado ssl	185
Anexo 3: Instalación de servidor solid.....	192

Índice de tablas

Tabla 1. Matriz de congruencia Metodológica	28
Tabla 2. Grupo de Control	33
Tabla 3. Pasos para elaborar la cadena de búsqueda	35
Tabla 4. Estudio Seleccionados.....	36
Tabla 5. Etapas del protocolo de un sistema descentralizado.....	56
Tabla 6. Docker vs Kubernetes.....	60
Tabla 7. Características React y Ionic.....	62
Tabla 8. Actores para la construcción de e-profile.	73
Tabla 9. Clasificación de métodos para clasificar habilidades socioemocionales.....	74
Tabla 10. Configuración establecida dentro del servidor Solid de Pancka Ruray	87
Tabla 11. Etiquetas RDFS para manejo de currículos vitae	100
Tabla 12. Etiquetas de VCARD usados en Pancka Ruray	108
Tabla 13. Características y métricas de calidad sistémica del producto Pancka Ruray	114
Tabla 14. Cuadro de resumen de pruebas.....	127
Tabla 15. Servidor Solid Pancka Ruray. Registro de un usuario	133
Tabla 16. Servidor Solid Pancka Ruray. Inicio de sesión de un usuario	135
Tabla 17. Servidor Solid Pancka Ruray. Ingreso de información de un usuario	137
Tabla 18. Servidor Solid Pancka Ruray. Manejo de permisos de acceso.....	140
Tabla 19. Servidor Solid Pancka Ruray. Agregar grupos de control dentro de los accesos.....	141
Tabla 20. Servidor Solid Pancka Ruray. Gestión de archivos en POD.....	143
Tabla 21. Servidor Solid Pancka Ruray. Manejar permisos de acceso en un POD	145
Tabla 22. Cliente Pancka Ruray. Inicio de sesión de un estudiante	147
Tabla 23. Cliente Pancka Ruray. Agregar habilidad de un estudiante	149
Tabla 24. Cliente Pancka Ruray. Editar información de un estudiante	151

Tabla 25. Servidor Solid Pancka Ruray. Certificar habilidad de estudiante	154
--	-----

Índice de figuras

Figura 1 Árbol de problemas	21
Figura 2. Tráfico de la red computrabajo y multitrabajo en Google	25
Figura 3. Proceso de construcción del Estado del arte	31
Figura 4. Fases de la metodología Design Science Research	52
Figura 5. Categorías de la variable independiente.....	54
Figura 6. Categorías de la variable dependiente	55
Figura 7. Patrones de tendencia para marcos de sistemas	56
Figura 8. Diseño y protocolos para el intercambio de información	60
Figura 9. Formato CV por columnas.....	64
Figura 10. Arquitectura descentralizada con características de software.....	80
Figura 11. Pancka Ruray primera arquitectura propuesta.....	84
Figura 12. Pancka Ruray segunda arquitectura propuesta	85
Figura 13. Configuración para tener HTTPS en servidor Solid	91
Figura 14. Configuración para utilizar el entorno gráfico mashlib.....	91
Figura 15. Dockerfile de nuestro servidor Solid Pancka ruray.....	92
Figura 16. Dockercompose configuración del servidor web nginx proxy	93
Figura 17. Dockercompose letsencrypt certificado https.....	94
Figura 18. Dockercompose configuración para imagen del servidor Solid Pancka Ruray	95
Figura 19. Pancka ruray dentro de Docker HUB de forma pública.....	96
Figura 20. Ciclo de vida del prototipado evolutivo.....	97
Figura 21. Diagrama de flujo para el registro de actividades extracurriculares Pancka Ruray ..	99
Figura 22. Microservicios de Pancka Ruray.....	100
Figura 23. Diagrama secuencial para el manejo de información en Solid	101

Figura 24. Diagrama secuencial para el registro de habilidades	101
Figura 25. Configuración para usar Firebase en React.....	104
Figura 26. Funciones guardar, consultar información de Firestore	104
Figura 27. Manejo de sesión por proveedores Solid.....	105
Figura 28. Navegación del prototipo Pancka Ruray	106
Figura 29. Inicio de sesión en Pancka Ruray.....	106
Figura 30. Atributos de tarjetas de presentación de estudiantes, docentes y supervisores dentro de Pancka Ruray	107
Figura 31. Petición fetch para agregar habilidades	109
Figura 32. Validación de habilidad del estudiante.....	109
Figura 33. Nombre del archivo donde se almacena la información	110
Figura 34. Servidor Solid con mashlib	111
Figura 35. Inicio de sesión servidor Solid mediante LDP	111
Figura 36. Registro de usuario mediante LDP	112
Figura 37. Comenzar host en firebase.....	112
Figura 38. Pancka Ruray dentro de los dominios de firebase	113
Figura 39. Perfil estándar de e-profile de los estudiantes	113
Figura 40. Registro de estudiante dentro del servidor.....	115
Figura 41. Formulario de registro con los campos del Pancka Ruray	115
Figura 42. Pasos para iniciar sesión dentro del servidor Pancka Ruray.....	116
Figura 43. Ventana de inicio de sesión basado en credenciales por email y contraseña	116
Figura 44. Ventana de token mediante OAuth2.....	117
Figura 45. Mostrar datos del perfil	117
Figura 46. Acceder a los datos de perfiles mashlib.....	118
Figura 47. Añadir información personal en VCARD con mashlib	119

Figura 48. Volvemos a la ventana inicial del servidor Solid Pancka Ruray	119
Figura 49. Abrimos los archivos raíz con todos los usuarios registrados	120
Figura 50. Seleccionamos la carpeta con nombre del POD	120
Figura 51. Pasos para añadir permisos a todo el mundo para ver la foto de perfil de usuario	121
Figura 52. Añadir permisos de un grupo para editar la información del perfil de usuario	121
Figura 53. Añadir el permiso específico para editar la información del perfil de usuario	122
Figura 54. Agregar un archivo nuevo dentro del POD de un estudiante	122
Figura 55. Agregar nombre y extensión del archivo nuevo	123
Figura 56. Añadir información dentro de un archivo JSON	123
Figura 57. Agregar todos los roles dentro de un archivo.....	123
Figura 58. Seleccionar un servidor Solid para inicio de sesión	124
Figura 59. Página con autorización de POD para Pancka Ruray.....	124
Figura 60. Perfil de estudiante en Pancka Ruray.....	125
Figura 61. Añadir habilidades para certificación de docentes y supervisores	125
Figura 62. Editar información del perfil en Pancka Ruray	126
Figura 63. Pregunta 1 de la encuesta sobre la estructura de la interfaz de Pancka Ruray	129
Figura 64. Pregunta 2 de la encuesta sobre diseño, accesibilidad y navegación en Pancka Ruray.....	129
Figura 65. Pregunta 3 de la encuesta sobre la estructura de Pancka Ruray.....	130
Figura 66. Pregunta 4 de la encuesta sobre la sobrecarga de datos en Pancka Ruray	130
Figura 67. Pregunta 5 de la encuesta sobre los elementos multimedia utilizados en Pancka Ruray.....	131
Figura 68. Bytes de descarga de componentes multimedia del cliente Pancka Ruray.....	131
Figura 69. Total, de actividades por días de usuarios en cliente Pancka Ruray.....	132
Figura 70. Tiempo promedio del uso para crear un e-profile por parte de estudiantes.....	132

Figura 71. Total, de eventos dentro del cliente Pancka Ruray	133
Figura 72. Pregunta N°1 análisis tiempo de respuesta	135
Figura 73. Pregunta N°2 análisis tiempo de respuesta	137
Figura 74. Pregunta N°3 análisis tiempo de respuesta	139
Figura 75. Pregunta N°4 análisis tiempo de respuesta	141
Figura 76. Pregunta N°5 análisis tiempo de respuesta	143
Figura 77. Pregunta N°6 análisis tiempo de respuesta	145
Figura 78. Pregunta N°7 análisis tiempo de respuesta	147
Figura 79. Pregunta N°8 análisis tiempo de respuesta	148
Figura 80. Pregunta N°9 análisis tiempo de respuesta	151
Figura 81. Pregunta N°10 análisis tiempo de respuesta	153
Figura 82. Análisis total tiempo de respuesta	153

Resumen

El mundo académico está avanzando dentro de las instituciones de educación superior al momento de registrar certificados y títulos de estudiantes buscando obtener un acceso confiable a la información, en Ecuador en el año 2019 se filtró 20.8 millones de registros ciudadanos donde el gobierno plantea la ley de protección de los datos personales la cual es aprobada en mayo del año 2021. Los reclutadores de empresas buscan información veraz dentro de plataformas gubernamentales y aun dentro de las misma se presentan inconsistencia de la información, por esta razón implementar un software para construir perfiles académicos online (E-profiles) mediante el uso de elementos del proyecto Solid que permita el accesos seguro, confiable y controlado por terceros a la información extraacadémica y profesional de estudiantes implementando la metodología de Design Science Research permite obtener resultados relacionados al uso de los datos que son manejados mediante el Marco de Descripción de Recursos RDF y los permisos de autenticación mediante OpenID Connect OIDC usado por la parte lógica de perfiles en línea de estudiantes (e-profile) y en la parte gráfica se implementó Aplicaciones de página única SPA, que se pueda implementar una aplicación de web progresiva PWA.

Los resultados obtenidos validan que el acceso de la información es seguro, confiable y controlado para los terceros, se demuestra de forma práctica características de PWA, siendo favorable para una aplicación descentralizada que cuenta con un servidor de almacenamiento Solid utilizando la forma gráfica Mashlib creado por la comunidad utilizando la configuración más básico en el menor precio de DigitalOcean y el cliente Pancka Ruray desplegado de forma gratuita dentro del servicio Firebase de Google.

Palabras clave: E-profile, Descentralizado, Solid Community, Aplicaciones de página única SPA, Aplicaciones de web progresiva PWA.

Abstract

The academic world is advancing within higher education institutions at the time of registering certificates and degrees of students seeking to obtain reliable access to information, in Ecuador in 2019 leaked 20.8 million citizen records where the government raises the law of protection of personal data which is approved in May 2021. Company recruiters seek truthful information within government platforms and even within the same there is inconsistency of information, for this reason implement software to build online academic profiles (E-profiles) using elements of the Solid project that allows secure access, For this reason, implementing a software to build online academic profiles (e-profiles) through the use of elements of the Solid project that allows secure, reliable and controlled access by third parties to extra academic and professional information of students by implementing the Design Science Research methodology allows obtaining results related to the use of data that are managed through the RDF Resource Description Framework and authentication permissions through OpenID Connect OIDC used by the logical part of online student profiles (e-profile) and in the graphic part was implemented SPA single page applications, which can implement a progressive web application PWA.

The results obtained validate that the access of information is secure, reliable, and controlled for third parties, it is demonstrated in a practical way PWA features, being favorable for a decentralized application that has a Solid storage server using the graphical form Mashlib created by the community using the most basic configuration at the lowest price of DigitalOcean and the Pancka Ruray client deployed for free within the Google Firebase service.

Key words: E-profile, Decentralized, Solid Community, Single page application SPA, Progressive web application PWA.

Capítulo I:

Introducción

En este capítulo se describe el problema, el alcance, los objetivos y la hipótesis del presente trabajo de titulación.

Antecedentes

Una aplicación con elementos del proyecto Solid permiten gestionar los datos en formato de POD que son lugares para almacenar información mediante un mecanismo de control para acceder utilizando el Macro de Descripción de Recursos (RDF¹) o Plataforma de Datos Vinculados (LDP²), dentro de un ecosistema interoperable tiene diferentes aplicaciones por ejemplo: en vez de utilizar el correo electrónico y el servicio de notificación de retiro bancario, los servicio de facturación de teléfono, se puede almacenar la información en un POD otorgando acceso para leer la información de correo electrónico a los servicios y aplicaciones, similar a lo que es el ID, DNI o Cédula de ciudadanía (solidproject.org, 2021). Los temas de certificados académicos como lo es títulos profesionales, participación de concursos o eventos dentro de las instituciones educativas, la información generada como datos de los participación, puesto de ganador, estudiante destacado se mantiene dentro de la unidad organizadora la cual ofrece: calidad, credibilidad, gobernabilidad y funciones administrativas, el modelo centralizado no es demasiado flexible mostrando dificultades al buscar reconocer los logros de un individuo o estudiante en particular, sin importar el tipo de reconocimiento formal e informal (Mikroyannidis, 2020).

¹ RDF: abreviación del inglés “*Resource Description Framework*”, es un modelo de datos para metadatos.

² LDP: abreviación del inglés “*Linked Data Platform*”, define un conjunto de reglas para operaciones HTTP

En la era digital que maneja grandes sistemas, software y soluciones en entornos centralizados pasa a ser soluciones de arquitecturas robustas con gran inversión de capital para almacenar la información en perfiles de usuarios, mediante datos que se replican en otra aplicación como los sistemas de: educación, salud, sindicatos, sistemas de transporte, redes energéticas de electricidad en general entidades y departamentos gubernamentales (© The Tapscott Group, 2017) en los temas de transparencia y conciencia pública de la información, los procesos se realizan de una forma jerárquica, las empresas que cuenta con mayor información de los usuarios tiene un privilegio sobre otras compañías, viviendo lo que se denominada “guerra de la información” donde se ha visto necesario aumentar las capacidades de búsqueda para obtener datos que sean relevantes y confiables dentro de la web validando los perfiles recolectados (Tapscott, 2018), es por eso que la mayoría de entrevistadores consideran las cuentas de LinkedIn un lugar donde se expone información esperando sea veraz y de forma pública el currículum vitae es un acceso online, fiel y formal de las personas donde almacenan datos esenciales para el contacto básico, informaciones de trabajos anteriores y preparación profesional, el casos del voluntariado no se puede comprobar, con el simple hecho de seleccionar la organización, colocar el rol que se desempeñó y una descripción de las actividades que se realizó. Esto puede ser cierto o falso al no tener comprobantes de finalización de una forma adecuada la vinculación social o cualquier tipo de voluntariado (Fernández Quintanilla, 2019).

Un ejemplo de una aplicación descentralizada se llevó a cabo con estudiantes de licenciatura que proporcionaron la información sobre la titulación, datos adicionales los documentos colocados directamente en formato digital por los estudiantes que lo obtienen a través de la universidad como un solo archivo, el acceso es limitado a cada estudiante se lo puede descargar a través de la plataforma universitaria, el estudiante puede incluir o excluir los datos. Se lo usa para enviar a una universidad diferente que busque postulantes para el máster, está revisa el material entregado por el estudiante puede tomar la decisión de aceptarlo como un nuevo alumno y finalmente el estudiante recibe la decisión de la universidad que postuló para

temas del máster donde ha sido aprobado o negado dentro de la interfaz de la universidad (Mainini & Laube-Rosenpflanzner, 2016).

Problemática

El reconocimiento positivo que aportan las actividades blandas también llamadas habilidades socioemocionales³ (HSE) dentro del ámbito educativo, profesional y cultural suele parecer que no conlleva a ser el problema, el hecho de no poder tener un registro confiable y con acceso de forma descentralizada de las distintas actividades tanto académicas como sociales que contribuyen al desarrollo y formación de las actividades socioemocionales dentro de las diferentes clasificaciones, dimensiones y las habilidades.

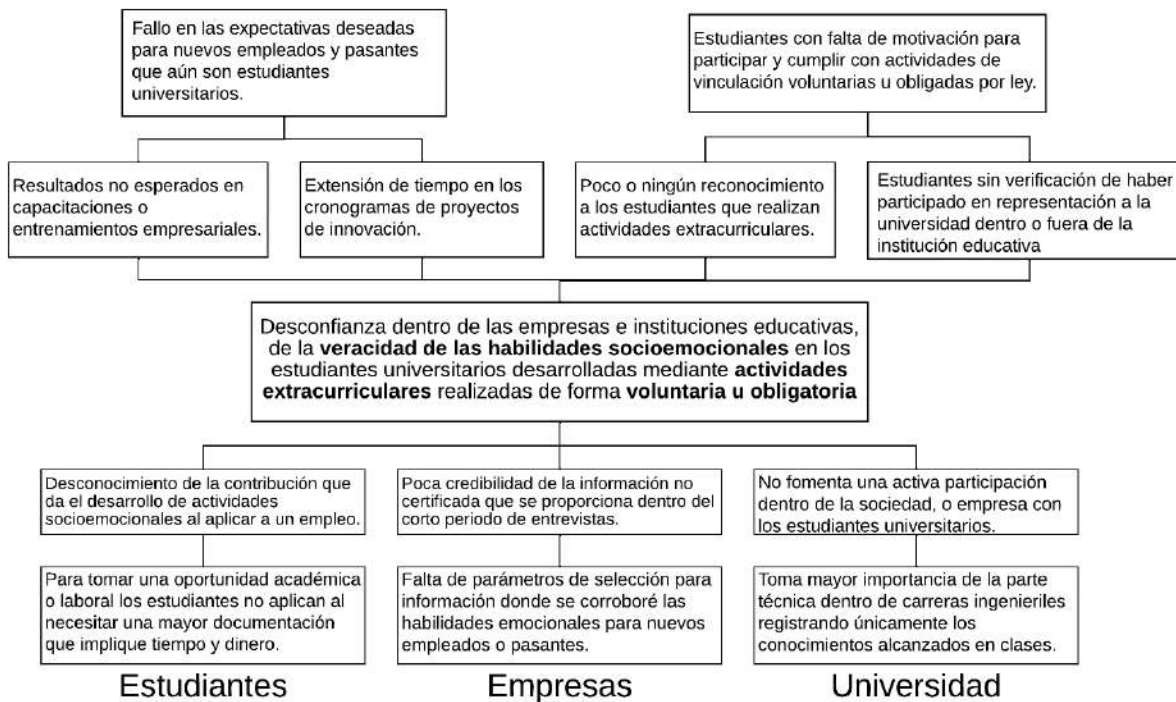
Se puede señalar como causantes a: La falta de interés de los estudiantes al participar dentro de eventos, cursos, charlas o proyectos de vinculación donde se desarrollan estas habilidades, al desconocer sobre las ventajas y beneficios que proporciona la participación dentro de proyectos para la mejora del conocimiento, desarrollando de habilidades técnicas y adquiriendo habilidades socioemocionales.

Se definieron las causas divido en tres posibles actores los estudiantes, las empresas o las instituciones educativas en este caso Universidades las cuales provocan posibles efectos y una problemática descrita al realizar el planteamiento del problema, se detalla el diagrama de “Árbol de Problemas” presente en la figura 1. Que es la representación gráfica del problema descrito a continuación.

Figura 1

Árbol de problemas

³ Habilidades socioemocionales: son características que permite manejar sus propias emociones y la de los demás como el compromiso, empatía o el trabajo en equipo



Nota. En el gráfico se representa un diagrama causa efecto elaborado del autor donde la parte superior se encuentra los efectos, la parte media el problema y la parte inferior las causantes del problema dividido en tres entidades.

Los estudiantes al culminar una carrera de tercer nivel buscan aplicar a un trabajo o a su vez continuar con su formación académica en ambos casos es necesario presentar un currículum vitae donde se resume su experiencia académica, muchos de los reclutadores proponen vincular sus certificados, de reconocimiento y participaciones de cursos, charlas, o concursos mostrando de una forma simple, sencilla y acceso libre respaldado de instituciones de carácter educativo o de empresas del ámbito público o privado donde se compruebe la información de títulos académicos, prácticas profesionales, o vinculación con la sociedad ya que al participar en actividades extracurriculares se desarrollan muchas de las habilidades blandas y si es posible tener un certificado que avale una correcta participación o reconocimiento honorífico del desarrollo dentro de una empresa es muy valorado. Esto es debido al no ser muy común contar con el reconocimiento de actividades de voluntariado, participación de eventos dentro de la

institución de forma individual de cada estudiante detallando el desarrollo de habilidades socioemocionales adquiridas por los estudiantes al participar.

En el mundo profesional se valora el voluntariado sin fines lucrativos al poder realizar trabajos como: organización, planificación, labor comunitaria sin remuneración, compromiso con el trabajo y realizar las actividades de la mejor manera. Existe una red de contactos entre voluntarios a nivel nacional e internacional que solventan problemas de una realidad nacional, el problema es que muchos de los voluntariados son obligatorios, no lo realizan de forma grata más bien es por el cumplimiento, al sentirse amenazados por llevar una sanción y si rara vez cumple con las expectativas no se ven reconocidos dentro del currículum vitae.

Cuando se realiza actividades extracurriculares con una participación excepcional, destacando del resto de participantes no se puede demostrar la existencia de este lo cual representa una ventaja desfavorable al tener las capacidades, aptitudes y conocimientos necesarios, pero sin el respaldo de un certificado emitido por una institución educativa o empresa. Esto provoca una falta de interés dentro de los participantes sean empleados o pasantes que siguen cursando sus estudios, al ser muchos de los proyectos o prácticas no remuneradas donde muchas de las veces la razón principal es terminar y cumplir el total de las horas, más no realizar un buen trabajo.

Una vez culminada la participación toda la información es centralizada, almacenada dentro de la respectiva institución encargada, el estudiante o profesional que requiere un certificado que valide un total de horas de trabajo toma mucho tiempo para obtenerse. Haber realizado labores dentro de un proyecto respaldado por una institución no ocurre todo el tiempo, lo que sí sucede a menudo es aplicar esfuerzo que no se respalda en ningún certificado impidiendo poder corroborar dicha información al ser añadido dentro del currículum vitae.

Las instituciones educativas al tener gran cantidad de estudiantes que participan en proyectos de vinculación invierten gran cantidad de dinero, personal y bienes en certificar de la participación de estudiantes dentro de proyectos de voluntariado, una forma general y unívoca

recopilan la información entregando por primera vez el certificado de cumplimiento pero al solicitar por segunda vez este proceso puede tomar una cantidad de tiempo moderada, al tener todos los datos que un estudiante requiere de forma centralizada.

Al momento de que los estudiantes tienen una participación dentro de proyectos multidisciplinarios entre departamentos y otro tipo de empresas estos pierden tiempo esencial al inicio del proyecto, mostrar sus aptitudes y capacidades, al distribuir tareas, y elegir encargados. En todas las actividades se valora más la experiencia dentro de la empresa, gestionando proyectos y si se tiene algún dominio técnico dentro de proyectos similares, se los considera posibles candidatos para ser líder de todo el proyecto esto debe influir al tener personal que cumple con experiencia dentro de trabajo en equipo y al implementar cierta tecnología. La información suele tomarse de perfiles de los empleados que trabajan en la empresa dificultado al momento de querer reclutar nuevos miembros los cuales se basa en perfiles online los cuales no garantizan la información.

Al organizar los equipos se busca un equilibrio proporcionando ayuda para los gerentes dentro de la red empresarial y universitaria mejorando el crecimiento estudiantil de forma académica y social dentro de un ámbito laboral o de voluntariado.

La falta de motivación estudiantil es causada por no tener un reconocimiento, un valor agregado que sustente la participación e intervención en los proyectos. Que permite un crecimiento laboral mostrando la obtención de experiencia fuera del aula de clases. No existe un certificado que garantiza la participación lo cual impide explorar todo el campo de actividades dentro de la universidad donde se facilita a los estudiantes obtener nuevos conocimientos actualizados, desenvolvimiento social, habilidades que se desarrollan tras haber ingresado y participado dentro de un club, realizando concursos, capacitaciones, socializando con empresas que valoran las actividades extracurricular o participando en eventos académicos o sociales representando y enaltecendo el nombre de la institución educativa.

Justificación

En los últimos cinco años desde el 2017 al 2022 en Ecuador existe un interés de búsqueda de trabajos usando plataformas digitales enfatizando en dos plataformas computrabajo (<https://www.computrabajo.com.ec/>) y multitrabajos (<https://www.multitrabajos.com/>) las cuales tiene una gran variedad de puestos de trabajo en distintas áreas profesionales en la siguiente figura 2 se muestra la cantidad de búsquedas desde Google que tienen estas dos páginas:

Figura 2.

Tráfico de la red computrabajo y multitrabajo en Google



Nota. En el gráfico lineal representa la cantidad de usuarios en Google que buscan el nombre del dominio computrabajo en azul y multitrabajo en rojo, elaborado por la herramienta [Google Trends](https://trends.google.com/trends/explore?cat=958&date=today%205-y&geo=EC&q=computrabajo,multitrabajos)⁴, se obtuvo como media de la popularidad de búsqueda un total de 50 para computrabajo provincia con mayor interés es Guayas y 54 para multitrabajo siendo Pichincha la provincia de mayor interés.

El 11 de mayo del 2021 la asamblea ecuatoriana aprobó la Ley de Protección de Datos Personales el cual busca asegurar los derechos ciudadanos y activar la economía digital entrando en vigor a partir del 2023 para lograr un correcto proceso y trato de los datos a través de la Dirección Nacional de Registro de Datos Público (Dinardap) y crear una Superintendencia

⁴ Google Trends herramienta para medir el tráfico de una o varios términos buscados en Google se compara multitrabajo y computrabajo <https://trends.google.com/trends/explore?cat=958&date=today%205-y&geo=EC&q=computrabajo,multitrabajos>

de Datos con el objetivo de vigilar el resguardo de la información de los ecuatorianos en contra del mal uso de información (Imbaquingo, 2021).

En la era digital se abre la posibilidad a los reclutadores de empleos usar sitios de trabajo para buscar dentro de un abanico de opciones para futuros trabajadores donde se puede reducir el costo total de contratación (Arghyadeep, 2021), si relacionar el currículum vitae a las redes digitales se obtiene una relevancia mediante la conectividad, generando y compartiendo el conocimiento de las actividades laborales que son bien valorados al momento de reclutar personal para nuevos puestos de trabajo (Rivero Amador y otros, 2020).

Uno de los mayor exponentes para almacenar información profesional es LinkedIn con más de 690 millones de miembros, 50 mil empresas, utilizado para buscar oportunidades laborales basado en candidatos potenciales adecuados para oportunidades de trabajo (Arghyadeep, 2021) es similar a un Currículum Vitae con dos niveles, formación académica de cursos haciendo uso del identificador del certificado y la parte del voluntariado que es agregado por cada individuo enlazando a la empresa o entidad donde realizó el trabajo, esta información es usado en la parte educativa en perfiles de docentes, investigadores y estudiantes relacionándolo a las redes digitales para generar y compartir información (Sobrido Prieto y otros, 2019) se tiene algunos ejemplos a nivel latinoamericano para los sistemas y bases de datos, para la gestión y análisis de currículos vitae en expertos demostrando indicadores métricos relevantes al utilizar el Currículum en proyección de políticas en cuanto a la toma de decisiones en procesos de ciencia y tecnología (Rivero Amador y otros, 2020).

Los reclutadores dentro de los perfiles de personas que buscan empleo valoran el tiempo que el usuario usa la plataforma, la edad, la formación académica, actividad extracurricular, actualización de la formación personal, con que empresas o personas tiene relación afirmando los beneficios y recomendaciones al tener un perfil en línea construyendo una marca personal sobre el resto de los postulantes (Arghyadeep, 2021) buscando generar una interacción confiable

entre los usuarios para no tener que verificar la información perdiendo tiempo y al no hacerlo causando problemas a largo tiempo.

Se analizó una encuesta enfocada a la opinión de los reclutadores sobre la importancia que ponen a la opción de “Mis habilidades han sido respaldadas por otros” y “Referencia de trabajo recibida” dentro de las plataformas digitales asignaron como un valor a no tomar en cuenta el respaldo de habilidades por otros y la referencia de trabajo reciba que bien puede ser interpretado como la experiencia laboral de los participantes, estos supieron manifestar que no existe un control de esta información por parte de las plataformas. (Arghyadeep, 2021) La razón que tienen las plataformas de usar una forma tan abierta y sin control para el registro de actividades es por la gran variedad de respuestas que se da en este campo uno puede tener varios trabajos a la vez, puede tener proyectos voluntarios que no tenga certificación similar a los certificados de cursos, por lo que este campo es libre (Tingting y otros, 2018).

Objetivos

En este apartado presentamos el objetivo general y los objetivos específicos que guían este trabajo de titulación.

Objetivo general

Implementar un software para construir perfiles académicos online (E-profiles) mediante el uso de elementos del proyecto Solid que permita el acceso seguro, confiable y controlado por terceros a la información extraacadémica y profesional de los estudiantes.

Objetivos específicos

- i. Realizar un análisis de la relación actual entre los e-profiles y currículum vitae dentro de las actividades de reclutamiento empresarial para salvaguardar la información de los estudiantes recién graduados.
- ii. Verificar la participación dentro de proyectos, empleos, cursos y eventos extraacadémicos mediante la entrega de un prototipo software para la

construcción de perfiles con información extraacadémica online descentralizados de los estudiantes, universidades y empresas.

- iii. Validar el prototipo software, ejecutando un plan de pruebas basado en la evaluación de prototipos que permita comprobar la eficiencia de la construcción de e-profile y conexión con aplicaciones de terceros a través de una aplicación que gestiona la información.

Alcance

El proyecto tiene como objetivo integrar elementos del proyecto Solid y una infraestructura de blockchain trabajando con tres nodos Estudiante, Empresa y Universidad, que proveerá de información al prototipo de software de perfiles académicos online, se agregará información académica certificada desde la Universidad o información extracurricular generado por la Empresa dentro del POD del Estudiante.

Los datos se muestran a través de una aplicación de una sola página también conocida como un Single Page Application (SPA).

Para conocer la base de currículum vitae según la perspectiva del mundo profesional, la estructura y los datos que se colocara en los perfiles académicos online se realizó una revisión de la literatura, encontrando información relacionada con posibles problemas al representar la información, la seguridad que se incorpora en perfiles online y otros fines dentro del uso de blockchain para gestionar la información.

Con el fin de contar con una guía dentro del alcance, se propone la Tabla 1 llamada matriz de congruencia metodológica, que permite plantear las preguntas de investigación relacionadas a cada uno de los objetivos específicos.

Tabla 1.

Matriz de congruencia Metodológica

Objetivo específico	Preguntas de Investigación
<p>i. Realizar un análisis de la relación actual entre los e-profiles y currículum vitae dentro de las actividades de reclutamiento empresarial para salvaguardar la información de los estudiantes recién graduados.</p>	<p>RQ1. ¿Cuál es el valor diferenciador de un e-profile y un currículum vitae?</p> <p>RQ2. ¿Qué aplicaciones y propuestas existen para la creación de e-profiles?</p> <p>RQ3. ¿Cómo se clasifican los tipos de red Blockchain de acuerdo con las ventajas y desventajas en sistemas similares?</p> <p>RQ4. ¿Qué herramientas se utilizan para la implementación de un ecosistema tecnológico basado en sistemas descentralizados?</p>
<p>ii. Verificar la participación dentro de proyectos, empleos, cursos y eventos extraacadémicos mediante la entrega de un prototipo software para la construcción de perfiles con información extraacadémica online descentralizados de los estudiantes, universidades y empresas.</p>	<p>RQ5. ¿Qué flujo de la información se debe seguir al trabajar con estudiantes, universidades y empresas?</p> <p>RQ6. ¿Cuáles son los elementos que se deben implementar dentro del ecosistema tecnológico basado en sistemas descentralizados?</p>
<p>iii. Validar el prototipo software, ejecutando un plan de pruebas basado en la evaluación de prototipos que permita comprobar la eficiencia de la construcción de e-profile y conexión con aplicaciones de terceros a través de</p>	<p>RQ7. ¿Cuáles son las métricas aceptables para validar el correcto funcionamiento de un prototipo descentralizado?</p> <p>RQ8. ¿Qué posibles limitaciones tiene la arquitectura descentralizada?</p> <p>RQ9. ¿Cuáles son los niveles de rendimiento,</p>

una aplicación que gestiona la información.	latencia y disponibilidad para un software funcional?
--	---

Nota. La tabla contiene preguntas de investigación (RQ) planteadas por cada objetivo específico planteado por el autor.

Hipótesis

Un sistema para la construcción de perfiles online implementado a través del proyecto Solid permite un acceso seguro, confiable y controlado a la información de estudiantes universitarios a los terceros

1.6. Señalamiento de variables

- Variable independiente: un sistema para la construcción de perfiles online implementado a través del proyecto Solid
- Variable dependiente: un acceso seguro, confiable y controlado a la información de estudiantes universitarios a terceros.

Capítulo II

Marco teórico y marco metodológico

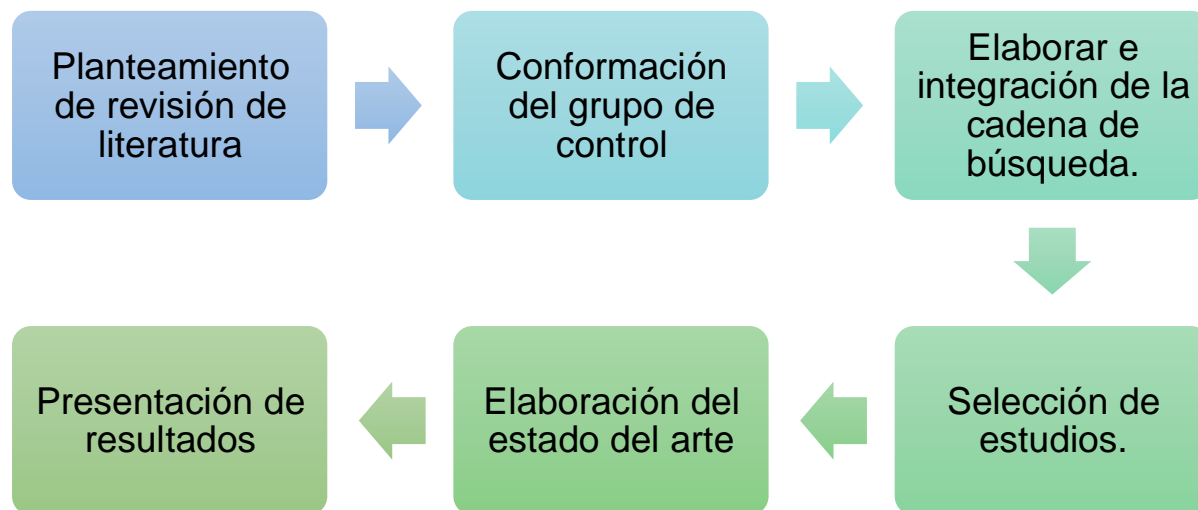
Este capítulo presenta el fundamento teórico y el estado de arte que sustenta al trabajo de titulación.

Estado del Arte

Dentro del proceso de análisis del estado de arte que se relaciona con el manejo de currículum vitae online de actividades académicas, sociales y voluntarios a través de tecnologías descentralizadas, se realizó una revisión de literatura preliminar. Las actividades consideradas para este proceso se describen en la figura 3.

Figura 3.

Proceso de construcción del Estado del arte



Nota. En la figura 3 se colocan los 6 pasos que se siguen para la elaboración del estado del arte, partiendo de un grupo de control y obteniendo artículos científicos que ayudarán al desarrollo del trabajo de titulación.

Planteamiento de la revisión de literatura

En esta fase se realizó una descripción del problema utilizando la figura 1 permitiendo contextualizar una búsqueda de estudios con criterio científico aplicado en las tecnologías de la información.

Criterios de inclusión y exclusión

Se plantea las características que se toma en cuenta y se obvia dentro de la búsqueda de los artículos científicos del proceso de revisión de literatura preliminar. Determinando si será considerado (o no) para la investigación. A continuación, detallamos los criterios de inclusión (CI) y exclusión (CE).

- *CI1*: Estudios que detallen el proceso, diseño y arquitectura de soluciones de software enfocado en la gestión y el uso de información de currículos vitae dentro perfiles en línea.
- *CI2*: Estudios en los que se evidencia usar los perfiles en línea comparado con los currículos vitae mostrando el impacto que se tienen dentro de los reclutadores para conseguir un puesto de trabajo.
- *CI3*: Estudios donde se muestra el impacto de usar perfiles en línea de estudiantes del área de la computación manejando información pública para conseguir una propuesta laboral.
- *CI4*: Estudios de propuestas o implementación de una plataforma digital que gestione la información académica de forma descentralizada y detallando las características de las herramientas que conforman la aplicación.
- *CI5*: Estudios donde permita entender las ventajas y desventajas de la solución con características diferentes a las plataformas digitales ya implementadas dentro de la red para manejar perfiles online.
- *CE1*: Estudios que presentan soluciones con software propietario que implique realizar un pago, sin licencia GLP que sea reutilizable o replicable.
- *CE2*: Estudios con arquitecturas centralizadas enfocadas en la recopilación de datos para ser utilizados en analítica de datos para realizar sistemas de recomendación.

- **CE3:** Estudios publicados antes del 2017, fuera del campo de la educación, publicaciones en conferencias, reseñas, editoriales, estudios, cartas de reporte, noticias, ensayo, estudios fuera de la base digital semantic scholar⁵
- **CE4:** Estudios donde las soluciones se enfoque en el manejo de la seguridad de la red, dispositivos IoT, criptográfica de la información y soluciones enfocadas en el uso de monedas digitales.

Conformación del grupo de control (GC) y extracción de palabras relevantes para la investigación

Basado en las causas del árbol de problemas presentes en la figura 1 se realiza la revisión de literatura no esquematizada, basada en 11 artículos relacionados con la temática, se propone 6 grandes variedades de los temas abordando en los de puntos descritos dentro de la problemática definidos en la Tabla 2.

Tabla 2.

Grupo de Control

Título	Cita	Palabras Clave
Towards A Blockchain-based Decentralized Educational Landscape	(Ramachandran y otros, 2020)	Blockchain, decentralization, verification, education.
Diseño de un Solid POD Siguiendo técnicas DCU	(Fernández Quintanilla, 2019)	Diseño centrado en el usuario, experiencia de usuario, identidad online, datos personales, POD
Blockchain-based approach to create a model of trust in open and	(Lizcano y otros, 2019)	Blockchain, Digital certificate, Competencies, Mined by consensus, Confidence, P2P

⁵ <https://www.semanticscholar.org/>: base digital que recopila varias bibliotecas digitales contando como socios a Springer Nature, IEEE, arXiv, Microsoft Academic, PubMed, Taylor & Francis, SAGE, Wiley, ACM y Unpaywall.

ubiquitous higher education		
Blockchain Applications in Education: A Case Study in Lifelong Learning	(Mikroyannidis, 2020)	Lifelong learning; blockchain; decentralization; smart badge; personalized recommendation.
GDPR-Compliant Personal Data Management: A Blockchain-Based Solution	(Truong y otros, 2020)	Blockchain, data management, GDPR, personal data, smart contract.
Access Control in Linked Data Using WebID	(Mainini & Laube-Rosenpflanzler, 2016)	Semantic Web, WebID, Linked Data, Access Control, Authentication, Authorization

Nota. En la tabla 2 se muestran tres columnas título el nombre del artículo científico, Cita el formato en el cual se hace referencia al artículo científico y Palabras claves las palabras que identifican a los artículos científicos. El grupo de control fue seleccionado por el autor bajo su criterio de los artículos con temáticas relacionada a los perfiles en línea usando el buscador semanticsholar. Teniendo en cuenta las veces que se citó cada artículo, sin importar el idioma, ni el lugar, teniendo en cuenta solo el acceso libre y en formato digital PDF.

Después de analizar los estudios del Grupo de Control, se obtuvo una selección de palabras relevantes respecto al objetivo de búsqueda, presentando los siguientes términos: blockchain, CV, resume, curriculum vitae, e-profile, profile, personal data.

Construcción y afinación de la cadena de búsqueda

Se va a conformar la cadena utilizando varios términos relevantes obtenidos al analizar el grupo de control, permitiendo encontrar los artículos en su respectiva base de datos digital.

Para establecer el objetivo de búsqueda se plantea 4 contextos que agrupa las palabras clave con sus sinónimos indicados a continuación:

- Objeto: Propósito de las acciones
- Contexto: El lugar donde se desarrolla la investigación
- Propósito: Detalle del porque se realiza la investigación
- Mecanismos: Herramientas utilizadas

La cadena de búsqueda se considera ideal al cumplir las siguientes características:

- Los artículos que se obtienen son relacionados con la temática y cumple los criterios de inclusión y exclusión.
- El total de artículos es manejable para realizar un análisis.
- Dentro del resultado de los estudios que se obtienen, aparecen la mayor cantidad de los estudios de control.

Tabla 3.

Pasos para elaborar la cadena de búsqueda

Cadena de Búsqueda	Total, de artículos obtenidos
("profile" OR "cv") AND "decentralized"	519000 artículos obtenidos
((“profile” OR “data”) AND (“solid” OR “pod”)) AND (“blockchain” AND “e-profile”)	74300 artículos obtenidos
“blockchain” AND ((“profile” OR “data”) AND (“solid” OR “pod”)) AND (“e-profile” OR "curriculum system" OR "job market")	3740 artículos obtenidos
“blockchain” AND ((“extracurricular activities” OR “soft skills” OR "hard skills") AND (“solid” OR “pod”)) AND (“e-profile” OR "curriculum system" OR "job market")	1310 artículos obtenidos
“blockchain” AND ((“extracurricular activities” OR “soft skills” OR "hard skills") AND (“SOLID” AND “POD”)) AND (“e-profile” OR "curriculum system" OR "job market") AND ("access control" OR "decentralization") NOT (“Solidity” or “Solidifier” or “crypto wallet”)	95 artículos obtenidos

Nota. En la tabla 3 tiene 2 columnas, la cadena de búsqueda donde se detalla los términos que se van agregando y el término booleano que se utilizó, el total de artículos obtenidos es la cantidad de resultados que se obtiene al aplicar la cadena de búsqueda.

Selección de estudios

La cadena seleccionada se aplica dentro de Semantic Scholar obteniendo 95 artículos relacionados con el tema a desarrollar. Siendo considerado suficiente, y adecuado para el análisis.

De estos 95 artículos se aplicaron tres filtros, descritos a continuación los cuales eran fáciles y aplicables dentro del sitio web:

- a. **Fecha de Publicación:** se seleccionó artículos realizados hasta 10 años atrás es decir desde el 2012, debido al cambiante ámbito de la tecnología en temas como lo es Blockchain de igual forma siendo actividades recientes donde muestra el impacto de aplicar tecnologías para el manejo de perfiles online.
- b. **Tipo de Estudio:** Seleccionando solo estudios de tipo: Journals y Conferences.
- c. **PDF:** es usado para seleccionar aquellos resultados que son de libre acceso y en un formato PDF.

Del uso de la cadena de búsqueda y los filtros se obteniendo un total de 54 artículos, de los cuales después de realizar una comparativa, se eliminan los resultados dentro de áreas fuera de la educación soluciones con una misma tecnología tomando a las propuestas más actuales se reducen a 9 artículos de los cuales se procede a la lectura de su temática, resumen y conclusiones de los seleccionados se construye la base para realizar el estudio del estado del arte, mostrado en la Tabla 4:

Tabla 4.

Estudio Seleccionados

Código	Título	Cita
EP1	Towards Complete Decentralized Verification of Data with Confidentiality: Different ways to connect Solid Pods and Blockchain	(Ramachandran y otros, 2020)
EP2	User-Controlled Privacy-Preserving User Profile Data Sharing based on Blockchain	(Shrestha y otros, 2017)
EP3	GDPR-Compliant Personal Data Management: A Blockchain-Based Solution	(Truong y otros, 2020)
EP4	Creating student's profile using blockchain technology	(Juricic y otros, 2019)

EP5	Security analysis of a blockchain-based protocol for the certification of academic credentials	(Baldi y otros, 2019)
EP6	Blockchain Based Framework for Driver Profiling in Smart Cities	(Pramod & Sankaran, 2019)
EP7	A Blockchain-Based Scheme for Secure Storage and Sharing of Student Digital Profiles	(Jin y otros, 2020)
EP8	Proofware: Proof of Useful Work Blockchain Consensus Protocol for Decentralized Applications	(Dong y otros, 2019)
EP9	Exploring the Innovative Aspects of CV Distributed Ledgers Based on Blockchain	(Silva y otros, 2020)

Nota. En la tabla 4 se observa tres columnas título el nombre del artículo científico, Cita el formato con el cual se hace referencia al artículo científico y las palabras que identifican a los artículos científicos. Los estudios que se seleccionan tienen el peculiar de ser parte del grupo de control y al tener la cadena de elaboración se agrega artículos científicos relacionados y un criterio de exclusión que fue detallado con anterioridad.

Resumen de los Estudios Primarios

EP1. Towards Complete Decentralized Verification of Data with Confidentiality:

Different ways to connect Solid Pods and Blockchain (Ramachandran y otros, 2020).

El artículo presente propone la implementación de un sistema llamado LinkChains que basado en blockchain verifica las credenciales académicas mientras almacena la credencial y los datos para respaldar el título, siendo usado cuando una credencial académica es emitida en la plataforma LinkChains, se procesa la credencial y luego almacena el hash en una red blockchain, al mismo tiempo la credencial académica es subido al POD del estudiante en Solid se proporciona un token al estudiante con el almacenamiento de los metadatos de forma descentralizada y modular. Reduciendo significativamente el tiempo de verificación de la información que realizan los empleadores.

Las soluciones que centralizan la información de forma desmesurada se considera un problema al perder el control de los datos de usuarios causando alteración de la información por

actualizaciones de los propietarios del sistema, corrupción dentro de las compañías y una de las más notorias la manipulación intencional por administradores de control que modifican o eliminan datos sin el consentimiento o conocimiento del propietario. Otro tema importante es la confidencialidad de los datos que se pierde cuando los datos son compartidos, vistos o vendidos por los poseedores. Una propuesta que ha demostrado ser beneficiosa es el almacenamiento de datos con carácter financiero, médico, científico, personal y confidencial donde se requiere una integridad y acceso.

Existen varios enfoques de aplicaciones descentralizados probados de forma práctica y literaria incluyendo almacenamiento de datos descentralizados punto a punto, almacenamiento de datos en registros distribuidos y una combinación que contiene datos y luego hash garantizando la integridad de los primeros. Se tiene un número mayor de personas que falsificaron credenciales académicas y reclutadores de empleos gastan recursos para autenticar la validez de títulos educativos, no se diga aquellos certificados de participación o voluntariado. Se realizó un análisis de 5500 currículos realizado por el Grupo Asesor de Riesgos encontrando que el 44% de estos tienen discrepancias en las afirmaciones educativas y un 10% tienen calificaciones falsas

EP2. User-Controlled Privacy-Preserving User Profile Data Sharing based on Blockchain (Shrestha y otros, 2017)

En el presente documento de investigación se construyó una arquitectura de intercambio de datos descentralizada con blockchain usando multicadena (MultiChain) en ámbito de viajes siendo aplicable dentro de temas de educación, salud y deportes. La propuesta es para empresas que operan en la industria del turismo como agencia de viajes y excursiones, hoteles y resorts, centros comerciales entre otros. El principal incentivo son los contratos inteligentes que son aplicados en blockchain para la protección de los datos, donde el usuario o propietario gestiona a quien desea compartir sus datos

Conectándose mediante MultiChain para realizar reservas en hoteles ficticios los nodos de la blockchain, recopilando información de datos del perfil de usuario con su respectiva validación y permita decidir cuales datos desea compartir asegurando al usuario el control y preservación de la información.

Existen tres categorías de blockchain, el tipo de red privada tiene una comunidad cerrada que almacena las transacciones que son de interés solo para los participantes presentes en la cadena, el público donde cualquier persona puede participar en la cadena y contribuir al proceso de creación de consenso, el consorcio permite que los nodos preseleccionados controlen el proceso de consenso que es una mezcla de red pública y privada.

En la solución se utilizó la red privada en forma de MultiChain al admitir una mayor cantidad de datos, resuelve problemas de minar de forma privada y gestiona permisos de usuarios existiendo solo cadenas de interés garantizando que sea visible los datos para los participantes elegidos, introduciendo controles sobre las transacciones y permite que la minería se lleve a cabo de forma segura sin pruebas de trabajo y costos asociados.

EP3. GDPR-Compliant Personal Data Management: A Blockchain-Based Solution
(Truong y otros, 2020).

En el presente artículo propone el concepto del diseño para una plataforma que gestiona datos personales basado en blockchain compatible con el Reglamento General de Protección de Datos (RGPD) sobre el marco Hyperledger Fabric Blockchain framework (HLF) interactuando entre un Servidor de Recursos (RS) y un Proveedor de Servicios (SP) de redes sociales con ayuda del Procesador de Datos (DP) y Propietario de Datos (DS) garantizando que las actividades sobre la información de perfiles almacenados sean viables y eficaces dentro del RGPD.

Se hace uso de blockchain y contratos inteligentes de forma descentralizada a propietarios de datos y proveedores de servicios, permitiendo a los usuarios imponer el consentimiento de uso designando de los datos se pueden utilizar y registrar guardando los en

blockchain para que sean inmutables. La arquitectura del sistema de alto nivel busca transferir a una red blockchain desde un servidor centralizado la autorización y autenticación, IdM y control de acceso, y los componentes de registro y procedencia implementados sobre un contrato inteligente

Para decidir cuales son los líderes de la tecnología en blockchain que debemos utilizar primero dividimos en redes blockchain públicas con eso en mente comparamos el rendimiento limitado que se tiene en varios proyectos similares Bitcoin obtiene 7 transacciones por segundo (tps) usando Blocktime 10 minutos y Ethereum Enterprise Alliance 15 tps mediante Blocktime 22 minutos y 15 segundo, ahora cambiamos a las redes de consenso donde teóricamente tenemos mayor rendimiento FabricCoin sobre un marco HLF alcanza 3500 tps con un punto de referencia de rendimiento de Hyperledger Caliper

EP4. Creating student's profile using blockchain technology (Juricic y otros, 2019).

El presente artículo recopila información sobre distintos casos de estudio donde la tecnología blockchain está presente en el área de la educación con universidades que lo aplican de forma parcial o completa. El proyecto Blockcert un software de código abierto desarrollado por Media Lab en la universidad de MIT y la empresa Learning Machine permite a las instituciones educativas emitir certificados académicos dando a los usuarios el control, de los documentos oficiales el cual no necesita agregar información privada para respaldar la integridad del certificado, Sony corporation y Sony education desarrollaron un sistema educativo de certificados utilizando blockchain.

Los currículos vitae requieren de un respaldo físico el cual cuenta con un identificador, una prestigiosa o por lo menos reconocidas entidades que emita el certificado para garantizar la credibilidad de los conocimientos y si estos no son completamente cumplidos no se emite ningún reconocimiento aun cuando las habilidades del estudiante mejoraran, la dependencia de un reconocimiento es lo único que toman en cuenta para poder continuar las siguientes fases del reclutamiento, los sistemas actuales para currículos no registran datos personales con

blockchain, solo valores hash de los certificados debido al ser un proceso unidireccional es imposible recuperar la información original

Los e-profiles tienen al dueño de la información como aquel que controla los datos independientes de la institución, mediante registros inmutables de la documentación oficial y con certificados de total confianza. Para un empleador fiarse de la información puede ser una espada de doble filo por lo que al manejar perfiles en línea los estudios proporcionados por estudiantes mediante, las evaluaciones del conocimiento que rinden son fiables con una calidad del aprendizaje adquirido. Para los estudiantes a que de manejar datos privados y públicos puede ver solo valores que ellos otorgan el permiso dentro de la universidad ya no manejan grandes volúmenes de información y tendría menores costos para la gestión de trámites administrativos, llevando a un máximo rendimiento de las cadenas de bloques los problemas burocráticos en plan de estudios, resultados error en los, cambios de mallas se lograría convalidar materias basado en el aprendizaje del estudiante, plataformas de aprendizaje, actividades y estudios extracurriculares de cursos gratuitos, pagados, presenciales o en línea al registrar las habilidades y conocimientos que se obtienen. Con esto poder postular y que los empleadores tengan bases fiables para potenciales empleos

EP5. Security analysis of a blockchain-based protocol for the certification of academic credentials (Baldi y otros, 2019).

En el presente artículo científico detalla una solución donde une dos formas para validar información de certificados de perfiles online, el primer paso para verificar las competencias en línea lo realizó la Fundación Mozilla colaborando con la fundación McArthur mediante el proyecto llamado Open Badges tiene como objetivo proporcionar a las organizaciones e instituciones un sistema para emitir insignias digitales a los propietarios de competencias para el reconocimiento, no solo del aprendizaje oficial sino también de habilidades blandas, diseñado para proporcionar un perfil detallado del destinatario recopilando todos los registros académicos de un estudiante.

La falta de validación de certificados compatibles dentro de Open Badges se utiliza un sistema basado en blockchain usado internacionalmente por MIT y Learning Machine llamado Blockcerts firmando criptográficamente los datos de certificados de títulos académicos dentro de una cadena de bloques públicos descentralizados y auto consistente. Para entender como funciona la insignia es un archivo de imagen que contiene metadatos incrustados, representa el alcance describiendo por que se otorgó, quien emite, y quien obtuvo el reconocimiento.

Se maneja tres secciones Profile, BadgeClass y Assertion combinados se genera el OpenBadge, primero crean badgeclass y otorga afirmaciones a los destinatarios, luego el destinatario comparte en redes sociales, correo electrónico o algún medio digital, por último, todo el mundo puede comprobar la autenticidad de la credencial donde se afirma que se otorga. La verificación se lo realiza mediante el hash del certificado calculado localmente y coincida con el receptor del certificado, compara el valor raíz de Merkle escrito en el recibo del certificado donde la raíz de Merkle escrita en la cadena de bloques, finalmente comprueba la recepción del certificado que se está analizando, formando parte del árbol Merkle.

EP6. Blockchain Based Framework for Driver Profiling in Smart Cities (Pramod & Sankaran, 2019).

En la presente investigación enfocada en gestionar los datos de conductores para utilizar en ciudades inteligentes donde permita monitorear patrones de conducción, infracciones de tránsito y resolver problemas legales, basado en blockchain para la creación de perfiles de conductores está compuesto de varios servicios Registro del conductor y validación de identidad mediante el servicio de autenticación, Proporcionar al conductor un token único basado en el tiempo para registrar datos basados en eventos para garantizar la autenticidad de los datos y Evaluar el sistema midiendo rendimiento y consumo energético analizando ventajas y desventajas como la medición de energía llevado a cabo usando la unidad de medida de fuente Keithleys serie 2461 para saber precisión y voltaje

Cuenta con el marco de dos cadenas de bloques que alojan diferentes servicios, la gestión de servicios es el responsable de autenticación de usuarios y generar tokens de validación de usuarios al registrarse y la cadena de bloques de manejo de eventos donde registra las acciones de los conductores, contiene datos de vehículos y conductores que se pueden compartir con agencias gubernamentales.

Para la creación de perfiles del conductor cuenta con el registro exitoso donde el minero proporcionará el pseudo certificado al conductor, la transacción se crea para el evento de registro correspondiente y se almacena en el servicio de manejo de Blockchain. Manejo de eventos y almacenamiento de datos se lleva a cabo mediante contratos inteligentes escritos para identificar eventos, registrando eventos enviados a Blockchain junto con el token validado y cuenta con dos partes el encabezado del bloque tiene tres conjuntos metadatos del bloque combinados para formar la cabecera de bloque es un hash actual más un hash del bloque anterior, la dificultad marcada por el tiempo y el *nonce*⁶ utilizado, y por último la raíz del árbol Merkle con una estructura de datos utilizada para resumir todas las transacciones y las transacciones son verificadas por los nodos mineros en caso de que se maneja blockchain para verificar la autenticidad de registros

EP7. A Blockchain-Based Scheme for Secure Storage and Sharing of Student Digital Profiles (Jin y otros, 2020).

En el documento de investigación propone un esquema seguro de almacenamiento e intercambio de perfil digital para estudiantes basado en la tecnología blockchain, con las características de descentralización, seguras, creíbles y resistentes a la manipulación de la cadena de bloques un consenso de nodos de la red a través del protocolo llamado Tolerancia Práctica a Fallas Bizantinas en inglés Delegated Byzantine Fault Tolerance (DBFT)

⁶ Nonce: número arbitrario utilizado una sola vez dentro de un proceso criptográfico.

El almacenamiento de información educativa en China es uno de los países que ha implementado el formato digital llegando al mismo valor legal que tiene el formato en papel permitiendo a los estudiantes compartir sus currículum vitae en estos dos formatos, pero el formato digital no llega a ser un e-profile descentralizado por tener presente problemas como la asimetría con la emisión de los certificados de autenticación, la inserción y modificación de los datos que no se puede rastrear. Los sistemas de información que gestionan las instituciones educativas trabajan de forma independiente entre sí impidiendo tener un perfil digital personal con la información completa y por último los datos son almacenados de forma centralizada.

Al aplicar tecnología blockchain se pueden establecer perfiles digitales completos de los estudiantes que incluya todo tipo de información de los estudiantes como estados de registro, calificaciones académicas, certificados obtenidos, rendimiento escolar entre otras actividades que se pueda registrar con un respaldo digital, la autenticidad de perfiles estudiantiles se verifica a través de datos de la cadena, reduciendo el costo de verificación de perfiles para los reclutadores empresariales.

Una de las partes primordiales es el uso del protocolo de consenso usado para la verificación de los nodos dentro de blockchain dependiendo la aplicación uno es más beneficios que otro destacando los siguientes: Prueba de Trabajo (PoW), Prueba de Participación (PoS), Prueba de participación delegada (DPoS), Tolerancia Práctica a Fallas Bizantinas (DBFT) es usado de forma similar a DPoS los nodos tienen derecho a voto para elegir delegados o tenedores para verificar nuevos bloques reduciendo la cantidad de nodos de verificación, resolviendo problemas de escalabilidad.

DBFT adopta dos tipos de nodos los ordinarios son nodos no participantes en la verificación y los de contabilidad son responsables de la verificación, pueden ver proceso de consenso y necesitan sincronizar la información determinada por nodos ordinarios mediante votación según la proporción de participación. Esto garantiza el consenso de los nodos de la red en el sistema de perfiles digitales de los alumnos, intercambio seguro de perfiles de estudiantes

mediante el almacenamiento de datos en base de datos fuera de la cadena con protocolos de control de acceso

EP8. Proofware: Proof of Useful Work Blockchain Consensus Protocol for Decentralized Applications (Dong y otros, 2019).

En el presente documento se presenta Proofware el que está diseñado para los desarrolladores y construyan aplicaciones descentralizadas (dApps), pueden probar protocolos de consenso de Prueba de Trabajo Útil (PoUW), al trabajar con un sistema de moneda incorporado para mantener el sistema de incentivos descentralizados, en este caso se creó una aplicación para compartir vídeos basada en multitudes denominándose OurTube.

Se analiza los diferentes protocolos de consenso para usar en con blockchain Prueba de trabajo (PoW) que resuelve problemas de escalabilidad con dificultad al consumir recursos computacionales (electricidad) para mantener seguridad de la red y el consumo de energía, no aplicable a los negocios, Prueba de Participación (PoS) propuesto por Peercoin no requiere un proceso de minería para llegar a consenso pidiendo a los usuarios que demuestran cierta cantidad de moneda, PoS está diseñado para resolver la desventaja del consumo energético a cambio reduce la escalabilidad causando un gran problema al no poder coordinar la apuesta entre todos los nodos y Prueba de Participación Delegada (DPoS) para solventar el último problema de escalabilidad y no limitar los nodos se genera un número impar de nodos para participar en el proceso de elección, el beneficio alcanzando una Transacción por Segundo (TPS).

Las pruebas de la capacidad o prueba de espacio es la cantidad significativa del espacio en disco en lugar de computación PoW implementa como Metadisk de Storj una plataforma de almacenamiento descentralizada en la nube. Los costos de ejecución de Proofware es alrededor de 5.4% de ejecución en Amazon EC2. El Grid Computing utiliza un programador de tareas distribuidas para dividir las tareas, distribuye a las computadoras voluntarias y fusiona el resultado que es enviado, Crowdfare similar a Grid Computing agregando un mecanismo de

incentivo financiero, Proofware aprovecha el sistema de créditos basado en blockchain, realiza un incentivo autocontrolado, usa contratos inteligentes autónomos para el servicio y un kit de herramientas dApp

Una comparación de plataformas de cómputo a gran escala como Karma, Cloud Service, Bitcoin Network, Grid Computing, Crowdware y Proofware está organizado de la más baja para Resistencia a la censura, Descentralización, Elasticidad, Rendimiento Autónomo, Fiabilidad, Rentabilidad, Costo eficiencia donde la más popular puede no ser la más alta en estos parámetros pero es donde tiene mayor adeptos es la solución de Bitcoin Network y la que tiene mayor alcance en los parámetros es Proofware y puede que no sea la más conocida y aplicada en los proyectos para realizar los consensos dentro de una solución causando problemas futuros.

EP9. Exploring the Innovative Aspects of CV Distributed Ledgers Based on Blockchain (Silva y otros, 2020).

El artículo gira en torno a identificar los aspectos innovadores de sistemas de gestión de registros de CV basados en un paradigma de cadena de bloques mediante el uso de revisión asertiva de la literatura del Servicio de Información Comunitaria sobre Investigación y Desarrollo (CORDIS).

Se plantean preguntas de investigación que son respondidas una de ellas es: ¿Qué y cuántos proyectos científicos relevantes se están desarrollando en torno a este tema? y ¿Cuáles son las arquitecturas propuestas? Se obtiene seis proyectos relevantes Bloomem usado para manejo de transacciones de diferentes usuarios en línea utilizando base de datos distribuidas, QualiChain crea, evalúa y manejo de plataformas descentralizadas donde se almacena, comparte y verifica las calificaciones educativas y laborales, KOPAR investiga la naturaleza de documentos científicos desde una perspectiva semántica integradora-interoperable, SmartCertificate genera y comparte documentos certificados verifica el acceso de los documentos ResearchProof deposita, de forma pública y poder comparar trabajos científicos

dentro de una plataforma donde los investigadores pueden almacenar cualquier tipo de artefacto de investigación.

Al utilizar blockchain permite una implementación de bases de datos distribuidas confiables como es el registro de currículum Vitae (CV) como bloques en una cadena vinculada, puede ser accesible, transparente para todos los Participantes propietarios de CV, certificadores y empleadores siendo una forma global y universal de CV de personas u organizaciones. Se puede registrar CV con autoría y derechos digitales de contenido de medios, al tener base de datos distribuida para alimentar plataformas de CV ejemplo un título de doctorado, un artículo de investigación, un registro de investigación, la comunidad científica, los Objetos de Investigación (RO) son elementos importantes del CV de un investigador.

QualiChain es un proyecto con enfoque evaluativo en la verificación de equivalencia de títulos entre universidades, diseño curricular inteligente, dotación de personal del sector público, prestar servicios de consultoría en recursos humanos y gestión de competencias. Permite cargar CV para tener recomendaciones, así como todas las actividades de quien es estudiante y/o busca trabajo, los proveedores de información son las instituciones educativas las cuales brinda datos del plan de estudio, evaluaciones y certificados, los reclutadores crean descripciones de puestos, requisitos, criterios de selección, verificación de calificaciones, y temas relacionados con procesos de contratación. Siendo una de las plataformas más versátiles, relevantes y exitosas para una plataforma universitaria, el proyecto KOPAR contribuye debido al enfoque de los metadatos de forma semántica interoperable de publicaciones científicas y todos los datos agregados a ellos.

Respuesta a las preguntas de investigación.

¿Cuál es el valor diferenciador de un e-profile y un currículum vitae?

Un e-profile es manejar información personal del usuario mediante la tecnología (Juricic y otros, 2019), (Jin y otros, 2020), (Silva y otros, 2020), uno de los países que busca igualdad de

la validez de la información para hacer más sencillo sus trámites académicos, reduciendo costos por la gran cantidad de personas es China donde se puede presentar títulos académicos mediante una referencia electrónica siendo de igual valor que mostrar un papel físico, certificado y con todas las validaciones correspondientes (Juricic y otros, 2019), (Jin y otros, 2020).

Se busca al desarrollar y elaborar un e-profile la información proporcionada de forma seguro, confiable, fidedigna, inmutable y esta pueda ser manejada por el dueño de la misma permitiendo contar con la autoría que se busca dentro del Reglamento General de Protección de Datos RGPD (Truong y otros, 2020), (Juricic y otros, 2019), es algo que para los reclutadores significa reducción de tiempo y esfuerzo al contratar a un nuevo personal contando con la certeza de que los conocimientos colocados son verídicos y además recopila toda la información laboral en un solo sitio (Juricic y otros, 2019), (Baldi y otros, 2019).

¿Qué aplicaciones y propuestas existen para la creación de e-profiles?

Para el Desarrollo de un e-profile que busca automatizar la información, descentralizar la información, permitir confiar en la información registrada se cuenta con aplicaciones para los certificados como Blockcert (Juricic y otros, 2019), (Baldi y otros, 2019) y puede usarse en conjunto con una base de datos descentralizada para que la información que está en una red privada pueda accederse a través de permisos de vista usando Solid Server (Ramachandran y otros, 2020) si se desea tener una guía para consumir se una aplicación descentralizada se debe buscar QualiChain que es uno de los proyectos más relevantes (Silva y otros, 2020).

Una plataforma que ayuda a generar soluciones que se requiere añadir datos inmutables y que sea de forma privada Multichain permitiendo al desarrollador tener una fácil integración, de sistemas descentralizados cuyo aporte es ser uno de los más flexibles que se puede usar como un proyecto para reserva de hoteles o información de conductores con temas diferentes al tener que comprimir la información en hash que son almacenados por transacciones de información (Shrestha y otros, 2017), (Pramod & Sankaran, 2019).

Un punto que se toma en cuenta es el modelo del negocio que tendrán las propuestas para minar monedas digitales (Truong y otros, 2020), grupos de consenso para garantizar una participación y un crecimiento continuo (Dong y otros, 2019) y en otros casos consumir monedas ya existentes dentro de su dAPP (Silva y otros, 2020).

¿Cómo se clasifican los tipos de red Blockchain de acuerdo con las ventajas y desventajas en sistemas similares?

Existen tres tipos de redes blockchain de forma general privada aquella que cuenta con una comunidad cerrada almacenando transacciones de interés para los participantes ejemplo de esta red Multichain para el desarrollo de dApps (Shrestha y otros, 2017) y para manejar pagos con la novedad de contratos inteligentes es Ethereum (Truong y otros, 2020) esta red suele usar protocolos de consenso PoS que no consumen mucha energía, pero cuenta con nodos limitados (Dong y otros, 2019). Red pública es aquella donde cualquier persona interesada en el proyecto puede ser parte del mismo creando contenido transaccional o minando las transacciones (Shrestha y otros, 2017) la cantidad de usuarios suele ser superior a los procesos utilizando el protocolo de consenso PoW solucionando el problema de la escalabilidad pero todos los nodos compiten para realizar la verificación de una transacción siendo una desventaja el desgaste computacional sin recibir nada a cambio (Jin y otros, 2020), (Dong y otros, 2019) un ejemplo es bitcoin donde cualquier persona puede usar y minar la criptomoneda (Truong y otros, 2020) y por último la red de consorcio la cual toma las ventajas de las redes privadas y públicas (Shrestha y otros, 2017) la mayoría de proyectos que se incorporan como Blockcert, QualiChain, KOPAR son una mezcla de redes privadas y públicas (Silva y otros, 2020) utilizando protocolos de consenso como DPoS (Jin y otros, 2020), (Dong y otros, 2019) o utilizan el marco Hyperledger Fabric Blockchain Framework (Truong y otros, 2020).

Dentro de las dApps y que usan blockchain para descentralizar los datos cuentan con características en común al mantener los datos inmutables, confiables, auditables y un almacenamiento distribuido manejando diferentes tipos de datos como la información personal,

certificados académicos, información de rutas de conductores, vídeo, reservas de hotel lo cual demuestra que dependiendo del objetivo el aplicativo puede implementar una u otra arquitectura para manejar usuario.

¿Qué herramientas se utilizan para la implementación de un ecosistema tecnológico basado en sistemas descentralizados?

Un sistema descentralizado permite incorporarse con uno o más aplicaciones y/o servidores los cuales se encuentran distribuidos incorporando blockchain (Truong y otros, 2020) que no tienen propiedad ni una institución que los regula la información todo se guarda y almacena entre la distribución de los servidores (Baldi y otros, 2019), existe la posibilidad de descentralizar la información dentro de blockchain teniendo un problema la replicación de la información en la red (Truong y otros, 2020) se vuelve muy difícil mantener usuarios y transacciones dentro de un mismo ecosistema al tener la verificación de la información el número arbitrario usado para establecer una comunicación criptográfica que incrementa la dificultad en generar el bloque perteneciente a la cadena (Pramod & Sankaran, 2019), (Dong y otros, 2019) la alternativa es el proyecto SOLID Server creado por el fundador de la Web Tim Berners-Lee donde se tiene un servidor descentralizado que puede ser usado desde distintas aplicaciones reutilizando la información almacenada en POD (Ramachandran y otros, 2020) y por último hay una opción de solución para el almacenamiento masivo en nodos blockchain con ayuda del protocolo de Tolerancia Practica a Fallas Bizantinas donde se tiene dos nodos los ordinarios que no participan en la minería para verificar datos dentro de la prueba de consenso y los nodos correctos que se cuenta con garantías para el consenso (Truong y otros, 2020), (Jin y otros, 2020).

Dentro del tipo correcto que se debería usar basado en resultados para manejar temas de escalabilidad debería ser el protocolo de consenso PoW lo cual estaría bien mientras la cantidad de mineros sea baja y con un dificultad de procesamiento fácil, se tienden de a desperdiciar mucha energía por esa causa se usa DPoS solventando las dificultades futuras

(Dong y otros, 2019) y si se busca acoplar proyectos que se han desarrollado para ciertas áreas como emisión de certificados Blockcert es una gran alternativa (Juricic y otros, 2019) (Baldi y otros, 2019) (Silva y otros, 2020).

Resumen general y conclusión del estado del arte.

Mediante el estudio del estado del arte se conoce la diferencia entre e-profiles y currículum vitae en los países donde aplican soluciones descentralizadas mediante tecnologías blockchain, se resalta proyectos como Blockcert o QualiChain utilizados para manejar perfiles en línea.

Por lo que se determina la factibilidad al implementar una arquitectura descentralizada adquiriendo el conocimiento utilizado en los sistemas distribuidos, contando con un abanico de herramientas para solventar el problema al no poder almacenar información utilizada de distintos certificados confiables que un estudiante con el fin de mejorar las condiciones para conseguir un empleo reduciendo el tiempo invertido por los reclutadores utilizando información validada.

Metodología

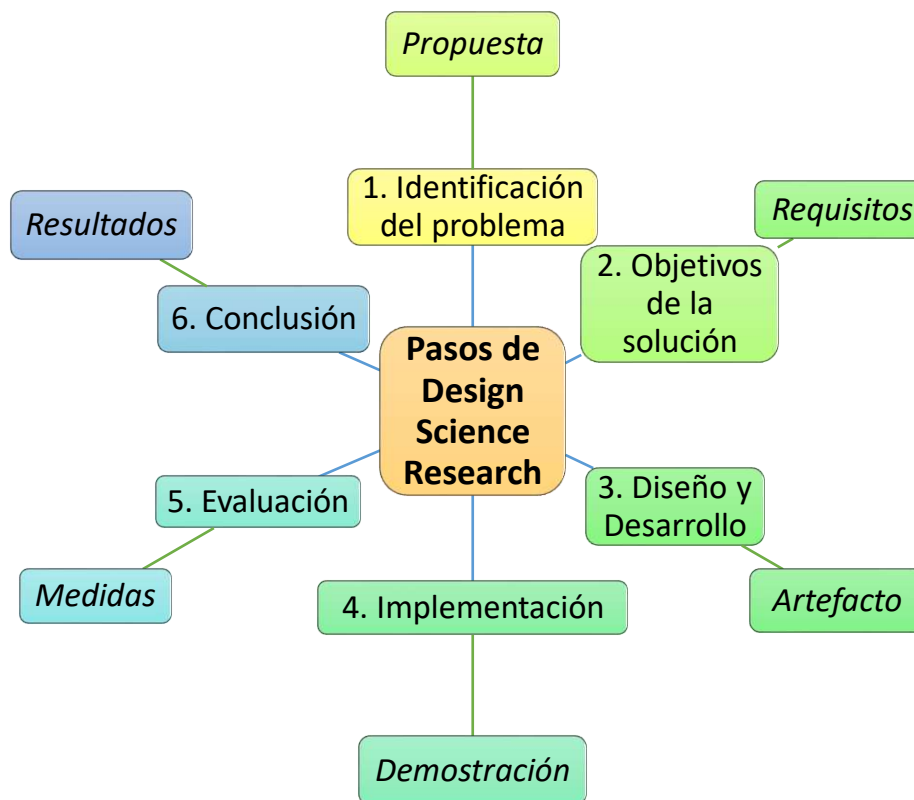
En la elaboración del prototipo se consideró aplicar Design Science Research (DSR), al ser de ámbito tecnológico permite obtener resultados innovadores dentro de proyectos de investigación (Brocke & Maedche, 2019) el requisito fundamental es disponer de suficiente literatura sobre el tema a investigar (Rossales Sengik y otros, 2022) dentro de los investigadores de Sistemas de Información apoya su equilibrio la relevancia y el rigor dentro de una investigación científica (Deng & Ji, 2018).

Una de las grandes ventajas es poder analizar de mejor forma el problema, objetivos de la solución, diseño del artefacto obteniendo una mejor solución enfocado a solventar el problema, mediante un diseño y desarrollo implementado obteniendo resultados mediante la evaluación de un prototipo funcional.

Las etapas de DSR son seis las cuales se detallan en la figura 4 colocando el orden y su respectivo resultado que se relacionan con los objetivos planteados para el desarrollo del proyecto

Figura 4.

Fases de la metodología Design Science Research



Nota. La figura 4 es una representación del cuadro de la metodología Design Science Research realizada por el autor (Mtsweni y otros, 2014) donde se detallan los pasos a seguir para el desarrollo de la metodología de investigación

Identificación del problema y motivación:

Dentro del presente proyecto se utilizó el estudio de la situación actual ecuatoriana sobre la filtración de datos personales de manera masiva dentro de los servicios gubernamentales, los ciudadanos buscan tener el control de sus propios datos sobre quienes pueden visualizar su información para no tener suplantaciones de identidad.

En la parte educativa la obtención de títulos falsos por entidades de renombre causa dificultades a los reclutadores generando desconfianza en los postulantes obtenido una tasa de desempleo dentro de estudiantes recién egresados al considerar los años de experiencia como factor determinante para conseguir el trabajo.

Definición de los objetivos de la solución

Se realizó la matriz de congruencia metodológica donde se describe los objetivos específicos en conjunto con sus preguntas de investigación (RQ) que ayuda a determinar la dirección que lleve al diseño y desarrollo de un prototipo funcional que busca solventar la problemática planteada delimitando el alcance que tendrá al momento de implementarse y evaluarse.

Diseño y desarrollo

Se aplica una metodología del desarrollo de modelos evolutivos de software, donde se plantea la arquitectura que se utiliza, los diagramas de flujo de la información con las ventajas de las herramientas tecnológicas elegidas sobre las demás obteniendo como resultado un prototipo funcional con una arquitectura descentralizada.

Demostración

El prototipo consta de dos etapas de desarrollo, pruebas locales donde se corrigen las fallas que se encuentran dentro de un entorno controlado con la mayor cantidad de privilegios al manejar la información y producción donde se gestiona la información de forma real, siendo un escenario con fallos que se desconocen.

Evaluación

En esta fase se evaluará la calidad de software obtenido en base a los tiempos de respuesta, la optimización de recursos, la facilidad de uso de los usuarios, mediante la prueba de un taller con estudiantes de los últimos niveles pertenecientes al Departamento de Ciencias de la Computación.

Comunicación

Una vez terminado la evaluación y con los resultados obtenidos se presenta las conclusiones dentro del área de investigación y desarrollo de software listando las siguientes ventajas al utilizar la metodología:

- Generar nuevas alternativas de arquitecturas descentralizadas.
- Implementar nuevos proyectos utilizando como base la arquitectura del prototipo
- Métodos rigurosos para evaluar prototipos descentralizados
- Continuar agregando más seguridad, diseño web, funcionalidades que aporten al crecimiento de plataformas con licencias libres.

Marco Teórico

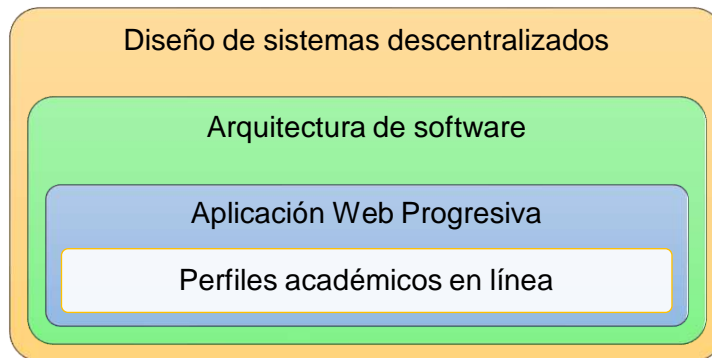
El objetivo del marco teórico tiene como propósito limitar el problema, obteniendo un resultado que se garantiza mediante el análisis del conocimiento sobre el material existente que guía los procesos de investigación.

Categorización de variables

Para desarrollar el marco teórico es necesario la red de categoría de la variable dependiente e independiente que sustentan la propuesta planteada, encontrando coherencia para el fundamento teórico del presente proyecto que se muestran en la figura 5 y 6.

Figura 5.

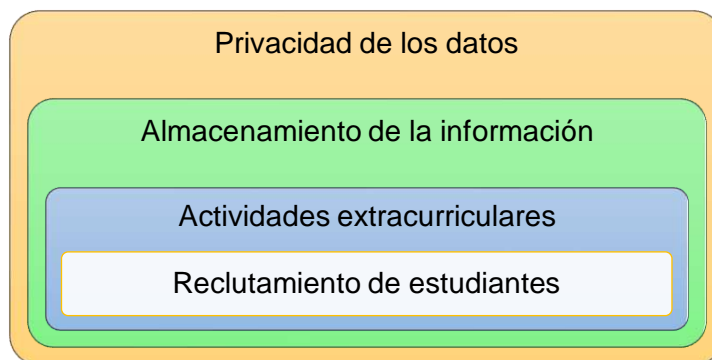
Categorías de la variable independiente.



Nota. Representa los contextos que se tratan dentro de la variable independiente, figura creada por el autor.

Figura 6.

Categorías de la variable dependiente.



Nota. Representa los contextos que se tratan dentro de la variable dependiente, figura creada por el autor.

Fundación científica de la variable independiente

Este apartado presenta información de diferentes autores presentado en artículos de investigación, donde cada uno de los temas siguientes pertenece a la red de categórica de la variable independiente

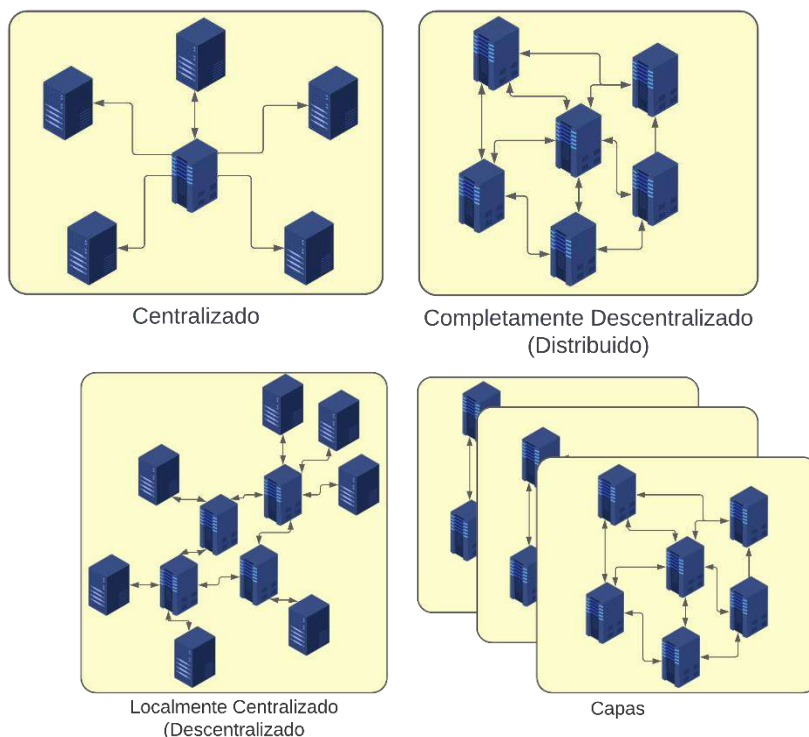
Diseño de sistemas descentralizados

Dentro del mundo digital para solventar la latencia y sobrecarga de hardware se utiliza el cómputo edge, fog y cloud. Con dispositivos perimetrales para nodos de enlace que se agregan y empaquetan los datos utilizando recursos de la nube, obteniendo de la forma óptima para los sistemas de información (Song y otros, 2018), se propone un diseño novedoso para los sistemas

de almacenamiento descentralizado preservando la privacidad basada en firmas y direcciones únicas (Kopp y otros, 2017) en la figura 7 se ve de forma gráfica la diferencia de los patrones para marcos de los distintos sistemas.

Figura 7.

Patrones de tendencia para marcos de sistemas.



Nota. La figura muestra los distintos recursos computacionales para manejar la persistencia de datos que aproveche arquitecturas en los servicios de la nube verificando un mayor rendimiento de la información, gráfico adaptado de los marcos Blockchain del artículo de (Song y otros, 2018).

Dentro del diseño de sistemas descentralizados utilizan seis etapas para la elaboración de protocolos que maneja la información, acceso y control de los usuarios descritos a continuación en la tabla 5 utilizando contratos inteligentes que son aplicados en dApp (Lyu y otros, 2019)

Tabla 5.

Etapas del protocolo de un sistema descentralizado.

Registro	Los usuarios establecen un par de claves pública/privada y envía de forma privada a un administrador.
Configuración	El administrador envía la lista de claves públicas a todos los nodos mediante un contrato inteligente, publicando la dirección del contrato.
Generación de claves	Con el contrato inteligente en una cadena de bloques se verifica la firma de cada nodo mediante la clave pública, si falla el contrato transmite haber encontrado un error.
Generación subsecreta	Recibir la verificación pública de los contratos inteligentes.
Carga secreta	Los contratos inteligentes reconstruyen la clave secreta usando los acuerdos con el sistema de cifrado del umbral.
Tally	El contrato inteligente descifra todos los datos mediante claves secretas y obtiene información para poder ser compartidos con los permisos correspondientes a usuarios o aplicaciones.

Nota. La tabla cuenta con 6 filas una por cada etapa su nombre y descripción basado en la información proporcionada por (Lyu y otros, 2019) dentro de los detalles del protocolo utilizado en su proyecto de investigación.

Dentro de un sistema de almacenamiento distribuido basado en recompensas financieras cuenta con propiedades privadas que manejan datos personales describiendo a continuación tres funciones avanzadas:

- **Usuario:** es un nodo el cual trabaja en conjunto con la red para acceder y controlar datos de un sistema. Un usuario crea contratos inteligentes para guardar información en forma de compromisos que se pagarán a un proveedor el cual conserva los datos en un periodo de tiempo se procesa una transacción para verificar la información al ser veraz transmite y replica el nuevo nodo generado a la blockchain sin importar ser pública, privada o federada.
- **Proveedor de almacenamiento:** es un nodo de la red que busca una retribución económica al proporcionar espacio de almacenamiento digital para los usuarios, pública las pruebas con el que puede recuperar la información de archivos demostrando haber cumplido con el contrato de almacenamiento.

- Minero: busca una recompensa realizando operaciones lógicas matemáticas dentro de los sistemas que validan las transacciones y contratos de almacenamiento donde garantizan haber agregado correctamente a la cadena de bloques, estos pueden ser compatibles con grandes empresas de monedas digitales como Ethereum y Bitcoin.

Los nodos pueden ser combinaciones de roles con gestión de tiempos, ayudando con el almacenamiento o incluyendo nuevos bloques dentro de la cadena de bloques (Kopp y otros, 2017)

Los contenedores de información es el lugar donde suelen almacenar la información descentralizada esto incluye tecnologías blockchain ya que al usar esta tecnología toda la información se descentraliza al momento de implementarse y dentro de estos proyectos existe una variedad de herramientas digitales con distintas aplicaciones y para poder lograr crear varios servidores estos proyectos se utiliza contenedores como Docker, orquestadores para entornos como Docker Swarm o Kubernetes (Song y otros, 2018).

Solid es un proyecto el cual permite a las personas almacenar sus datos de forma privada. Mediante el uso de POD que es un lugar para almacenar documentos con un mecanismo de control para acceder, el consumo se lo realiza mediante HTTP y LDP usado para intercambiar protocolos multiprotocolo intercambio de información de mapeo de etiquetas (W3C Solid Community Group., 2021).

Es de importancia destacar que dentro los diversos usos y aplicaciones del proyecto Solid (Social Linked Data) mediante un intercambio de datos de forma segura manteniendo la privacidad controlada por el usuario y dueño de los datos y cuyo objetivo es construir un ecosistema descentralizado que cambie el funcionamiento de las aplicaciones web. En el tema de solución que se plantea es la autenticación del sistema mediante accesos de confianza para que se mantenga un seguridad y control de acceso detallado mediante el uso de Blockchain y Solid donde se aumenta múltiples propiedad de seguridad, funciones de autenticación con toda

la documentación, de la implementación con las políticas que se van a manejar específicamente integración de firma RSA en cadenas de bloques habilitando un esquema de firmas distribuidas tolerante a fallas, fortifica el tema de seguridad.

Se recomienda que una alternativa es utilizar las firmas RSA dentro de la cadena de bloques en aplicaciones basadas en Solid en conjunto con contratos inteligentes para administrar políticas de control de acceso, donde se evaluó el desempeño siendo un resultado favorable contando como requerimiento transacciones rápidas y eficientes donde se busca a un futuro mejorar el rendimiento de blockchain en estas soluciones implementando un grupo de control para validar la exploración de firmas de umbral (Cai y otros, 2020).

Arquitectura de software

Se usa para referirse a los aspectos relacionados con la Tecnologías de la Información y de acuerdo con Software Engineering Institute (SEI) lo define como “las estructuras de un sistema, compuestas de elementos con propiedades visibles de forma externa y las relaciones que existen entre ellos” (Cervantes, 2010), el propósito es cumplir con los requisitos de software gestionando la seguridad de los datos bajo los campos de datos y el módulo que controla los datos (Kunifuji, 2013).

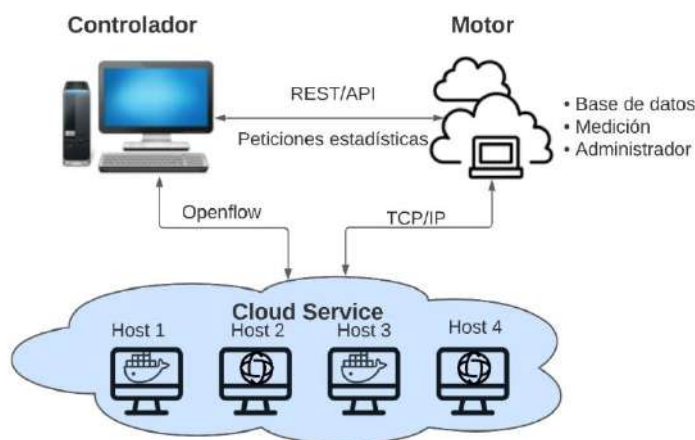
Un modelo de un sistema tiene configuración estos pueden ser directo o mediante contenedores que proporciona un contexto y tiempo de ejecución integrado y distribuido (Telschig y otros, 2018), es posible ejecutar múltiples aplicaciones en una misma máquina o servidor reuniendo todos los archivos y bibliotecas, para manejo de redes utiliza un host existente y una interfaz puente llamada Docker que maneja los contenedores con una red privada donde Docker proporciona tres redes: cerradas, puente y abierto (Shah & Dubaria, 2019).

Kubernetes es una arquitectura que cuenta con un nodo maestro el cual gestiona el clúster permitiendo la entrada de peticiones a realizar tareas administrativas que aloja el proceso de control disponible con los entornos de servicios (Shah & Dubaria, 2019), una arquitectura de

software descentralizado bien realizado debe poder contestar tres preguntas 1) ¿Cuál servicio de almacenamiento Cloud debe usarse? 2) ¿En qué momento debe ocurrir el despliegue de la aplicación? 3) ¿Qué archivos se deben subir al servicio?, en la figura 8 se puede observar como se establece un sistema descentralizado mediante contenedores (Tao y otros, 2019).

Figura 8.

Diseño y protocolos para el intercambio de información.



Nota. Gráfico de como los contenedores, servicios de la nube, motor y controlador se relacionan dentro de un entorno de aplicaciones descentralizadas representado en (Tao y otros, 2019).

Para lograr analizar las diferencias entre contenedores se debe conocer las ventajas sobre el uso Kubernetes al ser utilizado en implementación y monitoreo, uso de grandes clústeres, desarrollo con gestión de varios contenedores, tiempo de respuesta rápido y fiable, Docker se usa para crear contenedores virtuales, puede ser ejecutado y administrado de forma constante, puede usar una máquina virtual que ejecuta varios contenedores utilizando menos memoria y el contenedor es una parte responsable para la ejecución rápida de procesos de software la comparación de características se presenta en la siguiente tabla 6.

Tabla 6.

Docker vs Kubernetes

Características	Kubernetes	Docker
-----------------	------------	--------

Instalación y grupo de configuración	Instalación complicada con cluster muy fuerte.	Instalación simple con cluster débil.
Interfaz gráfica de usuario	Tablero GUI.	No hay GUI.
Escalabilidad	Altamente escalable y rápido.	Altamente escalable y 5 veces más rápido que Kubernetes.
Escalado automático	Permite auto escalar.	Docker Swarm no permite auto escalar
Balanceo de carga	Intervención manual para equilibrar carga del tráfico entre diferentes contenedores.	Equilibra el tráfico de la carga de forma automática entre contenedores.
Actualizaciones continuas y reversiones	Implementar actualizaciones y reversiones continuos.	Implementar actualizaciones, pero no reversiones.
Volúmenes de datos	Compartir volúmenes con otros contenedores del mismo usuario.	Compartir volúmenes con cualquier contenedor.
Inicio de sesión & vigilancia	Herramienta para registro y monitoreo.	Uso de terceros como ELK para registro y supervisión.

Nota. En la tabla comparativa se muestra la característica y como cumple o no los contenedores Docker o Kubernetes basado en el artículo (*Shah & Dubaria, 2019*)

Aplicaciones Web Progresivas

Una Aplicación Web Progresiva (PWA) es una aplicación desarrollada con tecnología web como HTML5, CSS3 y un lenguaje de programación usualmente JavaScript (Malavolta y otros, 2017) para integrar PWA existen distintas maneras dentro de aplicaciones nativas con ventajas y desventajas, donde las principales características son funcionar fuera de línea, notificaciones

push, acceso directo desde pantallas de inicio, navegación precargada, almacenamiento persistente, gestión de credenciales entre otras (Steiner, 2018)

Para implementar PWA se puede utilizar diferentes framework del mercado por ejemplo React que es desarrollado y mantenido por la empresa Facebook desde el 2015 tiene soporte para dispositivos iOS y Android. Otro framework es Ionic desarrollado y con soporte de la empresa Google, proporcionando un módulo HTTP utilizado para la obtención de información remota mediante la inyección de dependencias mediante Angular. La comparación de características, tecnologías utilizadas se lo detalla en la tabla 7 (Majchrzak y otros, 2017).

Tabla 7.

Características React y Ionic

	REACT	IONIC
Paradigma	Interpretado	Híbrido
Versión de JavaScript	ES2015	ES2015
Motor de vista	JSX	Angular.js
Acceso a Cámara	Si	Si
Acceso a galería de imágenes	Si	Si
Implementación de navegación	Intermedia	Sencilla
Obtención de datos remotos	Sencillo	Sencillo
Depuración	Sencilla	Sencilla
Instalación	Sencilla	Sencilla

Nota. Características técnicas, nivel de dificultad y desarrollo para manejar un framework que desarrolla aplicaciones PWA en base al resumen informativo de (Majchrzak y otros, 2017)

Las aplicaciones PWA busca mejorar en las siguientes 4 perspectivas dentro de los dispositivos móviles:

- **Conversiones:** busca tener una estrategia para mejorar siempre que el navegador se lo permita partiendo desde un simple HTML estático avanzando a usar animaciones y contar con un acceso a la red de forma asíncrona.

- **Fiabilidad:** puede cargar en un instante, con baja o nula conexión a la red, al almacenar el caché mediante archivos precargados eliminando la dependencia de red.
- **Rendimiento:** se puede usar procesamiento en segundo plano garantizando una experiencia instantánea y confiable dentro de los usuarios en conjunto con el almacenamiento en caché.
- **Compromiso:** se puede instalar un acceso en dispositivos de escritorio o como aplicaciones móviles que acceden directamente desde su pantalla de inicio, usar notificaciones desde la nube para involucrar a los usuarios con una mejor experiencia hasta con el modo pantalla completa.

Una aplicación PWA se puede ejecutar desde un servidor remoto, accediendo a la aplicación web desde un navegador donde el usuario puede decidir si instalar la aplicación basado en un carácter técnico se debe cumplir con tres aspectos 1) acceder a través de un HTTPS, 2) cuenta con un archivo llamado manifest donde declara los metadatos como nombre de la aplicación, iconos y tamaños de cuándo usarlos y 3) contar con un trabajador de servicios o en inglés (service worker) que es un conjunto de API usado por desarrolladores para almacenar caché, precargar datos, realizar notificaciones push entre otras cosas (Malavolta y otros, 2017)

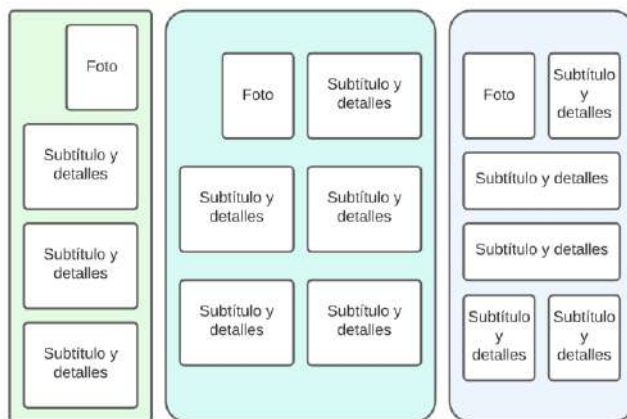
Perfiles académicos en línea

Un estudiante o profesional cuenta con un currículum vitae (CV) donde destaca la trayectoria académica conteniendo información escrita que indica calificaciones y logros a empleadores o reclutadores (Teixeira da Silva y otros, 2020) realiza una selección inicial para contratar a trabajadores usando el CV como una autopresentación (Wijayanti & Sulistyaningsih, 2018) para seleccionar a los candidatos más adecuados pasando a entrevistarlos y contratarlos lo que puede afectar un desempeño laboral de forma general en la empresa (Gajanayake y otros, 2020).

Existen tres tipos de formatos de CV en países en vías de desarrollo al no tener un estándar de cuantas columnas debe tener, en cambio lo que uno busca es la facilidad de lectura, contener información relevante y un lenguaje formal de forma gráfica se representa los tres tipos en la siguiente figura 9.

Figura 9.

Formato CV por columnas



Nota. Basado en el estudio recolectado de 300 CV el 96% usan un formato cronológico y un 61% a dos columnas y un 87% en una sola página, la adaptación del gráfico y los datos se obtuvieron de (Wijayanti & Sulistyaningsih, 2018)

Los reclutadores al evaluar a un candidato suelen usar un modelo de personalidad llamado Big Five o conocido como el modelo OCEAN usado para predecir los rasgos descritos a continuación:

- **Apertura (Curiosidad, imaginativo, inteligente):** al obtener altos puntajes será una persona creativa con gustos avanzados, reconociendo un sinnúmero de pensamientos, perspectivas y encuentros.
- **Escrupulosidad (Perseverante, responsable, organizado):** personas confiables, que realizan diligencias organizando a los empleados.
- **Extraversión (Amistosos, asertivos y extrovertida):** gente vivaz, cordial y que motivan de forma social al resto de empleados

- **Amabilidad (Cariñoso, cooperativo, servicial):** los puntajes altos tienden a ser armoniosos, demostrando esperanza y confianza en los demás.
- **Neuroticismo (Ansioso, sensible, inseguro):** personas irritables con tensión y posiblemente con sentimientos contradictorios.

Los rasgos descritos pueden ser localizados a partir de una llamada telefónica, entrevista o video llamada utilizando analítica de audios el documento de texto puede utilizar técnicas avanzadas para lograr un aprendizaje automático e identificar la personalidad con cuatro preguntas 1) ¿Qué experiencia posee para desempeñar, esta posición?, 2) ¿Por qué piensa dejar su organización actual? 3) ¿Cuáles son los retos que busca para este cargo en su organización? 4) ¿Cuál es su vida profesional en los siguientes 5 años? (Gajanayake y otros, 2020)

Fundación científica de la variable independiente

En la presente sección se utiliza información de estudios científicos con temas correspondientes a la red categórica de la variable independiente

Privacidad de los datos

Las empresas y los gobiernos se sumergen en la información personal de los datos de los usuarios de formas cada vez más innovadoras, especialmente rastreando qué son, qué hacen, con quién interactúan y adónde van. La explotación comercial de datos personales ha derrotado en gran medida la filosofía de trabajo de www. Al proponer el proyecto Solid, Sir Tim Berners-Lee ha iniciado una acción revolucionaria para devolver al usuario el control de sus propios datos. Como el sistema está en fase de desarrollo, lo realiza con servidor Debian 9 comandos NodeJs usando angular generator (Solanki, 2020)

La manipulación y el intercambio de la desinformación del usuario sin el consentimiento del mismo, donde se analiza las distintas soluciones que se aplican al tema del almacenamiento distribuido con tecnología Blockchain haciendo énfasis en el uso de Solid para lograr una mayor

control de los datos usando varias configuraciones de soluciones, demostrando que la verificación de datos sin pérdida del control de usuarios o terceros dentro de un entorno descentralizado con una variedad de dominios.

El manejo de los datos divididos en dos confidenciales con verificación y aquellos que se puede acceder de forma pública dentro de aplicaciones móviles y web los cuales pueden ser beneficiosas de la construcción de aplicaciones distribuidas (Ramachandran y otros, 2020).

Se habla sobre temas como MultiChain es una plataforma blockchain específica que ofrece "flujos de datos" integrados en la cadena para un almacenamiento de datos rápido y seguro se diseñó un protocolo de almacenamiento aprovechando las claves en los flujos de datos de MultiChain y creando un marco de datos de la cadena que permite una consulta eficiente. La solución se planteó para una competencia enfocado en la Integración de Datos para Análisis, Anonimización y uso Compartido (iDASH) ganando un taller realizado en San Diego, CA en octubre de 2018. Aunque siendo la ganadora tiene presente algunos inconvenientes como descargar todos los datos de la cadena y guardarlos localmente en la memoria para una consulta rápida. Para solucionar este problema, se proporciona una alternativa de "bigmem" que utiliza índices en lugar de almacenamiento local para consultas rápidas.

Al final los resultados de la investigación obtienen archivos de registro de acceso genómico donde se pueden almacenar y consultar de manera eficiente con blockchain. Un protocolo que podría aplicarse a otros tipos de datos de salud, como los registros de salud electrónicos. (Gürsoy y otros, 2018).

Ventajas en la nube como acceder a los datos del usuario interactuando con el mecanismo de autenticación. Se busca proporcionar un mecanismo de autenticación seguro mediante el uso de la tecnología Blockchain para bases de datos en la nube. Al valorar las ventajas que tiene Blockchain cambiando los detalles de las credenciales de inicio de sesión del usuario en el proceso de autenticación del usuario por parte de una persona con información privilegiada. Dentro de esto destaca que la persona con información privilegiada no puede

acceder a los datos de autenticación del usuario debido al esquema de autenticación distribuido. La actividad de la información privilegiada es algo rastreable y que no se puede alterar mientras que los usuarios internos como los externos se autentican mediante identificaciones y firmas individuales (Deep y otros, 2019).

Algunos protocolos de consenso utilizados en aplicaciones blockchain no están diseñados para realizar trabajos útiles para la sociedad debido a demasiada competencia y limitación de escalabilidad. Se han desperdiciado enormes recursos informáticos y de energía eléctrica, incluidos CPU, RAM, almacenamiento y sensores para ejecutar la red blockchain basada en estos protocolos de consenso. Los marcos y middleware actuales para la creación de Aplicaciones Descentralizadas (dApps) se limitan en gran medida a trabajos simples y menos útiles. Los marcos de blockchain actuales, como lo es Ethereum, son difíciles de admitir la diversidad de aplicaciones descentralizadas debido principalmente a la realización redundante de "hash". Dentro del trabajo de investigación se presenta Proofware, que está diseñado para que los desarrolladores creen sus dApps fácilmente con los recursos informáticos existentes públicos / basados en la multitud. Los desarrolladores pueden desarrollar y probar sus propios protocolos de consenso de Prueba de Trabajo Útil (PoUW). (Dong y otros, 2019).

Almacenamiento de la información

Los currículums vitae, hojas de vida o currículum se han convertido en instrumentos valiosos para obtener indicadores de medición del conocimiento resultantes de las diferentes actividades científicas, ya que permiten describir el comportamiento interdisciplinario del conocimiento y su influencia en el desarrollo de la ciencia en la institución o región donde se aplique esta herramienta. En la actualidad, deslumbran experiencias de integración regional e interinstitucional que marcan pautas a seguir para la elaboración de indicadores normalizados, así como el establecimiento de formas de gestionar la ciencia en instituciones universitarias desde una perspectiva integrada en sus procesos académicos, investigativos y extensionistas

vinculados con la región que la rodea. La presente revisión profundiza en estas cuestiones con el objetivo de analizar las experiencias que existen en Iberoamérica en este tipo de plataforma y su utilización como herramienta en la gestión de la ciencia y la tecnología a nivel institucional y regional. Mediante el análisis documental se logró mostrar las características de estos sistemas favorables a la medición y a la gestión de la ciencia a nivel institucional y regional. Estas herramientas seguirán siendo verdaderos instrumentos integradores que valoran el comportamiento de la ciencia a nivel individual-institucional-regional y poseen la bondad de integrar los procesos universitarios tradicionales: academia investigación extensión (Rivero Amador y otros, 2020).

Actividades extracurriculares

Las habilidades extracurriculares o actividades de segundo plano son actividades que permite a los estudiantes desarrollar competencias genéricas las cuales se define como habilidades transferibles, que se puede lograr a través del aprendizaje aplicado en temas de estudio, trabajo o la vida diaria de cada uno (Nghia, 2017) dentro del impacto de las habilidades extracurriculares en temas de empleabilidad se ha determinado algunas actividades como lo son comunicación oral, motivación, resolución de problemas, trabajo en equipo, confianza en sí mismo, independencia, adaptación y flexibilidad, disponibilidad de aprender cosas nuevas, planificación, creatividad e innovación, liderazgo, iniciativa y trabajar bajo presión teniendo la finalidad de demostrar que los estudiantes que fueron miembros de actividades extracurriculares como grupos sociales, trabajo de voluntario, arte y música y deporte, demostrando que tienen más probabilidades ser evaluados positivamente en un empleo gracias a sus habilidades blandas que se han desarrollado (Mefteh, 2021).

Un estudio sobre la relevancia de las actividades Co-curriculares de los graduados buscando evidenciar el desempleo con una muestra de 500 graduados del año 2010 empleando un modelo de riesgo proporcional de Cox para estimar la duración del desempleo encontrando

que es insignificante en los graduados (Shahiri & Adnan, 2015) en marzo 2021 en Tunisia se analizó el impacto de actividades extracurriculares de graduados llegando a la conclusión que la relación entre las actividades blandas que se detallan tiene un impacto positivo dentro de los temas de empleo detallando que las actividades con mayor impacto se llegan a agregar a las mallas curriculares de los estudiantes convirtiéndose en una actividad académica dentro de sus estudios llegando a la conclusión que no importa si las actividades extracurriculares que se realizan son con fines económicos, relacionados con el título que se está obteniendo estos van a ser fructíferos en el tema del desarrollo de actividades blandas desde tener que planificar un horario de clases que no interfiera con las reuniones o actividades que con lleva el voluntariado o el horario del trabajo donde los encuestadores en el año 2021 están tomando muy en cuenta como un punto fundamental la organización del tiempo de cada empleado (Mefteh, 2021).

Reclutamiento de estudiantes

Si bien existe un beneficio detrás de la educación y formación de cada uno de los estudiantes pero directa o indirectamente estos forman parte de una sociedad donde la van a beneficiar tarde o temprano si esto no se lo realiza pone en riesgo al aprendizaje de servicios convirtiéndose en servicios poco éticos al tener relacionado los entornos con estándares éticos y desarrollo civil es por todo eso que al aplicar un voluntariado donde se aplique actividades de entretenimiento como lo puede ser la música, danza, deportes, la hermandad y fraternidad permite obtener resultados más amplios dentro de una comunidad (Polmear y otros, 2020).

Los graduados que participan en actividades extracurriculares dentro de grupos sociales, trabajos de voluntarios y deporte se tiene mayor probabilidad de evaluar positivamente habilidades de empleabilidad como ejemplo la participación de grupos sociales donde se desarrolla habilidades que fomenta la responsabilidad individual, manejo de personas, autoevaluación e iniciativa, si esta consta de una vinculación con la sociedad puede complementarse con actividades de arte y música donde se desarrolla una comunicación oral,

resolviendo problemas que se adaptan y son flexibles dependiendo con el grupo que se trabaja descubriendo el líder que cada uno lleva en su interior esto puede resumirse en mayor participación de actividades voluntarias se desarrolla el trabajo en equipo, flexibilidad e iniciativa, independencia y adaptabilidad dentro del ámbito laboral y social de cada persona (Mefteh, 2021).

Capítulo III

Análisis y diseño de la arquitectura descentralizada para el desarrollo de e-profile

Se detalla los pasos para montar un prototipo descentralizado de nombre Pancka (Hoja de papel en Kichwa) Ruray(Trabajo en Kichwa) junto Pancka Ruray (Hoja para buscar trabajo), analizando los actores involucrados junto con la interacción de los servicios de verificación de datos correspondiente e implementados.

Introducción

Se analiza y diseña el sistema Pancka Ruray en base a la metodología de desarrollo de software modelo de prototipos detallando los actores que intervienen dentro del sistema para la construcción de e-profile, las herramientas digitales que se utilizan para construir la descentralización de la información, la interacción con los distintos servicios para almacenamiento de datos, los diagramas UML y tablas de resúmenes.

Análisis

En el análisis se describen los actores que trabajan en el sistema para la construcción de perfiles en línea, la integración de los sistemas descentralizados y sus aplicaciones, que permite el acceso seguro, confiable y controlado de los datos a terceros mediante aplicaciones, cuentas de usuario donde tendremos limitaciones de uso entre los usuarios o grupos

Actores del sistema para construcción de e-profile

El manejo de currículum vitae (CV), resume CV o perfiles en línea donde se pública información personal cuenta con varios actores al momento de ser utilizado para conseguir un empleo previo a ser utilizado, el proceso de construcción de uno CV por defecto es recopilación de información distribuida al no contar con una educación formal única dentro de una misma escuela, colegio, universidad, cursos de actualización, experiencia. Siendo la razón de que la información no se debe quedar dentro de esta institución porque genera un almacenamiento de

datos masivo el cual puede causar una demanda excesiva de solicitudes para la entrega del título certificado mientras que, si este fuera propiedad del dueño, los tiempos de trámites no existieran para temas de solicitud ya que uno mismo manejar la información. Al descentralizar los datos y analizar como garantizar la veracidad de estos se debe estudiar quienes son los actores que intervienen en la creación de certificados académico y extracurriculares.

Para el diseño de un perfil en línea contamos con la intervención activa de los siguientes actores que desarrollan la información profesional:

- **Estudiante:** actor que es parte de un proceso universitario que agrega información al CV y busca su primer empleo. Previamente ha desarrollado una o varias habilidades duras o blandas con la participación de actividades académicas o extracurriculares como cursos en línea, concursos, voluntariado, grupos o clubes relacionados con su desarrollo personal o profesional.
- **Institución de Educación Superior:** actor que autoriza el ingreso de estudiantes y docentes dentro del aula de clases o proyectos que beneficien a ambos, esta permite validar que los estudiantes cumplan con los requisitos para obtener un título universitario, certificado de participación que agrega experiencia al estudiante proporcionando un documento que es inalterable y verificable dentro de las actividades académicas.
- **Docente:** actor que brinda clases en una Institución de Educación Superior, incentiva aprender cosas nuevas, valida el área del conocimiento adquiridos por estudiante durante un periodo académico, dentro o fuera de clase mediante la participación mutua de proyectos de voluntariado o prácticas profesionales
- **Empresa:** actor que permite el ingreso de estudiante a su primer trabajo o sus prácticas laborales, el que puede ser mediante la institución de educación

superior o postulando mediante el uso del perfil en línea, donde se agregara la información de haber participado correctamente.

- **Reclutador:** actor el cual entrevista y recibe la información de estudiantes y profesionales determinando si es o no apto para desempeñar la vacante dentro de la empresa.
- **Supervisor:** actor que mide el progreso laboral y los resultados obtenidos de los nuevos estudiantes contratados mediante el uso de perfiles en línea agregando el desarrollo y mejora que tiene el empleado al desempeñar nuevas tareas dentro de la empresa.

En base lo descrito anteriormente se puede realizar un cuadro con los atributos y funciones que desempeña cada actor dentro del sistema para construcción de e-profile en la tabla 8

Tabla 8.

Actores para la construcción de e-profile.

Actor	Atributos	Funciones
Estudiante	<ul style="list-style-type: none"> • Identificador del perfil • Información personal, académica y extracurricular • Acceso de Instituciones de Educación Superior • Acceso de empresas 	<ul style="list-style-type: none"> • Maneja los permisos para agregar, acceder y visualizar la información • Añadir información de datos personales públicos
Institución de Educación Superior	<ul style="list-style-type: none"> • Identificador del grupo • Información pública 	<ul style="list-style-type: none"> • Ingreso de actividades académicas y extracurricular a estudiantes • Ingreso de docentes verificadores • Añadir información pública de la Institución de educación superior
Docente	<ul style="list-style-type: none"> • Identificador del perfil 	<ul style="list-style-type: none"> • Validación la participación y mejora de estudiantes

	<ul style="list-style-type: none"> • Información, personal y académica 	<ul style="list-style-type: none"> • Agregar información para identificar al docente
Empresa	<ul style="list-style-type: none"> • Identificador del grupo • Información pública 	<ul style="list-style-type: none"> • Ingreso de supervisor verificador • Ingreso de cargo desempeñado • Añadir información pública de la empresa
Reclutador		<ul style="list-style-type: none"> • Revisar perfiles de estudiantes
Supervisor	<ul style="list-style-type: none"> • Identificador del perfil • Información personal 	<ul style="list-style-type: none"> • Validación del desenvolvimiento del estudiante • Agregar información para identificar al supervisor

Nota. Tabla que representa los atributos que contiene cada actor y las funciones que desempeña dentro de la creación de e-profile creado por el autor.

Como se clasifican las habilidades socioemocionales (habilidades blandas)

Dentro del marco teórico se contó con un claro representante para manejar las habilidades socioemocionales siendo el más usado Big Five pero este no es respaldado por ninguna institución es por eso que para lograr abarcar un mayor alcance internacional se representa las habilidades en la siguiente tabla 9.

Tabla 9.

Clasificación de métodos para clasificar habilidades socioemocionales

Clasificación	Institución / Organización	Dimensión	Habilidades
Big Five		Colaboración	Confiable, colaborativo y empático.
		Responsabilidad	Con autocontrol, responsable, persistente y orientado a lograrlo.
		Apertura a la experiencia	Curioso, creativo y tolerante.

		Estabilidad Emocional	Controla sus emociones, es optimista y resiste el estrés.
		Extraversión	Asertivo, con energía y sociable.
Aprendizaje socioemocional	Collaborative for Academic Social, and Emotional Learning (CASEL)	Toma de decisiones	Identifica el problema y lo evalúa de forma ética y responsable.
		Autoconciencia	Identifica la confianza, eficiencia y percepción de uno mismo.
		Conciencia del entorno social	Empático, perspective que aprecia la diversidad con respeto a los demás.
		Relaciones sociales	Trabaja en equipo para resolver conflictos mediante la comunicación y el compromiso social
		Autorregulación	Controla el estrés y los impulsa automotivado hasta realizar los logros que se plantea.
Habilidades para el siglo XXI	National Research Council	Interpersonal	Colabora trabajando en equipo para resolver conflictos mediante el liderazgo
		Cognitivo	Resuelve problemas mediante el pensamiento crítico de forma creativa e innovativa
		Intrapersonal	Adaptable, responsable y curioso con una

			estabilidad emocional autorregulada
Habilidades socioemocionales	Organización para la Cooperación y el Desarrollo Económico (OCDE)	Trabajo con otros	Una persona social, respetuosa
		Manejo de emociones	Optimista y con autoconfianza
		Logro de metas	Tiene auto control perseverando hasta cumplir sus metas

Nota. En la tabla sin importar como se clasifica la habilidad socioemocional las habilidades que se registran representan parámetros similares es por ello que en el sistema se registrará de forma independiente e individual las habilidades tanto duras como blandas para un análisis de perfiles, tabla adaptada de (Arias Ortiz & Hincapié, 2020)

Planteamiento de historia de usuario

Teniendo claro los actores con la interacción y generalizado el contenido para manejar las habilidades que serán registradas dentro de los e-profile sin importar la clasificación entre duras o socioemocionales, se plantea los requerimientos funcionales que son implementados en el sistema. Usaremos la técnica de Historia de Usuario

1) Registro de datos públicos de estudiantes

Código 001	Título Registro de datos públicos de estudiantes	
Descripción	Como estudiante me gustaría que mi información pública pueda ser vista por la mayor cantidad de personas para obtener un empleo.	
Usuario	Estudiante	
Datos de entrada	<ul style="list-style-type: none"> • Nombre Completo • Rol • Empresa • Correo electrónico 	<ul style="list-style-type: none"> • Número telefónico • Fecha de nacimiento • Descripción
Datos de salida	<ul style="list-style-type: none"> • Uri del POD del usuario • Carta de presentación 	
Observaciones	Esta información la puede ver cualquier aplicativo es pública, pero inalterable solo el dueño del POD puede cambiarla.	

2) Agregar permisos para acceso seguro, confiable y controlado a la información del e-profile

Código 002	Título Agregar permisos para acceso seguro, confiable y controlado a la información del e-profile
Descripción	Como estudiante que participa en actividades dentro y fuera de la universidad que genera cualidades en mi CV busco que reconozcan, registren y validen mis actividades académicas y extracurriculares integradas en mi CV.
Usuario	Estudiante
Datos de entrada	<ul style="list-style-type: none"> • Usuario con un POD • Archivo ttl dentro de carpeta PanckaRurya
Datos de salida	<ul style="list-style-type: none"> • Archivo Profile Data con permiso de Posters
Observaciones	Los permisos extra que se conceda para manejar la información del sistema pueden causar problemas al momento de verificar la información

3) Agregar información dentro del e-profile

Código 003	Título Agregar información dentro del e-profile
Descripción	El docente puede añadir información dentro de los estudiantes al estar añadido dentro del grupo de universidad y este a su vez como Poster en Data Profile del estudiante
Usuario	Docente
Datos de entrada	<ul style="list-style-type: none"> • Identificador docente y del e-profile del estudiante. • Habilidades duras y socioemocionales con una entrada numérica de base porcentual.
Datos de salida	<ul style="list-style-type: none"> • Habilidad dura o socioemocional validada por el docente dentro del perfil del estudiante
Observaciones	Debido a la gran variedad de habilidades socio emocionales y al desconocimiento de los modelos este campo es abierto y gracias al conocimiento de que docente agrega la información se puede inculcar una capacitación relámpago.

4) Compartir el e-profile con terceros

Código 004	Título Compartir el e-profile con terceros
Descripción	Como estudiante puedo ir a charlas, capacitaciones, concursos o empleos en distintas empresas o universidades a más de las que pertenezco necesito tener la posibilidad de integrarse a los sistemas.
Usuario	Estudiante
Datos de entrada	<ul style="list-style-type: none"> • La URL personalizable basado en niveles de permisos donde se almacena la información POD
Datos de salida	<ul style="list-style-type: none"> • Integración con sistemas de terceros con un nivel de permisos personalizable • Agregar información de archivos o nuevas carpetas de sistemas con otro fin distinto al académico.
Observaciones	El sistema no se puede hacer responsable por generar cambios erróneos dentro de los datos del usuario por exceso de permisos

5) Revisar las actividades extracurriculares o académicas pendientes

Código 005	Título Revisar las actividades extracurriculares o académicas pendientes
Descripción	Como docente no tengo el tiempo para agregar de cero todas las habilidades, pero si puedo comenzar con que el proyecto desarrolla X habilidades duras y socioemocionales en Y porcentaje agregando la validación estándar a los estudiantes que cumplen con el proyecto
Usuario	Docente
Datos de entrada	<ul style="list-style-type: none"> • Identificador docente y del e-profile del estudiante. • Validar las habilidades agregadas por los estudiantes.
Datos de salida	<ul style="list-style-type: none"> • Cambio del estado de una habilidad dura o socioemocional agregado por un estudiante validando o descartando la veracidad de esta.
Observaciones	N/A

6) Utilizar información pública en otros sistemas descentralizados compatibles

Código 006	Título Utilizar información pública en otros sistemas descentralizados compatibles
Descripción	Como estudiante estoy cansado de replicar la información y si se puede llamar para agregar un sistema descentralizado compatible con el almacenamiento que se utiliza ahorraría mucho tiempo y almacenamiento de terceros.
Usuario	Estudiante
Datos de entrada	<ul style="list-style-type: none"> • Identificador de e-profile del estudiante • Sistema con ingreso de sesión mediante POD de Solid Server
Datos de salida	<ul style="list-style-type: none"> • Datos públicos que cualquier persona solo puede ver sin alterarlos
Observaciones	N/A

7) Acceder al e-profile público para reclutar estudiantes

Código 007	Título Acceder al e-profile público para reclutar estudiantes
Descripción	Como reclutador tener un sistema que respalda la información académica y extracurricular es de mucha ayuda al no tener que investigar para corroborar la información colocada.
Usuario	Reclutador
Datos de entrada	<ul style="list-style-type: none"> • Identificador del e-profile del estudiante.
Datos de salida	<ul style="list-style-type: none"> • Información académica y extracurricular de estudiantes en formato digital e-profile con estructura de CV.
Observaciones	N/A

Diseño

En esta sección se plantea la arquitectura y diseño del sistema para manejar e-profile, donde se controle el acceso a la información, explicando el uso que se da a cada componente dentro del prototipo con la configuración que se utiliza

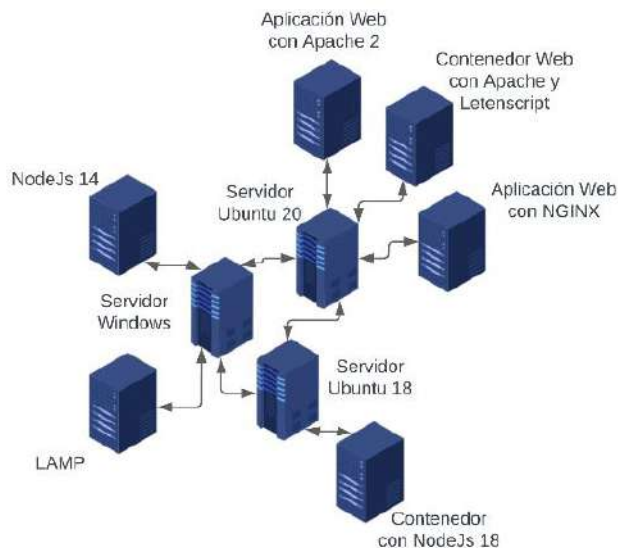
Razón para usar un sistema descentralizado

Para lograr cumplir con los requisitos planteados debe recopilar gran cantidad de información de distintos sistemas académicos, presentando la dificultad al integrar aplicaciones que consumen datos en servidores centralizados los cuales cuentan con la información, pero al ser los responsables directos de fugas en los datos la posibilidad de obtener permiso a las bases de almacenamiento de una forma rápida y con grandes privilegios para actualizar y agregar nueva información es casi imposible.

Al proponer usar un sistema centralizado para la posible solución del problema esta arquitectura demandaría gran cantidad de recursos, donde se almacene la cantidad de información verificada y consensuada similar a los sistemas gubernamentales centralizados que respalda los datos en casos de auditorías informáticas y por último el acceso confiable y seguro del prototipo Pancka Ruray puede ser vulnerado al ser atractivo para los atacantes robando o manipulando la información. A continuación, en la figura 10 se coloca de forma gráfica las tecnologías usadas en un sistema descentralizado donde un atacante no se enfrentaría a una sola tecnología sino a varias con versiones diferentes dificultando el robo y acceso a los datos.

Figura 10.

Arquitectura descentralizada con características de software



Nota. Arquitectura descentralizada para una aplicación web con servidores que manejan el almacenamiento y protocolos de mensajería, elaborado por el autor.

En la figura 10 se cuenta con 6 aplicaciones descentralizadas las cuales pueden utilizar 3 tipos de servidores para el almacenamiento de información de los que varía el sistema operativo y su versión. Es una red privada que los atacantes no pueden acceder, se dificulta saber a qué servidor realiza una petición y cuál es la vulnerabilidad respectiva del mismo. Las aplicaciones pueden ejecutarse dentro de contenedores el cual puede manejar una versión diferente, con cualidades únicas que el atacante al realizar una petición puede enviarse a cualquier servidor cliente y si encuentra el canal directo para alterar la información suele manejar la información entre servidores que respalda la información dejando una inconsistencia por haber sufrido un cambio inesperado a diferente de la arquitectura centralizada donde el atacante manipula la fuente de la información pasando por desapercibidos los cambios realizados.

Los beneficios de usar sistemas descentralizados manejando contenedores dentro de aplicaciones de varios servidores es beneficioso para el tráfico, la velocidad de carga y almacenamiento. Actualizar cambios de seguridad, temas de nuevas interfaces visuales hasta agregar un cambio simple como otro nombre puede ser un verdadero dolor de cabeza el tema de versiones se solventa de forma automática al realizar actualizaciones mediante el uso de

repositorios Git o a su vez lograr subir el contenedor a los repositorios Docker HUB donde se guarda un entorno de producción de forma instantánea.

Se debe usar un contenedor, cuando el sistema desarrollado va a ser distribuido en gran medida como lo son las aplicaciones descentralizadas, o se busca tener una portabilidad de ejecución al compartir varios sistemas, donde se comprime todo el entorno de ejecución como los puertos de la red, consumo de servicios web, bases de datos de una aplicación sin que interfiera con el entorno de ejecución de la máquina anfitriona que usa frameworks y librerías con versiones diferentes y no son soportan para el proyecto migrado tomando tiempo volver a ponerlo en ejecución. Para la migración de proveedores de Cloud es un dolor de cabeza cambiar el sistema donde se solicita cierto software con versiones específicas de máquinas personalizables implicando un tiempo extra para estar listo los contenedores solo instalamos la aplicación Docker buscamos en los contenedores el nuestro y lo instalamos y ejecutamos sin problema.

Una vez que analizamos por qué no usar un sistema centralizado, dentro del desarrollo de aplicaciones descentralizadas muchas se centran en el uso de blockchain lo cual abre un abanico de posibilidades como los mostrados dentro del análisis de literatura donde cumple con el almacenamiento de datos de distribuida, transparencia con transferencia, validación de la información que es una de la parte del sistema Pancka Rurya más no el manejo de acceso a los datos con permisos de forma gratuita sin uso de plataformas de criptomonedas, es por ello que se buscó proyectos descentralizados que almacenan datos llegando a encontrar el Proyecto Solid creado por el fundador de la web Tim Berners-Lee que busca que cada persona pueda ser propietario de sus datos con la mayor privacidad que se pueda tener dentro de la web y las distintas aplicaciones.

SOLID server proyecto de almacenamiento descentralizado

Solid es una derivación de "Social Linked Data" tiene una cantidad de herramientas para crear aplicaciones descentralizadas con el principio de manejar datos vinculados esto se refiere

a poder almacenar los datos dentro de uno o varios servidores los cuales pueden ser consultados por cualquier aplicación y poder crear, actualizar, eliminar y consultar datos públicos, privados con los permisos necesario, estos son datos y manejados por cada propietario de los datos.

Para poder modular y seguir creciendo la información de los datos se basa en los estándares y protocolos W3C que existen para el manejo de etiquetas por ejemplo para datos personales o en el caso de estudio para currículos vitae utiliza Resource Description Framework (RDF) y para el esquema de que nomas contiene un CV <http://rdfs.org/resume-rdf/cv.rdfs>.

La compañía Inrupt el más grande proveedor de almacenamiento del proyecto Solid seguido de la propia comunidad de Solid y si uno desea puede establecer su propio servidor en el caso de desconfiar en los que se dispone. Una de las mayores ventajas es tener una licencia GLP que permite manipular el código fuente hacer cambios y personalizando el servidor.

El servidor cuenta con distintas formas de configurarlo, teniendo mayor o menor control de la información tanto lógica dentro del servidor, donde almacenar, los protocolos que se debe seguir al consultar y utilizar la información, se tiene requisitos para tener un servidor Solid en producción como el nombre del dominio, certificado de Seguridad de la Capa de Transporte SSL, manejar subdominios y sufijos para los usuario si no se cumplen las aplicaciones pueden tomar la decisión de utilizar o no ese servidor como almacenamiento. Por eso la mayor ventaja es tratar de cumplir con todas las formas de seguridad para obtener la mayor ventaja de usar servidores Solid descentralizados

Para poder montar un servidor no se cuenta con un hardware mínimo este puede ser de forma local como remota, el sistema operativo no es un impedimento usar distribuciones de Linux, Windows o Mac OS, puede ser una máquina virtual como un contenedor, por todas estas cualidades flexibles permite poder lograr un sistema descentralizado con gran cantidad de acceso dentro de aplicaciones desarrolladas, con una múltiple cantidad de propósitos. Para las aplicaciones cliente cuenta con una comunidad que expande y comparte sus distintos avances con varias tecnologías web estas pueden ser javascript o un framework derivado y en la parte

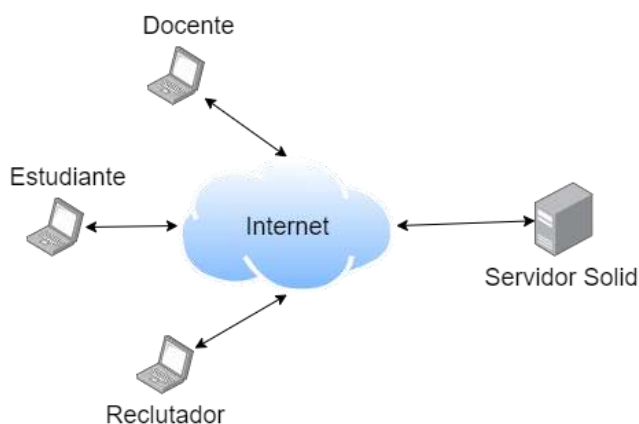
del frontend puede ser utilizado desde HTML y CSS puro como React o Angular para manejar aplicaciones de una sola página con PWA.

Diseño de la arquitectura descentralizada para el registro de e-profile

Parte de hardware y software implementado en el prototipo para el registro de e-profiles Pancka Rurya plasmado en diagramas de como convivir dentro de un mismo entorno de desarrollo los actores involucrados en el sistema determinando la primera arquitectura propuesta con una conexión directa del estudiante al servidor mediante internet como se muestra en la figura 11.

Figura 11.

Pancka Rurya primera arquitectura propuesta



Nota. Es una arquitectura simple que se trabaja sobre un solo servidor Solid, gráfico elaborado por el autor

El estudiante genera su e-profile con habilidades que aún no son validadas registrado en el servidor Solid, el docente accede al perfil del estudiante donde muestra cuales habilidades no han sido verificadas, pudiendo verificar y agregar el tipo de curso, charla o voluntariado realizó, y el reclutador puede registrar el cargo desempeñado por el estudiante.

Los limitantes dentro de esta arquitectura radica sobre las funciones y falta de algunos actores listados a continuación:

- No se puede acceder a la información sin un previo registro dentro del servidor que conceda los permisos.

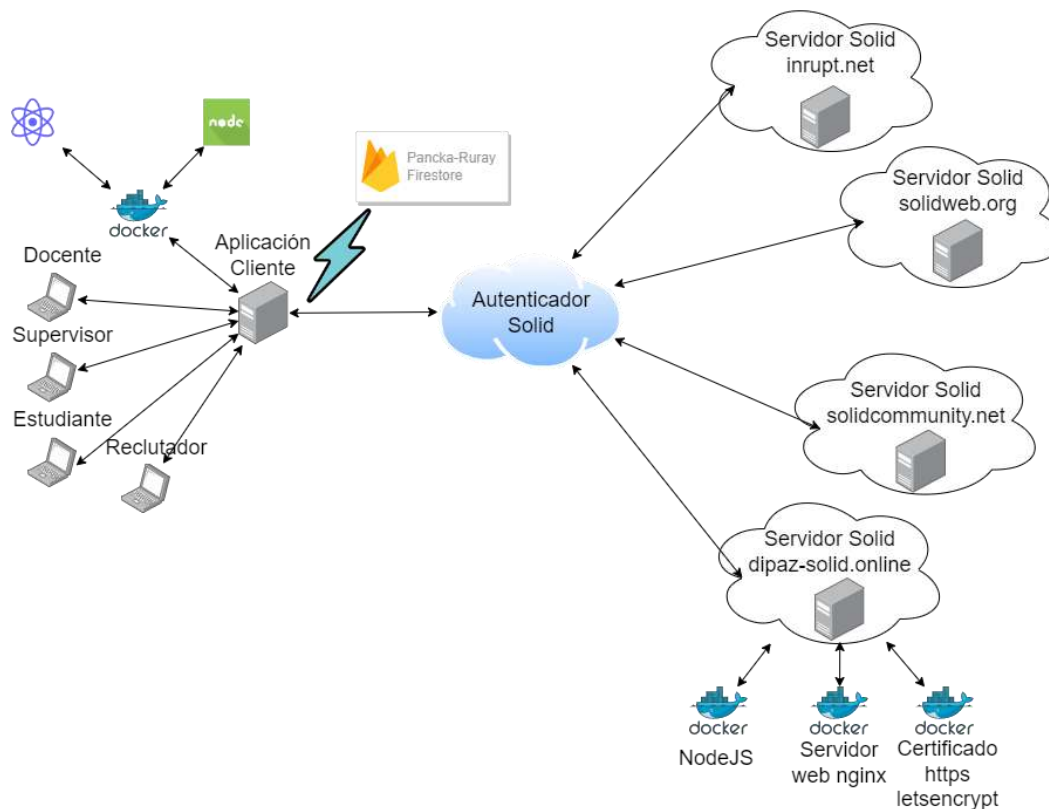
- Se debe contar con tres aplicativos separados para poder agregar información solo de estudiantes, agregar certificados de cumplimiento por parte de docentes y agregar prácticas profesionales por empresas.
- El consumo de información por terceros se dificulta limitada por almacenar información dentro de un solo servidor.
- El tráfico de la red se satura por las peticiones de los tres actores estudiante, docente y reclutador
- Para replicar la arquitectura dentro de otros sistemas universitarios es complejo estandarizar el lugar donde proveer información en los casos de actividades fuera de la institución educativa donde pertenece el estudiante, aunque el docente agregue estudiantes ajenos a la institución educativa causa una duplicidad de la información

Para poder resolver el problema de la primera propuesta arquitectónica de Pancka Ruray donde se registre e-profile de estudiantes se opta por manejar distintos servidores Solid el cual permite aligerar el tráfico de la red para poder llevar a cabo las funciones de validación de actividades

Para solventar los problemas de la arquitectura presente en la figura 12 es necesario agregar una arquitectura más robusta con herramientas que permita manejar sistemas descentralizados que facilitan la migración y duplicidad de aplicaciones en los distintos servidores cloud permitiendo una mayor escalabilidad de la información y de los servicios futuros mejorando el acceso seguro y control de los datos.

Figura 12.

Pancka Ruray arquitectura final propuesta



Nota. Dentro del esquema de la arquitectura se muestra gran cantidad de servidores, las herramientas que se van a utilizar para la parte del almacenamiento de la información como la parte visual que los actores utilizan, elaborado por el autor.

Con esta arquitectura permite mantener la solución planteada con un entorno descentralizado, con herramientas para desarrollar software que proporciona los siguientes beneficios sobre la arquitectura anterior:

- Gestiona el acceso a través del protocolo Solid de autenticación de la información la cual permite trabajar con varios servidores que comparten datos.
- Los contenedores permiten tener una red interna para manejar el servidor Solid sobre una red virtual, que permite cumplir con los requisitos para usar POD, hacer portable la aplicación para registrar e-profile.
- La base de datos cloudstorage permite generar una uri con el nombre del POD y el servidor donde se guarda la información que los reclutadores pueden consultar.

- Se necesita una herramienta de software la cual pueda tomar decisiones donde renderiza la información basada en datos obtenidos mediante permisos y el nivel de rol de acceso de lo contrario no mostrar un mensaje de falta de autorización por ello React y NodeJS es una de las mejores opciones generando la forma estándar para un PWA.
- Al tener una forma de identificar el lugar donde se guarda la información podemos con solo tener el link del perfil lograr consultar datos para agregar o actualizar certificados o títulos dentro o fuera de la institución educativa, en empresas fuera directamente relacionada con el estudiante.
- Los docentes y supervisores serán manejados por grupos los cuales pueden manejar información dentro y fuera de su servidor con el nivel de rol de permisos por archivo, carpeta o perfil público.

La comunidad Solid permite instalar servidores para manejar POD, estos proporcionan una configuración estándar para una sola persona, el uso sea de forma local donde pueda probar los beneficios de una manera rápida y sencilla es por lo que en la tabla 10 se analiza el archivo config basado en las posibles opciones a seleccionar con la descripción y razón de por qué utilizar dicha configuración

Tabla 10.

Configuración establecida dentro del servidor Solid de Pancka Ruray

Etiquetas	Descripción
/app/main/default.json	El archivo contiene como inicializar, detener y finalizar el servidor Solid.
/app/init/initialize-root.json	Permite saber cuál será la forma que tendrá el manejador de datos dentro del servidor en este proyecto almacenará a los usuarios desde la raíz del servidor.

/app/setup/disabled.json	La instalación y configuración del servidor no está disponible para cambiar su configuración que maneja la forma de almacenar los datos.
/app/variables/default.json	Maneja el cliente donde configura la forma de guardar la información de una forma persistente o volátil y dentro del servidor los tokens que maneja para autorizar a las aplicaciones de terceros, el puerto y la raíz para almacenar datos.
/http/handler/default.json	Maneja las configuraciones que pasan por una solicitud para las peticiones estáticas, OIDC, peticiones http, identificador del proveedor, LDP.
/http/middleware/websockets.json	El gestor que maneja las peticiones entrantes agrega encabezados a las respuestas, el orden de ejecución y cuándo cerrar una conexión.
/http/static/default.json	Las direcciones para archivos estáticos como el icono del servidor, los archivos de hojas de estilos en Cascada para el diseño del servidor.
/identity/access/public.json	Los permisos para quienes pueden crear un nuevo POD dentro del servidor.
/identity/email/default.json	Es la forma con la cual permite manejar el sistema de autenticación en nuestro servidor a partir del correo electrónico.
/identity/handler/default.json	Permite configurar la redirección de archivos para peticiones para el proveedor de identificadores de IDP, permite o no a las aplicaciones de terceros conectarse leyendo los permisos y el nivel necesario de autorización.
/identity/ownership/token.json	Permite solicitar que se agregue un valor específico a los identificadores web o URL de aplicaciones que consumen el almacenamiento Solid.
/identity/pod/static.json	Permite almacenar los recursos de un nuevo POD dentro del servidor de una forma estática.

/identity/registration/enabled.json	Se configura la vista que se tendrá para el registro, el controlador de petición y el tipo de autenticación en este caso contraseña.
/ldp/authentication/dpop-bearer.json	El tipo de autenticaciones admitidas en el sistema permite la demostración de pruebas de posesión (DPoP), Bearer y acceso sin credenciales.
/ldp/authorization/webacl.json	Usar el acceso a la web mediante WebACL para la autorización que maneja los recursos mediante autorización, permisos de lectura para enviar el acceso externo a contenedores que utilizan almacenamientos internos, acceso a la lectura de archivos que estén fuera al contenedor.
/ldp/handler/default.json	Maneja las peticiones, autorizadas, error de petición, operación de metadatos, poder escribir, dividir la información enviada.
/ldp/metadata-parser/default.json	Convierte los metadatos de peticiones en RDF.
/ldp/metadata-writer/default.json	Agrega respuestas de metadatos en RDF.
/ldp/modes/default.json	Coloca las funciones necesarias para HTTP, extrae módulos de acceso para solicitudes PATCH dentro del body de la solicitud, y el extraer el acceso basado en HTTP.
/storage/backend/file.json	El almacenamiento de los datos en el backend dentro del servidor es guardado en archivos y carpetas por los usuarios.
/storage/key-value/resource-store.json	La forma de almacenamiento es escrita en bajo nivel, con valor interno, bloqueando el acceso externo a los contenedores de almacenamiento evitando la exposición de los datos y marcó el almacenamiento como un recurso auxiliar que se oculta de los contenedores.
/storage/middleware/default.json	Maneja el caché para evitar recursos duplicados, configura un almacenamiento en pila del servicio utilizado por las instancias y cuando accede a un

	contenedor con un archivo index.html es usado como vista.
/util/auxiliary/acl.json	Contiene referencia a los archivos auxiliares cuando aplicaciones usan acl lleva al archivo reg.
/util/identifiers/suffix.json	La dirección http://dipaz-solid.online/nombrePOD para usuarios nuevos registra el nombre del POD después del nombre del dominio del servidor, y funciona solo para archivos guardados en backend.
/util/logging/winston.json	Uso de librería Winston que maneja el inicio de sesión.
/util/representation-conversion/default.json	Conversión de datos, encuentro de rutas para convertir una interfaz de usuario predeterminada por una representación de los recipientes de la comunidad.
/util/resource-locker/memory.json	Usado para múltiples operaciones de lectura simultánea, bloquea almacenamiento en memoria y bloqueos que caducan con la inactividad.
/util/variables/default.json	Llama a las configuraciones establecida, como el puerto del servidor, la autenticación y configuración estableciendo el dominio del servidor, establece el nivel máximo de mensajes de registros, mostrar los errores de respuesta del servidor, rutas de archivos JSON para manejar los POD

Nota. Dentro de esta configuración no se utiliza websockets como factor del servidor, al añadir manualmente, y dentro de las utilidades diferentes tipos de conexión en este caso la más segura mediante https.

Las configuraciones establecidas permiten crear usuario y el ingreso de aplicaciones sin poder cambiar el nivel de permiso es por lo que se debe añadir dos configuraciones extra manejar HTTPS para poder confiar en la información dentro de las aplicaciones Solid y un recipiente que maneje de forma gráfica los permisos de las carpetas y archivos del POD de usuarios, la comunidad de Solid cuenta con dos Mashlib y Penny. La decisión para el proyecto es usar Mashlib al poder manejar los permisos de la mejor manera al tener automáticamente el ingreso de campos basados en RDF. Para poder usar HTTPS basado en las llaves y el certificado son

creados por un contenedor fuera del servidor que genera los archivos .pem de forma estática en el en la ruta cert y agregando el siguiente texto de la figura 13 en config:

Figura 13.

Configuración para tener HTTPS en servidor Solid

```
"@id": "urn:solid-server:default:ServerFactory",
"@type": "WebSocketServerFactory",
"baseServerFactory": {
  "@id": "urn:solid-server:default:HttpServerFactory",
  "@type": "BaseHttpServerFactory",
  "handler": { "@id": "urn:solid-server:default:HttpHandler" },
  "options_showStackTrace": { "@id": "urn:solid-server:default:variable:showStackTrace" },
  "options_https": true,
  "options_key": "/certs/privkey.pem",
  "options_cert": "/certs/fullchain.pem"
},
"websocketHandler": {
  "@type": "UnsecureWebSocketsProtocol",
  "source": { "@id": "urn:solid-server:default:ResourceStore" }
}
```

Nota. Los archivos privkey y fullchain son generados por letsencrypt el tipo de configuración es el factor del servidor websocket que cambia el HTTP por HTTPS es por eso la configuración extra, figura elaborada por el autor

Figura 14.

Configuración para utilizar el entorno gráfico mashlib

```

"@id": "urn:solid-server:default:DefaultUiConverter",
"@type": "ConstantConverter",
"contentType": "text/html",
"filePath": "./node_modules/mashlib/dist/databrowser.html",
"options_container": true,
"options_document": true,
"options_minQuality": 1,
"options_disabledMediaRanges": [
  "image/*",
  "application/pdf"
]

"comment": "Serve Mashlib static files.",
"@id": "urn:solid-server:default:StaticAssetHandler",
"@type": "StaticAssetHandler",
"assets": [
  {
    "StaticAssetHandler:_assets_key": "/mash.css",
    "StaticAssetHandler:_assets_value": "./node_modules/mashlib/dist/mash.css"
  },
  {
    "StaticAssetHandler:_assets_key": "/mashlib.min.js",
    "StaticAssetHandler:_assets_value": "./node_modules/mashlib/dist/mashlib.min.js"
  },
  {
    "StaticAssetHandler:_assets_key": "/mashlib.min.js.map",
    "StaticAssetHandler:_assets_value": "./node_modules/mashlib/dist/mashlib.min.js.map"
  }
]

```

Nota. Mashlib permite manejar la información personal, crear carpetas públicas o privadas, con un diseño gráfico para personas que está iniciando a manejar permisos mediante RDF, figura elaborada por el autor.

Lista la configuración del servidor es necesario colocar un Dockerfile el cual se muestra en la figura 15 que es el encargado de elaborar una imagen de nuestro servidor Solid listo para ejecutar como contenedor

Figura 15.

Dockerfile de nuestro servidor Solid Pancka ruray

```

FROM node:lts AS build
WORKDIR /community-server
COPY package*.json ./
COPY . .
RUN npm ci --unsafe-perm && npm run build

FROM node:lts-alpine
LABEL maintainer="Solid Community Server Docker Image Maintainer
<matthieubosquet@gmail.com> and editing by Diego Ignacio Paz Naula
for create a server Solid Pancka Ruray"

RUN mkdir /config /data
WORKDIR /community-server
COPY --from=build /community-server/package.json .
COPY --from=build /community-server/bin ./bin
COPY --from=build /community-server/config ./config
COPY --from=build /community-server/dist ./dist
COPY --from=build /community-server/node_modules ./node_modules
COPY --from=build /community-server/templates ./templates

EXPOSE 3000
ENTRYPOINT [ "node", "bin/server.js" ]
CMD [ "-c", "config/my-config-mashlib.json", "-f", "/data" ]

```

Nota. El archivo dockerfile cuenta con una configuración de un contenedor de nodejs, copiando direcciones necesarias para los estilos, librerías, configuración y dirección para almacenamiento de los datos, figura elaborada por el autor.

El archivo Dockerfile está realizado en base al proporcionado por la comunidad Solid donde se agrega la configuración planteada en los pasos anteriores, se colocó la propiedades extras como dirección raíz para el almacenamiento de los datos al ser restaurado o reiniciando el servidor Solid, cuenta con la configuración estándar del entorno de desarrollo NodeJs con la versión estable alpine, y copiar las carpetas de configuración, estilos, librerías y temas para mostrar el servidor con el recipiente Mashlib ejecutándose como aplicación que maneja los permisos de usuarios.

Figura 16.

Dockercompose configuración del servidor web nginx proxy

```

services:
  nginx-proxy:
    image: jwilder/nginx-proxy
    restart: always
    ports:
      - "80:80"
      - "443:443"
    volumes:
      - /var/run/docker.sock:/tmp/docker.sock:ro
      - certs:/etc/nginx/certs:ro
      - vhostd:/etc/nginx/vhost.d
      - html:/usr/share/nginx/html
    labels:
      - com.github.jrcs.letsencrypt_nginx_proxy_companion.nginx_proxy

```

Nota. La figura contiene la configuración para montar un contenedor de nginx con un proxy por defecto, elaborado por el autor.

En la figura 16 se tiene una imagen de un contenedor por jwilder quien permite añadir los archivos de certificados, nombres de dominio y una configuración para HTML como página de inicio que comparta la información con otros contenedores además de tener el puerto donde se ejecuta las peticiones http transformándose a https automáticamente.

Figura 17.

Dockercompose letsencrypt certificado https

```

letsencrypt:
  image: jrcs/letsencrypt-nginx-proxy-companion
  restart: always
  environment:
    - NGINX_PROXY_CONTAINER=nginx-proxy
  volumes:
    - certs:/etc/nginx/certs:rw
    - vhostd:/etc/nginx/vhost.d
    - html:/usr/share/nginx/html
    - /var/run/docker.sock:/var/run/docker.sock:ro

```

Nota. Letsencrypt permite crear certificados que validan dominios web por un tiempo, y si se ejecutan de forma gratuita y válida al navegar por la red para proyectos pequeños, figura elaborada por el autor

La configuración permite usar la imagen de jracs donde usa letsencrypt para validar un servidor web nginx con proxy inverso manteniendo una red privada entre contenedores para letsencrypt y el servidor web nginx compartiendo la información que necesitan estos servicios para la configuración HTTPS.

Figura 18.

Dockercompose configuración para imagen del servidor Solid Pancka Ruray

```

www:
  image: css
  restart: always
  expose:
    - "3000"
  volumes:
    - ~/Solid:/data
  environment:
    - VIRTUAL_HOST=dipaz-solid.online,www.dipaz-solid.online
    - LETSENCRYPT_HOST=dipaz-solid.online,www.dipaz-solid.online
    - LETSENCRYPT_EMAIL=diegopaznaula@gmail.com
  depends_on:
    - nginx-proxy
    - letsencrypt

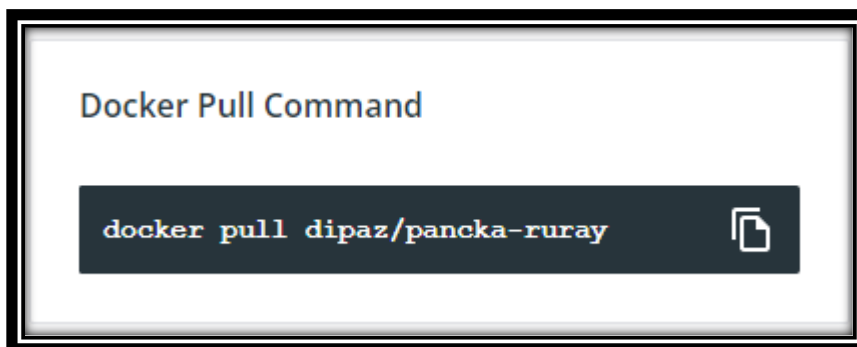
```

Nota. La configuración necesaria para unir el servidor web, la certificación HTTPS y el proyecto Solid, elaborado por el autor.

El servicio se activa de manera automática una vez ejecutado por primera vez lo único que necesita es contar con la imagen creada por el dockerfile que es la figura 15 dentro de las imágenes del contenedor ejecutado en el servidor o a su vez usar la imagen cargada en los repositorios de Docker elaborado por el auto siguiendo el enlace a continuación <https://hub.docker.com/r/dipaz/pancka-ruray> y utilizar el comando de instalación mostrado en la figura 19 para montar la imagen dentro de cualquier servidor que maneja contenedores con esto podemos reemplazar el nombre de la imagen “css” en la figura 18 y colocar el nombre dipaz/pancka-ruray teniendo una instancia del servidor, instalando en cualquier servidor de forma más sencilla para descentralizar el almacenamiento en base de proyecto Pancak Ruray.

Figura 19.

Pancka ruray dentro de Docker HUB de forma pública



Nota. Comando para instalar imagen del servidor Solid Pancka Ruray elaborado por el autor y almacenado en los Hub de Docker

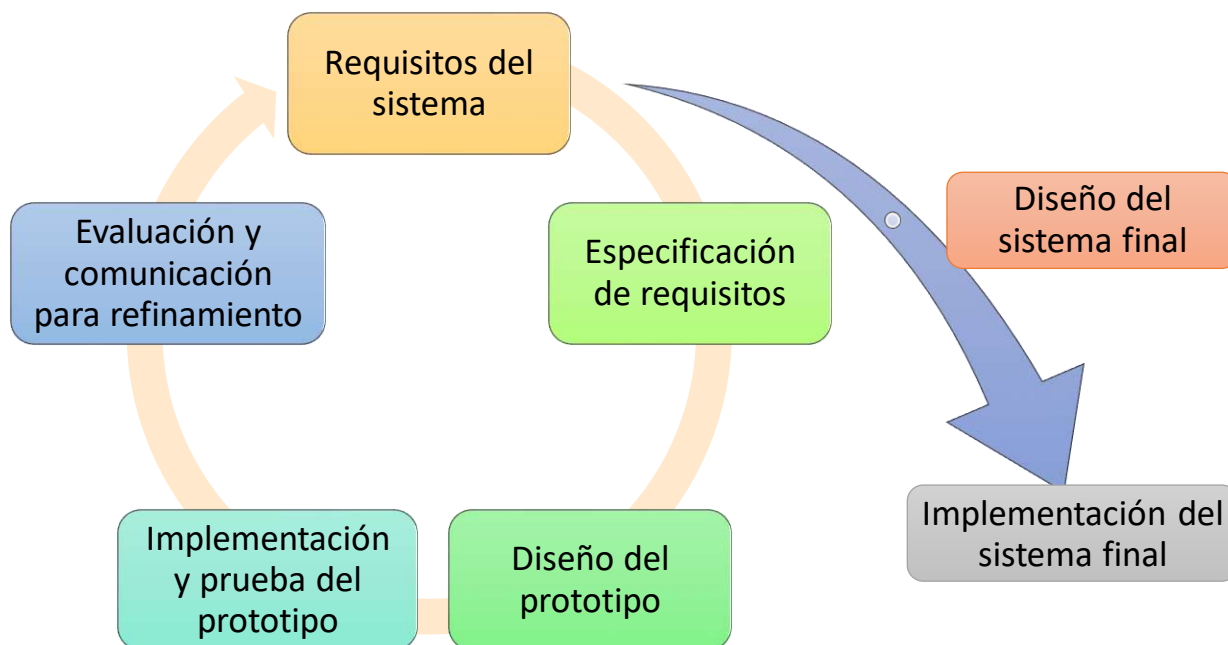
Tener la configuración, el proyecto de Solid de la comunidad editado y listo para usar solo cambiando el nombre del dominio del servidor permite instanciar una gran cantidad de servidores en el menor tiempo posible y de la manera más rápida usando todas las imágenes desde la nube de Docker como lo es el servidor web nginx con proxy, letsencrypt conectado al servidor web y ahora el proyecto Solid con los cambios realizados por el autor.

Diseño del proceso para el registro de e-profile

Para lograr cumplir con los requerimientos del proyecto se basará a través de patrones arquitectónicos y diseño de software. Mediante el uso del modelo de prototipos perteneciente a los modelos evolutivos y el cual cumple con el ciclo de vida del prototipado evolutivo al ser necesario la construcción en un lapso muy corto del tiempo, conocer cuales son las herramientas adecuadas y no incluir el uso de muchos recursos de hardware.

Nuestro cliente son los estudiantes para lo cual realizar un demo funcional que permite una retroalimentación clara de lo que necesitan para futuros requisitos de software, permitiendo conocer las perspectivas, tendencias de estilos y necesidades dentro de los estudiantes que son cambiantes dentro de una u otra institución educativa.

Para entender mejor el orden del desarrollo de Pancka Ruray dentro de la figura 20 se muestra el ciclo de vida del prototipado evolutivo

Figura 20.*Ciclo de vida del prototipado evolutivo*

Nota. La figura 20 representa el ciclo de vida del prototipado evolutivo estándar con las 5 etapas más el diseño y despliegue final

Requisitos del sistema

Los requisitos están establecidos dentro del alcance, donde se establece lo que ofrece el sistema y las restricciones que tendrá dentro del funcionamiento.

Especificación de requisitos

Esta establecido dentro de las historias de usuario en el análisis para el desarrollo de e-profile, se plantea los actores para construir el registro de Pancka Ruray

Diseño del prototipo

El diseño está establecido por partes la del servidor de almacenamiento que proporciona el acceso, seguro, confiable y controlado de los datos y el cliente Pancka Ruray encargado de presentar la información y un esquema de e-profile de los estudiantes a docentes y reclutadores de las habilidades certificadas junto con información personal.

Implementación y prueba del prototipo

La implementación y prueba fue desarrollada de forma local las pruebas de conectividad, manejo de protocolos seguros para conectar la información del servidor y el prototipo dentro de un entorno controlado.

Evaluación y comunicación para refinamiento

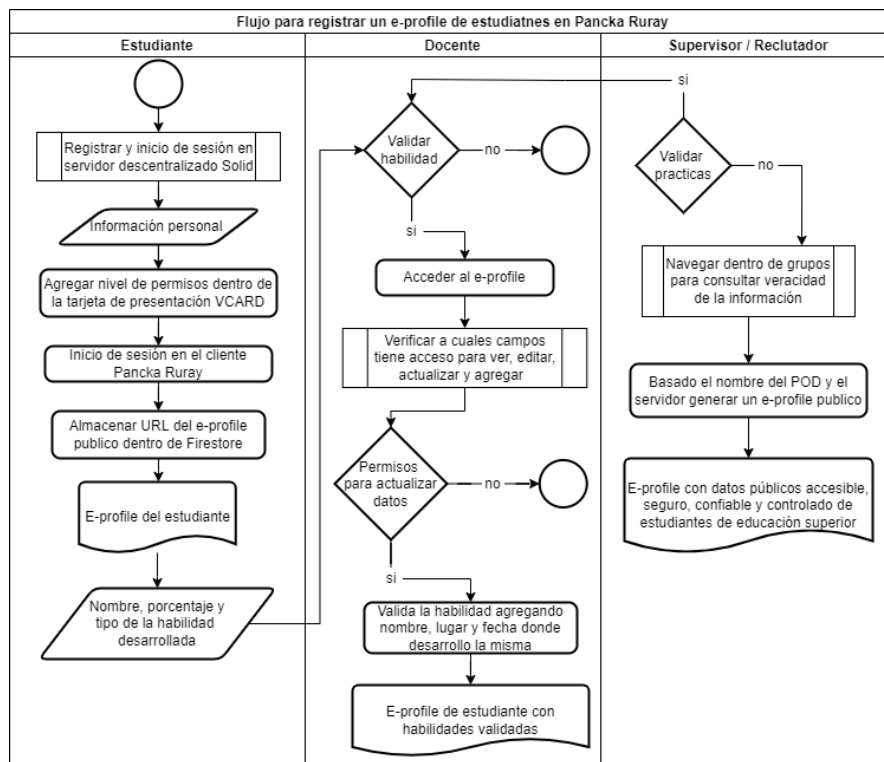
El sistema tuvo dificultades en el despliegue por contar con distintas versiones de librerías utilizados en el prototipo, el tráfico de la red saturó el límite gratuito del servidor necesitando una solución para disminuir el consumo archivos multimedia, refinando el prototipo hasta lograr la integración de una aplicación de página única y con manejo de una web progresiva.

El ciclo de prototipado evolutivo tuvo lugar a dos ciclos el primero usando aplicaciones estáticas basado en tecnologías HTML, CSS y Javascript, mientras que el segundo ciclo tuvo lugar al desarrollo de la web progresiva usando React solventando el tráfico excesivo de datos por peticiones al editar y consultar información como estudiante, los docentes al verificar la información saturando el servidor Solid, mientras que los reclutadores no podían revisar la información por falta de medios para enrutar los e-profile. Una vez solventado los objetivos planteados en las historias de usuario la arquitectura de Pancka Ruray es la figura 12.

Para plasmar cual es el flujo de la información y la utilización del sistema Pancka Ruray se lo realiza mediante un diagrama de flujo el cual cuenta con los actores estudiante, docentes, empleados y supervisores, con el objetivo de registrar actividades extracurriculares dentro del e-profile pertenecientes a los estudiantes de instituciones de educación superior diagramado en la figura 21.

Figura 21.

Diagrama de flujo para el registro de actividades extracurriculares Pancka Ruray



Nota. El gráfico muestra el proceso lineal dentro de Pancka Ruray para el registro de actividades académicas y extracurriculares

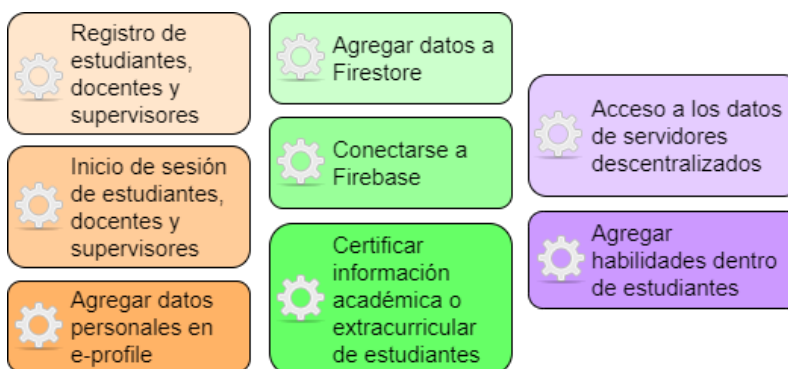
Las aplicaciones de página única utilizan una arquitectura basada en componentes dentro de los cuales puede usarse como contenedores de información y vista para presentar los datos. Tener un sistema cliente servidor y querer manejar solicitudes bajo permisos, web semántica, distintos servidores es algo impensable mientras que el patrón de arquitectura de microservicios permite ordenar y organizar a los actores dividiendo en mini programas que se añaden dependiendo el rol que desempeña el usuario autenticado.

Tiene ventajas como el fácil escalamiento, permite un fácil entendimiento para realizar futuros cambios, un despliegue independiente sin tener una coordinación con el resto de los equipos del sistema. La arquitectura de microservicios es usado cuando los sistemas requieren un desarrollo rápido dentro de temas de tendencia como es la privacidad de los datos manejados por protocolos de W3C y por último es un tipo de desarrollo que se extiende por todo el mundo

al ser tendencia usar tecnologías que se programan con una lógica de componentes .Reescribir cambios bajo los protocolos que maneja Solid server como es el inicio de sesión mediante OIDC,o IDP en la figura 22 mostraremos los microservicios que se utiliza en Pancka Ruray.

Figura 22.

Microservicios de Pancka Ruray



Nota. Se cuenta con un total de 8 microservicios los cuales permite conectar las herramientas tecnológicas con los actores y con otras herramientas tecnológicas para lograr cumplir con los requerimientos planteados, figura elaborada por el autor.

Una vez analizados los servicios que se incorpora en el sistema se describe cual será el contenido manejado a través de etiquetas RDF dentro de los e-profile en la tabla 11:

Tabla 11.

Etiquetas RDFS para manejo de currículos vitae

Etiquetas	Descripción
SkillLastUsed	Es la última fecha en que se utiliza una habilidad certificada por un docente o supervisor dentro o fuera de la institución educativa o empresa http://rdfs.org/resume-rdf/cv.rdfs#skillLastUsed
SkillName	Es el nombre de la habilidad dura o socioemocional que contiene los estudiantes en sus e-profile este validada o no un docente o supervisor http://rdfs.org/resume-rdf/cv.rdfs#skillName
OtherInfoType	Usado para agregar el tipo de nueva información que no esté contemplada dentro de un RDF para CV en el proyecto actual es conocer qué tipo de habilidad se agregó una blanda (soft skill) dura (hard skill) http://rdfs.org/resume-rdf/cv.rdfs#otherInfoType

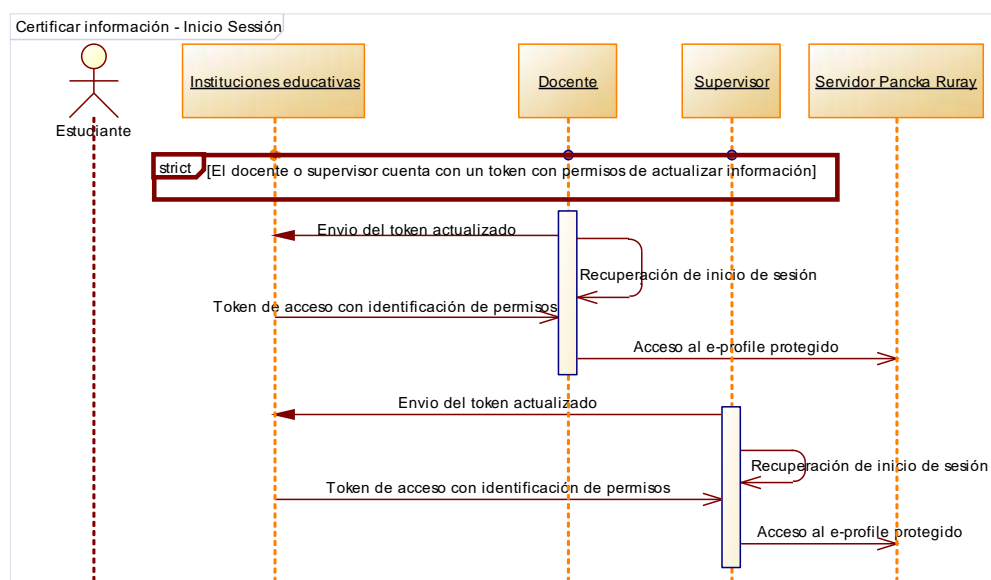
SkillLevel	Para manejar el nivel de habilidad recomienda una escala de Likert de 0 a 5 pero en este caso se maneja de forma porcentual para el manejo del crecimiento personal de los estudiantes, donde los docentes pueden añadir un nuevo valor http://rdfs.org/resume-rdf/cv.rdfs#skillLevel
HasSkill	Permite validar la existencia de una habilidad y sirve para poder descartar o no la certificación de habilidades de estudiantes por docentes o empleados http://rdfs.org/resume-rdf/cv.rdfs#hasSkill

Nota. Para agregar permisos dentro de los archivos, la información y agregar un identificador para elaborar un CV se utiliza RDF y para agregar

Una de las herramientas principales es la conexión que autentifica y autoriza a los actores para poder realizar cambios dentro de la información esto es mediante el uso de solid-client-authn-js la cual cuenta con un flujo de datos que se representa de forma gráfica en la figura 23 y figura 24

Figura 23.

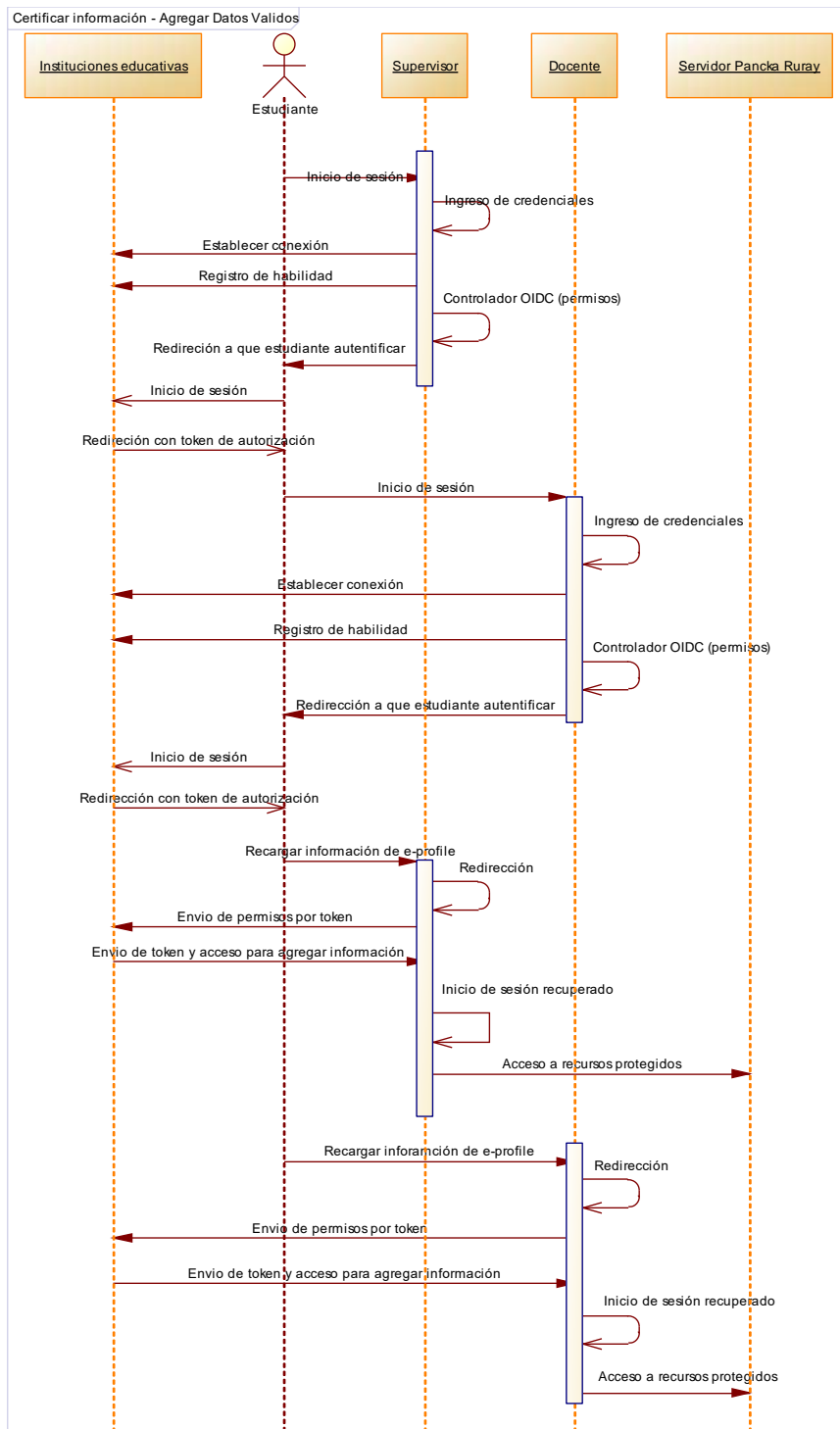
Diagrama secuencial para el manejo de información en Solid



Nota. Diagrama UML secuencia elaborado por el autor para iniciar sesión dentro del sistema Pancka Ruray basado en el protocolo OIDC

Figura 24.

Diagrama secuencial para el registro de habilidades



Nota. Diagrama UML secuencial elaborado por el autor para certificar las habilidades mediante un acceso seguro y confiable de la información dentro de Pancka Ruray

Capítulo IV

Desarrollo e implementación del prototipo de software

Muestra del flujo de la red, almacenamiento y resultados al poner el prototipo funcional en producción, permitiendo generar el registro de e-profile de estudiante de los últimos niveles.

Introducción

Dentro del capítulo se realiza el desarrollo de la aplicación cliente Pancka Ruray que es una aplicación usando el servidor Solid como almacenamiento permitiendo gestionar e-profile de los estudiantes almacenando información en dipaz-solid y luego añadiendo los permisos dentro de los grupos los cuales pueden agregar información sin alterar otra para los docentes y empleados, certificando las habilidades mediante la participación que desarrolla la experiencia de los estudiantes.

Se establece como desplegar la aplicación cliente de forma gratuita y se podrá verificar los beneficios que se tiene al usar un PWA dentro de las pruebas con la ayuda de analytics una herramienta de Google y que se puede usar con firebase dentro de aplicaciones web que tiene control para marketing digital mediante disparadores específicos que los usuarios realizan como visitar a la página web principal, total de clic en respectivos componentes, solicitudes realizadas a un servidor de almacenamiento de información entre otras.

Desarrollo

Desarrollo del sistema para estudiantes donde se registra los e-profile

El prototipo Pancka Ruray cliente para el registro de e-profile de estudiantes tiene tres grandes componentes para manejar los 8 microservicios planteados en la sección anterior por ello se los describe en la presente sección

Desarrollo necesario para la configuración de Firebase

La descripción del manejo de la base de datos firestore, analytics dentro de un aplicativo estándar de React se puede observar en la figura 25

Figura 25.

Configuración para usar Firebase en React

```
import { initializeApp } from "firebase/app"
import { getFirestore } from "firebase/firestore"
import { getAnalytics } from "firebase/analytics"

const firebaseConfig = {
  apiKey: process.env.REACT_APP_API_KEY,
  authDomain: process.env.REACT_APP_AUTH_DOMAIN,
  projectId: process.env.REACT_APP_PROJECT_ID,
  storageBucket: process.env.REACT_APP_STORAGE_BUCKET,
  messagingSenderId: process.env.
  REACT_APP_MESSAGING_SENDER_ID,
  appId: process.env.REACT_APP_APP_ID,
  measurementId: process.env.REACT_APP_MEASUREMENT_ID
}

initializeApp(firebaseConfig)
export const analytics=getAnalytics()
export const db=getFirestore()
```

Nota. La configuración estándar representada en la figura es generalmente usada dentro de javascript, teniendo las credenciales desde un archivo .env para no subir en texto plano las credenciales, código elaborado por el autor.

Se tiene dos librerías firestore para manejar datos los datos dentro de la colección de PODs, luego documentos que tiene el nombre del POD y el servidor donde se almacenó, por último, cuenta con una colección con la información de la URL del servidor, la URL del perfil WebID del estudiante, docente o empleado y el nombre del POD. Permite tener datos esenciales los cuales sirven para poder obtener los datos del perfil de forma pública y sin inicios de sesión.

Después de establecer la conexión a firebase se debe poder realizar las peticiones de agregar, consultar y actualizar la información en la cual de forma estática se tiene dos funciones mostradas en la figura 26

Figura 26.

Funciones guardar, consultar información de Firestore

```
import { doc, getDoc, setDoc } from "firebase/firestore"
import { db } from "../config"

const collectionName="PODs"

export const saveData=(data,uri)=>setDoc(doc(db,
collectionName,uri),data)
export const getData=(id)=>getDoc(doc(db,collectionName,id))
```

Nota. Dentro del archivo se establece la cadena de carácter de la colección y dos funciones para manejar la información, código elaborado por el autor.

El nombre de la colección permite tener una forma de consultar todos los usuarios ingresados donde guardar datos, el nombre utilizado cuenta con dos parámetros datos en un formato JSON, y la URI con el nombre y servidor del POD, obtener datos usa el parámetro id que es para identificar los POD del usuario obteniendo la información pública dentro de un e-profile que genera la vista para los reclutadores de empresas.

Conexión con servidores Solid para consultar perfiles

Para poder consultar la información de e-profile de estudiantes se configura dentro de React el ingreso del sistema, restaurando o no información proporcionada almacenada en caché el código para esto es mostrado en la figura 27

Figura 27.

Manejo de sesión por proveedores Solid

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';
import { SessionProvider } from "@inrupt/solid-ui-react";

const root = ReactDOM.createRoot(document.getElementById(
'root'));
root.render(
  <SessionProvider restorePreviousSession>
    <App />
  </SessionProvider>
);
```

Nota. El manejo de sesión este puesto en el index de la aplicación react para controlar los inicios de sesión en proveedores Solid, código usado por el autor

El acceso público a los e-profile o de forma privada para agregar, certificar y controlar la información se lo maneja por Navegación de Rutas que permiten mantener el rol de cada individuo registra o no con el nivel de privilegios de la información implementado en la figura 28

Figura 28.

Navegación del prototipo Pancka Ruray

```
import './App.css';
import { useSession } from '@inrupt/solid-ui-react';
import LoginForm from './components/LoginForm';
import Dashboard from './components/Dashboard';
import Profile from './components/Solid';
import { BrowserRouter, Routes, Route } from 'react-router-dom';

function App() {
  const { session } = useSession()
  if (!session.info.isLoggedIn){
    return (
      <BrowserRouter>
        <Routes>
          <Route path="/" element={<LoginForm/>}/>
          <Route path="/profile/:id" element={<Profile/>}/>
          <Route path="*" element={<h1>No encontrado</h1>}/>
        </Routes>
      </BrowserRouter>
    )
  }
  return <Dashboard/>
}
export default App;
```

Nota. Las routes permite manejar los tres casos, no estar registrado, pero necesitar ver la información de un e-profile, que la ruta no exista y poder iniciar sesión dentro de Pancka Ruray

Una vez posible gestionar los distintos privilegios del sistema es necesario poder acceder o escalar en el mismo bajo las reglas lógicas planteadas en la sección del diseño y análisis es por ello por lo que la figura 29 tiene el ejemplo práctico de cómo se realizó

Figura 29.

Inicio de sesión en Pancka Ruray

```
<div className="login-btns">
  <LoginButton className="login-btns-btn"
    oidIssuer="https://dipaz-solid.online"
    redirectUrl={redirect}
    restorePreviousSession
  >>button>Sesión Dipaz</button></LoginButton>
```

Nota. El código utiliza el botón de inicio de sesión para cada proveedor este debería ser sustentado para múltiples servidores.

Los tres campos para iniciar sesión dentro del registro de e-profile: 1) la URL del servidor Solid los cuales están descentralizados, 2) la redirección permite ingresar una vez se confirme el token de OAuth 2, y 3) restablecer sesiones previas iniciadas sesión dentro de proveedores Solid.

Todas las configuraciones previas permiten contar con un control de acceso, seguro, confiable a los datos de un e-profile estudiantil por cualquier el desarrollo de aplicación puede utilizar la configuración dentro de cualquier sistema por defecto para almacenar datos dentro de servidores Solid.

Manejo de la información de estudiantes con e-profile

Para acceder a datos dentro de POD de usuarios existen distintas maneras por ejemplo Friend of A Friend (FOAF) que es amigo de un amigo, Resource Description Framework en español marco de descripción de recursos (RDF) y para información personal se maneja dentro de tarjetas de presentación electrónica (VCARD) el cual utilizamos para almacenar información personal de los estudiantes, maestros y supervisores, como se muestra en la figura 30

Figura 30.

Atributos de tarjetas de presentación de estudiantes, docentes y supervisores dentro de Pancka Ruray

```

<section className="header-left-section">
  <h1><Text property={VCARD.fn}/></h1>
  <hr />
  <h4><Text property={VCARD.role}/></h4>
  <h4><Text property={VCARD.
organization_name.iri.value} edit=
{editing} autosave/></h4>
</section>
<section className="header-middle-section">
  <Image property={VCARD.hasPhoto.iri.
value}
className="header-middle-section-img"
alt="Foto de perfil" width="40%"/>
  <a href={webId}>Perfil Solid</a>
  <Button size="small" color="primary"
onClick={() => setEditing(!editing)}>
  Editar Información
  </Button>
</section>

```

Nota. Para usar datos de contactos está la etiqueta VCARD permitiendo a los sistemas conocer fácilmente, nombres, teléfonos, correos electrónicos entre otra información personal

Dentro de la amplia categoría disponible de etiquetas por VCARD, manejando las clases se tiene los comentarios para usar información el link <https://www.w3.org/2006/vcard/ns#> usaremos los más comunes para CV descritos dentro de la tabla 12

Tabla 12.

Etiquetas de VCARD usados en Pancka Ruray

Etiqueta	Descripción
fn	Este campo es usado para añadir los nombres y apellidos del estudiante, docente o supervisor
Role	El campo que determina si el actor es estudiante, docente o empleado
Organization name	El nombre de la organización puede ser la universidad de los estudiantes o docentes o el nombre de la empresa para los empleados
hasPhoto	El campo que permite saber si existe una foto retornando la URL con la imagen o las imágenes del perfil
Note	Es usado para escribir una descripción de las experiencias, atributos y descripciones de un estudiante, docente o empleador
hasTelephone	Permite agregar números telefónicos de trabajo, casa o no definido
hasEmail	Permite agregar correos electrónicos de trabajo, casa o no definido

Nota. Las etiquetas usadas para manejar la información general y pública de los estudiantes, docentes y empleados, tabla elaborada por el autor

Todas las etiquetas mencionadas en la tabla 12 se usa para la vista de información de los reclutadores y para agregar datos en el servidor mediante mashlib, y algunos de estos datos se puede cambiar dentro del cliente como lo es la descripción, los teléfonos, los correos, el rol y la institución que es información cambiante de forma más común en el tiempo.

La información para añadir una habilidad la cual se puede validar por un docente cuenta con las etiquetas descritas en la tabla 12, donde se añade mediante peticiones fetch como se muestra en la figura 31

Figura 31.*Petición fetch para agregar habilidades*

```

const addSkill=async(text,percentage)=>{
  const indexUrl=getSourceUrl(skillList)
  const skillWithText=addStringNoLocale(
    | createThing(),NAME_PREDICATE,text
  )
  const skillWithPercentage=addStringNoLocale(
    | skillWithText,PERCENTAGE_PREDICATE,percentage
  )
  const skillWithDate=addDatetime(
    | skillWithPercentage,CREATED_PREDICATE,new Date()
  )
  const skillWithClass=addStringNoLocale(
    | skillWithDate,SKILL_PREDICATE,type
  )
  const skillWithType=addUrl(skillWithClass,
  TYPE_PREDICATE, SKILL_CLASS)
  const updatedSkillList=setThing(skillList,skillWithType)
  const updatedDataset=await saveSolidDatasetAt(indexUrl,
  updatedSkillList,{
    | fetch: session.fetch,
  })
  setSkillList(updatedDataset)
}

```

Nota. El código elaborado por el autor dentro de la figura 31 muestra cómo agregar información dentro de una petición para actualizar las habilidades.

La función que actualiza para que una habilidad sea certificada bajo el perfil del docente tiene como código la figura 32

Figura 32.*Validación de habilidad del estudiante*

```

const updatedSkills=async(skills)=>{
  const skillsUrl=getSourceUrl(skillList)
  const date=new Date()
  const doneSkills=addDatetime(skills,UPDATE_PREDICATE,date)
  const updatedSkills=setThing(skillList,doneSkills,
  {fetch});
  const updatedList=await saveSolidDatasetAt(skillsUrl,
  updatedSkills,{fetch})
  setSkillList(updatedList)
  return
}

```

Nota. La función updatedSkills creado por el autor mostrado en la figura 32, actualiza la información y registra la fecha de modificación

Para finalizar toda la información debe mantener un orden, donde los datos puedan ser controlados, accesible, seguros y confidenciales para ello se maneja un archivo .ttl que almacena los datos agregados y validados, cuenta con la configuración del código de la siguiente figura 33

Figura 33.

Nombre del archivo donde se almacena la información

```
import { createSolidDataset, getSolidDataset,
saveSolidDatasetAt } from "@inrupt/solid-client"

export async function getOrCreateList(containerUri, fetch){
  const indexUrl=`${containerUri}ProfileData.ttl`
  try{
    const list=await getSolidDataset(indexUrl, {fetch})
    return list
  }catch (error){
    if(error.statusCode===404){
      const list=await saveSolidDatasetAt(
        indexUrl, createSolidDataset(), {fetch}
      )
      return list
    }
  }
}
```

Nota. La configuración de almacenar archivos dentro del archivo ProfileData.ttl creado por el autor agregado mediante la librería util dentro de los componentes

Despliegue del servidor Solid community

Para tener una instalación de un servidor Solid usado en Pancka Ruray visitamos el GitHub del autor con el repositorio copiado, actualizado y modificado creado por la comunidad Solid modificada en la rama educChain.

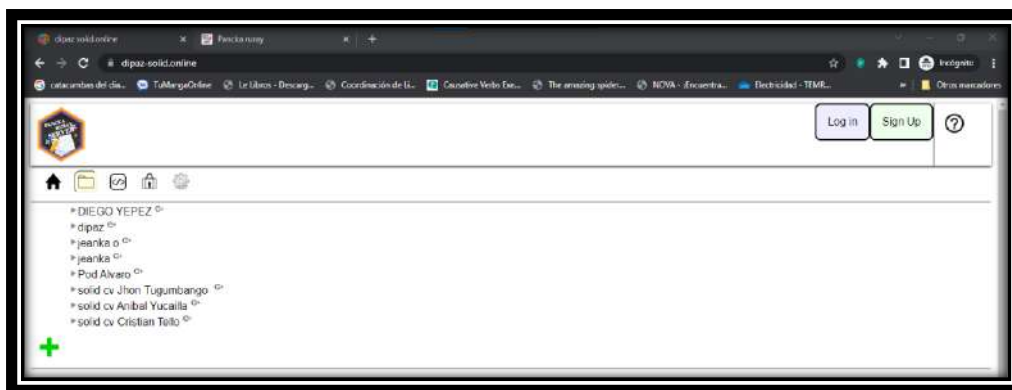
- Original: <https://github.com/diego40g/CommunitySolidServer>
- educChain: <https://github.com/diego40g/CommunitySolidServer/tree/educChain>

Se cuenta con un manual de instalación para instalar un servidor Solid en el Cloud de DigitalOcean adjuntado en la sección de anexos con el nombre de “Anexo 1: Manual de instalación para servidor Solid dentro de DigitalOcean” para implementar se requiere tener acceso al repositorio de GitHub.

Una vez implementado todos los pasos del manual de forma correcta se obtiene como resultado un servidor con un dominio, usando un protocolo seguro y almacena la información por carpetas y archivos como se muestra en la figura 34

Figura 34.

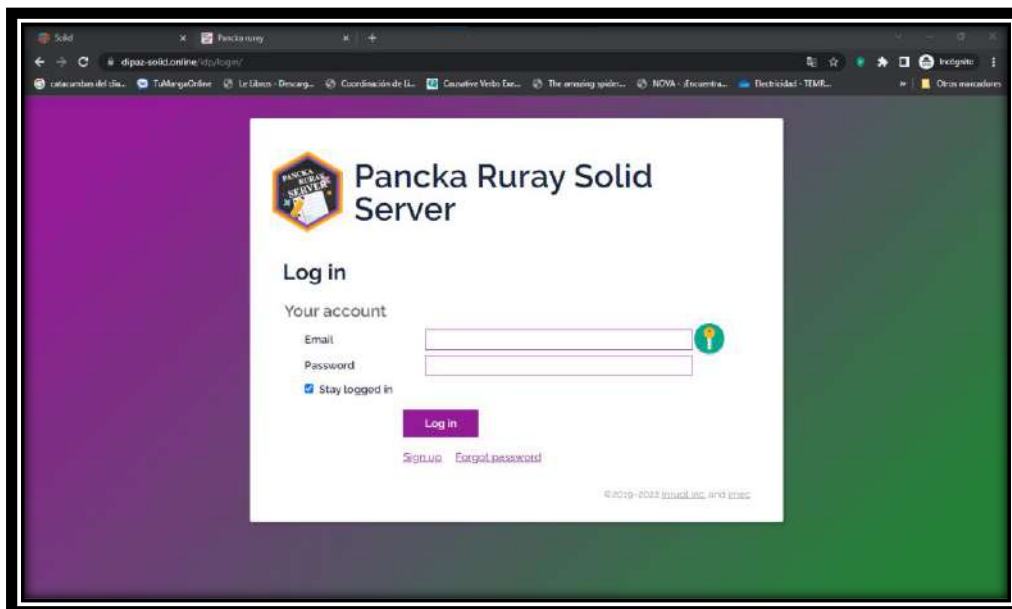
Servidor Solid con mashlib



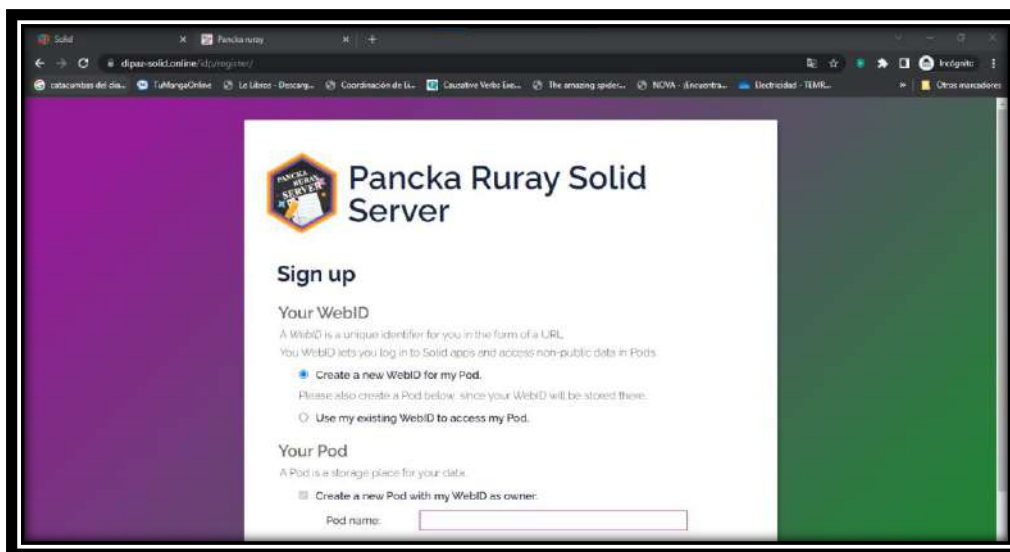
Nota. La figura 34 muestra los usuarios registrados dentro del servidor dipaz-solid implementado por el autor.

Figura 35.

Inicio de sesión servidor Solid mediante LDP



Nota. La interfaz gráfica de mashlib utiliza las credencialitas LDP configurado por el autor, basado en ingreso de correo electrónico y contraseña

Figura 36.*Registro de usuario mediante LDP*

Nota. El servidor permite un registro mediante LDP configurado por el autor, donde proporciona el nombre del POD y se registra usando correo electrónico para identificar a los usuarios.

Puesta en marcha del cliente Pancka Ruray en host firebase

Para poder registrar una aplicación dentro de firebase se debe tener una cuenta de Gmail, para usar la consola de firebase creando un proyecto y accediendo a hosting para usar el servicio después de haber copiado el proyecto Pancka Ruray cliente dentro del Github del autor siguiendo el link <https://github.com/diego40g/pancka-ruray> y los pasos como se muestra en la figura 37

Figura 37.*Comenzar host en firebase*

Nota. Imagen perteneciente a firebase para iniciar configuración del host
Los pasos para poner la producción se siguen los siguientes pasos:

- Instalar el CLI de firebase con el siguiente comando “npm install -g firebase-tools”
- Iniciamos sesión con el comando “firebase login”
- Instanciamos el servicio de firebase “firebase init”
- Luego creamos la carpeta build dentro de nuestro proyecto
- Y finalmente colocamos la producción del sistema “firebase deploy”

Figura 38.

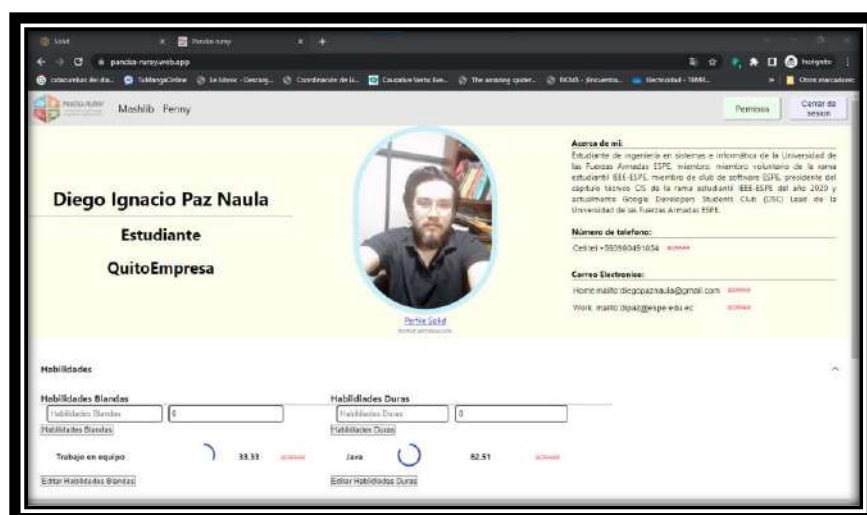
Pancka Ruray dentro de los dominios de firebase



Nota. Página de inicio elaborado por el autor donde estos distintos servidores Solid

Figura 39.

Perfil estándar de e-profile de los estudiantes



Nota. Estándar de e-profile de estudiantes que maneja la información personal y las habilidades duras o socioemocionales que son válidas por docentes o supervisores.

Capítulo V

Evaluación de prototipo de software

El objetivo del capítulo es verificar el acceso, control de los datos es por ello que se plantea una evaluación del prototipo el cual mide la dificultad basa en los tiempos y la encuesta del diseño basado en la escala de Likert es por eso que detalla el proceso a seguir donde los estudiantes podrán evaluar el prototipo y el servidor Pancka Ruray

Proceso de evaluación del prototipo

Para Pancka Ruray no se cuenta con un modelo específico de evaluación de prototipos, utilizando aquel con mayor cantidad de métricas de las cuales se puede seleccionar las que más se adapte al producto de software. Implementando el Modelo Sistémico de Calidad (MOSCA) cuyo fin es aproximar la calidad dentro de un desarrollo de software y cuenta con un total de 587 métricas.

El proceso de evaluación de software cuenta las siguientes seis categorías del producto: 1) Funcionalidad, 2) Fiabilidad, 3) Usabilidad, 4) Eficiencia 5) Mantenibilidad y 6) Portabilidad, cuentan con aspectos contextuales e internos del producto donde resaltan la selección de los contextuales resumiendo la selección en la tabla 13

Tabla 13.

Características y métricas de calidad sistémica del producto Pancka Ruray

Categoría	Aspecto Contextual	
Funcionalidad	<ul style="list-style-type: none"> • Interoperabilidad • Ajuste a los propósitos 	<ul style="list-style-type: none"> • Seguridad
Fiabilidad	<ul style="list-style-type: none"> • Recuperación 	<ul style="list-style-type: none"> • Tolerancia a fallas
Usabilidad	<ul style="list-style-type: none"> • Conformidad con los estándares • Capacidad de aprendizaje 	<ul style="list-style-type: none"> • Facilidad de comprensión • Interfaz gráfica
Eficiencia	<ul style="list-style-type: none"> • Utilización de recursos 	

Mantenibilidad	<ul style="list-style-type: none"> • Capacidad de prueba • Estabilidad 	<ul style="list-style-type: none"> • Facilidad de cambio
Portabilidad	<ul style="list-style-type: none"> • Coexistencia 	<ul style="list-style-type: none"> • Capacidad de instalación

Nota. Las métricas colocadas son basadas en la revisión de literatura y que pueden ser factores decisivos para la evaluación de un prototipo descentralizado

Pasos para desarrollo del taller con estudiantes

Previo al análisis de las preguntas se detalla un taller paso a paso desde la figura 40 hasta la 62 que sirve para evaluar algunos aspectos contextuales de la tabla 13

Figura 40.

Registro de estudiante dentro del servidor



Nota. Imagen del servidor Pancka ruray para iniciar sesión, tomada por el autor del prototipo

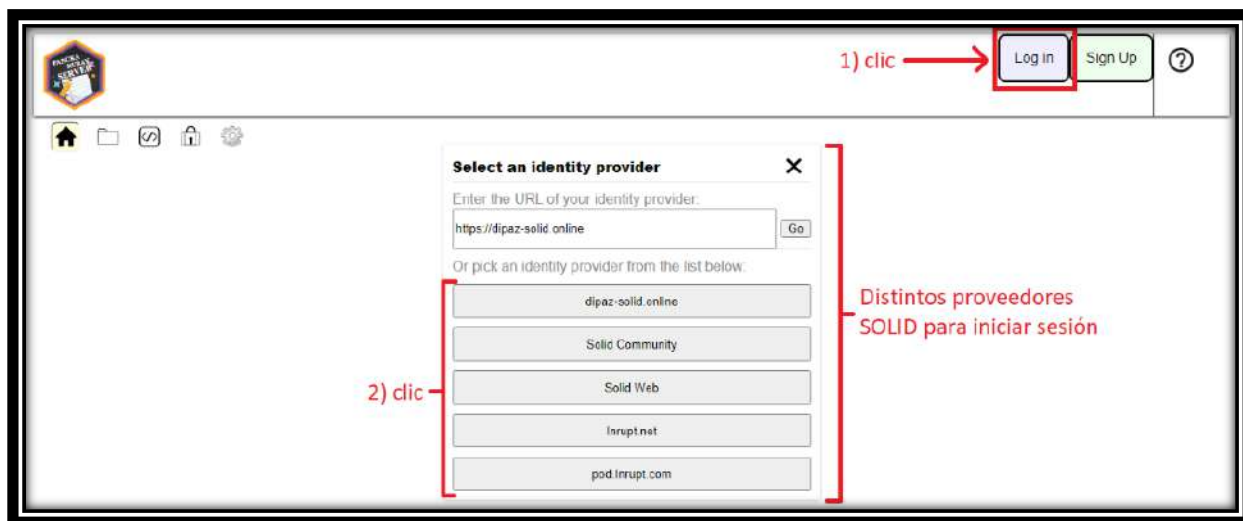
Figura 41.

Formulario de registro con los campos del Pancka Ruray

Nota. Campos mandatorios para el registro de Pancka Ruray

Figura 42.

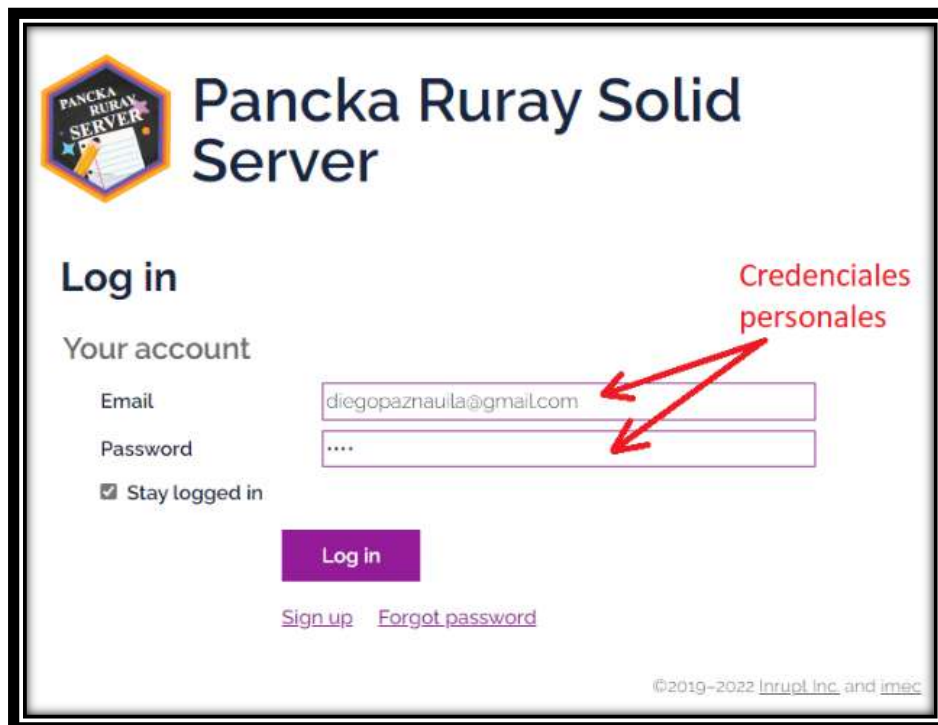
Pasos para iniciar sesión dentro del servidor Pancka Ruray



Nota. Selección de los servidores descentralizado Solid para manejar información dentro del servidor Pancka Ruray

Figura 43.

Ventana de inicio de sesión basado en credenciales por email y contraseña



Nota. Uso del protocolo OIDC para inicio de sesión dentro del servidor Pancka Ruray utilizando la configuración de correo electrónico como credencial de autenticación

Figura 44.

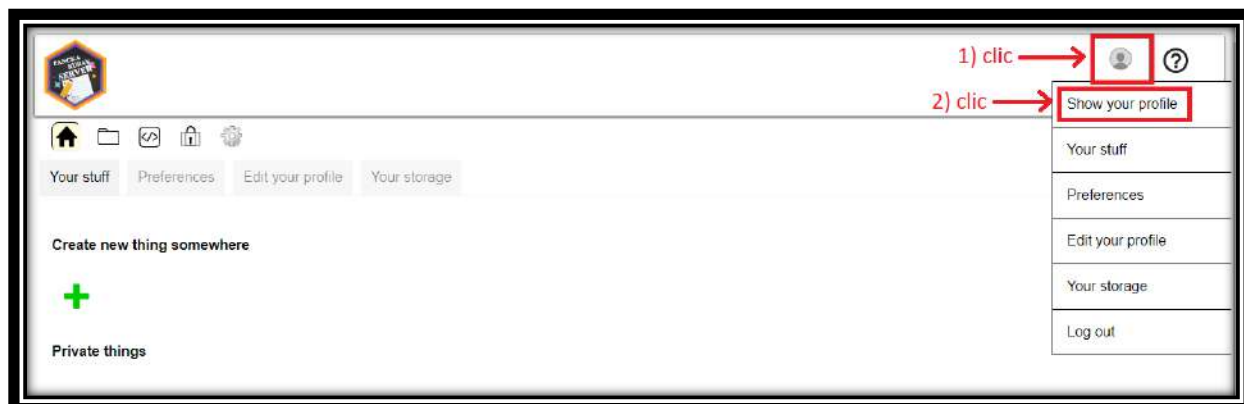
Ventana de token mediante OAuth2



Nota. Ventana de autorización, generado mediante un token OAuth2 para usar servidores Solid dentro de aplicaciones gráficas en este caso Mashlib.

Figura 45.

Mostrar datos del perfil



Nota. Pasos para acceder al perfil donde se almacena la información pública de los datos de los estudiantes, docentes y empleados que utilizan el prototipo Pancka Ruray

Para acceder al VCARD para insertar fotografía, nombre, teléfonos, correo electrónico, fecha de nacimiento, descripción es necesario dar clic en el icono de la figura 46 desplegando el formulario con los datos ingresados que se actualiza al pasar por los distintos campos.

Figura 46.

Acceder a los datos de perfiles mashlib



Nota. El icono tiene una forma de tarjeta de presentación al tener el formulario para datos de VCARD que es una carta de presentación para los estudiantes, docentes y empleados.

El formulario para la información personal puede dejar los campos en blanco, agregando los datos que se reflejan dentro del sistema cliente de Pancka Ruray, permitiendo a los estudiantes tener datos que pueden ser compartidos dentro de aplicaciones sin tener que volverlos a escribir, se puede enviar como un formato general que las aplicaciones entienden similar al caché que se genera basado en las etiquetas HTML dentro de las aplicaciones que piden un correo electrónico pero esta información no se almacena dentro de la PC si no dentro de un servidor descentralizado el cual tiene una mayor seguridad, control y accesos para limitar los datos que uno no desea compartir como puede ser la fecha de nacimiento, no maneja contraseñas pero si información útil al momento de llenar formularios. Con un mayor interés en mejorar la forma de conseguir empleo una descripción personal, formas de contactar, hasta cuales son las habilidades certificadas previamente por docentes y empleadores que trabajan en conjunto con los estudiantes o participado para una mejora curricular.

Figura 47.

Añadir información personal en VCARD con mashlib

The screenshot shows a web interface for adding personal information to a vCard. The interface is divided into several sections:

- Profile Card:** Displays a profile picture and a name field containing "Diego Paz".
- Form Fields:**
 - Name:** Diego Paz
 - Role:** Estudiante
 - Organization name:** ESPE
 - Address:** Work, email: dipaz@espe.edu.ec
 - Phone:** Work, Value: +593980491064
 - Birth:** 24/10/1997
 - Notes:** Fuerzas Armadas ESPE, miembro, miembro voluntario de la rama estudiantil IEEE-ESPE, miembro de club de software ESPE, presidente del capitulo técnico C15 de la rama estudiantil IEEE-ESPE del año 2020 y actualmente Google Developers Students Club (DSC) Lead de la Universidad de las Fuerzas Armadas ESPE.

Red arrows indicate the following actions:

- 1. clic:** Points to a cloud icon in the top navigation bar.
- 2. campos de texto:** Points to the text input fields for Name, Role, and Organization name.
- añadir correo:** Points to the email field.
- añadir telefono:** Points to the phone field.
- fecha de nacimiento:** Points to the birth date field.
- descripción personal:** Points to the notes field.

Nota. Los campos que se agregan dentro de un e-profile de estudiantes con información general

Figura 48.

Volvemos a la ventana inicial del servidor Solid Pancka Ruray



Nota. El paso para poder volver donde se almacenará la información dentro del servidor Pancka Ruray.

Figura 49.

Abrimos los archivos raíz con todos los usuarios registrados

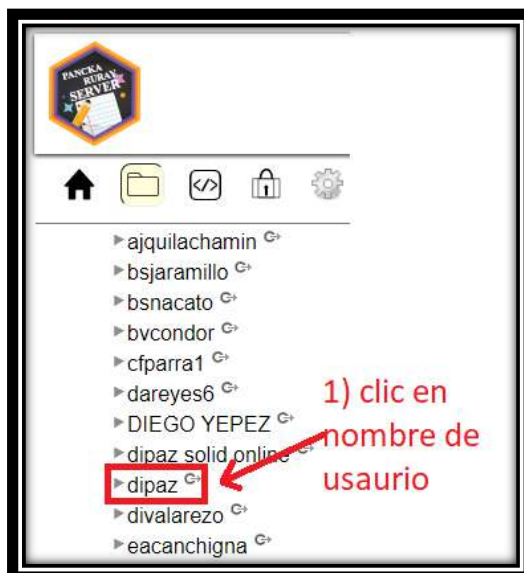


Nota. Para mostrar a todos los POD y archivos de usuarios muestra la carpeta raíz del servidor Pancka Ruray

Dentro de la carpeta raíz buscamos el nombre del POD del estudiante donde el docente o supervisor va a poder navegar agregando o verificando la información después de obtener los permisos necesarios para agregar información como se muestra en la figura 50

Figura 50.

Seleccionamos la carpeta con nombre del POD

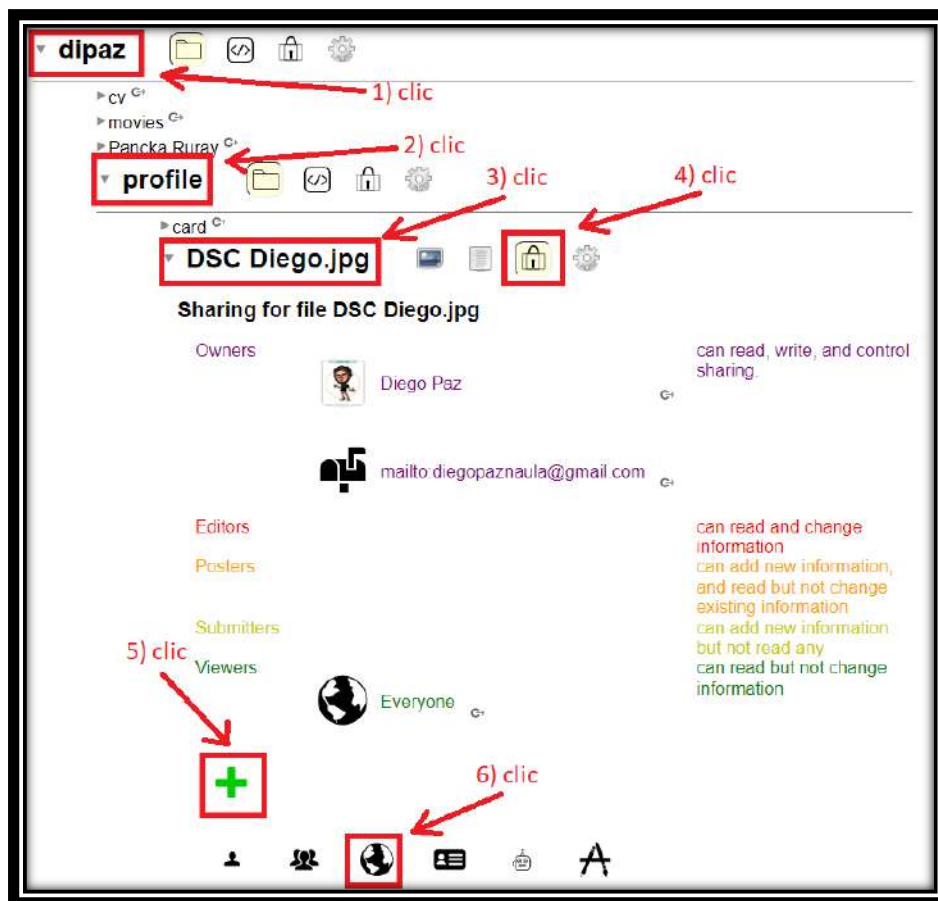


Nota. Buscamos el nombre del POD del estudiante al dar clic en la flecha o el icono para poder ver los archivos y carpetas de los estudiantes, docentes o empleados.

Un estudiante puede tener varias carpetas y ningún e-profile será igual a otro, los POD son los lugares donde se almacena la información, el lugar de registro crea carpetas individuales por aplicación esta información es pública y la llamamos profile como se ve en la figura 51

Figura 51.

Pasos para añadir permisos a todo el mundo para ver la foto de perfil de usuario

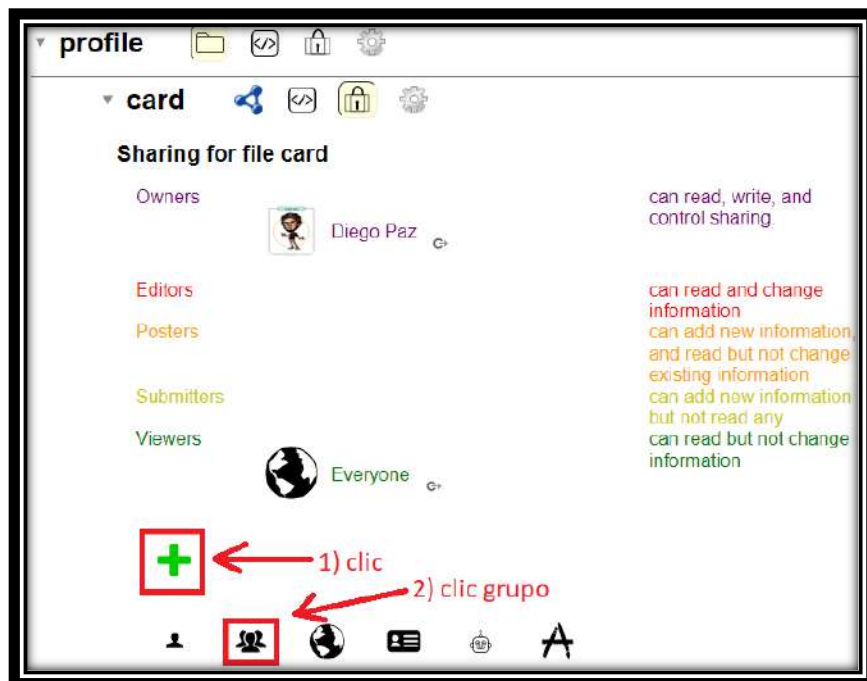


Nota. Los pasos para poder visualizar la información de forma pública recomendado por el autor para no tener que agregar los datos públicos de forma redundante.

El icono del mundo es acceso a todos por lo que debe tener los privilegios más bajos luego contamos con individuos que agrega permisos para un estudiante, docente o empleado en específico, los grupos permite una gestión de forma masiva donde puede agregar información las personas que pertenezca a estos grupos, luego está FOAF para manejar contactos de amigos que pueden agregar información y por último el manejo de aplicaciones donde podemos asignar un dominio específico que pueda gestionar la información como muestra la figura 52.

Figura 52.

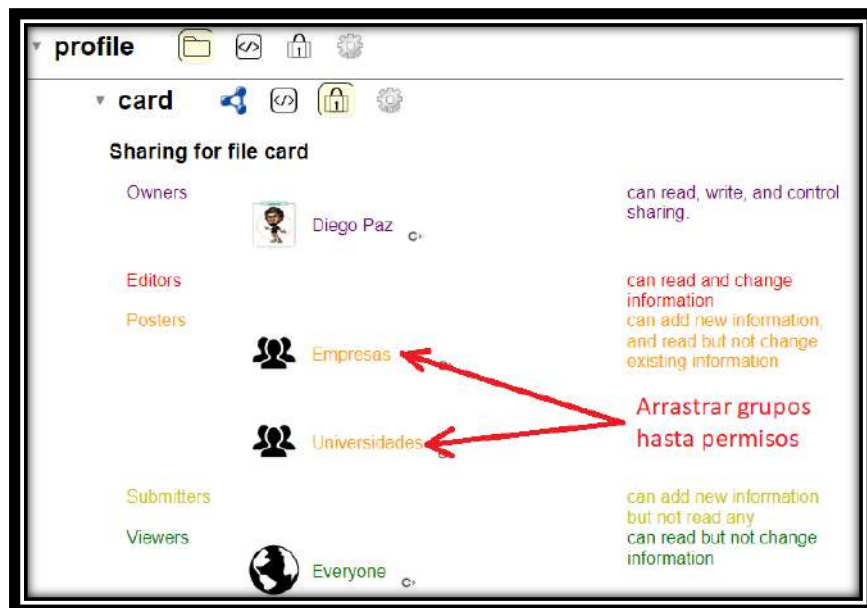
Añadir permisos de un grupo para editar la información del perfil de usuario



Nota. Añadir grupos para publicar nueva información dentro de los perfiles de estudiantes

Figura 53.

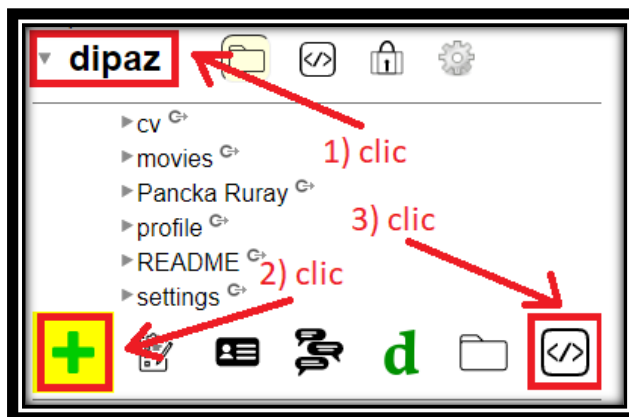
Añadir el permiso específico para editar la información del perfil de usuario



Nota. Los grupos añadidos escala hasta el nivel de poder agregar información, pero no poder editar la información ya existente dentro del e-profile de los estudiantes.

Figura 54.

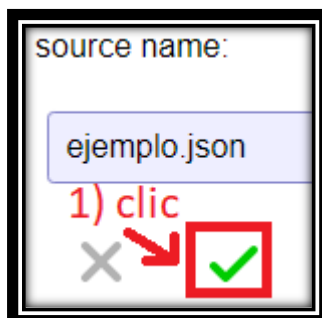
Agregar un archivo nuevo dentro del POD de un estudiante



Nota. Dentro de los POD se puede manejar información mediante distintos tipos de archivo los cuales se pueden agregar siguiente los pasos de la figura 54

Figura 55.

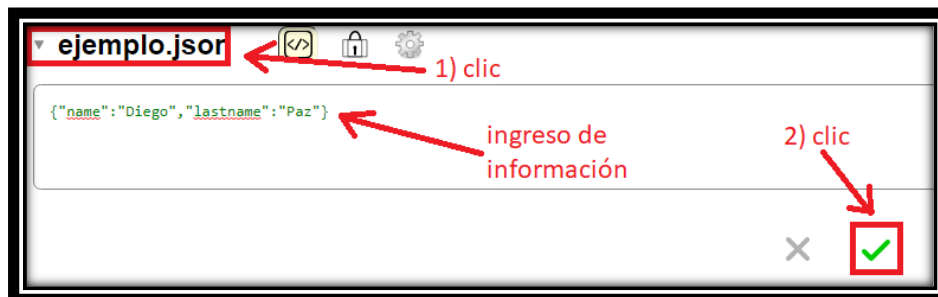
Agregar nombre y extensión del archivo nuevo



Nota. Podemos agregar nombre de archivos con extensiones como HTML, JSON, entre otros para ser manejados mediante páginas web.

Figura 56.

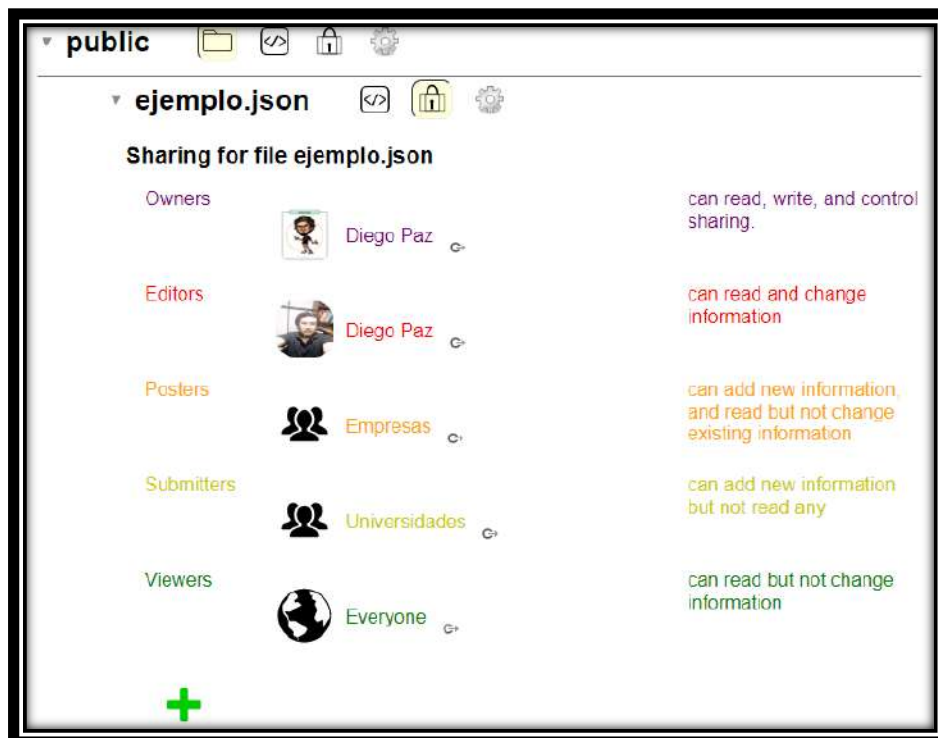
Añadir información dentro de un archivo JSON



Nota. Una vez creado el archivo se puede agregar el texto plano correspondiente el cual tiene una validación antes de guardar en la figura 56 el ejemplo es con una extensión JSON, lo mismo es con las demás extensiones.

Figura 57.

Agregar todos los roles dentro de un archivo



Nota. La tarea es colocar un actor en cada uno de los roles para manejar la información dentro del archivo recién creado.

Figura 58.

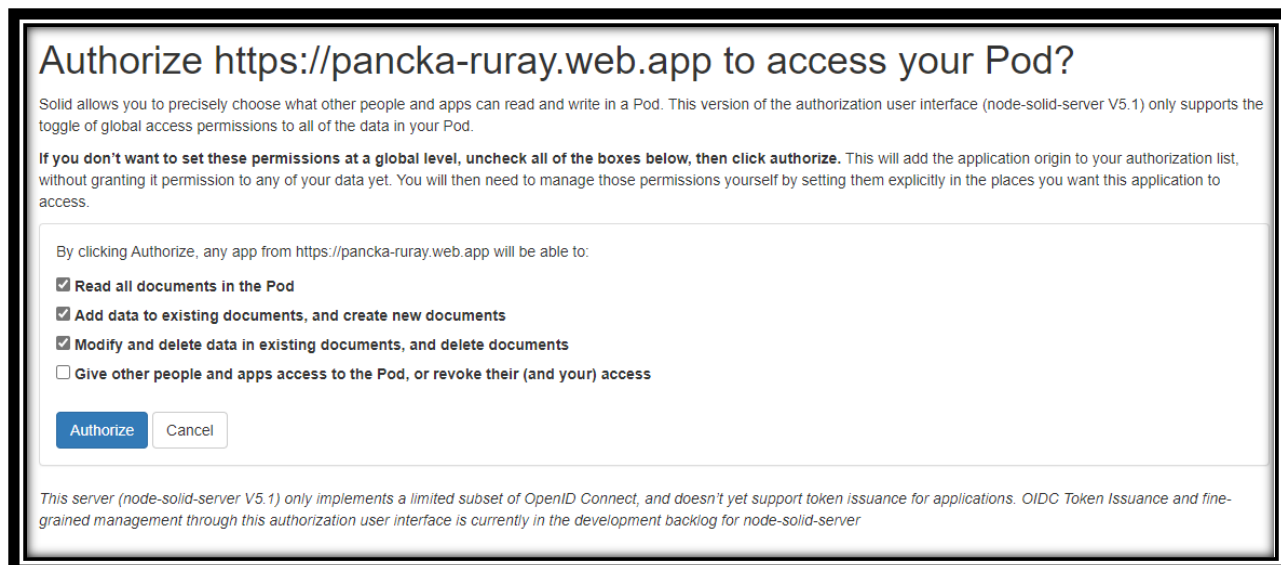
Seleccionar un servidor Solid para inicio de sesión



Nota. El inicio de sesión sobre el cliente que manipula los e-profile de los estudiantes y donde los reclutadores pueden consultar los perfiles de los estudiantes.

Figura 59.

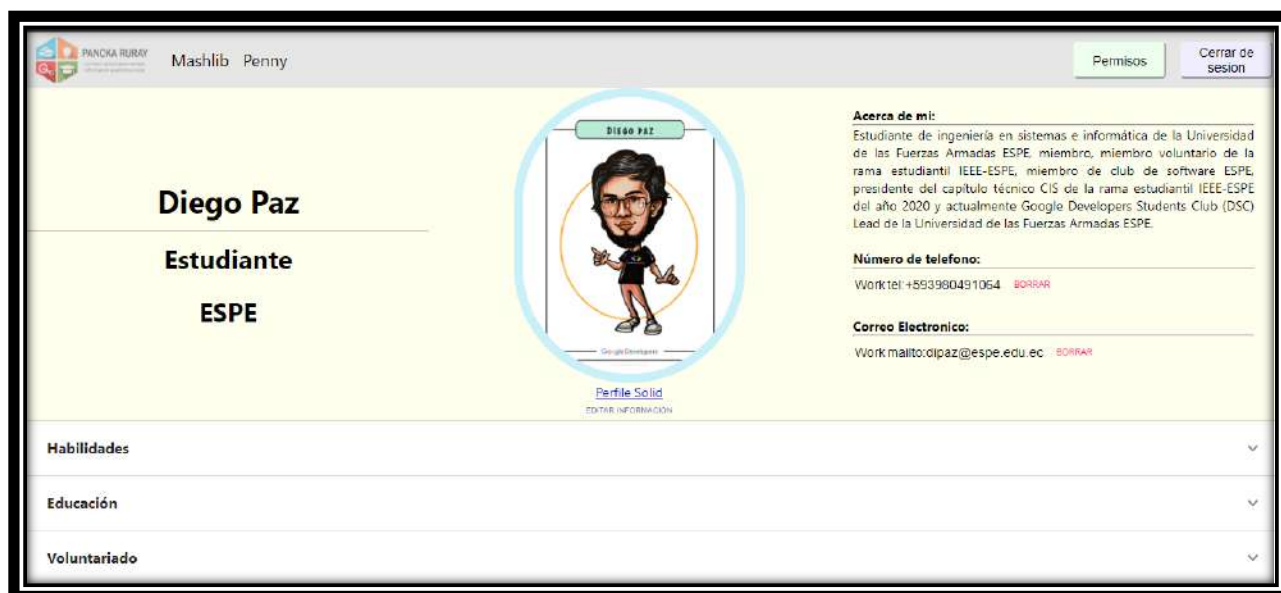
Página con autorización de POD para Pancka Ruray



Nota. El primer inicio de sesión dentro del sistema aparece el nivel de privilegios que se le concede a la aplicación, como lo son Lectura, Añadir, Modificar información y por último si puede manejar los privilegios dentro del almacenamiento Solid.

Figura 60.

Perfil de estudiante en Pancka Ruray



Nota. La primera imagen es del e-profile que cuenta con un menú de navegación para acceder a aplicaciones que gestiona permisos de archivos Solid, cerrar sesión y luego la estructura de un CV digital donde se tiene información personal, habilidades y la educación de los estudiantes, con sus certificados académicos, voluntariado o experiencia laboral.

Figura 61.

Añadir habilidades para certificación de docentes y supervisores

Habilidades Blandas		Habilidades Duras	
Habilidades Blandas	0	Habilidades Duras	0
Liderazgo	80 BORRAR	C++	75 BORRAR
Trabajo en equipo	99 BORRAR	Java	86 BORRAR

Nota. El estudiante puede añadir habilidades las cuales serán certificadas por docentes y supervisores.

Es conveniente que el e-profile puede actualizar, eliminar información que cambia en cortos periodos de tiempo, no como el nombre que es un caso especial, si no la descripción personal, números de teléfono, correos electrónicos, rol y organización.

Figura 62.

Editar información del perfil en Pancka Ruray

Nota. Para editar información dentro del e.profile solo se puede cambiar datos comunes en el tiempo como organización, rol, teléfono y correo electrónico.

Con esto se culmina el taller el cual ayuda a medir el tiempo promedio para tener un dato de cuál es la media para lograr realizar ciertas actividades dentro del prototipo Pancka Ruray y cuál es la opinión de los estudiantes para lograr crear un e-profile.

Plan de pruebas utilizado para evaluar Pancka Ruray

El plan de pruebas cuenta con un resumen de las pruebas que se van a realizar en la tabla 14

Tabla 14.

Cuadro de resumen de pruebas

Componentes para probar:	Módulos del sistema Pancka Ruray <ul style="list-style-type: none"> • Registro de usuarios en servidor Solid • Iniciar sesión en servidor Solid • Agregar información de presentación VCARD • Manejar permisos de acceso, público a VCARD • Agregar dos grupos de control para edición, agregar datos en POD • Crear, actualizar, modificar archivos dentro de un POD • Manejar permisos de acceso para edición, agregar datos en POD • Inicio de sesión en cliente Pancka Ruray • Agregar habilidad de estudiante • Editar información de e-profile
Objetivos de las pruebas	En los módulos se realiza pruebas para validar: <ul style="list-style-type: none"> • La conectividad entre sistemas • Efectuar de manera correcta cada proceso • La usabilidad de los estudiantes • lógica del proceso
Detalle del orden de ejecución de los componentes	Los componentes están ordenados por: <ul style="list-style-type: none"> • Uso del servidor Solid mediante mashlib • Utilizar cliente Pancka Ruray para e-profile
Responsabilidad de la Prueba	La responsabilidad de las pruebas recae sobre el autor que desarrolla e implementa Pancka Ruray

Nota. Tabla con el orden a ejecutar las pruebas, objetivos, los componentes que se pondrán a prueba y el responsable

Entorno y configuración de pruebas

- Servidor Solid
 - Ubuntu 20 con Docker 19.0.312
 - 1 GB de memoria RAM
 - 25 GB de almacenamiento
 - 1000 GB de transferencia de datos
- Cliente Pancka Ruray
 - 10 GB de almacenamiento
 - 10 GB de transferencia de datos

Criterios de aprobación/rechazo

- **Errores graves:** Información ingresada presentada con cambio en los valores, caída de la red descentralizada, irrumpir el funcionamiento entre servidores, falla en permisos de acceso a la información.
- **Errores medios (comunes):** Error al cargar elementos multimedia del aplicativo.
- **Errores leves:** Error de estructura del diseño al mostrar información, dificultad de operar el sistema.

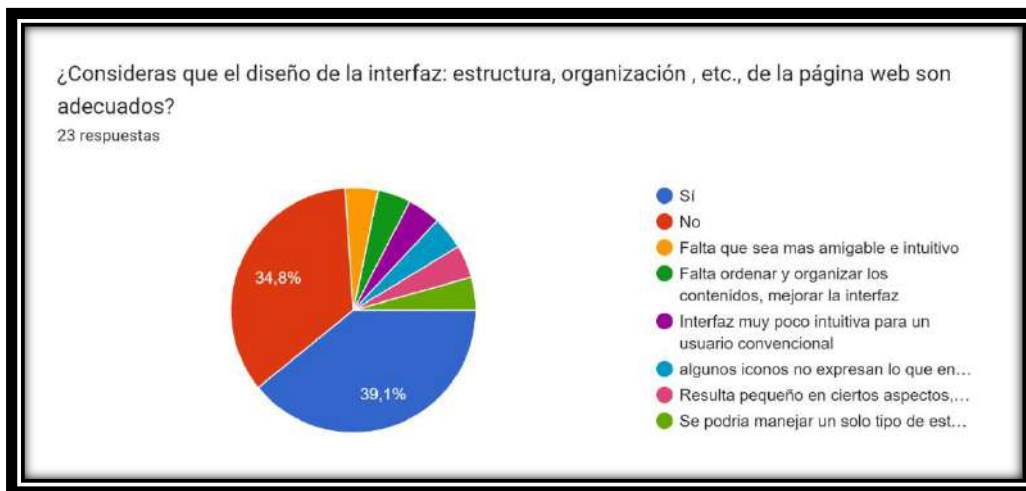
Criterio de aprobación, el proyecto con un 75% o más de las pruebas aprobará basado en la implementación de prototipos de software MOSCA, se cuenta con 10 preguntas por lo que llegar hasta el paso número 8 sin registrar errores cuenta como aprobado el prototipo Pancka Ruray. Caso contrario el rechazo no cumple con el nivel exigido dentro del proyecto rechazando el prototipo como funcional.

Resultados obtenidos en la encuesta de diseño y satisfacción de Pancka Ruray

Dentro de la presente sección estudiantes de los últimos niveles realizaron el taller para registrar habilidades duras y blandas donde se analiza el tiempo y el diseño de la interfaz del servidor Solid como el cliente Pancka ruray

Figura 63.

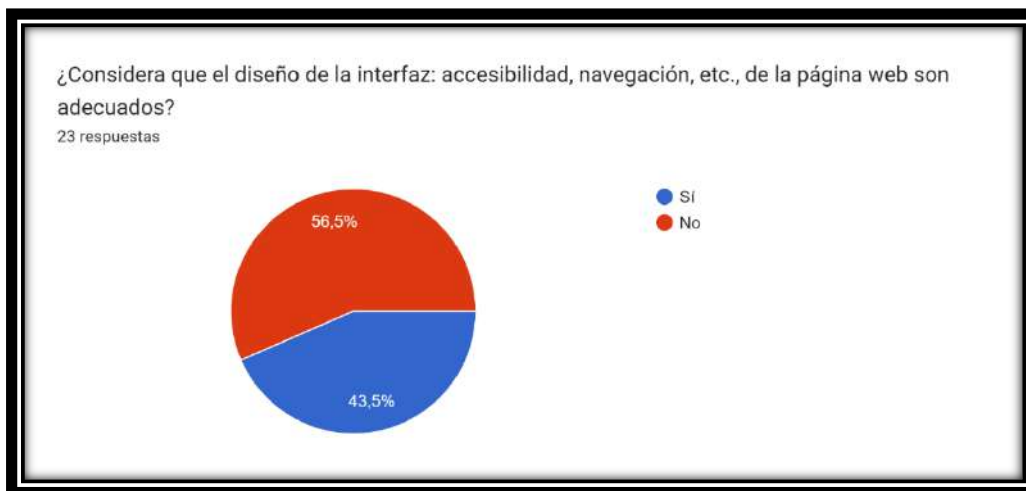
Pregunta 1 de la encuesta sobre la estructura de la interfaz de Pancka Ruray



Nota. El diseño de interfaz desarrollada por el autor no es intuitivo basado en las respuestas de estudiantes.

Figura 64.

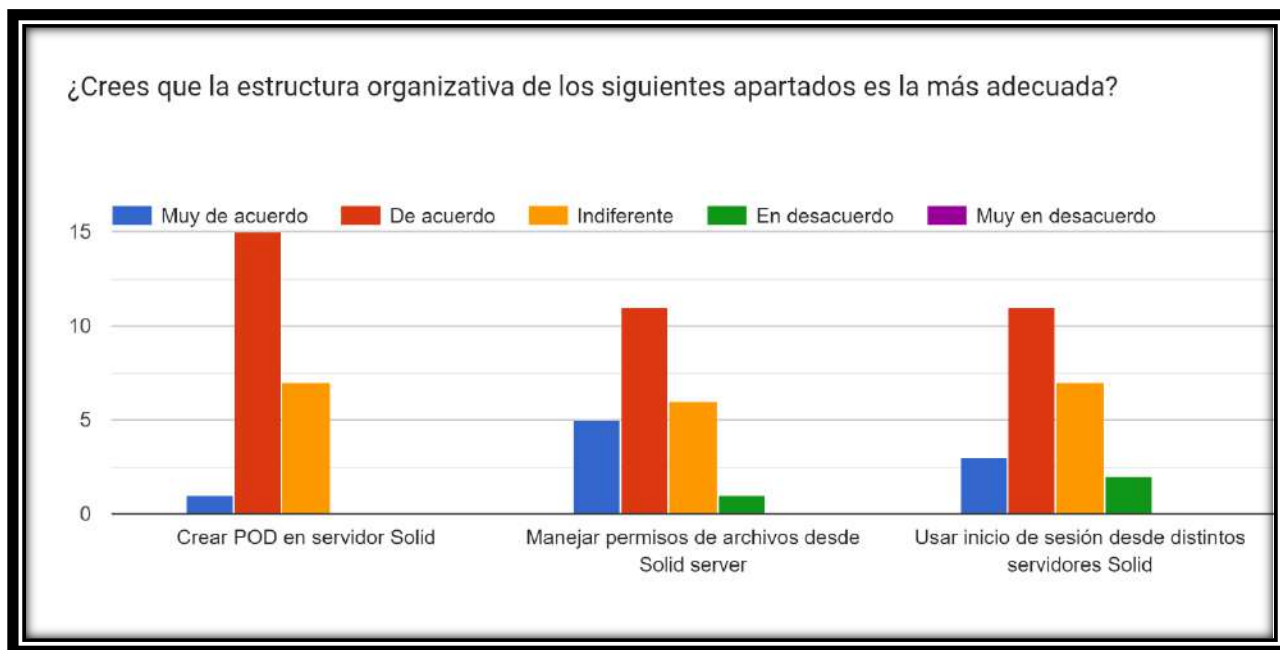
Pregunta 2 de la encuesta sobre diseño, accesibilidad y navegación en Pancka Ruray



Nota. La navegación dentro de la interfaz y el acceso basado en servidores descentralizados Solid ha sido complicado ser utilizada.

Figura 65.

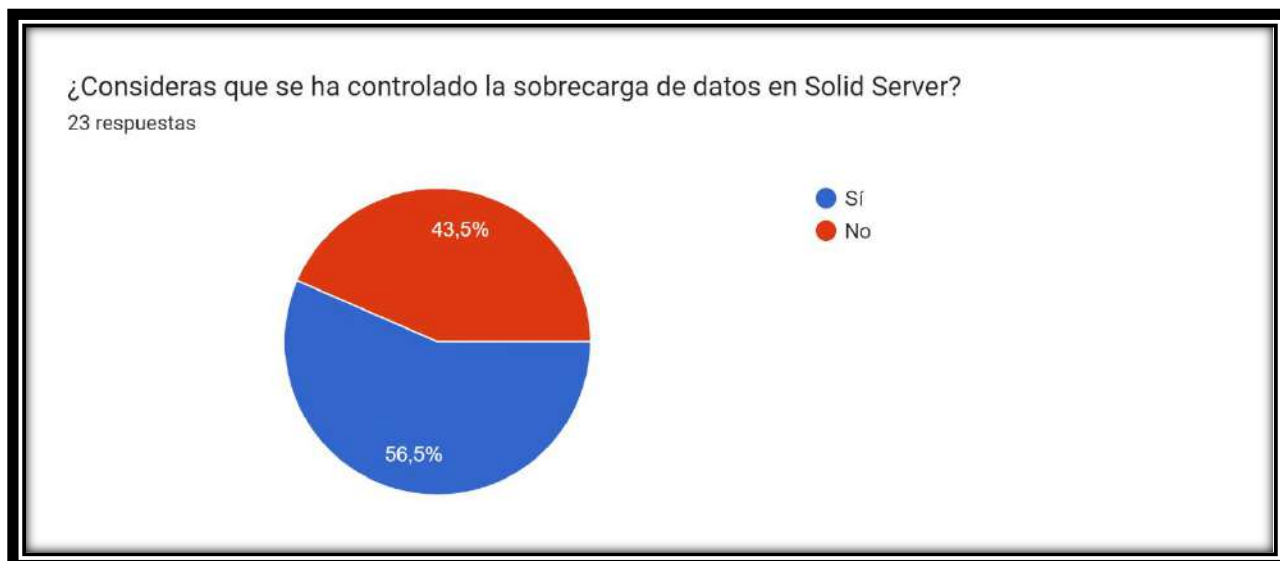
Pregunta 3 de la encuesta sobre la estructura de Pancka Ruray



Nota. Dentro de la página cliente ha permitido mejorar la comprensión de los estudiantes logrando un mayor entendimiento.

Figura 66.

Pregunta 4 de la encuesta sobre la sobrecarga de datos en Pancka Ruray



Nota. Tomando en cuenta de que los parámetros del servidor Solid son los más económicos y el cliente Pancka Ruray esta almacena de la forma gratuita se considera los resultados favorables al tener más de la mitad de los estudiantes con el sí.

Figura 67.

Pregunta 5 de la encuesta sobre los elementos multimedia utilizados en Pancka Ruray



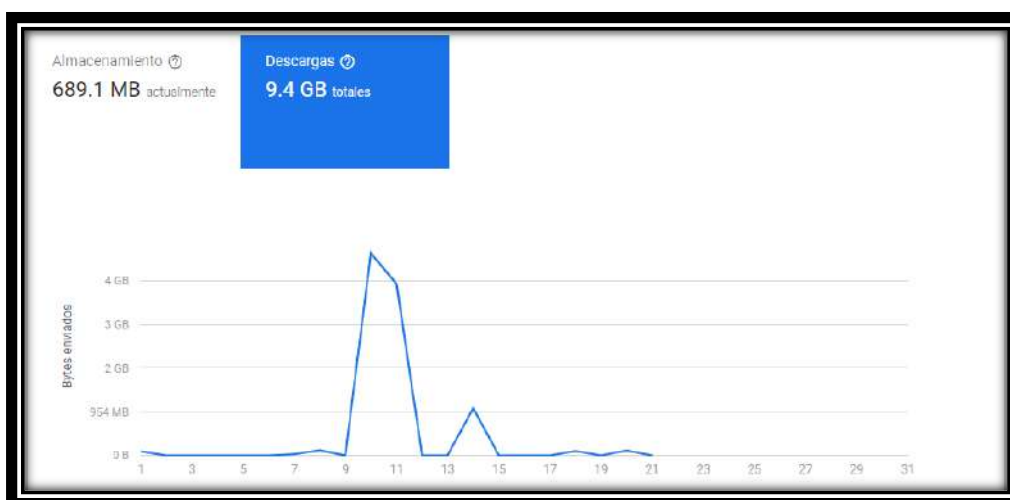
Nota. El valor gráfico de solo iconos a no es favorable, puede ser el causante de los demás factores, como la dificultad para navegar, poca conductividad del uso del servidor Solid.

Análisis de tráfico del cliente Pancka Ruray

Utilizamos la gráfica gracias a la incorporación de analytics que permite de forma visual entender el consumo de la aplicación

Figura 68.

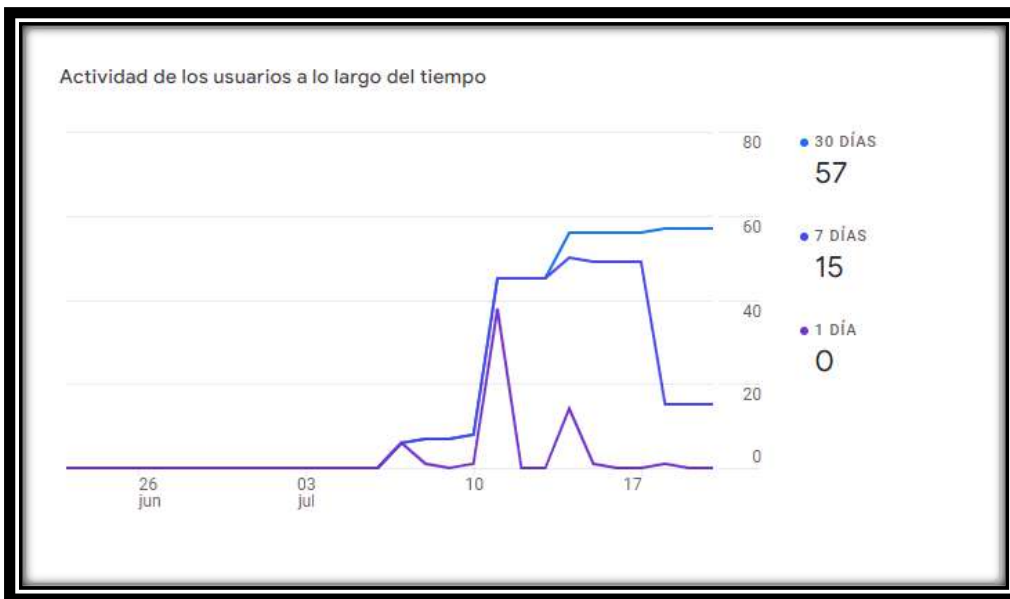
Bytes de descarga de componentes multimedia del cliente Pancka Ruray



Nota. En la gráfica cuenta con dos grandes picos los cuales se realizó el primer día de la prueba con más de 4 GB de tráfico de la red, pero el segundo día se redujo en la tercera parte con la misma cantidad de estudiantes ya que el cliente usa PWA almacenando datos en caché.

Figura 69.

Total, de actividades por días de usuarios en cliente Pancka Ruray



Nota. La gráfica tiene a los usuarios en el eje x mientras que por día en los y logrando un crecimiento a largo plazo contando con un crecimiento progresivo.

Figura 70.

Tiempo promedio del uso para crear un e-profile por parte de estudiantes



Nota. El uso promedió tiene tres picos el primero es de visitantes sin cuentas o reclutadores que toma 25 segundos para revisar e-profile de estudiantes, el segundo y tercer pico de estudiantes que ingresan a sus perfiles para agregar información sobre sus habilidades.

Figura 71.

Total, de eventos dentro del cliente Pancka Ruray

Número de eventos por Nombre del evento		
NOMBRE DEL EVENTO	NÚMERO DE EVENTOS	
page_view	364	-
user_engagement	323	-
session_start	67	-
first_visit	57	-

Nota. Se cuenta con 4 disparadores el primero es total de visitas a la página principal, y el total de usuario que regresan al prototipo de e-profile, el total de personas que inician sesión y por último personas que inician sesión para editar la información.

Registro del resultado de componentes y análisis de los tiempos dentro del proceso del taller del uso de Pancka Ruray

En el siguiente apartado se evalúa los 10 procesos del taller registrado en el plan de pruebas desarrollado para los estudiantes.

Registro de usuarios en servidor Solid

El registro de cualquier actor que busca tener una interacción para el registro de actividades extracurriculares mediante un servidor descentralizado llamado Solid los componentes para la prueba están registrados en la tabla 15

Tabla 15.

Servidor Solid Pancka Ruray. Registro de un usuario

Objetivos de la prueba	Validar el microservicio para el registro de POD de actores
Técnicas	Insertar los datos necesarios para el registro del servidor con datos reales, para validar los campos del nombre del POD, correo electrónico o contraseña.

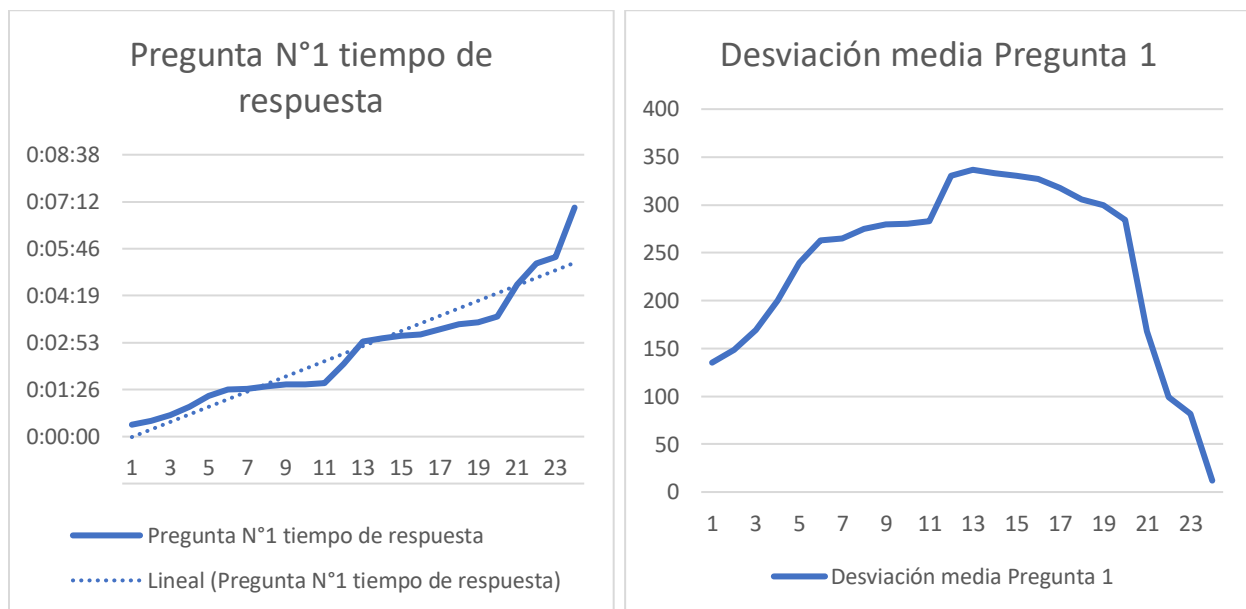
Código involucrado	<pre>describe('A CreateCredentialsHandler', (): void => { let operation: Operation; let body: CredentialsHandlerBody; let accountStore: jest.Mocked<AccountStore>; let credentialStorage: jest.Mocked<KeyValueStorage<string, ClientCredentials>>; let handler: CreateCredentialsHandler; beforeEach(async(): Promise<void> => { operation = { method: 'POST', body: new BasicRepresentation(), target: { path: 'http://example.com/foo' }, preferences: {}, }; body = { email: 'test@example.com', webId: 'http://example.com/foo#me', name: 'token', }; }); });</pre>
Caso de prueba	<p>Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,</p> <p>Objetivo: Ratificar si los campos requeridos cuentan con las seguridades para el tipo de dato, puede registrar un POD usando los protocolos de autenticación correo electrónico y contraseña</p> <p style="text-align: center;">Caso N° 1</p> <p>Descripción: Datos correctos, ingresando datos verificados.</p> <p>Entradas: Nombre del POD, correo electrónico, contraseña y confirmación de contraseña</p> <p>Salida esperada: Muestra la página principal donde muestra la ruta del e-profile</p>
Resultados	El caso de prueba se ha ejecutado, rectificando todos los defectos obtenidos en los casos de estudio.
Observaciones	N/A

Nota. En la tabla 15 prueba de componente para el registro se añade información utilizada por el autor del proyecto comprueba los microservicios de Pancka Ruray.

La figura 72 es un análisis gráfico de los resultados esperados al medir el tiempo que le toma a un estudiante promedio realizar el proceso de registro de usuario en un servidor Solid.

Figura 72.

Pregunta N°1 análisis tiempo de respuesta



Nota. La figura 72 cuenta con el total del tiempo para realizar la creación de un POD dentro de un servidor Solid y la segunda para analizar si el tiempo puede tener una correlación.

Iniciar sesión en servidor Solid

El inicio de sesión dentro de cualquier servidor Solid es la forma de mantener la privacidad de datos identificando al propietario de la información descentralizada los componentes para la prueba están registrados en la tabla 16

Tabla 16.

Servidor Solid Pancka Ruray. Inicio de sesión de un usuario

Objetivos de la prueba	Verificar el microservicio para el inicio de sesión dentro del servidor Solid
Técnicas	Insertar los datos dependiendo la configuración de autenticación en Pancka Ruray es correo electrónico y contraseña.

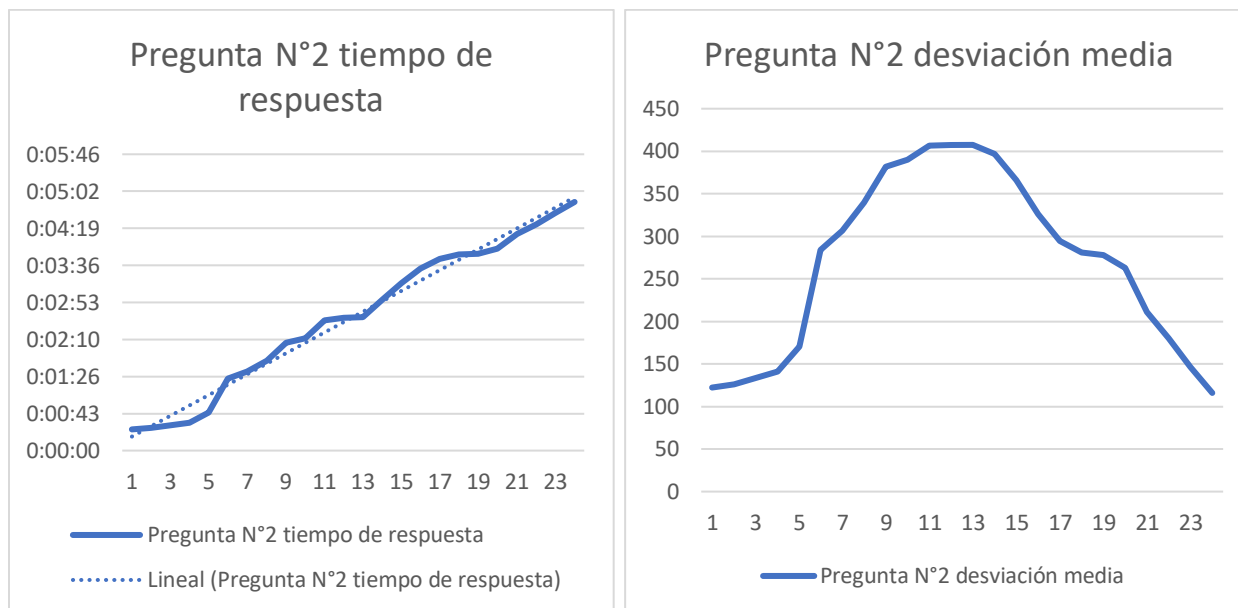
Código involucrado	<pre>describe('An EmailPasswordAuthorizer', (): void => { const email = 'test@example.com'; const password = 'super_secret'; const webId = 'http://example.com/profile#me'; let operation: Operation; let response: Representation; let accountStore: jest.Mocked<AccountStore>; let source: jest.Mocked<CredentialsHandler>; let handler: EmailPasswordAuthorizer; beforeEach(async(): Promise<void> => { operation = { method: 'POST', body: new BasicRepresentation(JSON.stringify({ email, password })), APPLICATION_JSON), target: { path: 'http://example.com/foo' }, preferences: {}, }; }); }</pre>						
Caso de prueba	<table border="1"> <tr> <td data-bbox="570 615 1398 751">Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,</td> </tr> <tr> <td data-bbox="570 751 1398 961">Objetivo: Comprobar la forma de autenticar al propietario del POD dentro del servidor Solid mediante la configuración correo electrónico y contraseña.</td> </tr> <tr> <td data-bbox="570 961 1398 1035" style="text-align: center;">Caso N° 1</td> </tr> <tr> <td data-bbox="570 1035 1398 1108">Descripción: Datos correctos, ingresando datos verificados.</td> </tr> <tr> <td data-bbox="570 1108 1398 1182">Entradas: Correo electrónico y contraseña.</td> </tr> <tr> <td data-bbox="570 1182 1398 1308">Salida esperada: Página de inicio con los permisos del usuario iniciado sesión dentro del POD de un servidor Solid</td> </tr> </table>	Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,	Objetivo: Comprobar la forma de autenticar al propietario del POD dentro del servidor Solid mediante la configuración correo electrónico y contraseña.	Caso N° 1	Descripción: Datos correctos, ingresando datos verificados.	Entradas: Correo electrónico y contraseña.	Salida esperada: Página de inicio con los permisos del usuario iniciado sesión dentro del POD de un servidor Solid
Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,							
Objetivo: Comprobar la forma de autenticar al propietario del POD dentro del servidor Solid mediante la configuración correo electrónico y contraseña.							
Caso N° 1							
Descripción: Datos correctos, ingresando datos verificados.							
Entradas: Correo electrónico y contraseña.							
Salida esperada: Página de inicio con los permisos del usuario iniciado sesión dentro del POD de un servidor Solid							
Resultados	El caso de prueba se ha ejecutado, rectificando todos los defectos obtenidos en los casos de estudio.						
Observaciones	N/A						

Nota. En la tabla 16 prueba de componente para inicio de sesión se añade información utilizada por el autor del proyecto y comprueba los microservicios de Pancka Ruray.

La figura 73 es un análisis gráfico de los resultados esperados al medir el tiempo que le toma a un estudiante promedio iniciar sesión dentro de un servidor Solid. Mediante una ilustración en orden ascendente del tiempo utilizado con una línea para distinguir si existe un error en los datos para obtener la desviación media.

Figura 73.

Pregunta N°2 análisis tiempo de respuesta



Nota. La figura 73 cuenta con el total del tiempo para iniciar sesión en un POD dentro de un servidor Solid y la segunda para analizar si el tiempo puede tener una correlación

Agregar información de presentación VCARD

Agregar información personal dentro perfil generado automáticamente en el servidor de almacenamiento descentralizados similar al registro de un nuevo usuario dentro de una red social, agregando imagen, nombre, fecha de nacimiento, rol entre los tres actores estudiante, docente, supervisor/reclutador con la diferencia que es ficha puede ser utilizado dentro de web semánticas los componentes para la prueba están registrados en la tabla 17

Tabla 17.

Servidor Solid Pancka Ruray. Ingreso de información de un usuario

Objetivos de la prueba	Probar el microservicio para añadir información personal dentro del servidor Solid
Técnicas	Ingresar nombre, rol, fecha de nacimiento, descripción del usuario, dirección, correo electrónico y número de teléfono de contacto

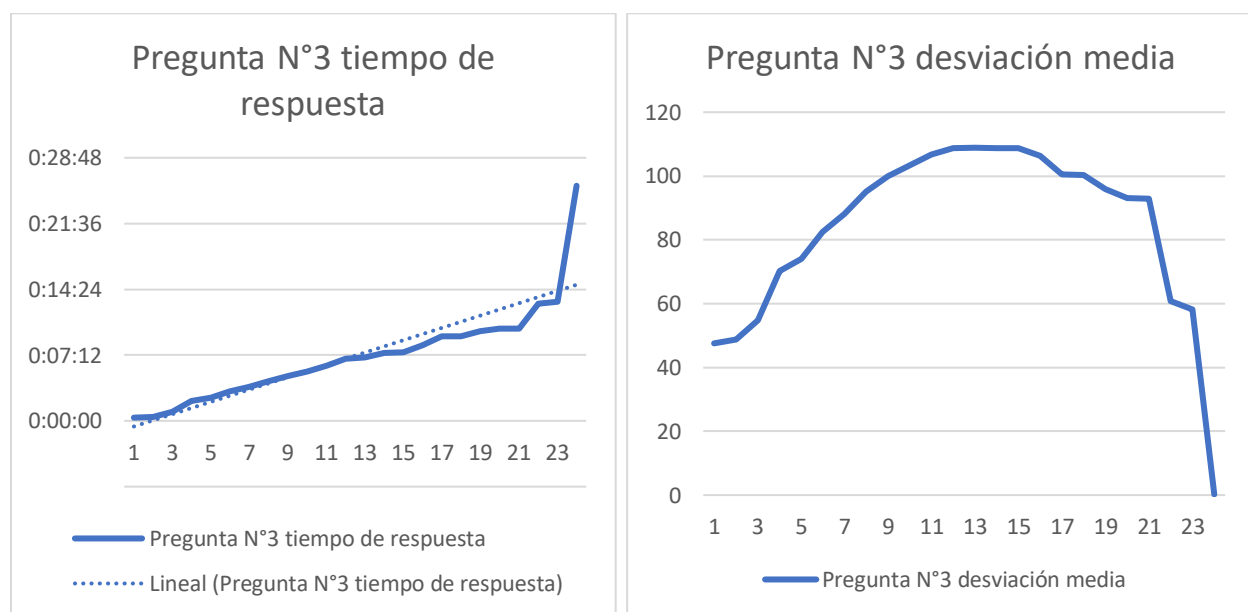
Código involucrado	<pre> section className="header"> <section className="header-left-section"> <h1><Text property={VCARD.fn}></h1> <hr /> <h4><Text property={VCARD.role}></h4> <h4><Text property={VCARD.organization_name.iri.value} edit={editing} auto </section> <section className="header-middle-section"> <Image property={VCARD.hasPhoto.iri.value} className="header-middle-sectid Perfile Solid <Button size="small" color="primary" onClick={() => setEditing(!editing)}> Editar Información </Button> </section> <section className="header-rigth-section"> Acerca de mi: <hr /> <p><Text property={VCARD.note} className='about-me' /></p>
 Número de telefono: <hr /> <ContentData property={VCARD.hasTelephone} edit={editing}></ContentData>
 </pre>						
Casos de prueba	<table border="1"> <tr> <td data-bbox="570 821 1404 961"> Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest, </td> </tr> <tr> <td data-bbox="570 961 1404 1171"> Objetivo: Validar los tipos de datos ingresados como número de teléfono, correo electrónico, fecha de nacimiento, imagen de perfil </td> </tr> <tr> <td data-bbox="570 1171 1404 1241" style="text-align: center;"> Caso N° 1 </td> </tr> <tr> <td data-bbox="570 1241 1404 1310"> Descripción: Datos correctos, ingresando datos verificados. </td> </tr> <tr> <td data-bbox="570 1310 1404 1520"> Entradas: Imagen de perfil, nombre, rol, nombre de la organización, correos electrónicos, teléfonos, fecha de nacimiento y descripción </td> </tr> <tr> <td data-bbox="570 1520 1404 1650"> Salida esperada: Permitir el registro de los campos de los datos </td> </tr> </table>	Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,	Objetivo: Validar los tipos de datos ingresados como número de teléfono, correo electrónico, fecha de nacimiento, imagen de perfil	Caso N° 1	Descripción: Datos correctos, ingresando datos verificados.	Entradas: Imagen de perfil, nombre, rol, nombre de la organización, correos electrónicos, teléfonos, fecha de nacimiento y descripción	Salida esperada: Permitir el registro de los campos de los datos
Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,							
Objetivo: Validar los tipos de datos ingresados como número de teléfono, correo electrónico, fecha de nacimiento, imagen de perfil							
Caso N° 1							
Descripción: Datos correctos, ingresando datos verificados.							
Entradas: Imagen de perfil, nombre, rol, nombre de la organización, correos electrónicos, teléfonos, fecha de nacimiento y descripción							
Salida esperada: Permitir el registro de los campos de los datos							
Resultados	El caso de prueba se ha ejecutado, rectificando todos los defectos obtenidos en los casos de estudio.						
Observaciones	N/A						

Nota. En la tabla 17 prueba de componente para agregar información en VCARD, los campos que utiliza el autor del proyecto comprueban los microservicios de Pancka Ruray.

La figura 74 es un análisis de los datos que muestra un gran incremento del tiempo promedio por los estudiantes existiendo una diferencia de casi el doble del tiempo pasando de 14 minutos aproximadamente a 28 minutos para realizar el registro de información donde demuestra una clara dificultad en esta actividad

Figura 74.

Pregunta N°3 análisis tiempo de respuesta



Nota. La figura 74 cuenta con el total del tiempo para agregar la información de un estudiante dentro de un POD en el servidor Solid y la segunda para analizar si el tiempo puede tener una correlación con los estudiantes.

Manejar permisos de acceso, público a VCARD

El VCARD al ser utilizado como una tarjeta de presentación debe ser algo público que maneja información del estudiante como nombre, institución educativa, descripción, foto de perfil, fecha de nacimiento, luego de esos datos permitir que los reclutadores vean las habilidades con las que cuenta un estudiante es lógico que ellos busquen un acceso sin registro, facilitando ver los datos de varios prospectos para posibles plazas de trabajo, los componentes para la prueba están registrados en la tabla 18

Tabla 18.

Servidor Solid Pancka Ruray. Manejo de permisos de acceso

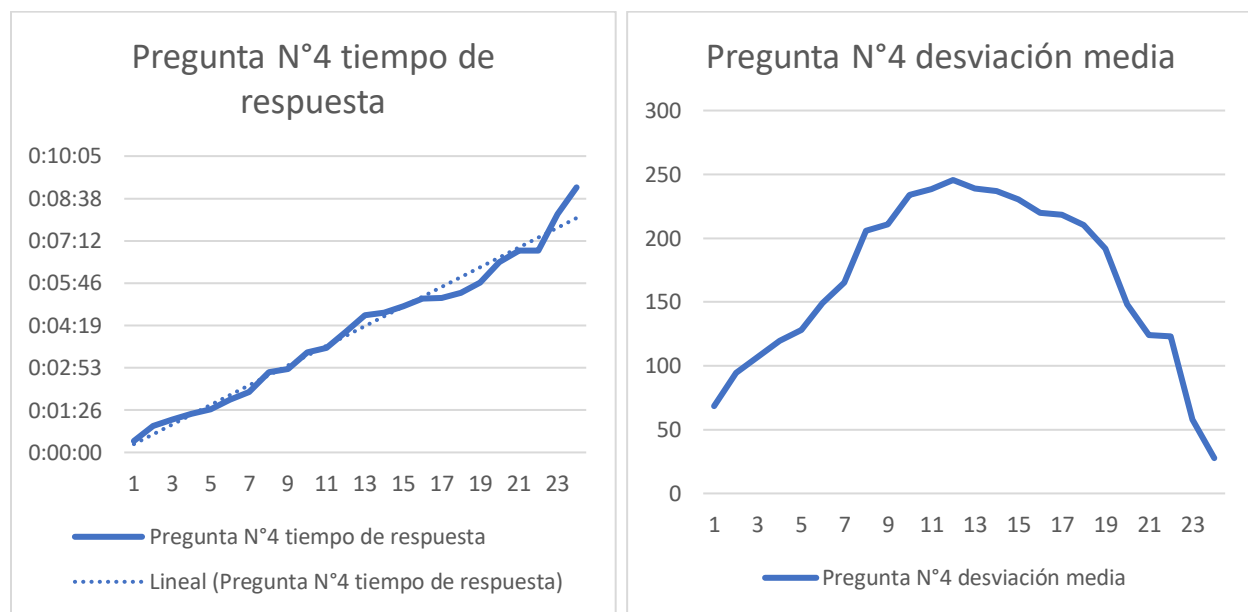
Objetivos de la prueba	Constatar el microservicio del manejo para permisos de acceso dentro del servidor Solid						
Técnicas	Incorporar todo el mundo para poder ver el e-profile sin tener acceso a agregar ni actualizar información mucho menos eliminarla.						
Código involucrado	<pre>it('allows access if the permissions are matched by the reader output.', async(): Promise<void> => { input.modes = new Set([AccessMode.read, AccessMode.write]); input.permissionSet = { [CredentialGroup.public]: { read: true, write: false }, [CredentialGroup.agent]: { write: true }, }; await expect(authorizer.handle(input)).resolves.toBeUndefined(); });</pre>						
Casos de prueba	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td data-bbox="570 867 1404 1003">Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,</td> </tr> <tr> <td data-bbox="570 1003 1404 1209">Objetivo: Permitir que todos los usuarios con y sin cuentas pueda leer la información pública como nombre, rol, descripción, formas de contacto y foto de e-profile</td> </tr> <tr> <td data-bbox="570 1209 1404 1283" style="text-align: center;">Caso N° 1</td> </tr> <tr> <td data-bbox="570 1283 1404 1356">Descripción: Datos correctos, ingresando datos verificados.</td> </tr> <tr> <td data-bbox="570 1356 1404 1493">Entradas: Especificación de etiquetas con permisos para compartir la información</td> </tr> <tr> <td data-bbox="570 1493 1404 1623">Salida esperada: Acceso a la información para personal autorizado</td> </tr> </table>	Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,	Objetivo: Permitir que todos los usuarios con y sin cuentas pueda leer la información pública como nombre, rol, descripción, formas de contacto y foto de e-profile	Caso N° 1	Descripción: Datos correctos, ingresando datos verificados.	Entradas: Especificación de etiquetas con permisos para compartir la información	Salida esperada: Acceso a la información para personal autorizado
Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,							
Objetivo: Permitir que todos los usuarios con y sin cuentas pueda leer la información pública como nombre, rol, descripción, formas de contacto y foto de e-profile							
Caso N° 1							
Descripción: Datos correctos, ingresando datos verificados.							
Entradas: Especificación de etiquetas con permisos para compartir la información							
Salida esperada: Acceso a la información para personal autorizado							
Resultados	El caso de prueba se ha ejecutado, rectificando todos los defectos obtenidos en los casos de estudio.						
Observaciones	N/A						

Nota. En la tabla 18 prueba de componente para agregar permisos de acceso en VCARD, los campos que utiliza el autor comprueban los microservicios de Pancka Ruray.

La figura 75 es muy constante del tiempo teniendo un crecimiento lineal con picos muy cortos permitiendo deducir que existe una relación entre el tiempo y el aprendizaje al tener una forma muy similar a la campana de Gauss, los estudiantes obteniendo como resultado un tiempo promedio de 5 minutos aproximado para realizar esta tarea de forma intuitiva sin la capacitación de un conocedor del sistema.

Figura 75.

Pregunta N°4 análisis tiempo de respuesta



Nota. La figura 75 cuenta con el total del tiempo para editar los permisos de acceso a la información de un estudiante de un POD dentro de un servidor Solid y la segunda para analizar si el tiempo puede tener una correlación con los estudiantes.

Agregar dos grupos de control para edición, agregar datos en POD

Para poder agregar la verificación de datos de los e-profile que cuentan con restricciones de acceso, es necesario manejar mediante grupos los cuales contiene a los actores respectivos brindándoles privilegios para añadir información y por ello los componentes para la prueba están registrados en la tabla 19

Tabla 19.

Servidor Solid Pancka Ruray. Agregar grupos de control dentro de los accesos

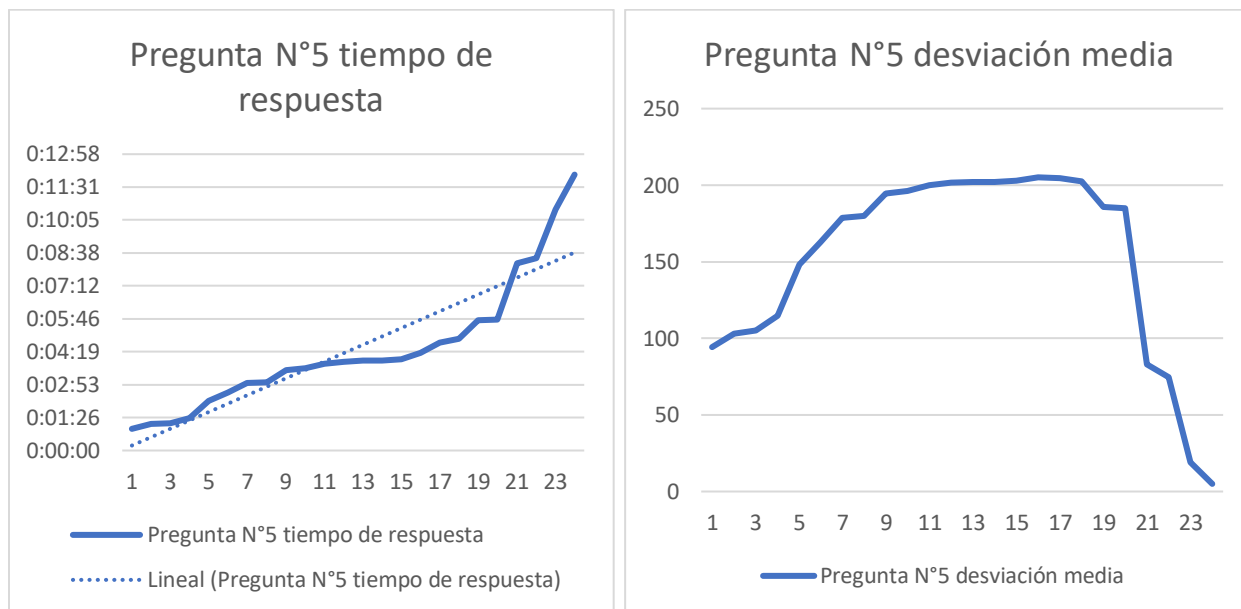
Objetivos de la prueba	Confirmar el microservicio para agregar grupos de control en los accesos dentro del servidor Solid						
Técnicas	Colocar la URI donde está el grupo desde un e-profile genérico que gestiona los grupos de empresas y universidades						
Código involucrado	<pre>describe('An AgentGroupAccessChecker', () : void => { const webId = 'http://test.com/alice/profile/card#me'; const groupId = 'http://test.com/group'; const acl = new Store(); acl.addQuad(namedNode('groupMatch'), ACL.terms.agentGroup, namedNode(groupId)); acl.addQuad(namedNode('noMatch'), ACL.terms.agentGroup, namedNode('badGroup')); let fetchMock: jest.SpyInstance; let representation: Representation; let cache: ExpiringStorage<string, Promise<Store>>; let checker: AgentGroupAccessChecker; beforeEach(async(): Promise<void> => { const groupQuads = [quad(namedNode(groupId), VCARD.terms.hasMember, namedNode(webId))]; representation = new BasicRepresentation(groupQuads, INTERNAL_QUADS, false); fetchMock = jest.spyOn(fetchUtil, 'fetchDataset'); fetchMock.mockResolvedValue(representation); fetchMock.mockClear(); }); });</pre>						
Casos de prueba	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td data-bbox="574 888 1409 1024">Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,</td> </tr> <tr> <td data-bbox="574 1024 1409 1234">Objetivo: Gestionar el ingreso de información por terceros dentro de e-profile de los estudiantes, sin tener privilegios de modificar o eliminar información.</td> </tr> <tr> <td data-bbox="574 1234 1409 1308" style="text-align: center;">Caso N° 1</td> </tr> <tr> <td data-bbox="574 1308 1409 1381">Descripción: Datos correctos, ingresando datos verificados.</td> </tr> <tr> <td data-bbox="574 1381 1409 1455">Entradas: URI del grupo dentro de un libro de direcciones</td> </tr> <tr> <td data-bbox="574 1455 1409 1581">Salida esperada: Confirmación de agregar permisos dentro de una carpeta o archivo</td> </tr> </table>	Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,	Objetivo: Gestionar el ingreso de información por terceros dentro de e-profile de los estudiantes, sin tener privilegios de modificar o eliminar información.	Caso N° 1	Descripción: Datos correctos, ingresando datos verificados.	Entradas: URI del grupo dentro de un libro de direcciones	Salida esperada: Confirmación de agregar permisos dentro de una carpeta o archivo
Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,							
Objetivo: Gestionar el ingreso de información por terceros dentro de e-profile de los estudiantes, sin tener privilegios de modificar o eliminar información.							
Caso N° 1							
Descripción: Datos correctos, ingresando datos verificados.							
Entradas: URI del grupo dentro de un libro de direcciones							
Salida esperada: Confirmación de agregar permisos dentro de una carpeta o archivo							
Resultados	El caso de prueba se ha ejecutado, rectificando todos los defectos obtenidos en los casos de estudio.						
Observaciones	N/A						

Nota. En la tabla 19 prueba de componente para agregar grupos dentro de los permisos de acceso en VCARD, los campos que utiliza el autor del proyecto comprueban los microservicios de Pancka Ruray.

La figura 76 cuenta con grandes desfases en la gráfica lineal mostrando que los estudiantes promedios tienen dificultades al encontrar los archivos POD donde se almacena la información y poder localizar la URI de los grupos pertenecientes a las universidades y empresas.

Figura 76.

Pregunta N°5 análisis tiempo de respuesta



Nota. La figura 76 cuenta con el total del tiempo para agregar dos grupos con permisos para añadir información dentro del e-profile de un estudiante en un servidor Solid y la segunda para analizar si el tiempo puede tener una correlación con los estudiantes.

Crear, actualizar, modificar archivos dentro de un POD

La gestión de archivos es importante al necesitar crear un lugar donde poder subir información como certificados externos, validados de forma digital o incorporar datos descargados por aplicaciones de terceros que deseamos dentro de nuestro e-profile, los componentes para la prueba están registrados en la tabla 20

Tabla 20.

Servidor Solid Pancka Ruray. Gestión de archivos en POD

Objetivos de la prueba	Ejecutar el microservicio para crear, actualizar, modificar archivos planos dentro del servidor Solid
-------------------------------	---

<p>Técnicas</p>	<p>Escribir el nombre y la extensión del archivo, donde se valida los tipos de archivos que se puede crear JSON o HTML</p>						
<p>Código involucrado</p>	<pre> jest.mock('fs'); jest.mock('fs-extra'); describe('A JsonFileStorage', (): void => { const rootFilePath = 'files/'; const jsonPath = 'storage.json'; let cache: { data: any }; let locker: ReadWriteLocker; let storage: JsonFileStorage; // todo: @Van-Hunguyen - hace 17 meses • feat: Create KeyValueStorage with a JSON file bac... beforeEach(async(): Promise<void> => { cache = mockFileSystem(rootFilePath); locker = { withReadLock: jest.fn(async(identifier: ResourceIdentifier, whileLocked: () => any): Promise<any> => await whileL withWriteLock: jest.fn(async(identifier: ResourceIdentifier, whileLocked: () => any): Promise<any> => await whileL }; storage = new JsonFileStorage(`\${rootFilePath}\${jsonPath}`, locker); }); </pre>						
<p>Casos de prueba</p>	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px;"> <p>Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,</p> </td> </tr> <tr> <td style="padding: 5px;"> <p>Objetivo: Crear archivos planos, lograr agregar contenido bajo la estructura de la extensión, modificando los datos agregados por defecto</p> </td> </tr> <tr> <td style="text-align: center; padding: 5px;"> <p>Caso N° 1</p> </td> </tr> <tr> <td style="padding: 5px;"> <p>Descripción: Datos correctos, ingresando datos verificados.</p> </td> </tr> <tr> <td style="padding: 5px;"> <p>Entradas: Nombre y extensión del archivo</p> </td> </tr> <tr> <td style="padding: 5px;"> <p>Salida esperada: El archivo con estructura en blanco</p> </td> </tr> </table>	<p>Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,</p>	<p>Objetivo: Crear archivos planos, lograr agregar contenido bajo la estructura de la extensión, modificando los datos agregados por defecto</p>	<p>Caso N° 1</p>	<p>Descripción: Datos correctos, ingresando datos verificados.</p>	<p>Entradas: Nombre y extensión del archivo</p>	<p>Salida esperada: El archivo con estructura en blanco</p>
<p>Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,</p>							
<p>Objetivo: Crear archivos planos, lograr agregar contenido bajo la estructura de la extensión, modificando los datos agregados por defecto</p>							
<p>Caso N° 1</p>							
<p>Descripción: Datos correctos, ingresando datos verificados.</p>							
<p>Entradas: Nombre y extensión del archivo</p>							
<p>Salida esperada: El archivo con estructura en blanco</p>							
<p>Resultados</p>	<p>El caso de prueba se ha ejecutado, rectificando todos los defectos obtenidos en los casos de estudio.</p>						
<p>Observaciones</p>	<p>N/A</p>						

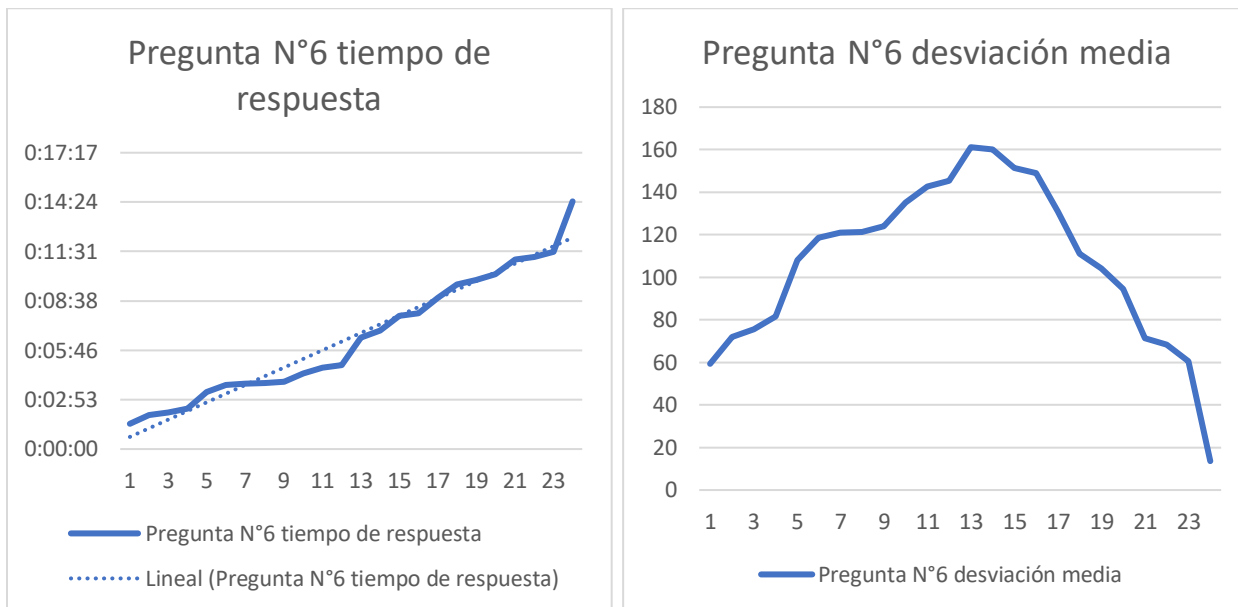
Nota. En la tabla 20 prueba de componente para gestionar un archivo plano dentro del POD de estudiantes Solid, los campos que utiliza el autor del proyecto comprueban los microservicios de Pancka Ruray.

La figura 77 permite divisar que existe una clara relación del proceso realizado por los estudiantes con los tiempos, donde se logra un aprendizaje directo siendo intuitivo al mostrar una

forma Gaussiana de la desviación media teniendo un tiempo promedio para realizar el proceso de 8 minutos.

Figura 77.

Pregunta N°6 análisis tiempo de respuesta



Nota. La figura 77 cuenta con el total del tiempo para agregar un nuevo archivo dentro del POD de un estudiante en un servidor Solid y la segunda para analizar si el tiempo puede tener una correlación con los estudiantes.

Manejar permisos de acceso para edición, agregar datos en POD

Registrar los roles que permiten manipular datos por terceros es necesario para realizar cambios desde aplicaciones descentralizadas fuera de los servidores de almacenamiento Solid y los componentes para la prueba están registrados en la tabla 21

Tabla 21.

Servidor Solid Pancka Ruray. Manejar permisos de acceso en un POD

Objetivos de la prueba	Efectuar el microservicio para manejar acceso a los archivos planos dentro del servidor Solid
Técnicas	Los actores sean individuos, grupos, todo el mundo, aplicaciones que fueron agregados mediante la URI

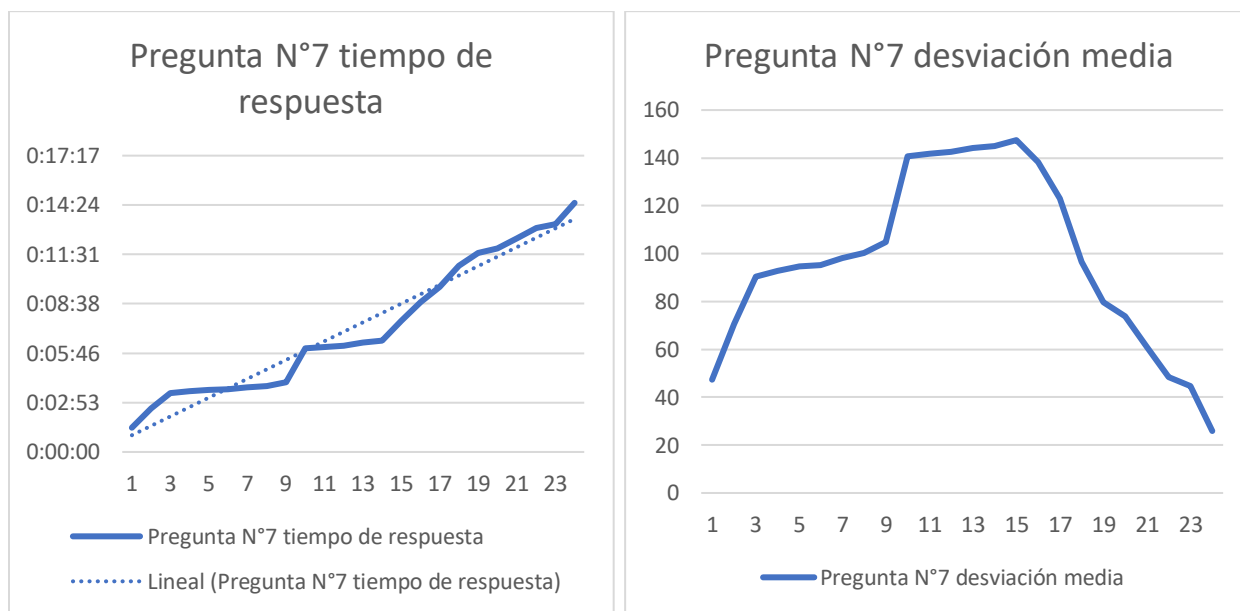
Código involucrado	<pre>describe('A AgentAccessChecker', (): void => { const webId = 'http://test.com/alice/profile/card#me'; const acl = new Store(); acl.addQuad(namedNode('match'), ACL.terms.agent, namedNode(webId)); acl.addQuad(namedNode('noMatch'), ACL.terms.agent, namedNode('http://test.com/bob')); const checker = new AgentAccessChecker(); it('can handle all requests.', async(): Promise<void> => { await expect(checker.canHandle(null as any)).resolves.toBeUndefined(); }); it('returns true if a match is found for the given WebID.', async(): Promise<void> => { const input: AccessCheckerArgs = { acl, rule: namedNode('match'), credential: { webId } }; await expect(checker.handle(input)).resolves.toBe(true); }); it('returns false if no match is found.', async(): Promise<void> => { const input: AccessCheckerArgs = { acl, rule: namedNode('noMatch'), credential: { webId } }; await expect(checker.handle(input)).resolves.toBe(false); }); });</pre>						
Casos de prueba	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px;"> Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest, </td> </tr> <tr> <td style="padding: 5px;"> Objetivo: Lograr tener distintos usuarios, grupos de usuario o aplicaciones con distintos niveles para gestión de archivos los cuales pueden variar entre el acceso de observar, crear, actualizar y eliminar información del POD </td> </tr> <tr> <td style="text-align: center; padding: 5px;"> Caso N° 1 </td> </tr> <tr> <td style="padding: 5px;"> Descripción: Datos correctos, ingresando datos verificados. </td> </tr> <tr> <td style="padding: 5px;"> Entradas: Listado de roles dentro de archivos (Propietario, Editor, Publicador, Remitente y Espectador) </td> </tr> <tr> <td style="padding: 5px;"> Salida esperada: Lista de control de acceso para un archivo o carpeta existente </td> </tr> </table>	Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,	Objetivo: Lograr tener distintos usuarios, grupos de usuario o aplicaciones con distintos niveles para gestión de archivos los cuales pueden variar entre el acceso de observar, crear, actualizar y eliminar información del POD	Caso N° 1	Descripción: Datos correctos, ingresando datos verificados.	Entradas: Listado de roles dentro de archivos (Propietario, Editor, Publicador, Remitente y Espectador)	Salida esperada: Lista de control de acceso para un archivo o carpeta existente
Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,							
Objetivo: Lograr tener distintos usuarios, grupos de usuario o aplicaciones con distintos niveles para gestión de archivos los cuales pueden variar entre el acceso de observar, crear, actualizar y eliminar información del POD							
Caso N° 1							
Descripción: Datos correctos, ingresando datos verificados.							
Entradas: Listado de roles dentro de archivos (Propietario, Editor, Publicador, Remitente y Espectador)							
Salida esperada: Lista de control de acceso para un archivo o carpeta existente							
Resultados	<p>El caso de prueba se ha ejecutado, rectificando todos los defectos obtenidos en los casos de estudio.</p>						
Observaciones	<p>N/A</p>						

Nota. En la tabla 21 prueba de componente para manejar permisos de acceso en un archivo o carpeta dentro del POD de estudiantes Solid, los campos que utiliza el autor del proyecto comprueban los microservicios de Pancka Ruray.

La figura 78 cuenta con un desfase entre los minutos 2 y 8, son claves para entender que poder cambiar los roles es complicado para los estudiantes al no colocar que debe hacer si moverlo mediante el uso del teclado, se debe detallar que los actores agregados deben darse clic y mover al rol deseado.

Figura 78.

Pregunta N°7 análisis tiempo de respuesta



Nota. La figura 78 cuenta con el total del tiempo para agregar un tipo de actor en cada uno de los tipos de acceso Solid y la segunda para analizar si el tiempo puede tener una correlación.

Inicio de sesión en cliente Pancka Ruray

Iniciar sesión en el cliente es una tarea simple lo que se busca con esta prueba es la interconectividad con distintos sistemas de almacenamiento Solid y los componentes para la prueba están registrados en la tabla 22

Tabla 22.

Cliente Pancka Ruray. Inicio de sesión de un estudiante

Objetivos de la prueba	Validar el microservicio de inicio de sesión de la aplicación descentralizada Pancka Ruray
-------------------------------	--

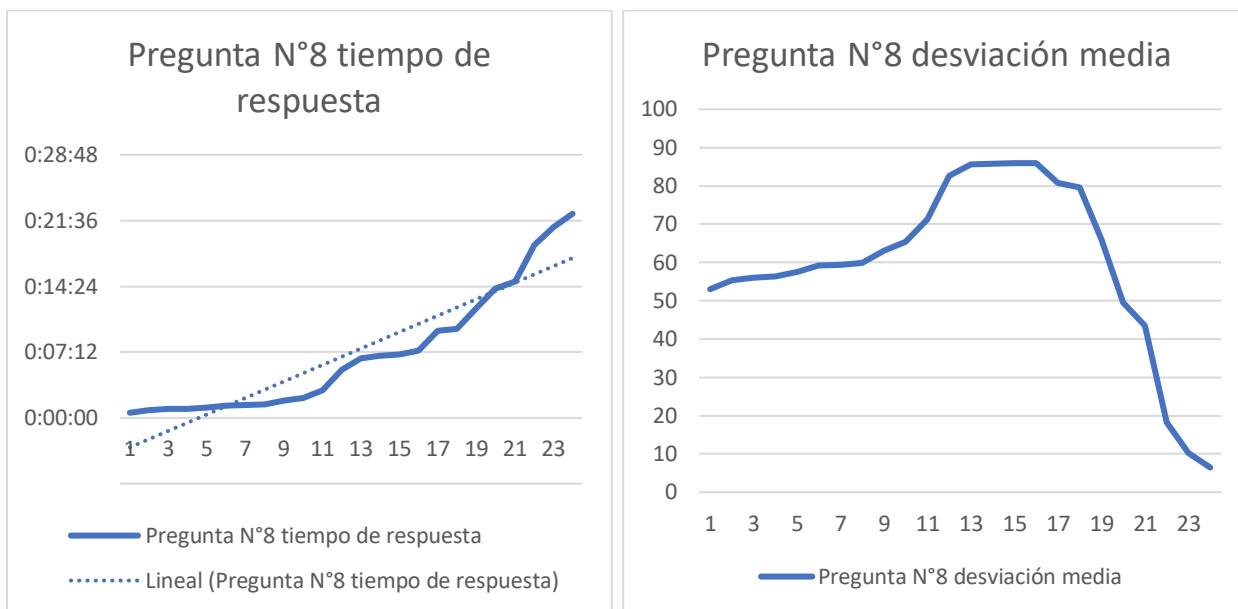
Técnicas	Seleccionar el servidor descentralizado de preferencia, redirecciona mediante OIDC y puede restaurar las credenciales previas mediante un token de autorización						
Código involucrado	<pre>import { SessionProvider } from "@inrupt/solid-ui-react"; const root = ReactDOM.createRoot(document.getElementById('root')); root.render(<SessionProvider restorePreviousSession> <App /> </SessionProvider>);</pre>						
Casos de prueba	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td data-bbox="570 699 1409 842">Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,</td> </tr> <tr> <td data-bbox="570 842 1409 984">Objetivo: Implementar inicio de sesión de la aplicación descentralizada Pancka Ruray</td> </tr> <tr> <td data-bbox="570 984 1409 1058" style="text-align: center;">Caso N° 1</td> </tr> <tr> <td data-bbox="570 1058 1409 1131">Descripción: Datos correctos, ingresando datos verificados.</td> </tr> <tr> <td data-bbox="570 1131 1409 1274">Entradas: Dirección del servidor, protocolo de acceso, y datos para autenticar al estudiante</td> </tr> <tr> <td data-bbox="570 1274 1409 1346">Salida esperada: Pantalla con el e-profile del estudiante</td> </tr> </table>	Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,	Objetivo: Implementar inicio de sesión de la aplicación descentralizada Pancka Ruray	Caso N° 1	Descripción: Datos correctos, ingresando datos verificados.	Entradas: Dirección del servidor, protocolo de acceso, y datos para autenticar al estudiante	Salida esperada: Pantalla con el e-profile del estudiante
Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,							
Objetivo: Implementar inicio de sesión de la aplicación descentralizada Pancka Ruray							
Caso N° 1							
Descripción: Datos correctos, ingresando datos verificados.							
Entradas: Dirección del servidor, protocolo de acceso, y datos para autenticar al estudiante							
Salida esperada: Pantalla con el e-profile del estudiante							
Resultados	El caso de prueba se ha ejecutado, rectificando todos los defectos obtenidos en los casos de estudio.						
Observaciones	N/A						

Nota. En la tabla 22 prueba de componente para iniciar sesión desde Pancka Ruray, los campos que utiliza el autor del proyecto comprueban los microservicios del cliente.

En la figura 79 la gráfica tiene un valor inicial y final con gran diferencia partiendo del primer minuto hasta 28 minutos esto es debido a que el exceso de peticiones terminó colapsando el servidor del cliente.

Figura 79.

Pregunta N°8 análisis tiempo de respuesta



Nota. La figura 79 cuenta con el total del tiempo para iniciar sesión en el cliente Pancka Ruray y la segunda para analizar si el tiempo puede tener una correlación con los estudiantes.

Agregar habilidad de estudiante

El registro de actividades es la parte fundamental ya que de aquí parte poder validar, crear y compartir perfiles en línea de estudiantes, este paso es previo a la certificación realiza por docentes o empleadores

Tabla 23.

Cliente Pancka Ruray. Agregar habilidad de un estudiante

Objetivos de la prueba	Validar el microservicio para agregar habilidades de estudiantes en la aplicación descentralizada Pancka Ruray
Técnicas	Escoger el tipo, redactar el nombre y porcentaje de la habilidad.

Código involucrado	<pre>const Index = ({skillList, setSkillList, type}) => { const { session }=useSession() const [skillText, setSkillText]=useState(""); const [skillPercentage, setSkillPercentage]=useState(0); const addSkill=async(text,percentage)=>{ const indexUrl=getSourceUrl(skillList) const skillWithText=addStringNoLocale(createThing(),NAME_PREDICATE,text) const skillWithPercentage=addStringNoLocale(skillWithText,PERCENTAGE_PREDICATE,percentage) const skillWithDate=addDatetime(skillWithPercentage,CREATED_PREDICATE,new Date()) const skillWithClass=addStringNoLocale(skillWithDate,SKILL_PREDICATE,type) const skillWithType=addUrl(skillWithClass, TYPE_PREDICATE, SKILL_CLASS) const updatedSkillList=setThing(skillList,skillWithType) const updatedDataset=await saveSolidDatasetAt(indexUrl,updatedSkillList,{ fetch: session.fetch,</pre>						
Casos de prueba	<table border="1"> <tr> <td data-bbox="574 844 1408 982">Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,</td> </tr> <tr> <td data-bbox="574 982 1408 1121">Objetivo: Proporcionar al estudiante agregar habilidades de mejora en su e-profile</td> </tr> <tr> <td data-bbox="574 1121 1408 1199" style="text-align: center;">Caso N° 1</td> </tr> <tr> <td data-bbox="574 1199 1408 1276">Descripción: Datos correctos, ingresando datos verificados.</td> </tr> <tr> <td data-bbox="574 1276 1408 1354">Entradas: Nombre y el porcentaje de la habilidad</td> </tr> <tr> <td data-bbox="574 1354 1408 1409">Salida esperada: La habilidad se añade a la interfaz grafica</td> </tr> </table>	Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,	Objetivo: Proporcionar al estudiante agregar habilidades de mejora en su e-profile	Caso N° 1	Descripción: Datos correctos, ingresando datos verificados.	Entradas: Nombre y el porcentaje de la habilidad	Salida esperada: La habilidad se añade a la interfaz grafica
Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,							
Objetivo: Proporcionar al estudiante agregar habilidades de mejora en su e-profile							
Caso N° 1							
Descripción: Datos correctos, ingresando datos verificados.							
Entradas: Nombre y el porcentaje de la habilidad							
Salida esperada: La habilidad se añade a la interfaz grafica							
Resultados	El caso de prueba se ha ejecutado, rectificando todos los defectos obtenidos en los casos de estudio.						
Observaciones	N/A						

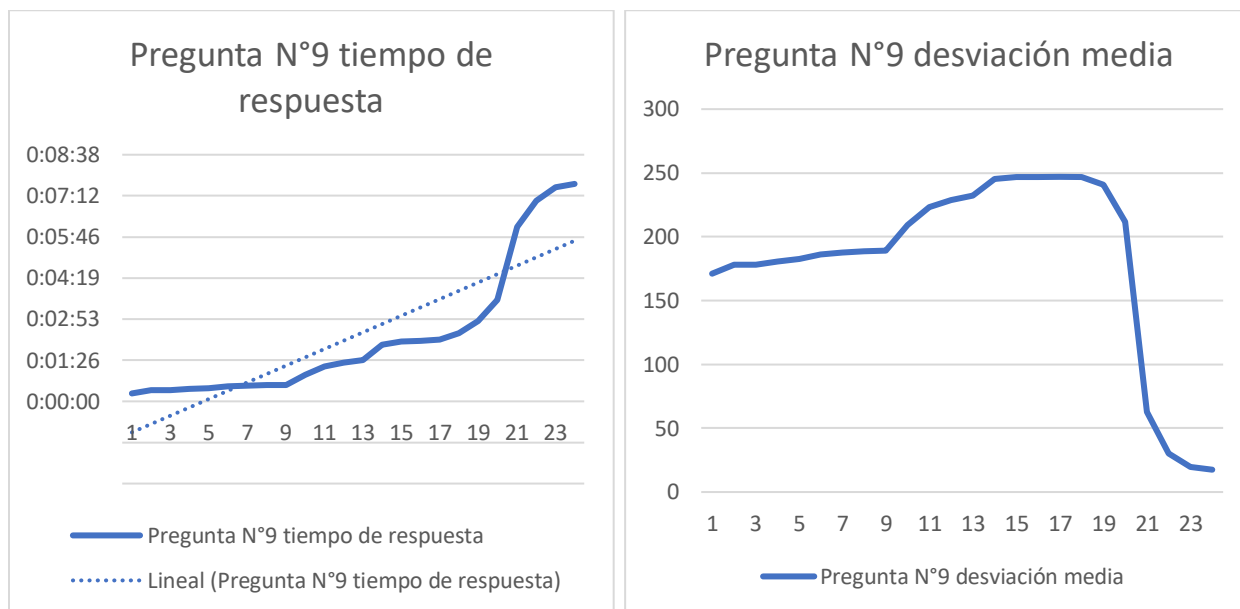
Nota. En la tabla 23 prueba de componente para agregar habilidad a un estudiante en la aplicación cliente, los campos que utiliza el autor del proyecto comprueban los microservicios de Pancka Ruray.

Al ser un proceso secuencial los estudiantes colocan 0 minutos al no contar con el acceso y descartando esa información el tiempo promedio no se correlaciona al contar con encuestas

de carácter individual un estudiante con mayor o menor aptitud al resto utiliza más tiempo para cumplir con la instrucción demostrado dentro de la figura 80.

Figura 80. *Pregunta N°9 análisis tiempo de respuesta*

Pregunta N°9 análisis tiempo de respuesta



Nota. La figura 80 cuenta con el total del tiempo para añadir las habilidades de un estudiante en el cliente Pancka Ruray y la segunda para analizar si el tiempo puede tener una correlación con los estudiantes.

Editar información de e-profile

Cambiar datos es de suma importancia, como es el rol de estudiante a empleado, modificar el cambio de un número de teléfono o correo electrónico para el contacto y la descripción de uno mismo va a cambiar durante el tiempo similar a las habilidades que se agregan a un e-profile, los componentes para la prueba están registrados en la tabla 24

Tabla 24.

Cliente Pancka Ruray. Editar información de un estudiante

Objetivos de la prueba	Corroborar el microservicio para editar información del estudiante en la aplicación descentralizada Pancka Ruray
-------------------------------	--

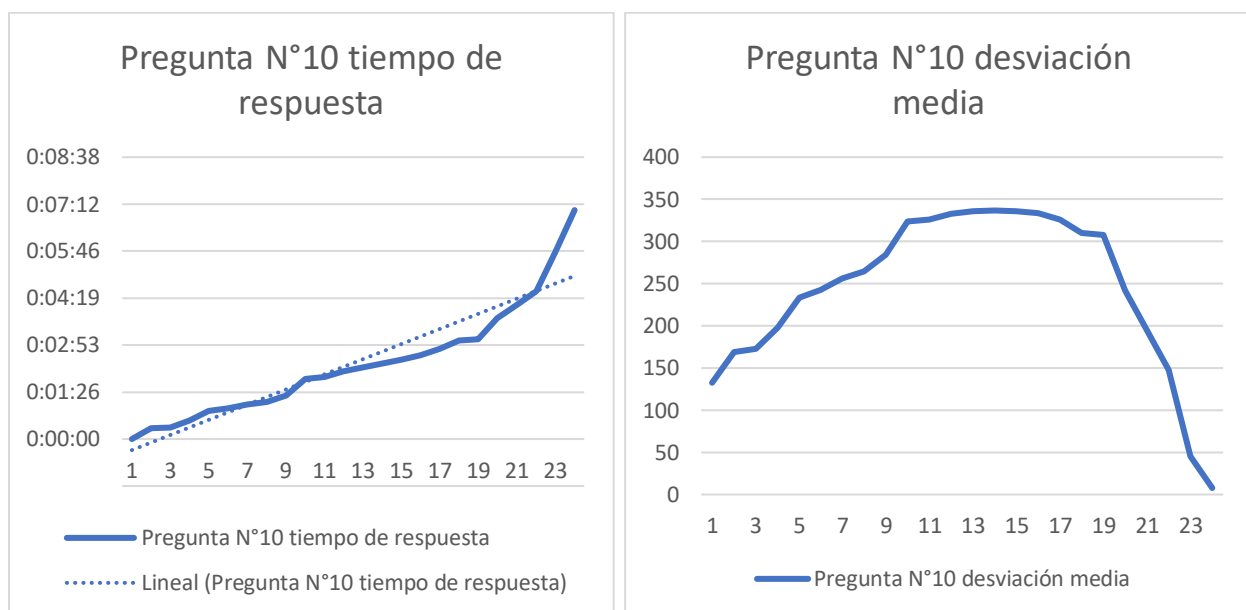
Técnicas	Se cambia el tiempo de vista como espectador, dando lugar a poder editar los campos con permisos.						
Código involucrado	<pre> <CombinedDataProvider className="card-container" datasetUrl={webId} thingUrl={webId} > <main> <section className="header"> <section className="header-left-section"> <h1><Text property={VCARD.fn}/></h1> <hr /> <h4><Text property={VCARD.role}/></h4> <h4><Text property={VCARD.organization_name.iri.value} edit={editing} autosave/> </section> <section className="header-middle-section"> <Image property={VCARD.hasPhoto.iri.value} className="header-middle-section-img" Perfil Solid <Button size="small" color="primary" onClick={() => setEditing(!editing)}> Editar Información </Button> </section> </main> </section> </pre>						
Casos de prueba	<table border="1"> <tr> <td data-bbox="574 800 1409 940">Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,</td> </tr> <tr> <td data-bbox="574 940 1409 1081">Objetivo: Comprobar el acceso a editar la información dentro de un e-profile del estudiante</td> </tr> <tr> <td data-bbox="574 1081 1409 1150" style="text-align: center;">Caso N° 1</td> </tr> <tr> <td data-bbox="574 1150 1409 1220">Descripción: Datos correctos, ingresando datos verificados.</td> </tr> <tr> <td data-bbox="574 1220 1409 1360">Entradas: Acción clic en opción de editar perfil, texto de componentes como nombre, rol, teléfono correo electrónico.</td> </tr> <tr> <td data-bbox="574 1360 1409 1501">Salida esperada: Mostrar los datos finales actualizado o mantenidos del e-profile</td> </tr> </table>	Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,	Objetivo: Comprobar el acceso a editar la información dentro de un e-profile del estudiante	Caso N° 1	Descripción: Datos correctos, ingresando datos verificados.	Entradas: Acción clic en opción de editar perfil, texto de componentes como nombre, rol, teléfono correo electrónico.	Salida esperada: Mostrar los datos finales actualizado o mantenidos del e-profile
Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,							
Objetivo: Comprobar el acceso a editar la información dentro de un e-profile del estudiante							
Caso N° 1							
Descripción: Datos correctos, ingresando datos verificados.							
Entradas: Acción clic en opción de editar perfil, texto de componentes como nombre, rol, teléfono correo electrónico.							
Salida esperada: Mostrar los datos finales actualizado o mantenidos del e-profile							
Resultados	El caso de prueba se ha ejecutado, rectificando todos los defectos obtenidos en los casos de estudio.						
Observaciones	N/A						

Nota. En la tabla 24 prueba de componente para editar la información de un estudiante en la aplicación cliente, los campos que utiliza el autor del proyecto comprueban los microservicios de Pancka Ruray.

Una vez aligerada la carga de lectura de datos permite tener un flujo casi lineal, teniendo solo como conclusión una falta de recursos de hosting al actualizar la información de forma simultánea se agrega 1 o 2 minutos al tiempo promedio para editar información.

Figura 81.

Pregunta N°10 análisis tiempo de respuesta

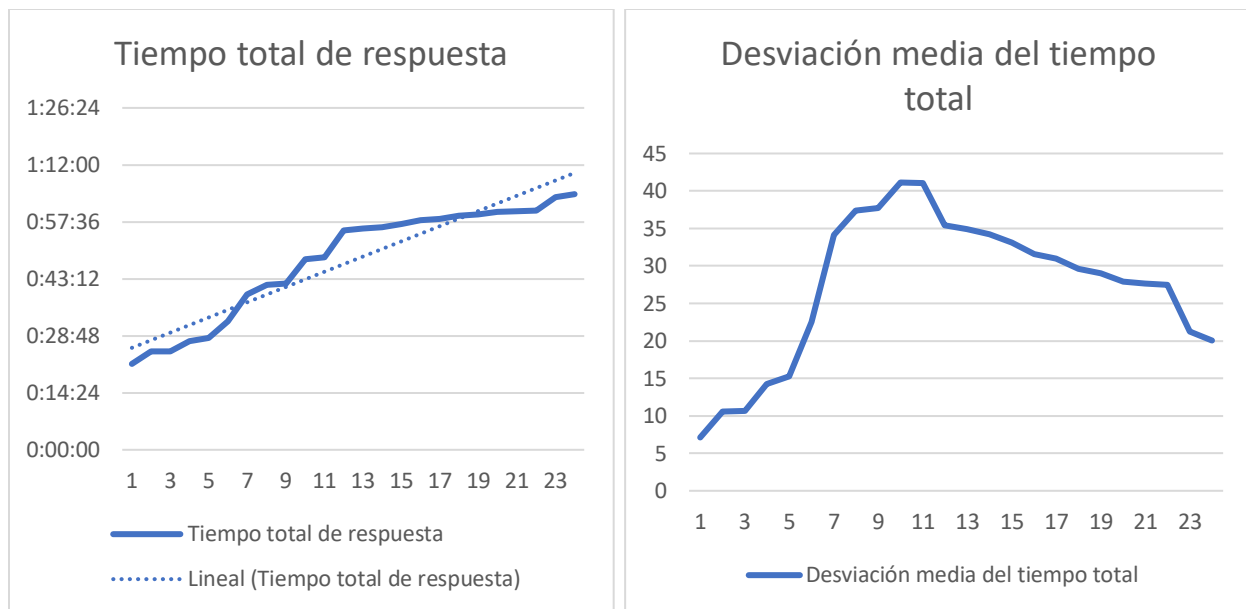


Nota. La figura 81 cuenta con el total del tiempo para editar la información de un e-profile estudiantil en el cliente Pancka Ruray y la segunda para analizar si el tiempo puede tener una correlación con los estudiantes.

El análisis total no puede tener una relación directa en toda la figura al existir una pregunta que no tiene relación se cuenta con un pico muy marcado en el paso de los 57 minutos al tiempo final de 1 hora con 26 minutos y se lo puede ver más claro en la gráfica lineal en la parte final donde ya no tiene un crecimiento más bien estancándose a partir del minuto 57.

Figura 82.

Análisis total tiempo de respuesta



Nota. La figura 82 cuenta con el total del tiempo por estudiante dentro del taller y la segunda para analizar si el tiempo puede tener una correlación con los estudiantes.

Certificación de habilidades

La validez que se dan a las habilidades agregadas por los estudiantes donde los docentes o supervisores verifican basado en las aptitudes de los estudiantes es de suma importancia al postular para un trabajo en esta prueba los componentes para la prueba están registrados en la tabla 25

Tabla 25.

Servidor Solid Pancka Ruray. Certificar habilidad de estudiante

Objetivos de la prueba	Probar el microservicio para certificar la información del estudiante en la aplicación descentralizada Pancka Ruray
Técnicas	Una vez agregada la habilidad el campo solo es visible para los que pueden editar la información, siendo posible certificar solo a miembros pertenecientes a los grupos de empresa o universidad

Código involucrado	<pre>const updatedSkills=async(skills)=>{ const skillsUrl=getSourceUrl(skillList) const date=new Date() const doneSkills=addDatetime(skills,UPDATE_PREDICATE,date) const updatedSkills=setThing(skillList,doneSkills, {fetch}); const updatedList=await saveSolidDatasetAt(skillsUrl, updatedSkills,{fetch}) setSkillList(updatedList) return }</pre>						
Casos de prueba	<table border="1"> <tr> <td data-bbox="574 625 1409 764">Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,</td> </tr> <tr> <td data-bbox="574 764 1409 909">Objetivo: Validar las habilidades ingresadas por los estudiantes dentro de un e-profile</td> </tr> <tr> <td data-bbox="574 909 1409 982" style="text-align: center;">Caso N° 1</td> </tr> <tr> <td data-bbox="574 982 1409 1056">Descripción: Datos correctos, ingresando datos verificados.</td> </tr> <tr> <td data-bbox="574 1056 1409 1201">Entradas: E-profile de estudiantes con el listado de habilidades sin respaldar.</td> </tr> <tr> <td data-bbox="574 1201 1409 1339">Salida esperada: Registro de la habilidad con el respaldo del actor encargado y su fecha de actualización</td> </tr> </table>	Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,	Objetivo: Validar las habilidades ingresadas por los estudiantes dentro de un e-profile	Caso N° 1	Descripción: Datos correctos, ingresando datos verificados.	Entradas: E-profile de estudiantes con el listado de habilidades sin respaldar.	Salida esperada: Registro de la habilidad con el respaldo del actor encargado y su fecha de actualización
Tipo de prueba: Unitaria (caja blanca) implementado con el framework Jest,							
Objetivo: Validar las habilidades ingresadas por los estudiantes dentro de un e-profile							
Caso N° 1							
Descripción: Datos correctos, ingresando datos verificados.							
Entradas: E-profile de estudiantes con el listado de habilidades sin respaldar.							
Salida esperada: Registro de la habilidad con el respaldo del actor encargado y su fecha de actualización							
Resultados	El caso de prueba se ha ejecutado, rectificando todos los defectos obtenidos en los casos de estudio.						
Observaciones	N/A						

Nota. En la tabla 25 prueba de componente para certificar las habilidades de un estudiante en la aplicación cliente, los campos que utiliza el autor del proyecto comprueban los microservicios de Pancka Ruray.

Resumen de los resultados obtenidos

El análisis de resultado se realiza para la parte estética de la aplicación donde se concluye un problema de manejo del flujo de datos por utilizar solo iconos sin una descripción a primera vista, la interfaz del servidor no es intuitiva dentro del proceso de registro de la información causando conflictos para realizar el proceso de registro de e-profile para estudiantes de instituciones de educación superior. Se obtiene que la distribución utilizada en el cliente Pancka Ruray es mucho más eficaz y enfocado a solventar problemas para manejo de información de los datos del CV.

Dentro del flujo de la red, con sobrecarga de datos los gráficos como la mayoría de los estudiantes pueden corroborar que existe un trabajo exitoso teniendo presente el total de descarga de información con dos grandes picos de 4 GB que es la primera vez que utilizan el cliente en el segundo consumo es reducido a 1 GB al tener el cliente que integra PWA donde los recursos son almacenados en caché del navegador, en el dispositivo móvil o en la computadora de acceso reduciendo en una cuarta parte el tráfico del servidor permitiendo implementar clientes descentralizados de forma gratuita al no exceder el límite de descarga siempre que el total de usuarios ronde entre los 15 por 7 días o 57 en todo el mes, por último manejar disparadores para saber que tipos de acceso y rol tiene dentro del sistema demuestra que se registró 67 inicios de sesión de estudiantes, fue visita por reclutadores 57 veces y con un total de 364 usuario que obtuvieron la primera vista. Los roles como reclutadores o visitantes pueden revisar un e-profile en 25 segundo, mientras que estudiantes pueden registran e-profile en 2 minutos con 30 segundo dentro de la aplicación cliente Pancka Ruray.

Las respuestas dentro del plan para el prototipo se pueden dividir en dos entornos: el de prueba y producción. Dentro del primer entorno se prueba la funcionalidad del prototipo llegando a alcanzar un 75% de las pruebas de caja blanca planteadas se realiza los mismos casos de uso, pero con estudiantes reales donde ya no se controla el entorno de ejecución y las

representaciones gráficas muestran el resultado de las pruebas en producción representando una correlación de tiempo estudiante en su gran mayoría de preguntas descartando las preguntas 8 y 9 que tiene valores dispersos.

Síntesis de las preguntas de investigación

Una vez recopilada la información, desarrollado, implementado y evaluado un prototipo funcional cumpliendo con las fases de la metodología de investigación, se procede a dar una respuesta a las preguntas de investigación planteadas en la tabla 1.

1. ¿Cuál es el valor diferenciador de un e-profile y un currículum vitae?

Si nos enfocamos solo en el Ecuador un e-profile puede ser el inicio para los registros académicos en papel que se maneja dentro de las instituciones de educación superior, la gran referencia es el país con mayor población en el mundo el cual buscó alternativas y un digitalizado los certificados teniendo el mismo valor un digital que uno físico.

2. ¿Qué aplicaciones y propuestas existen para la creación de e-profiles?

QualiChain es un proyecto el cual utiliza webs semánticas, mejora el proceso de optimización de universidades, mejora la relación con los reclutadores compartiendo el currículum vitae físico transformado a digital permitiendo reclutar y manejar los servicios componentes para posibles empleados. Blockcert es una aplicación que se integra en las aplicaciones para emitir certificados validados dentro de una red blockchain.

3. ¿Cómo se clasifican los tipos de red Blockchain de acuerdo con las ventajas y desventajas en sistemas similares?

Blockchain cuenta con cuatro clases redes públicas permite el acceso a cualquier usuario para minar y ser parte de la lógica del negocio, privadas solo cierta parte de nodos o servidores pueden validar las transacciones que se realizan, federadas ya no cuenta con la libertad de los otros tipos pertenece a una entidad que respalda los procesos y de consorcio que una la privada

y pública permitiendo elegir entre ser un minero o solo consumir las ventajas de pertenecer a la red blockchain.

4. ¿Qué herramientas se utilizan para la implementación de un ecosistema tecnológico basado en sistemas descentralizados?

Las herramientas de un sistema descentralizado se lo incorporan mediante blockchain o un servidor de almacenamiento de datos descentralizado permitiendo el consumo, integración y replicación de la información puede utilizar software para facilitar cumplir los requerimientos al procesar información en distintos sistemas se utiliza base de datos distribuidas puede ser SQL o noSQL, pero es limitado a los servidores que uno mismo instancia por ello Solid es un proyecto que ya cuenta con distintos servidores activos por empresas o comunidades.

5. ¿Qué flujo de la información se debe seguir al trabajar con estudiantes, universidades y empresas?

El principal propósito es que los dueños de la información puedan manejar los roles de acceso con eso en cuenta todos los procesos parte desde la participación de los estudiantes al controlar la incorporación o eliminar datos, modificación de archivos. La utilización de un sistema descentralizado permite tener procesos asíncronos utilizando las habilidades del e-profile de un estudiante el docente o empleador pueden validar esta información y a su vez un reclutador puede ver el e-profile todo esto al mismo tiempo logrando tener un flujo línea que se puede interrumpir al tener la actualización de datos en tiempo real trabajados de forma directa con el servidor de almacenamiento.

6. ¿Cuáles son los elementos que se deben implementar dentro del ecosistema tecnológico basado en sistemas descentralizados?

La arquitectura descentralizada es fácil de comprender esta consta con servidores que proporciona información a un sin número de aplicaciones clientes por lo que se puede dividir en estas dos, pero al mismo tiempo las aplicaciones pueden ser consumidas por otras aplicaciones

y los servidores de almacenamiento pueden compartir información comunicándose entre ellos mismos logrando generar una red de varios lugares para almacenar información y aplicaciones que usan dicha información.

7. ¿Cuáles son las métricas aceptables para validar el correcto funcionamiento de un prototipo descentralizado?

Existe un sin número de metodologías para validar software que cuentan con menor cantidad métricas que el Modelo Sistemico de Calidad (MOSCA), es por lo que se basado en las seis categorías se tomaron los aspectos contextuales que más se apegan a la idea de utilizar un sistema descentralizado la propuesta es que si no cumple un 75% de las métricas establecidas el prototipo no puede categorizarse como funcional y debe mantenerse en desarrollo sin ser puesto en producción, si se desea saber cuáles son estas métricas se encuentra en la tabla 13.

8. ¿Qué posibles limitaciones tiene la arquitectura descentralizada?

La mayor problema de una arquitectura descentralizada es que el sistema de almacenamiento se pierda con el tiempo no es que se caiga el servidor es que se deje de dar sustento por una empresa o comunidad con ello queda inutilizado cualquier aplicación cliente, la otra es el exceso de carga de trabajo sobre los servidores al tener distintas opciones el hardware utilizado es menor abaratando costos y si existe un exceso de peticiones esto genera gastos para mantener el flujo de datos entre el servidor y aplicaciones cliente.

9. ¿Cuáles son los niveles de rendimiento, latencia y disponibilidad para un software funcional?

Para un cliente descentralizado es más que suficiente redundancia de 100 usuario una disponibilidad 24/7, con un rendimiento del 75% de la aplicación, latencia en la red del cliente 70 ms hasta 155 ms mientras que el servidor es de 130ms a 210ms, con esto permite a un estudiante común registrar su e-profile.

Capítulo VI

Conclusiones, Recomendaciones y Trabajos Futuros

Este capítulo describe las conclusiones, recomendaciones y líneas de trabajo futuro, producto del trabajo de investigación.

Conclusiones

El sistema de registro de e-profile para estudiantes de instituciones académicas para un acceso seguro, confiable y controlado Pancka Ruray detallado en presente trabajo permite obtener las conclusiones siguientes:

- a. La evidencia presentada dentro de todo el trabajo investigativo demuestra que los e-profiles y currículum vitae son utilizados para el reclutamiento de futuros empleados, en las empresas se acepta compartir CV en formato digital PDF perdiendo el rastro y uso que se les da a nuestros datos privados mientras que con un e-profile mantiene el control sobre la información de los estudiantes al gestionar los privilegios de lectura.
- b. Basado en el despliegue del prototipo Pancka Ruray se expone la integración de un estudiante en diferentes empleos, cursos, proyectos, eventos dentro y fuera de la institución educativa a la que pertenece, certificando las habilidades obtenidas tras una correcta interacción, finalizando con el registro y la construcción de un e-profile con información extraacadémica.
- c. Finalmente, la validación del prototipo basado en plan de pruebas unitarias, el Modelo Sistémico de Calidad (MOSCA) enfocado en las métricas aceptables para un sistema descentralizado, el cumplimiento fundamental de un cliente descentralizado se puede afirmar que el prototipo Pancka Ruray es una aplicación apta para gestionar información de terceros manteniendo los datos seguros, confiables y controlados.

En base a los resultados obtenidos la hipótesis de mantener los datos con acceso, seguro, confiable y controlado se comprobó mediante la ejecución de un servidor Solid que gestiona

basado en roles de acceso la información de un servidor descentralizado, permitiendo acceder a los datos desde una vista en base a los datos públicos, si se desea agregar o actualizar la información puede ingresar como un docente o supervisor, los cuales validan la información académica o extracurricular, de una forma gráfica y personal.

Recomendaciones

1. Implementar un aplicativo blockchain para validar las transacciones realizadas entre los estudiantes y docentes, los estudiantes y los supervisores, certificando las habilidades que se desarrollan al trabajar en cursos, voluntariados y empleos. Una de las opciones con mayor probabilidad de integrar es el proyecto Blockcert donde se puede respaldar generando certificados digitales validados por la cadena de bloques.
2. Dentro del desarrollo de una aplicación cliente utilizando servidores Solid como almacenamiento es de tomar en cuenta tecnologías como PWA que almacena la información dentro de los dispositivos. Permite aligerar la carga de transacciones entre servidor y aplicación facilitando el crecimiento y para temas de portabilidad con distribución del aplicativo aligerando la transferencia de datos usar contenedores como Docker implementado sobre capas gratuitas de muchos servicios en la nube como Firebase o netlify almacenando aplicaciones de forma gratuita.
3. Para utilizar el prototipo Pancka Ruray la arquitectura puede ser implementada desde la institución educativa o empresa, se recomienda utilizar en las instituciones educativas como un proyecto para desarrollar habilidades dentro de los estudiantes manejando una arquitecturas descentralizadas y permisos W3C dentro de aplicaciones web. Para posibles casos de estudio de la calidad del software al contar con una arquitectura diferente a la que se aplica en universidades los resultados de calidad serán variantes, reales y con posibilidad de mejora de forma sencilla.

Trabajos Futuros

1. Realizar un recipiente para manejar un servidor Solid de mejor forma para la parte hispanohablante, mejorando los iconos y textos, que permitan una navegación más intuitiva y precisa para los estudiantes de Latinoamérica.
2. Evaluar las categorías de una arquitectura del software enfocándose en la calidad que permita disminuir el tiempo del ciclo de vida del desarrollo de aplicaciones distribuidas comparando las métricas obtenidas de un sistema centralizado comparando uno descentralizado.
3. Poder implementar arquitecturas descentralizadas en sistemas gubernamentales para mejorar el acceso de datos prioritarios de una forma compartida entre distintas entidades como la información de los registros médicos entre los servicios de salud como hospitales, clínicas, centros de salud ubicados en un país, región, provincia, ciudad o cantón.

Bibliografía

- © The Tapscott Group. (2017). *A Declaration of Interdependence Towards a New Social Contract for the Digital Economy*. A project done in collaboration with the Berkman Klein Center for the Internet & Society at Harvard University Draft 1.0.
<https://dontapscott.com/wp-content/uploads/New-Social-Contract-May-10-2017.pdf>
- Arghyadeep, B. (2021). Effect of personal branding efforts on job prospects of a LinkedIn user. *IT in Industry*, 9(2), 323-332. <https://doi.org/10.17762/itii.v9i2.350>
- Arias Ortiz, E., & Hincapié, D. (2020). ESCUELA PARA LA VIDA: HABILIDADES SOCIOEMOCIONALES EN EL CONTEXTO ESCOLAR. En E. Arias Ortiz, D. Hincapié, & D. Paredes, *Educación para la vida. El desarrollo de las habilidades socioemocionales y el rol de los docentes* (págs. 6-10). Banco Interamericano de Desarrollo (BID).
- Baldi, M., Chiaraluce, F., Kodran, M., & Spalazzi, L. (2019). Security analysis of a blockchain-based protocol for the certification of academic credentials. *arXiv*, 1-12.
<https://arxiv.org/pdf/1910.04622.pdf>
- Brocke, J. v., & Maedche, A. (2019). The DSR grid: six core dimensions for effectively planning and communicating design science research projects. *Electronic Markets*, 379-385.
<https://doi.org/10.1007/s12525-019-00358-7>
- Cai, T., Yang, Z., Chen, W., Zheng, Z., & Yu, Y. (2020). A Blockchain-Assisted Trust Access Authentication System for Solid. *IEEE Access*, 8, 71605-71616.
<https://doi.org/10.1109/ACCESS.2020.2987608>.
- Cervantes, H. (10 de marzo de 2010). *Arquitectura de Software*. SG Buzz:
<https://sg.com.mx/revista/27/arquitectura-software>
- Deep, G., Mohana, R., Nayyar, A., Sanjeevikumar, P., & Hossain, E. (2019). Authentication Protocol for Cloud Databases Using Blockchain Mechanism. *sensors*, 19(4444), 1-13.
<https://doi.org/10.3390/s19204444>

- Deng, Q., & Ji, S. (2018). A Review of Design Science Research in Information Systems: Concept, Process, Outcome, and Evaluation. *AIS Electronic Library (AISeL)*, 10(1), 1-35. <https://aisel.aisnet.org/cgi/viewcontent.cgi?article=1167&context=pajais>
- Dong, Z., Lee, Y. C., & Zomaya, A. Y. (2019). Proofware: Proof of Useful Work Blockchain Consensus Protocol for Decentralized Applications. *ArXiv abs/1903.09276*. <https://arxiv.org/pdf/1903.09276.pdf>
- Fernández Quintanilla, D. (5 de septiembre de 2019). *Diseño de un Solid POD siguiendo técnicas DCU*. Máster Universitario en Diseño de Experiencia de Usuario, Universidad Internacional de La Rioja (UNIR), Madrid. Retrieved 12 de junio de 2022, from <https://reunir.unir.net/bitstream/handle/123456789/9847/Fernandez%20Quintanilla%2c%20David.pdf?sequence=1&isAllowed=y>
- Gajanayake, R., Hiras, M., Gunathunga, P., Karunasenna, A., Bandara, P., & Supun, E. (2020). Candidate Selection for the Interview using GitHub Profile and User Analysis for the Position of Software Engineer. *2020 2nd International Conference on Advancements in Computing (ICAC)*, 1, 168-173. <https://doi.org/10.1109/ICAC51239.2020.9357279>.
- Gürsoy, G., Bjornson, R., Green, M. E., & Gerstein, M. (2018). Using blockchain to log genome dataset access: efficient storage and query. *iDASH Privacy and Security Workshop*, 7, 1-9. <https://doi.org/10.1186/s12920-020-0716-z>
- Imbaquingo, J. R. (12 de mayo de 2021). Ley para la protección de datos de los ecuatorianos; se creará una Superintendencia. *e/Comercio*. <https://www.elcomercio.com/actualidad/politica/ley-proteccion-datos-ciudadanos-ecuador.html#:~:text=La%20Ley%20de%20Protecci%C3%B3n%20de,datos%20digitalizados%20de%20los%20ecuatorianos.&text=La%20ley%20debe%20entrar%20en,tratamiento%20de%20datos%20adec%C3>

- Jin, M., Cui, C., Yu, G., Li, X., Zhang, Y., & Zeng, L. (2020). A Blockchain-Based Scheme for Secure Storage and Sharing of Student Digital Profiles. *2020 3rd International Conference on Smart BlockChain (SmartBlock)*, 209-214.
<https://doi.org/10.1109/SmartBlock52591.2020.00045>
- Juricic, V., Radosevic, M., & Fuzul, E. (2019). Creating student's profile using blockchain technology. *42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 521-525.
<https://doi.org/10.23919/MIPRO.2019.8756687>
- Kopp, H., Mödinger, D., Hauck, F., Kargl, F., & Bösch, C. (2017). Design of a Privacy-Preserving Decentralized File Storage with Financial Incentives. *2017 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, 14-22.
<https://doi.org/10.1109/EuroSPW.2017.45>.
- Kunifuji, T. (2013). A proposal of autonomous decentralized software architecture to realize gradual and short-term extension in safety related system. *2013 IEEE Eleventh International Symposium on Autonomous Decentralized Systems (ISADS)*, 1-6.
<https://doi.org/10.1109/ISADS.2013.6513400>
- Lizcano, D., Lara, J., White, B., & Aljawarneh, S. (Abril de 2019). Blockchain-based approach to create a model of trust in open and ubiquitous higher education. *Journal of Computing in Higher Education*(32), 109-134. <https://doi.org/10.1007/s12528-019-09209-y>
- Lyu, J., Jiang, Z., Wang, X., Nong, Z., Au, M., & Fang, J. (2019). A Secure Decentralized Trustless E-Voting System Based on Smart Contract. *18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, 570-577.
<https://doi.org/10.1109/TrustCom/BigDataSE.2019.00082>.

- Mainini, P., & Laube-Rosenpflanzler, A. (2016). Access Control in Linked Data Using WebID. *ArXiv, abs/1610.04405*. <https://arxiv.org/pdf/1611.03019.pdf>
- Majchrzak, T., Biørn-Hansen, A., & Grønli, T.-M. (2017). Comprehensive Analysis of Innovative Cross-Platform App Development Frameworks. *Proceedings of the 50th Hawaii International Conference on System Sciences (HICSS)*, 6162-6171. <https://doi.org/10125/41909>
- Malavolta, I., Procaccianti, G., Noorland, P., & Vukmirović, P. (2017). Assessing the Impact of Service Workers on the Energy Efficiency of Progressive Web Apps. *2017 IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (MOBILESoft)*, 35-45. <https://doi.org/10.1109/MOBILESoft.2017.7>
- Mefteh, H. (2021). Impact of extracurricular in the improvement of higher education graduates employability skills: The case of Tunisia. *Business Excellence and Management*, 11(1), 80-93. <https://doi.org/10.24818/beman/2021.11.1-06>
- Mikroyannidis, A. (2020). Blockchain Applications in Education: A Case Study in Lifelong Learning. *The 12th International Conference on Mobile, Hybrid, and On-line Learning (eLmL 2020)*, 21(25), 31-35. <http://oro.open.ac.uk/69593/1/QualiChain%20paper%20camera%20ready.pdf>
- Mtsweni, J., Biermann, E., & Pretorius, L. (2014). iSemServ: A model-driven approach to developing semantic web services. *South African Computer Journal*, 52, 55-70. <https://doi.org/10.18489/sacj.v52i0.185>
- Nghia, T. (2017). Developing generic skills for students via extra-curricular activities in Vietnamese universities: Practices and influential factors. *Journal of Teaching and Learning for Graduate Employability*, 8(1), 22-39.

- Polmear, M., Chau, A., & Simmons, D. (2020). Ethics as an outcome of out-of-class engagement across diverse groups of engineering students. *Australasian Journal of Engineering Education*. <https://doi.org/10.1080/22054952.2020.1836752>
- Pramod, N., & Sankaran, S. (2019). EP6. Blockchain Based Framework For Driver Profiling in Smart Cities. *2019 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, 1-6. <https://doi.org/10.1109/ANTS47819.2019.9117923>
- Ramachandran, M., Chowdhury, N., Third, A., Domingue, J., Quick, K., & Bachler, M. (2020). Towards Complete Decentralised Verification of Data with Confidentiality: Different ways to connect Solid Pods and Blockchain. *A Decentralised Web Workshop, 20(24)*, 645–649. <https://doi.org/10.1145/3366424.3385759>
- Rivero Amador, S., Díaz Pérez, M., & Rodríguez Font, R. J. (2020). Sistemas de información curricular en Iberoamérica y sus principales características. *Revista Cubana de Información en Ciencias de la Salud, 31(1)*, 1-20. <http://scielo.sld.cu/pdf/ics/v31n1/2307-2113-ics-31-01-e1244.pdf>
- Rossales Sengik, A., Lerch Lunardi, G., Scalabrin Bianchi, I., & Costa Wiedenhöft, G. (2022). Using design science research to propose an IT governance model for higher education institutions. *The Author(s), under exclusive licence to Springer Science+Business Media, LLC*, 1-21. <https://doi.org/10.1007/s10639-022-11088-3>
- Shah, J., & Dubaria, D. (2019). Building Modern Clouds: Using Docker, Kubernetes & Google Cloud Platform. *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*, 184-189. <https://doi.org/10.1109/CCWC.2019.8666479>
- Shahiri, H., & Adnan, S. (2015). The Relevancy of Co-Curricular Activities on Graduates Employability: Evidence from a Graduate Unemployment Duration Model. *Prosiding Perkem, 10*, 93-98.

- Shrestha, A. K., Deters, R., & Vassileva, J. (2017). User-Controlled Privacy-Preserving User Profile Data Sharing based on Blockchain. *Future Technologies Conference (FTC) 2017*, 31-40.
- Silva, F., Humberto Rui, P., Goncalves, M. J., Ferreira da Silva, A., & Silva, M. (2020). Exploring the Innovative Aspects of CV Distributed Ledgers Based on Blockchain. *WorldCIST 2020*, 14-24. https://doi.org/10.1007/978-3-030-45691-7_2
- Sobrido Prieto, M., Valverde, T., & Ángel, M. (2019). Nuevos modelos de currículum vitae en la era digital. *Index de Enfermería*, 27(3).
https://scielo.isciii.es/scielo.php?script=sci_arttext&pid=S1132-12962018000200010&lang=es
- Solanki, M. R. (2020). SOLID: A Web System to Restore the Control of Users' Personal Data. *ICT Systems and Sustainability*, 1, 257-268. <https://doi.org/10.1007/978-981-15-8289-9>
solidproject.org. (6 de enero de 2021). *Solid*. Solid Servers and Pods:
<https://solidproject.org/about>
- Song, J., Demir, M., Prevost, J., & Rad, P. (2018). Blockchain Design for Trusted Decentralized IoT Networks. *13th Annual Conference on System of Systems Engineering (SoSE)*, 169-174. <https://doi.org/10.1109/SYSOSE.2018.8428720>
- Steiner, T. (2018). What is in a Web View? An Analysis of Progressive Web App Features When the Means of Web Access is not a Web Browser. *Companion Proceedings of the Web Conference 2018*, 27(28), 789-796. <https://doi.org/10.1145/3184558.3188742>
- Tao, X., Esposito, F., Sacco, A., & Marchetto, G. (2019). A Policy-Based Architecture for Container Migration in Software Defined Infrastructures. *2019 IEEE Conference on Network Softwarization (NetSoft)*, 198-202.
<https://doi.org/10.1109/NETSOFT.2019.8806659>

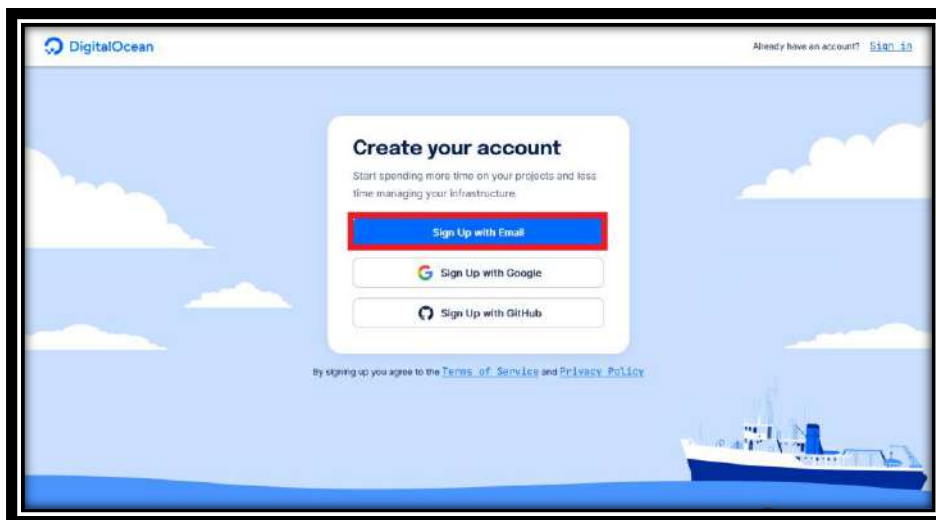
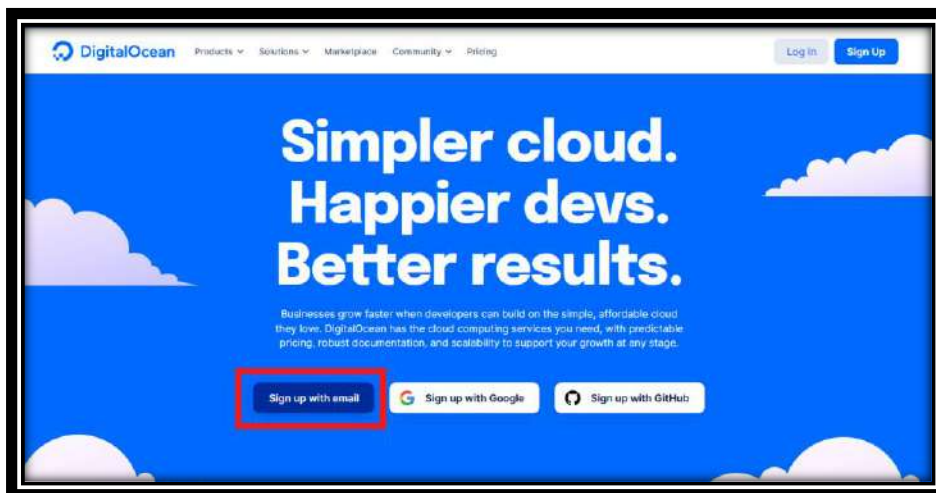
- Tapscott, D. (29 de enero de 2018). *A Declaration of Interdependence: Toward a New Social Contract for the Digital Economy*. Blockchain Research Institute:
https://resources.blockchainresearchinstitute.org/hubfs/Creative%20Commons/Tapscott%20Declaration%20of%20Interdependence_New%20Social%20Contract_Blockchain%20Research%20Institute.pdf
- Teixeira da Silva, J., Dobránszki, J., Al-Khatib, A., & Tsigaris, P. (2020). Curriculum vitae: challenges and potential solutions. *KOME – An International Journal of Pure Communication Inquiry*, 8, 109-127. <https://doi.org/10.17646/KOME.75672.52>
- Telschig, K., Schönberger, A., & Knapp, A. (2018). A Real-Time Container Architecture for Dependable Distributed Embedded Applications. *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*, 1367-1374.
<https://doi.org/10.1109/COASE.2018.8560546>
- Tingting, J. Z., Fritzler, A., & Jan, O. (noviembre de 2018). DATA INSIGHTS: JOBS, SKILLS AND MIGRATION TRENDS METHODOLOGY & VALIDATION RESULTS. En *World Bank Group | LinkedIn* (pág. 94).
<https://documents1.worldbank.org/curated/en/827991542143093021/pdf/World-Bank-Group-LinkedIn-Data-Insights-Jobs-Skills-and-Migration-Trends-Methodology-and-Validation-Results.pdf>
- Truong, N. B., Sun, K., Lee, G. M., & Guo, Y. (2020). GDPR-Compliant Personal Data Management: A Blockchain-based Solution. *IEEE TRANSACTION ON INFORMATION FORENSICS AND SECURITY*, 15, 1746-1761.
<https://doi.org/10.1109/TIFS.2019.2948287>.
- W3C Solid Community Group. (7 de Julio de 2021). *Solid Protocol*. Definitions:
<https://solidproject.org/TR/protocol#definitions>

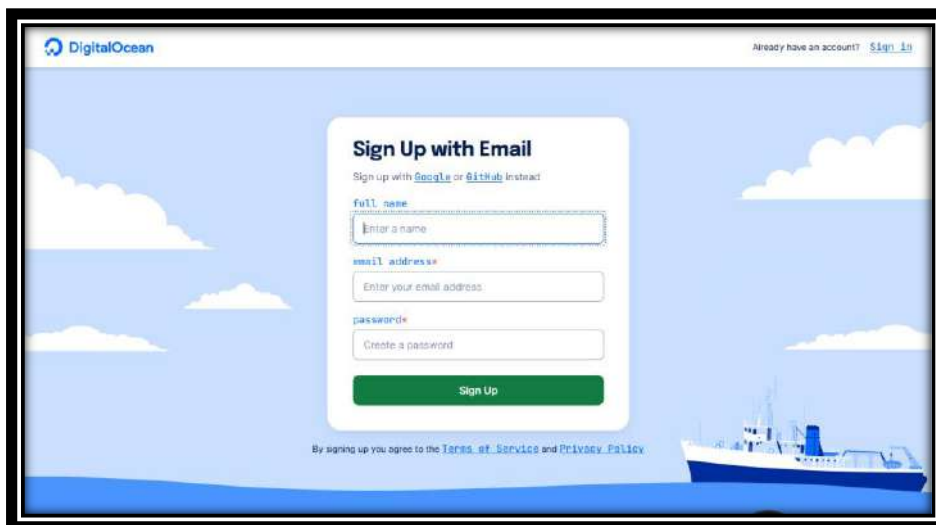
Wijayanti, S., & Sulistyarningsih, E. (2018). Format, Design and Content of Curriculum Vitae in a Developing Country. *Advances in Social Science, Education and Humanities Research / 1st Social and Humaniora Research Symposium (SoRes 2018)*, 307, 328-332.

<https://doi.org/10.2991/sores-18.2019.76>

Anexos

Anexo 1: Manual de instalación para servidor Solid dentro de DigitalOcean





DigitalOcean Already have an account? [Sign In](#)

Sign Up with Email

Sign up with [Google](#) or [GitHub](#) instead

full_name
Enter a name

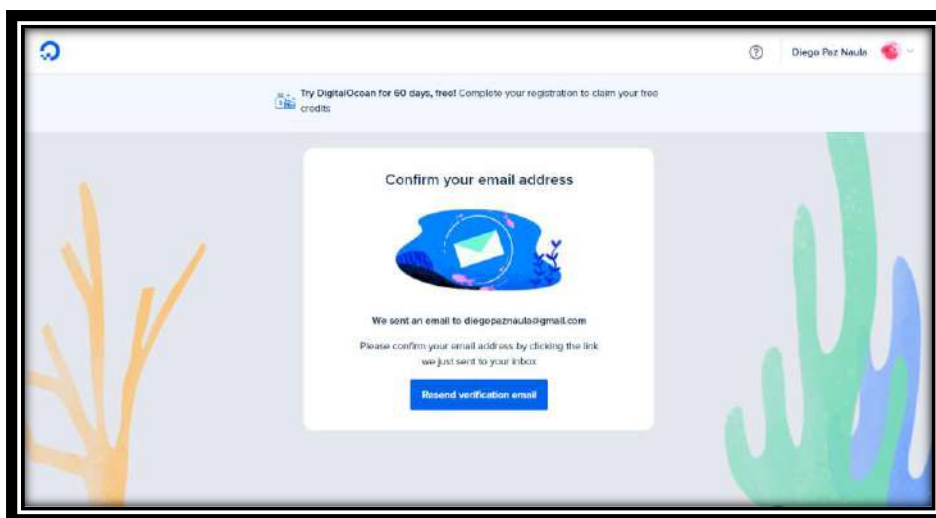
email_address*
Enter your email address

password*
Create a password

[Sign Up](#)

By signing up you agree to the [Terms of Service](#) and [Privacy Policy](#)


(Background illustration: a blue sky with white clouds and a white ship on a blue sea.)



? Diego Piz Naulo

Try DigitalOcean for 60 days, free! Complete your registration to claim your free credits

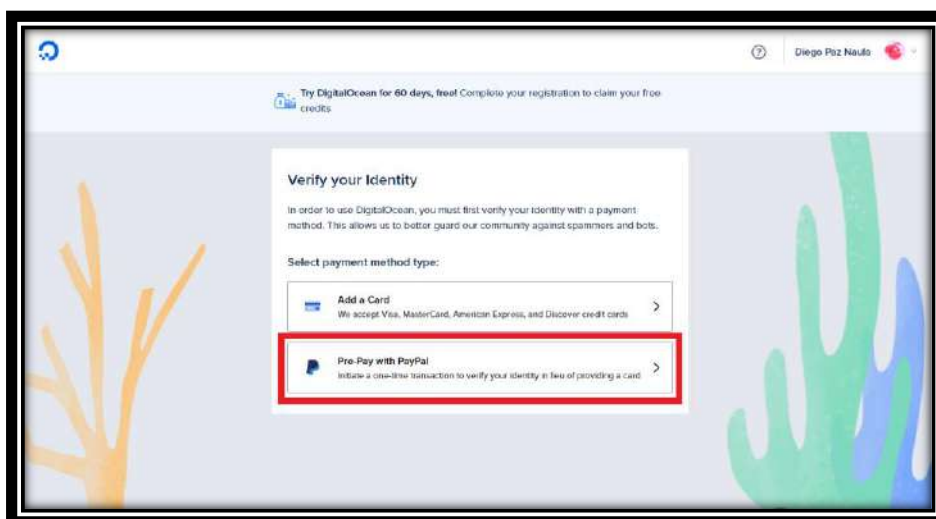
Confirm your email address



We sent an email to diegopeznaulo@gmail.com
Please confirm your email address by clicking the link we just sent to your inbox.

[Resend verification email](#)

(Background illustration: stylized coral reefs in orange and green.)



? Diego Piz Naulo

Try DigitalOcean for 60 days, free! Complete your registration to claim your free credits

Verify your Identity

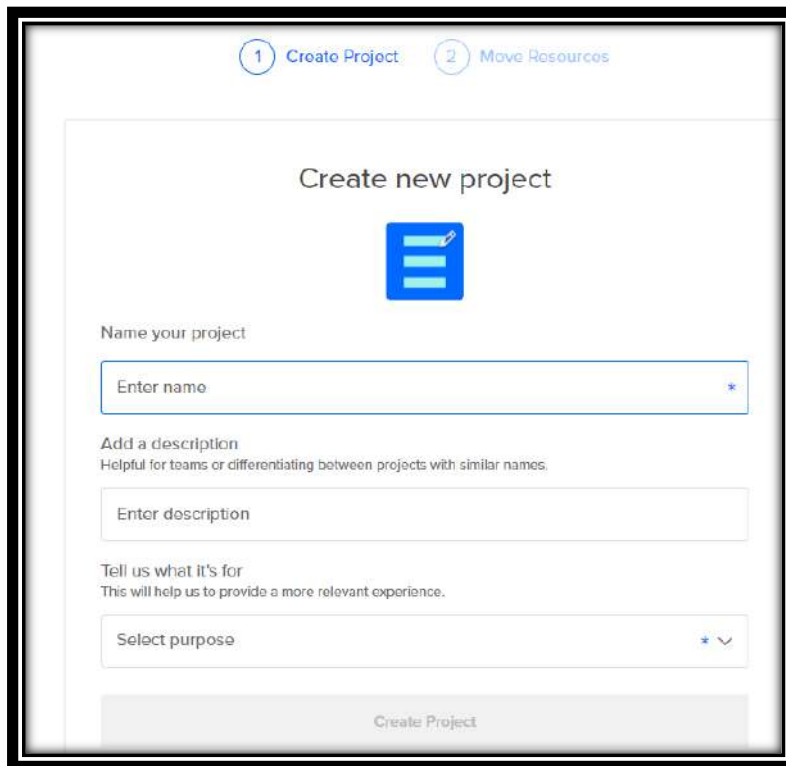
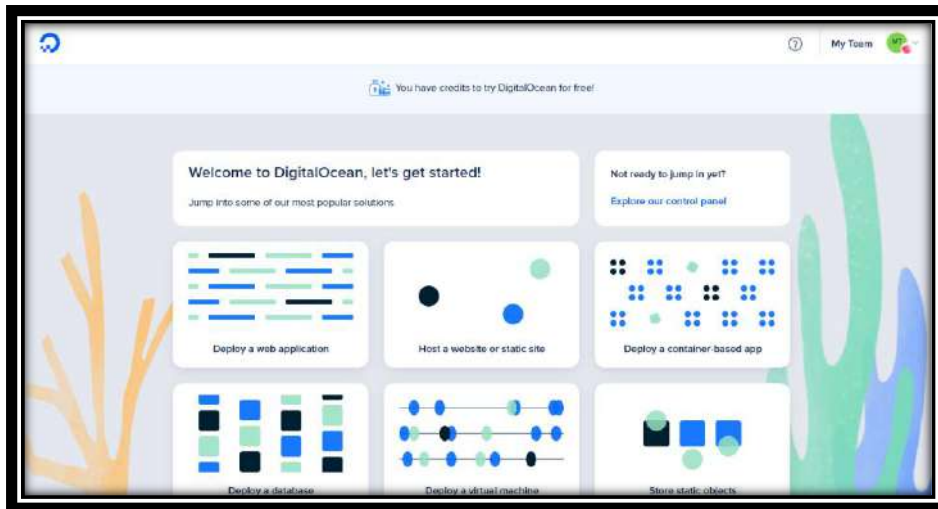
In order to use DigitalOcean, you must first verify your identity with a payment method. This allows us to better guard our community against spammers and bots.

Select payment method type:

- [Add a Card](#)
We accept Visa, MasterCard, American Express, and Discover credit cards >
- [Pre-Pay with PayPal](#)
Initiate a one-time transaction to verify your identity in lieu of providing a card >

(The 'Pre-Pay with PayPal' option is highlighted with a red border.)

(Background illustration: stylized coral reefs in orange and green.)



1 Create Project 2 Move Resources

Create new project

Name your project

Enter name
solid-server ✓

Add a description
Helpful for teams or differentiating between projects with similar names.

Enter description
Solid server to test application

Tell us what it's for
This will help us to provide a more relevant experience.

Web Application * v

Create Project

Agregar más recurso no saltar por ahora

1 Create Project 2 Move Resources

Move resources into solid-server

Move existing Droplets, Spaces, load balancers, domains or floating IPs into your new project.

Start typing to find and select existing

Move Resources

Skip for now

Agregamos un servidor o como lo llama digital ocean un droplet

Search by resource name or public IP (Ctrl+B)

Create v ? 🔔 My Team Credit: \$100.00 (expires in 59 days)

solid-server
Web Application / Solid server to test application → Move Resources

Resources Activity Settings

The sidebar contains a 'Create' button at the top, followed by a list of services: Droplets (highlighted with a red box), Kubernetes, Apps, Databases, Volumes, Spaces, Domains/DNS, Cloud Firewalls, Floating IPs, Load Balancers, and Resource Alerts.

Create Droplets

Choose an image

Distributions Container distributions Marketplace Custom Images

Search keyword [See all Marketplace Apps](#)

Recommended for you

- WordPress 5.8 on Ubuntu 20.04
- Plesk 18.0 on Ubuntu 20.04
- cPanel & WHM® RELEASE on ...
- Docker 19.03.12 on Ubuntu 20...
- LAMP on Ubuntu 20.04
- Laravel 7.20.0 on Ubuntu 20.04

Choose a plan

SHARED CPU DEDICATED CPU

Basic General Purpose CPU-Optimized Memory-Optimized Storage-Optimized

Basic virtual machines with a mix of memory and compute resources. Best for small projects that can handle variable levels of CPU performance, like blogs, web apps and dev/test environments.

CPU options: Regular with SSD Premium Intel with NVMe SSD Premium AMD with NVMe SSD

Plan	Price	CPU	RAM	Disk	Transfer
Basic	\$5/mo	1 GB / 1 CPU	25 GB SSD Disk	1000 GB transfer	
General Purpose	\$10/mo	2 GB / 1 CPU	50 GB SSD Disk	2 TB transfer	
CPU-Optimized	\$15/mo	2 GB / 2 CPUs	60 GB SSD Disk	3 TB transfer	
Memory-Optimized	\$20/mo	4 GB / 2 CPUs	80 GB SSD Disk	4 TB transfer	
Storage-Optimized	\$40/mo	8 GB / 4 CPUs	160 GB SSD Disk	5 TB transfer	
Storage-Optimized	\$80/mo	16 GB / 8 CPUs	320 GB SSD Disk	6 TB transfer	

Basic Droplet plans have replaced Standard Droplet plans. Users who have Standard Droplets using deprecated plans can still create Droplets with the API.

Each Droplet plan includes free outbound data transfer which is shared between all Droplets each billing cycle. Inbound bandwidth to Droplets is always free. [Learn more or try our price calculator.](#)

Add block storage

Add Volume

Choose a datacenter region

Region	Count
New York	1
San Francisco	3
Amsterdam	3
Singapore	1
London	1
Frankfurt	1
Toronto	1
Bangalore	1

Create Droplets

Choose an image

Distributions Container distributions Marketplace Custom images

Ubuntu

20.04 LTS x64

FreeBSD

Select version

Fedora

Select version

Debian

Select version

CentOS

Select version

Rocky Linux

Select version

Choose a plan Help me choose

SHARED CPU DEDICATED CPU

Basic General Purpose CPU-Optimized Memory-Optimized Storage-Optimized

Basic virtual machines with a mix of memory and compute resources. Best for small projects that can handle variable levels of CPU performance, like blogs, web apps and dev/test environments.

CPU options: Regular with SSD Premium Intel with NVMe SSD new Premium AMD with NVMe SSD new

Plan	Price	CPU	RAM	Disk	Transfer
Basic 1	\$5/mo \$0.007/hour	1 GB / 1 CPU	35 GB SSD Disk	1000 GB transfer	
Basic 2	\$10/mo \$0.015/hour	2 GB / 1 CPU	50 GB SSD Disk	2 TB transfer	
Basic 3	\$15/mo \$0.022/hour	2 GB / 2 CPUs	60 GB SSD Disk	3 TB transfer	
Basic 4	\$20/mo \$0.030/hour	4 GB / 2 CPUs	80 GB SSD Disk	4 TB transfer	
Basic 5	\$40/mo \$0.050/hour	8 GB / 4 CPUs	160 GB SSD Disk	5 TB transfer	
Basic 6	\$80/mo \$0.100/hour	16 GB / 8 CPUs	320 GB SSD Disk	6 TB transfer	

Basic Droplet plans have replaced Standard Droplet plans. Users who have Standard Droplets using deprecated plans can still create Droplets with those plans using the API.

Each Droplet plan includes free outbound data transfer which is shared between all Droplets each billing cycle. Inbound bandwidth to Droplets is always free. [Learn more](#) or [try our price calculator](#).

Add block storage

[Add Volume](#)

Choose a datacenter region

New York 1 2 3	San Francisco 1 2 3	Amsterdam 2 3	Singapore 1	London 1	Frankfurt 1
Toronto 1	Bangalore 1				

VPC Network

default-vc3 DEFAULT

All resources created in this datacenter will be members of the same VPC network. They can communicate securely over their Private IP addresses. [What does this mean?](#)

Heads up

Private networking is now automatically enabled. You can create new networks or just use the default.

OK [Learn more](#)

Crear llave ssh

Authentication

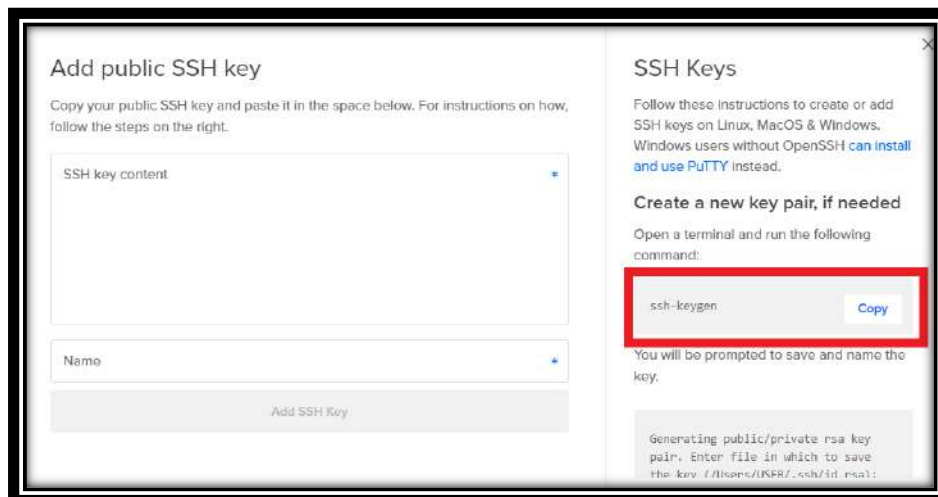
SSH keys
 A more secure authentication method

Password
 Create a root password to access Droplet (less secure)

Choose your SSH keys ❗ Select at least one key.

[New SSH Key](#)

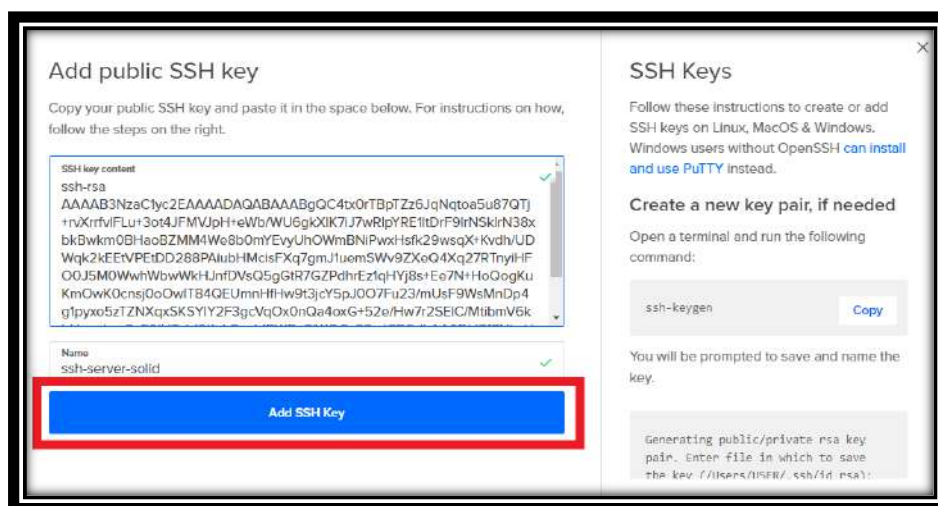
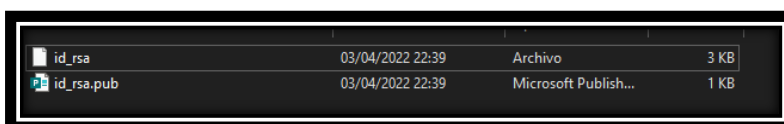
Abrimos una terminal y pegamos el comando “ssh-keygen”



En la terminal nos pide el directorio donde lo vamos a guardar, una frase para que la contraseña se más única y repetir la misma frase

```
H:\SolidProject\Solid-Server-DigitalOcean>ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\Intel Ext
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
```

Se crean dos archivos, .pub llave pública la cual debemos pegar en digital ocean



Finalize and create

How many Droplets?
Deploy multiple Droplets with the same configuration.

Choose a hostname
Give your Droplets an identifying name you will remember them by. Your Droplet name can only contain alphanumeric characters, dashes, and periods.

1 Droplet

solid-server-dipaz

Add tags
Use tags to organize and relate resources. Tags may contain letters, numbers, colors, dashes, and underscores.

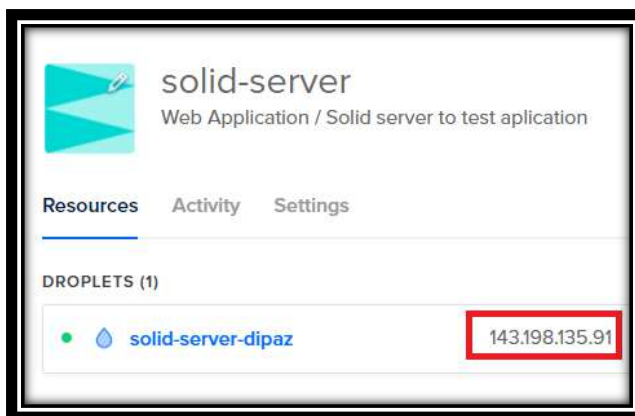
Typo tags here

Select Project
Assign Droplets to a project.

solid-server

Create Droplet

Progreso de creación



En el caso de saber usar la terminal para ingresar mediante ssh vamos una terminal y con la ip copiada ingresamos donde pondremos usuario root y la contraseña se pone gracias a la llave generada anteriormente

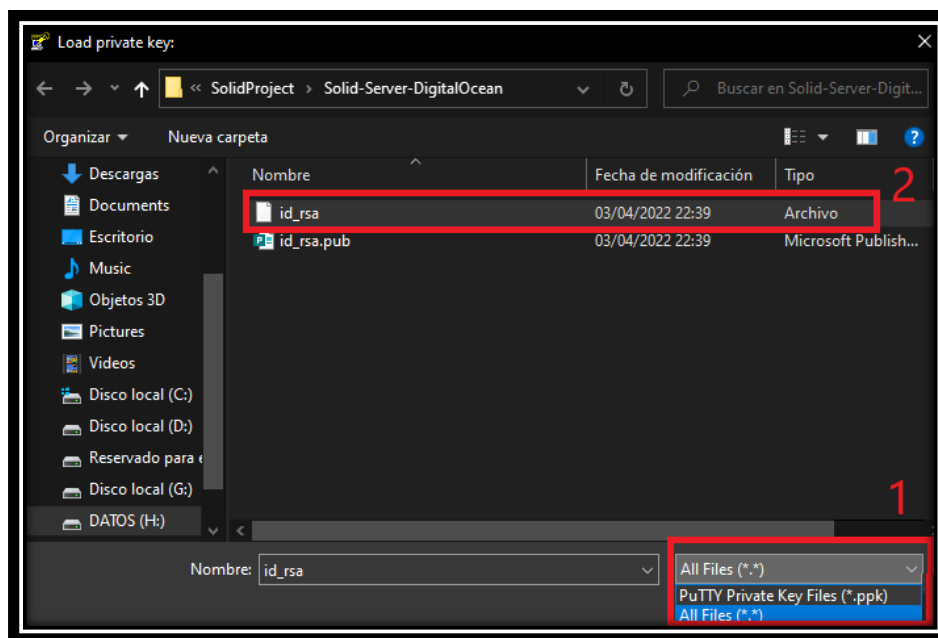
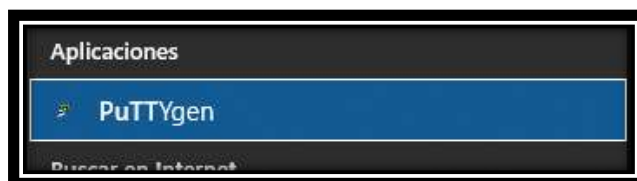
```

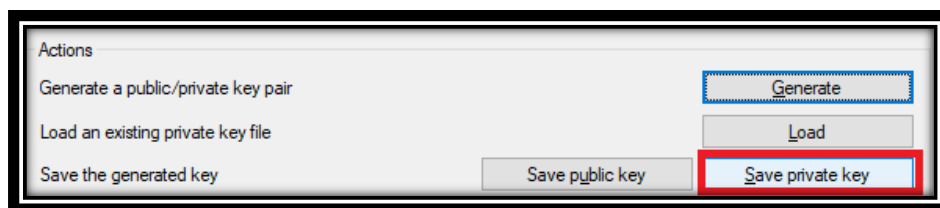
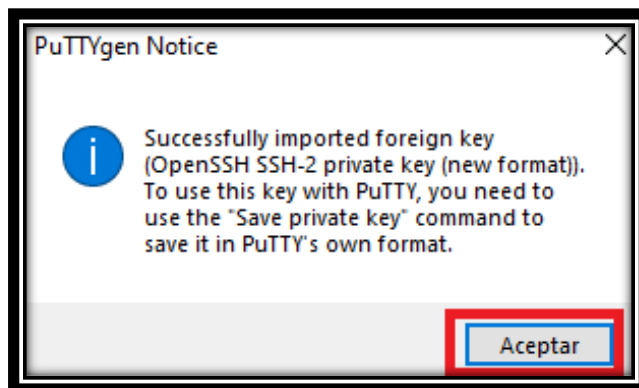
Intel Extreme@DESKTOP-70ILM6C MINGW64 ~
$ ssh root@143.198.135.91
The authenticity of host '143.198.135.91 (143.198.135.91)' can't be established.
ECDSA key fingerprint is SHA256:6CtDsZVSp2WK7QuI4xBkiyRcxORA5/TVxxzSWKdZFOY.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '143.198.135.91' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-97-generic x86_64)

```

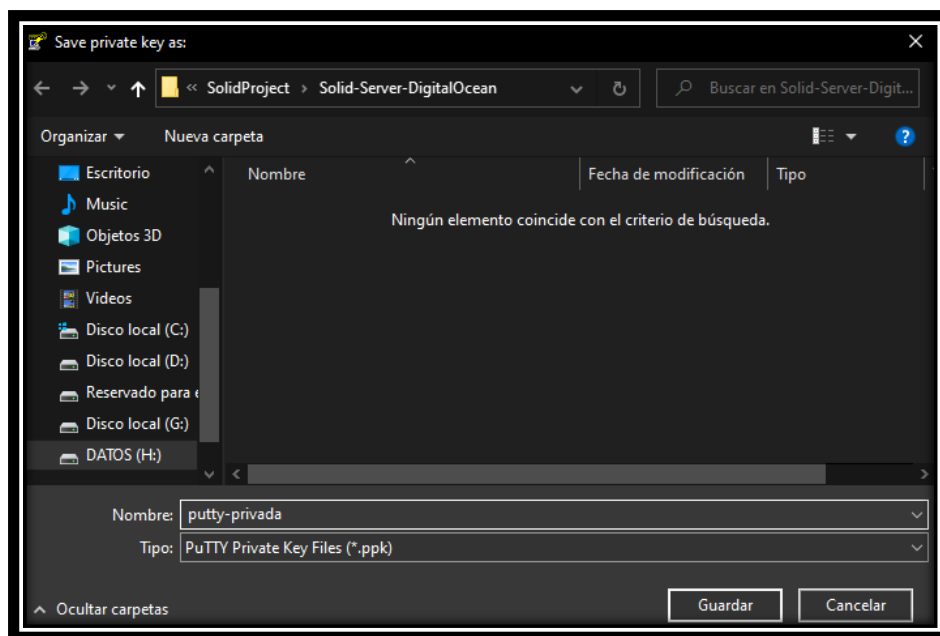
En el caso de usuarios novatos usaremos putty y puttygen

Abrimos puty gen y añadimos la llave privada nota la llave privada no se debe compartir con nadie

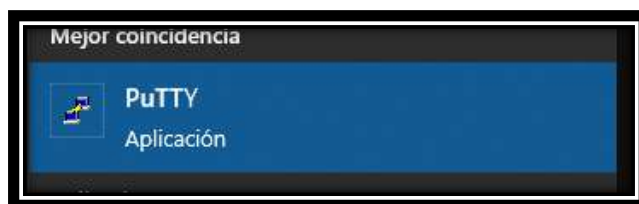




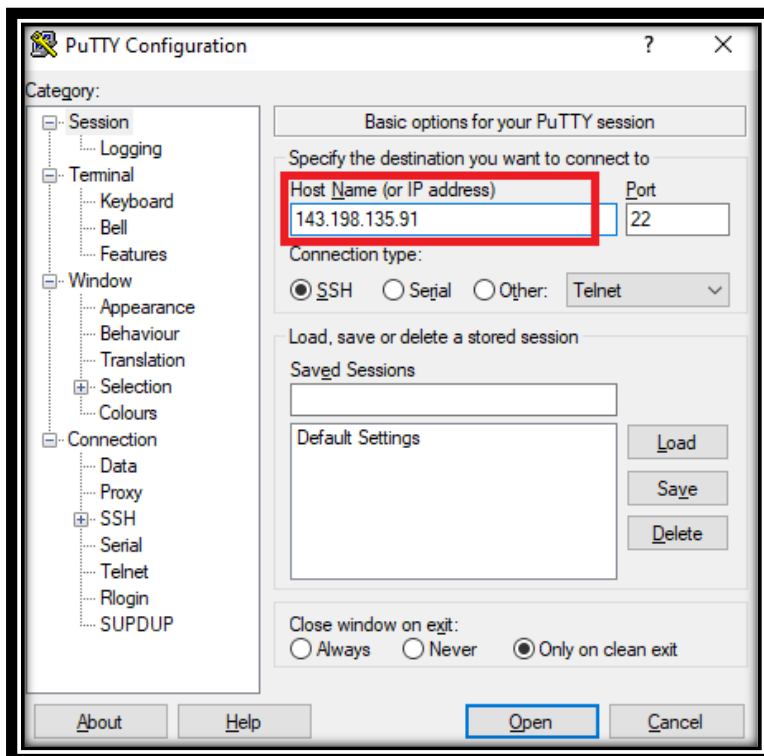
Guardamos las credenciales de putty



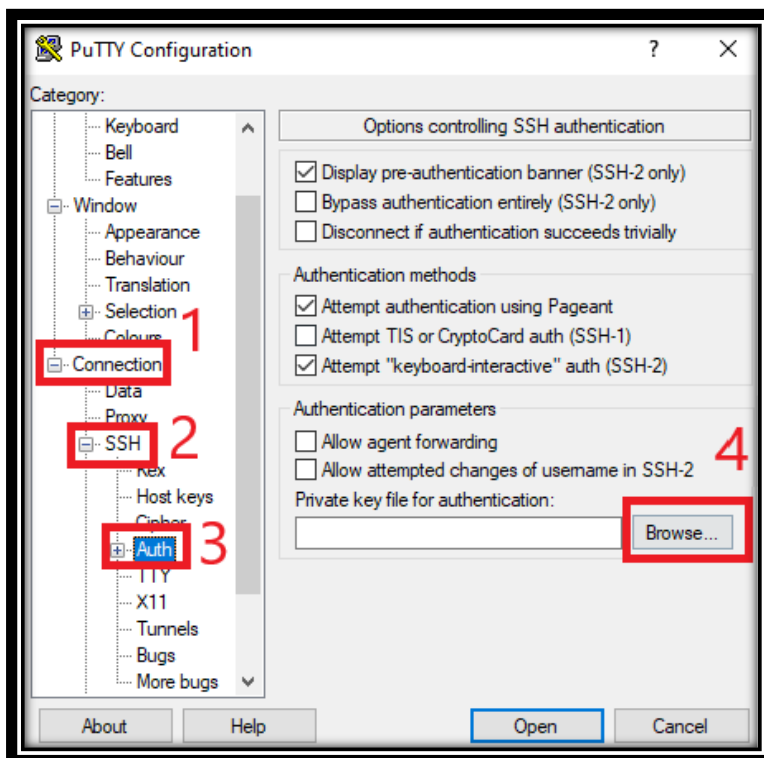
Ahora abrimos putty

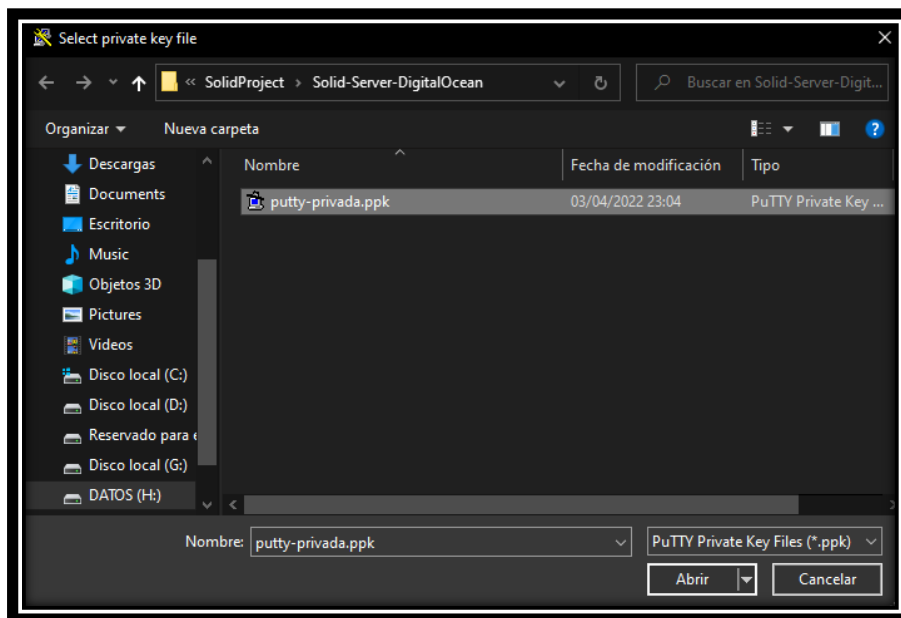


En la primera venta pegamos la ip que nos proporciona Digital Ocean

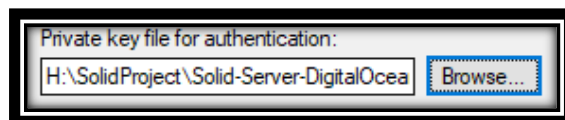


Seguimos para agregar la llave agregada en putty





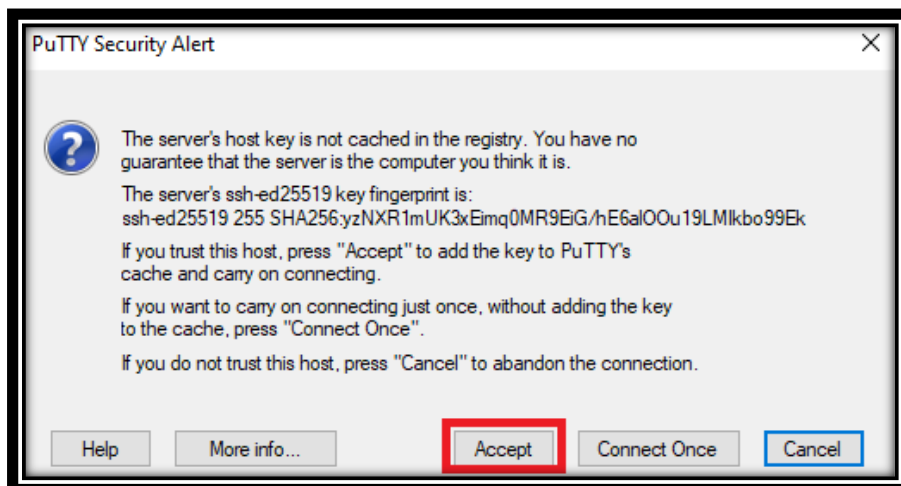
Podemos ver que se agregó la ruta



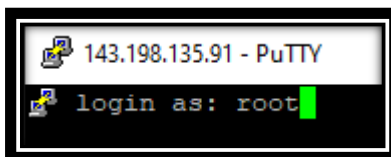
Damos clic en open



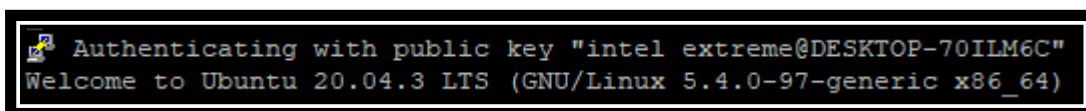
Clic en aceptar



Ingresamos con usuario root



Esperamos la autenticación y habremos ingresado



Una vez ingresado descargaremos las nuevas actualizaciones del servidor

```
root@solid-server-dipaz:~# sudo apt update
```

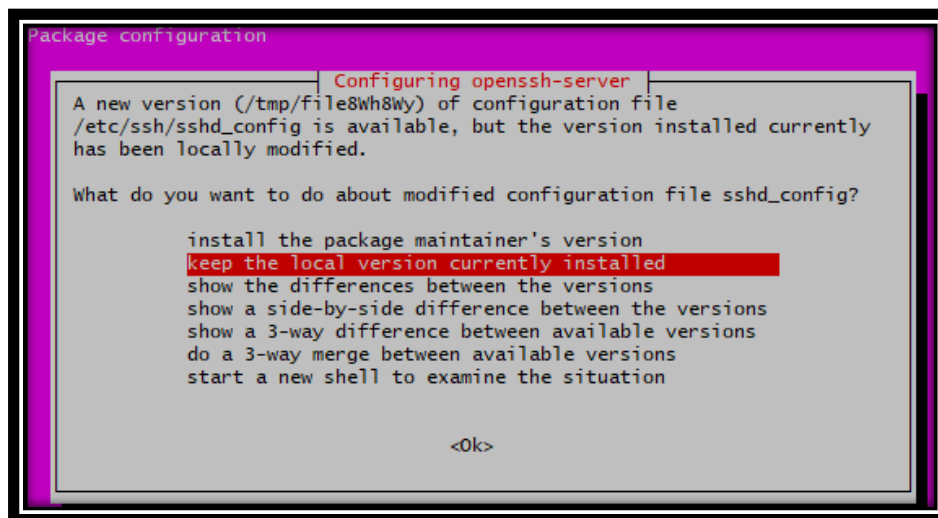
Ahora ejecutaremos tales actualizaciones

```
root@solid-server-dipaz:~# sudo apt upgrade
```

Se nos notifica la cantidad de espacio en disco que ocuparan y si deseamos continuar escribimos la letra "Y" continuando con la descarga de paquetes posteriormente instalandolos

```
After this operation, 203 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Dejamos por defecto para no dañar la parte local



Vamos a agregar el servidor web "nginx" damos a la letra "Y" para continuar con la instalación

```
root@solid-server-dipaz:~# sudo apt install nginx
```

Para comprobar si todo va bien podemos copiar la dirección ip del servidor de digital ocean y pegar en el navegador debería verse una página similar a la imagen que se ve a continuación



Activamos HTTP y HTTPS

```
root@solid-server-dipaz:~# sudo ufw allow 'Nginx HTTP'
```

```
root@solid-server-dipaz:~# sudo ufw allow 'Nginx HTTPS'
```

Creamos archivo de redirección proxy inverso

```
root@solid-server-dipaz:~# nano /etc/nginx/sites-available/dipaz-solid.online
```

```
server{
    listen 80;
    server_name dipaz-solid.online www.dipaz-solid.online

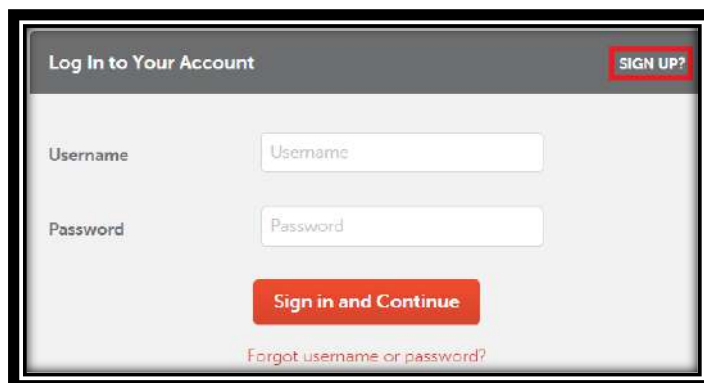
    location / {
        proxy_pass http://127.0.0.1:3000
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
    }
}
```

Enlazamos el archivo creado para sitios habilitados dentro de los sitios activados

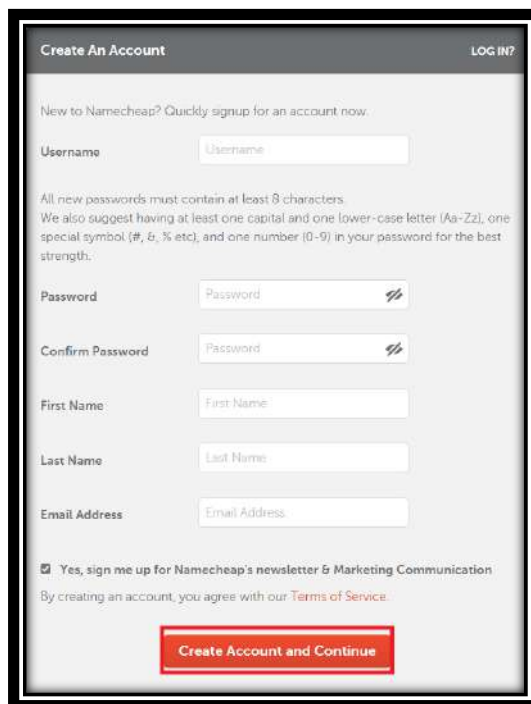
```
root@solid-server-dipaz:~# ln /etc/nginx/sites-available/dipaz-solid.online /etc/nginx/sites-enabled/dipaz-solid.online
```


Anexo 2: Compra de dominio y certificado ssl

Agregar dominio y https ingresar a “namecheap”



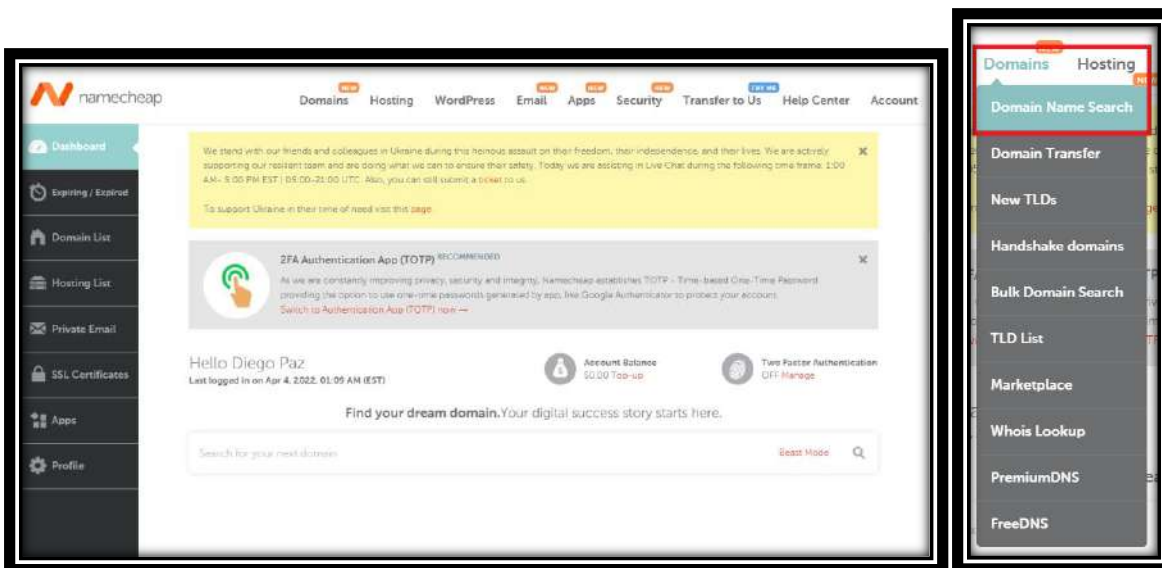
The screenshot shows the Namecheap login page. The title is "Log In to Your Account" and there is a "SIGN UP?" link in the top right corner. The form has two input fields: "Username" and "Password". Below the input fields is a red button labeled "Sign in and Continue". At the bottom of the form, there is a link that says "Forgot username or password?".



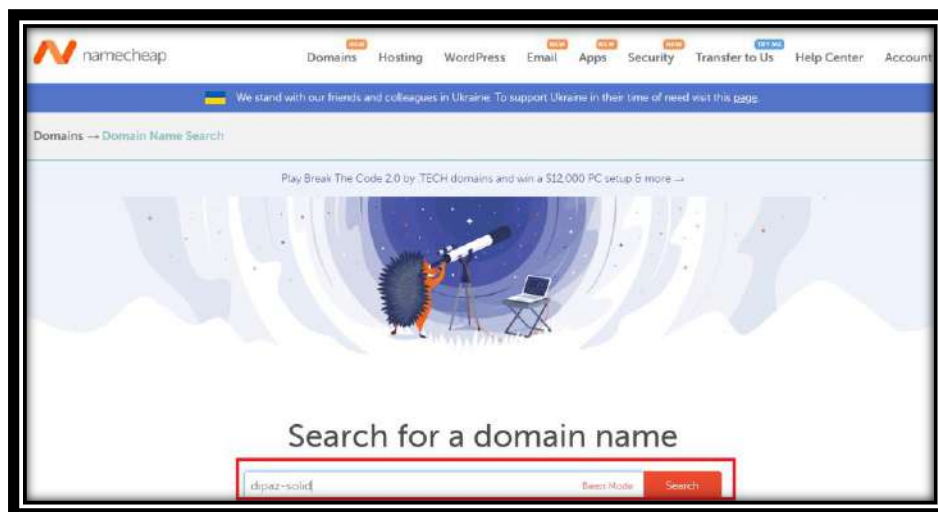
The screenshot shows the Namecheap account creation page. The title is "Create An Account" and there is a "LOG IN?" link in the top right corner. The form has several input fields: "Username", "Password", "Confirm Password", "First Name", "Last Name", and "Email Address". Below the input fields, there is a checkbox labeled "Yes, sign me up for Namecheap's newsletter & Marketing Communication". Below the checkbox, there is a link that says "By creating an account, you agree with our Terms of Service.". At the bottom of the form, there is a red button labeled "Create Account and Continue".

Una vez ingresado tendremos el siguiente dashboard

Nos colocamos en domains y clic en domain name search para buscar un dominio



Ingresamos un nombre sugestivo



Seleccionamos el que mejor se acomode a nosotros

Añadimos al carro de compras

Y agregamos el certificado ssl para tener uso de https

Suggested Results [Hide](#)

dipaz-solid.com	\$8.98/yr Retail \$13.98/yr	Add to cart
dipaz-solid.net	\$9.98/yr Retail \$12.98/yr	Add to cart
dipaz-solid.org	\$7.48/yr Retail \$12.98/yr	Add to cart
dipaz-solid.ai	\$69.98/yr Retail \$79.98/yr	Add to cart
dipaz-solid.to	\$29.98/yr Retail \$39.98/yr	Add to cart

Results [Explore More](#)

dipaz-solid.inc	\$1,098.00/yr Retail \$2,098.00/yr	Add to cart
dipaz-solid.xyz	\$1.00/yr Retail \$10.00/yr	Add to cart
dipaz-solid.art	\$3.88/yr Retail \$10.98/yr	Add to cart

dipaz-solid.online **\$8.88 FOR 2 YEARS** **\$1.88/yr** Retail \$32.98/yr [Add to cart](#)

dipaz-solid.online **\$8.88 FOR 2 YEARS** **\$1.88/yr** Retail \$32.98/yr [Add to cart](#)

Frequently bought together [Hide](#)

SSL Site security made simple	\$4.98/yr	Add to cart	VPN Get Safe & Fast Browsing!	Free trial	Add to cart
WordPress Hosting Managed WordPress Hosting	\$24.88/yr	Add to cart	Professional Email Custom email@your-domain	Free trial	Add to cart
Web Hosting Website Builder included	\$21.88/yr	Add to cart	Premium DNS DNS Speed & Security	\$4.88	Add to cart

Una vez listo continuamos con la orden

namecheap

[Bulk Settings](#) [Edit Cart](#)

Domain Registration	dipaz-solid.online	1 Year	Auto-renew	552.98 \$1.88	84% OFF	Remove
	ICANN fee			\$0.18		
Privacy and Uptime Protection	Domain Privacy [*]	1 year subscription	ON/OFF	\$0.00	FREE FOREVER!	
	Premium DNS ^{NEW}	1 Year	ON/OFF	\$4.88		
	PositiveSSL	1 Year	ON/OFF	\$12.00 \$4.98		Remove

Your Subtotal

Subtotal \$7.04

[Confirm Order](#)

Promo Code

98DOTCOM **REMOVE**

COM for \$0.98 for 1st-year (2-years registration)

Account Contact Information

First Name:

Last Name:

I'm registering on behalf of a company

Address Line 1:

Address Line 2:

City:

State/Province:

Zip/Postal Code:

Country:

Phone Number:

Add phone extension

Fax Number: OPTIONAL

Email Address:

Your Cart

	EDIT
dipaz-solid.online 1 year registration	\$1.88
ICANN fee	\$0.10
PositiveSSL 1 year	\$4.98
Subtotal	\$7.04

Continue

Whois Contact Information

ICANN requires the Whois database to store owners' contact information for all domains registered. [Learn more →](#)

Registrant Contact:

Diego Paz
Ambrosio Acosta, Prudencio Vasquez
Quito, Pichincha 170708
EC

tel: +593 980491064
fax: diegopaznaula@gmail.com

Administrative Contact:

Diego Paz
Ambrosio Acosta, Prudencio Vasquez
Quito, Pichincha 170708
EC

tel: +593 980491064
fax: diegopaznaula@gmail.com

Technical Contact:

Diego Paz
Ambrosio Acosta, Prudencio Vasquez
Quito, Pichincha 170708
EC

tel: +593 980491064
fax: diegopaznaula@gmail.com

Billing Contact:

Diego Paz
Ambrosio Acosta, Prudencio Vasquez
Quito, Pichincha 170708
EC

tel: +593 980491064
fax: diegopaznaula@gmail.com

Your Cart

	EDIT
dipaz-solid.online 1 year registration	\$1.88
ICANN fee	\$0.10
PositiveSSL 1 year	\$4.98
Subtotal	\$7.04

Continue

Forma de pago

Receipt Details

We'll provide your receipt. Please confirm the address.

Billing Address Use default account contact

Diego Paz
Ambrosio Acosta
Prudencio Vasconez
Quito, Pichincha, 170708
EC

tel: +593 960491064
diegopezmaule@gmail.com

Renewal Settings for Your Purchase

Some items in your purchase are eligible for auto-renewal. We've set all eligible items to auto-renew. If you'd prefer to personally select which items to auto-renew, please uncheck the box below.

Automatically renew all of my eligible new services before expiration. **RECOMMENDED**

Domain Registration dipaz-solid.online

Free Domain Privacy for dipaz-solid.online

PremiumDNS for dipaz-solid.online

Save the configuration above to my default payment settings

We'll save this information for future use.

[Continue](#)

Your Cart

[EDIT](#)

dipaz-solid.online	
1 year registration	\$1.88
ICANN fee	\$0.18
<hr/>	
PositiveSSL	
1 year	\$4.98
<hr/>	
Subtotal	\$7.04

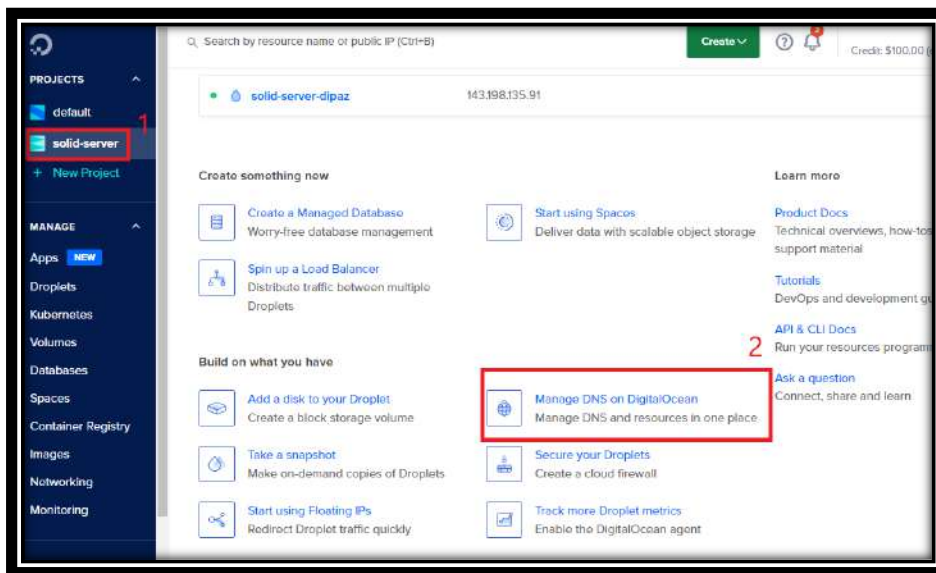
[Continue](#)

Después del pago tenemos el nombre del dominio y lo más importante el certificado ssl

Domain List REFRESH				Domains
Domains	Status	Auto-Renew	Expiration	
<input type="checkbox"/> dipaz-solid.online Domain Privacy protection is ON	ACTIVE	<input type="checkbox"/>	Apr 4, 2023	MANAGE

ID	Name	Status	Purchased for	Time remaining	
17976006	PositiveSSL <small>To use an SSL you first need to activate it and then install it.</small>	NEW	1 yr	.365 days	ACTIVATE

Abrimos digital ocean



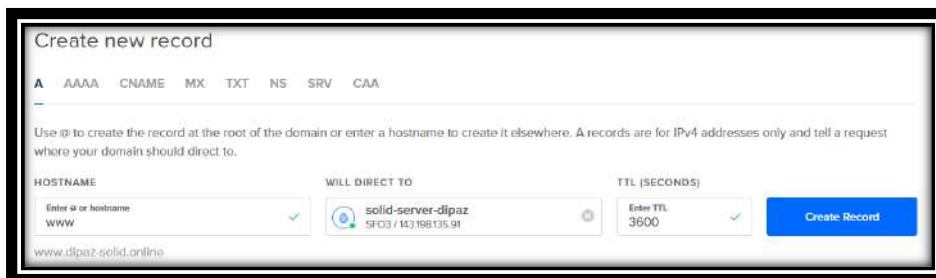
Agregamos el nombre del dominio



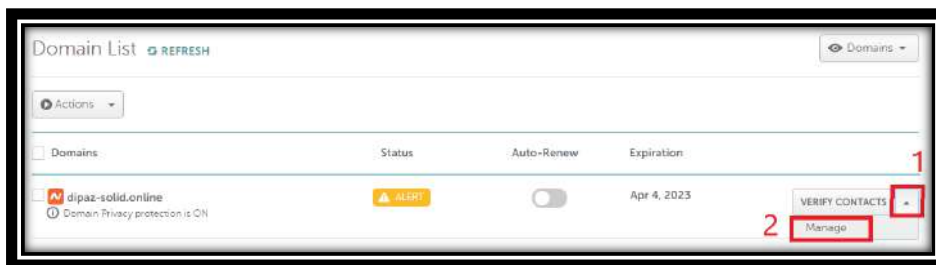
Creamos la redirección de nuestra aplicación



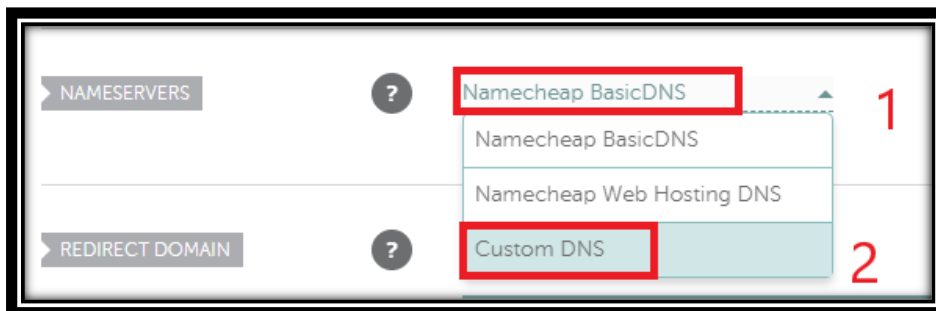
De la misma manera el www



Ahora en namecheap vamos a agregar los dominios de digital ocean



Bajamos hasta encontrar nameservers y damos clic eligiendo custom DNS



Quedando de la siguiente manera, ahora nuestro dominio está apuntando a digital ocean para que la configuración esté listo toma alrededor de 20 minutos a 1 hora



Anexo 3: Instalación de servidor solid

Entramos en nuestro cli dentro del nuevo droplets para comprobar si se tiene lo necesario que es Docker y GitHub

```
root@solid-server-dipaz:~# docker --version
Docker version 20.10.7, build f0df350
root@solid-server-dipaz:~# git --version
git version 2.25.1
root@solid-server-dipaz:~#
```

Crear carpeta para solid server

```
root@solid-server-dipaz:~# mkdir solid_server
```

Copiamos repositorio de solid server

```
root@solid-server-dipaz:~# cd solid_server/
root@solid-server-dipaz:~/solid_server# git clone https://github.com/solid/community-server.git
```

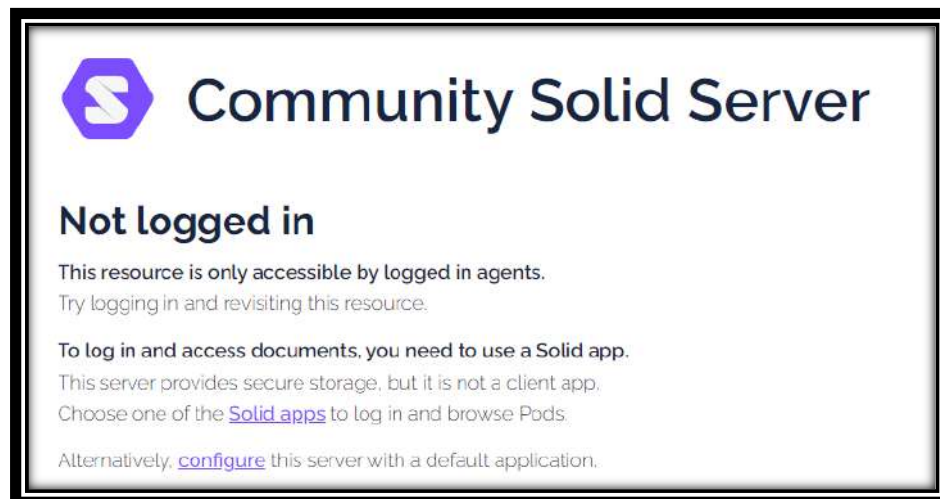
Montamos el proyecto dockerizado

```
root@solid-server-dipaz:~/solid_server# cd community-server/
root@solid-server-dipaz:~/solid_server/community-server# docker build --rm -f Dockerfile -t css:latest .
```

Se ejecuta en el puerto 3000

```
root@solid-server-dipaz:~/solid_server/community-server# docker run --rm -v ~/Solid:/data -p 3000:3000 -it css:latest
```

Obtenemos ventana de falta https



Agregar certificado SSL Certbot (gratis). Agregamos la librería para poder descargar paquetes de certbot. Sudo add-apt-repository ppa:certbot/certbot

```
root@solid-server-dipaz:~# sudo add-apt-repository ppa:certbot/certbot
The PPA has been DEPRECATED.

To get up to date instructions on how to get certbot for your systems, please see
https://certbot.eff.org/docs/install.html.
More info: https://launchpad.net/~certbot/+archive/ubuntu/certbot
Press [ENTER] to continue or Ctrl-c to cancel adding it.

Hit:1 https://repos.insights.digitalocean.com/apt/do-agent main InRelease
Hit:2 https://repos-droplet.digitalocean.com/apt/droplet-agent main InRelease
Get:3 http://mirrors.digitalocean.com/ubuntu focal InRelease [265 kB]
Get:4 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Ign:5 http://ppa.launchpad.net/certbot/certbot/ubuntu focal InRelease
Hit:6 https://download.docker.com/linux/ubuntu focal InRelease
```

Agregamos paquete para nginx

```
root@solid-server-dipaz:~# sudo apt install python3-certbot-nginx
```

El certificado caducara cada 3 meses se debe renovar.

```
root@solid-server-dipaz:~# sudo certbot --nginx -d dipaz-solid.online -d www.dipaz-solid.online
```

Correo para que llegen notificaciones de seguridad y aceptamos los términos

```
Enter email address (used for urgent renewal and security notices) (Enter 'c' to cancel): diegopaznaula@gmail.com
```

```
Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf. You must
agree in order to register with the ACME server at
https://acme-v02.api.letsencrypt.org/directory
-----
(A)gree/(C)ancel: A
```

Seleccionamos 2 que siempre queremos que se redireccione a https y renovar el ssl

```
Please choose whether or not to redirect HTTP traffic to HTTPS, removing HTTP access.
-----
1: No redirect - Make no further changes to the webserver configuration.
2: Redirect - Make all requests redirect to secure HTTPS access. Choose this for
new sites, or if you're confident your site works on HTTPS. You can undo this
change by editing your web server's configuration.
-----
Select the appropriate number [1-2] then [enter] (press 'c' to cancel): 2
```

```
root@solid-server-dipaz:~# certbot renew --dry-run
```

SSL Pagado

ID	Name	Status	Purchased for	Time remaining	
17976006	PositiveSSL ① To use an SSL you first need to activate it and then install it	NEW	1 yr	365 days	ACTIVATE