



UNIVERSIDAD TECNOLÓGICA CENTROAMERICANA

FACULTAD DE INGENIERÍA

PRÁCTICA PROFESIONAL

ACKLEN AVENUE | HERO UNIT

PREVIO A LA OBTENCIÓN DEL TÍTULO

INGENIERO EN SISTEMAS COMPUTACIONALES

PRESENTADO POR:

11641069 ILIANA MARÍA PINTO GALO

ASESOR: LIC. TANIA LUCILA MEZA AMADOR

CAMPUS TEGUCIGALPA; ABRIL, 2021

RESUMEN EJECUTIVO

El presente informe tiene como finalidad mostrar las actividades desarrolladas durante la práctica profesional en la empresa Acklen Avenue | Hero Unit de octubre del 2020 a marzo del 2021, donde se tuvo la oportunidad de conocer lo que se requiere en el ámbito laboral para un Ingeniero en Sistemas Computacionales.

Acklen Avenue | Hero Unit es una empresa que se ha dedicado a la entrega de software satisfaciendo las expectativas de clientes externos y cumpliendo con los objetivos, ha procurado entregar software de calidad con código limpio para la escalabilidad y el fácil mantenimiento. Igualmente cuenta con el desarrollo de varios proyectos internos.

En el año 2019 se comenzó con el desarrollo del proyecto interno Peeplo. Inicialmente esta aplicación era llamada Kiwi pero luego fue renombrada Peeplo. El propósito de esta aplicación consiste en permitir a los empleados agregar un registro de las actividades desarrolladas durante el día. En ella los administradores o los gerentes pueden tener un seguimiento de las actividades, los proyectos asignados y el tiempo libre de los empleados.

El puesto dentro de la empresa que se desempeñó fue el de desarrollador en el proyecto interno Peeplo. El trabajo realizado consistió en el desarrollo de nuevas funcionalidades, la corrección de errores y la implementación de nuevas herramientas para mejorar el flujo de trabajo. Cada actividad realizada fue aceptada por el equipo cumpliendo con los requisitos solicitados. Se utilizaron tecnologías como React, NestJS, Typescript, PostgreSQL, Smallchat y Zapier.

ÍNDICE DE CONTENIDO

I.	Introducción.....	1
II.	Generalidades de la Empresa.....	2
	2.1 Descripción de la Empresa.....	2
	2.2 Descripción del Departamento	2
	2.3 Objetivos del puesto.....	2
	2.3.1 Objetivo General.....	2
	2.3.2 Objetivos Específicos	2
III.	Marco Teórico.....	4
IV.	Desarrollo	12
	4.1 Descripción del Trabajo Realizado.....	12
	4.1.1 Inducción.....	12
	4.1.2 Entrenamiento con las Herramientas	13
	4.1.3 Introducción al Proyecto Peeplo	15
	4.1.4 Corrección de Redireccionamiento	15
	4.1.5 Eliminar al Primer Empleado Cuando se Crea una Nueva Organización	16
	4.1.6 Corrección de Error al Ordenar Fechas	17
	4.1.7 Implementación de un Chat de Soporte.....	18
	4.1.8 Implementación de un Indicador de Carga.....	19
	4.1.9 Implementación del Campo Contacto de Emergencia	20
	4.1.10 Implementación de la Herramienta Zapier	21
	4.1.11 Corrección de Cálculos en la Duración de Horas	26
	4.1.12 Mostrar la Duración de Horas en Formato Decimal.....	28
	4.2 Cronograma de Actividades.....	30

V.	Conclusiones.....	31
VI.	Recomendaciones.....	32
	Bibliografía.....	33

ÍNDICE DE ILUSTRACIONES

Ilustración 1 - Marco de Referencia Scrum.....	4
Ilustración 2 - Características de un Sprint.....	5
Ilustración 3 - Tablero Básico de Kanban.....	7
Ilustración 4 - Estructura Básica Front-end y Back-end.....	8
Ilustración 5 - Creación de un trigger	11
Ilustración 6 - Creación de una acción.....	11
Ilustración 7 - Insignias ganadas en el proceso de inducción	12
Ilustración 8 - Tabla de contenido para el curso de TypeScript Fundamentals	13
Ilustración 9 - Tabla de contenido para el curso de React Fundamentals	14
Ilustración 10 - Tabla de contenido para el curso de Node.js: Getting Started	14
Ilustración 11 - Modificación de una condición.....	15
Ilustración 12 - Documentación de cambios para eliminar al primer empleado	16
Ilustración 13 - Comportamiento original al ordenar las fechas	17
Ilustración 14 - Comportamiento al implementar el algoritmo customizado	18
Ilustración 15 - Implementación de un chat de soporte.....	19
Ilustración 16 - Validación para un indicador de carga.....	20
Ilustración 17 - Cambios realizados a la base de datos.....	20
Ilustración 18 - Endpoint para autenticar al usuario	21
Ilustración 19 - Endpoint para enviar un código de aprobación	22
Ilustración 20 - Endpoint para verificar credenciales	22
Ilustración 21 - Prueba de autenticación en Zapier	22
Ilustración 22 - Endpoint para enviar datos de prueba.....	23
Ilustración 23 - Datos de prueba enviados a Zapier.....	24
Ilustración 24 - Endpoint para obtener un webhook	25

Ilustración 25 - Endpoint para eliminar un webhook.....	25
Ilustración 26 - Llamado al endpoint de obtención de organizaciones.....	26
Ilustración 27 - Cálculos erróneos en la suma del total de horas	27
Ilustración 28 - Corrección en la suma del total de horas.....	28
Ilustración 29 - Cálculos en formato decimal.....	29
Ilustración 30 - Diagrama de Gantt.....	30

LISTA DE SIGLAS Y GLOSARIO

CEO	Director Ejecutivo
CRAI	Centro de Recursos para el Aprendizaje y la Investigación
CTO	Director de Tecnología
DOM	Modelo de Objetos del Documento
FP	Programación Funcional
FRP	Programación Reactiva Funcional
OOP	Programación Orientada a Objetos
QA	Quality Assurance
URL	Uniform Resource Locator
WIP	Trabajo en Proceso

Software Es el conjunto completo de programas, procedimientos y documentación relacionada asociada con un sistema mecánico o electrónico y especialmente un sistema informático (Merriam-Webster, s.f.)

Peeplo Aplicación web de fácil acceso para los empleados que permite registrar toda información que la empresa necesita en un solo lugar. (Peeplo, 2020)

Trello Es una solución fácil, gratuita, flexible y visual para gestionar proyectos y organizarlo todo, en la que confían millones de personas de todo el mundo. (Trello, s.f.)

Slack Es una plataforma de mensajería basada en canales. Con Slack, las personas pueden trabajar juntas de manera más efectiva, conectar todas sus herramientas y servicios de software y encontrar la

	información que necesitan para hacer su mejor trabajo, todo dentro de un entorno seguro de nivel empresarial. (Slack, s.f.)
Zoom	Plataforma en la nube fácil y confiable para video, voz, intercambio de contenido y chat, se ejecuta en dispositivos móviles, computadoras de escritorio, teléfonos y sistemas de salas. (Zoom, s.f.)
Google Meet	Plataforma para crear videoconferencias de calidad empresarial. (Google Meet, s.f.)
Google Calendar	Agenda y calendario electrónico, permite sincronizarlo con los contactos de Gmail de manera que podamos invitarlos y compartir eventos. (Google Calendar, s.f.)
Pluralsight	Plataforma que ayuda a personas y empresas a transformarse con cursos creados por expertos sobre las tecnologías más importantes de la actualidad. (Pluralsight, s.f.)
JavaScript	Es un lenguaje de programación interpretado que se utiliza principalmente para crear páginas web dinámicas. (Pérez, 2017)
JSX	Es una extensión de la sintaxis de JavaScript. (React, 2020)
GitLab	Es una plataforma de desarrollo de software de código abierto y gestor de repositorios. (GitLab, s.f.)
Zapier	Es un servicio web que automatiza el flujo de datos entre más de 500 aplicaciones web (Zapier, s.f.)
Ngrok	Aplicación que permite exponer un servidor web que se ejecuta en su máquina local a Internet (Ngrok, s.f.)
Smallchat	Aplicación que publica los mensajes de sus visitantes en un canal que creará en Slack (Smallchat, s.f.)
Webhook	Permite enviar datos en tiempo real de una aplicación a otra cada vez que ocurre un evento determinado. (Vero, s.f.)

I. INTRODUCCIÓN

En la actualidad, el software se ha vuelto indispensable, debido a que provee múltiples ventajas. Está integrado en casi todo tipo de productos y servicios modernos, ya sea en el trabajo, en la educación o de uso personal. A causa de ello, el cliente demanda una mejor calidad en las aplicaciones y por ello es necesario estar actualizado en el desarrollo de todo tipo de software.

Existen muchas plataformas que pueden ser de gran utilidad dentro de una empresa, sin embargo, no todas son de uso gratuito. Debido a que Acklen Avenue | Hero Unit es una empresa que se dedica a desarrollar software, ha creado sus propias aplicaciones. Esto trajo consigo, la reducción de costos dentro de la empresa y la creación de nuevas funcionalidades a su gusto y conveniencia.

Una de las aplicaciones desarrolladas se llama Peeplo que su funcionalidad consiste en que el equipo de trabajo pueda documentar las actividades realizadas en el día a día.

El presente informe incluye las generalidades de la empresa, que contiene una breve descripción de la empresa, la descripción del puesto en dónde fue desarrollada la práctica y los objetivos tanto específicos como generales.

En el marco teórico, se describe lo que es una plataforma web, las herramientas más importantes que posee el proyecto y las metodologías utilizadas dentro de la empresa para la gestión y el desarrollo del software.

En la sección de desarrollo se mostrará el progreso de las actividades realizadas por el practicante y una descripción a detalle de cada una de ellas, para tener un buen entendimiento del proceso.

II. GENERALIDADES DE LA EMPRESA

2.1 DESCRIPCIÓN DE LA EMPRESA

Acklen Avenue | Hero Unit crea productos digitales que se adaptan a las empresas. Al equipo de profesionales de desarrollo de software les apasiona el código limpio y las experiencias de usuario sencillas. Gracias a los heroicos esfuerzos del equipo y las increíbles ideas de los clientes, Acklen Avenue entrega con amor software diariamente desde Nashville y Honduras a socios de todo el mundo. (Sobre | Acklen Avenue, s.f.)

Posee su sede en Nashville Tennessee, Estados Unidos. Fue fundada en el año 2011 por Colin Callahan (CEO) y Byron Sommardahl (CTO).

2.2 DESCRIPCIÓN DEL DEPARTAMENTO

La práctica profesional se llevó acabo en el área de desarrollo de software, donde se trabajó con uno de los proyectos internos desarrollados por la empresa. El proyecto asignado fue Peeplo, que su funcionalidad consiste en registrar las actividades desarrolladas por los empleados.

2.3 OBJETIVOS DEL PUESTO

2.3.1 OBJETIVO GENERAL

Asistir, desarrollar e implementar soluciones de software para la aplicación web Peeplo cumpliendo con los requerimientos establecidos para satisfacer las necesidades de una organización.

2.3.2 OBJETIVOS ESPECÍFICOS

- Documentar los cambios realizados en el software para conocer a detalle el trabajo a realizar.
- Corregir los defectos que surjan en la plataforma Peeplo para mejorar la experiencia de usuario.

- Desarrollar funcionalidades utilizando los frameworks React en el front-end y NestJS en el back-end.

III. MARCO TEÓRICO

Acklen Avenue | Hero Unit utiliza Scrum "un marco de trabajo por el cual las personas pueden abordar problemas complejos adaptativos, a la vez que entregar productos del máximo valor posible, productiva y creativamente." (Sutherland & Schwaber, 2017)

En Scrum se realiza la entrega parcial de un producto "terminado", dando la flexibilidad de adaptarse a cambios y requisitos nuevos durante el desarrollo de un producto. Por ello, Scrum ha resultado ser efectivo en el desarrollo y gestión de productos complejos.

Scrum cuenta con roles, eventos y artefactos. Cada componente tiene un propósito en específico y es esencial para el éxito y uso de Scrum.

En la Ilustración 1 se pueden observar los eventos, artefactos y cómo encajan juntos.

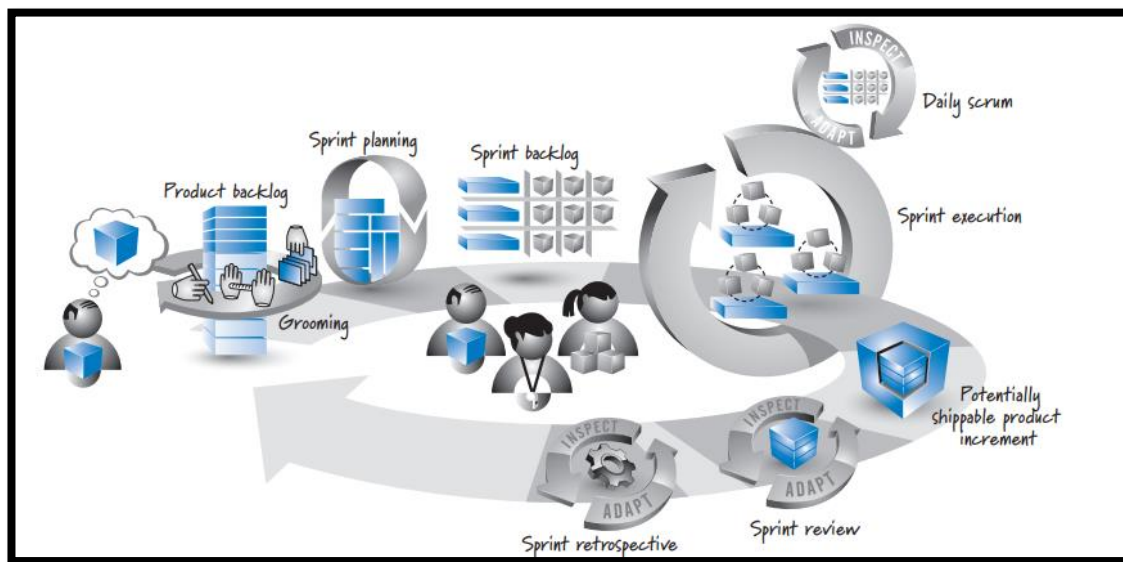


Ilustración 1 - Marco de Referencia Scrum

Fuente: (S. Rubin, 2012)

Se denomina Scrum Team a todas aquellas personas que tienen una relación con el desarrollo del producto. Está compuesto por el dueño del producto (Producto Owner), el equipo de desarrollo (Development Team) y un Scrum Master.

El Dueño de Producto es el responsable de maximizar el valor del producto resultante del trabajo del Equipo de Desarrollo. El cómo se lleva a cabo esto podría variar ampliamente entre distintas organizaciones, Equipos Scrum e individuos. (Sutherland & Schwaber, 2017)

El Equipo de Desarrollo consiste en los profesionales que realizan el trabajo de entregar un Incremento de producto "Terminado" que potencialmente se pueda poner en producción al final de cada Sprint. (Sutherland & Schwaber, 2017).

El Scrum Master es responsable de promover y apoyar Scrum como se define en la Guía de Scrum. Los Scrum Masters hacen esto ayudando a todos a entender la teoría, prácticas, reglas y valores de Scrum. (Sutherland & Schwaber, 2017)

Dentro de los eventos de Scrum se encuentra el Sprint que está compuesto por Sprint Planning, Daily Scrum, Sprint Review y Sprint Retrospective.

El corazón de Scrum es el Sprint, es un bloque de tiempo de un mes o menos durante el cual se crea un incremento del producto "terminado" utilizable y potencialmente desplegable. Es más conveniente si la duración de los Sprint es consistente a lo largo del esfuerzo de desarrollo. Cada nuevo Sprint comienza inmediatamente después de la finalización del Sprint anterior. (Sutherland & Schwaber, 2017)

En la Ilustración 2 se pueden observar las características de un Sprint.

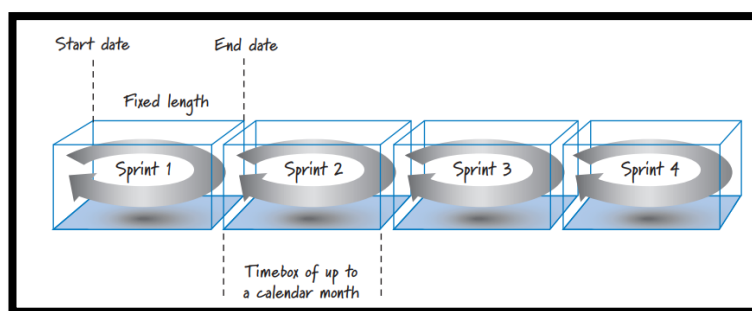


Ilustración 2 - Características de un Sprint

Fuente: (S. Rubin, 2012)

La estimación para la lista del producto (product backlog) puede tomar mucho tiempo, que representaría mucho más de lo que se podría completar en un sprint. Por ello, el Scrum Team selecciona un subconjunto de los elementos más importantes para luego trabajar en los mismos durante el Sprint.

El Daily Scrum es una reunión diaria de 15 minutos para el equipo de desarrollo. En ella cada miembro del equipo de desarrollo se encarga de responder 3 preguntas:

- ¿Qué logré desde el último Daily Scrum?
- ¿En qué planeo trabajar para el próximo Daily Scrum?
- ¿Qué obstáculos impiden avanzar?

Al final del Sprint se lleva a cabo una Revisión de Sprint para inspeccionar el Incremento y adaptar la Lista de Producto si fuese necesario. Durante la Revisión de Sprint, el Equipo Scrum y los interesados colaboran acerca de lo que se hizo durante el Sprint. Basándose en esto y en cualquier cambio a la Lista de Producto durante el Sprint, los asistentes colaboran para determinar las siguientes cosas que podrían hacerse para optimizar el valor. Se trata de una reunión informal, no una reunión de seguimiento, y la presentación del Incremento tiene como objetivo facilitar la retroalimentación de información y fomentar la colaboración. (Sutherland & Schwaber, 2017)

El Sprint Retrospective ayuda al equipo a inspeccionarse para luego crear un plan de mejoras para el siguiente Sprint.

El Sprint Retrospective tiene lugar después de la Revisión de Sprint y antes de la siguiente Planificación de Sprint. Se trata de una reunión de, a lo sumo, tres horas para Sprints de un mes. Para Sprints más cortos el evento es usualmente más corto. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. (Sutherland & Schwaber, 2017)

Para la gestión de proyectos se utiliza Kanban que se define como “una estrategia para optimizar el flujo de valor a través de un proceso que utiliza un sistema visual basado en extracción. Existen varias formas de definir el valor, incluida la consideración de las necesidades del cliente, el usuario final, la organización y el medio ambiente.” (The Kanban Guide, 2020)

El tablero más básico de Kanban está compuesto por tres columnas: "Por hacer", "En proceso" y "Hecho". Si se aplica bien y funciona correctamente, serviría como una fuente de información, ya que demuestra dónde están los cuellos de botella en el proceso y qué es lo que impide que el flujo de trabajo sea continuo e ininterrumpido. (Kanbanize, s.f.)

En la Ilustración 3 se muestra un tablero básico de Kanban.

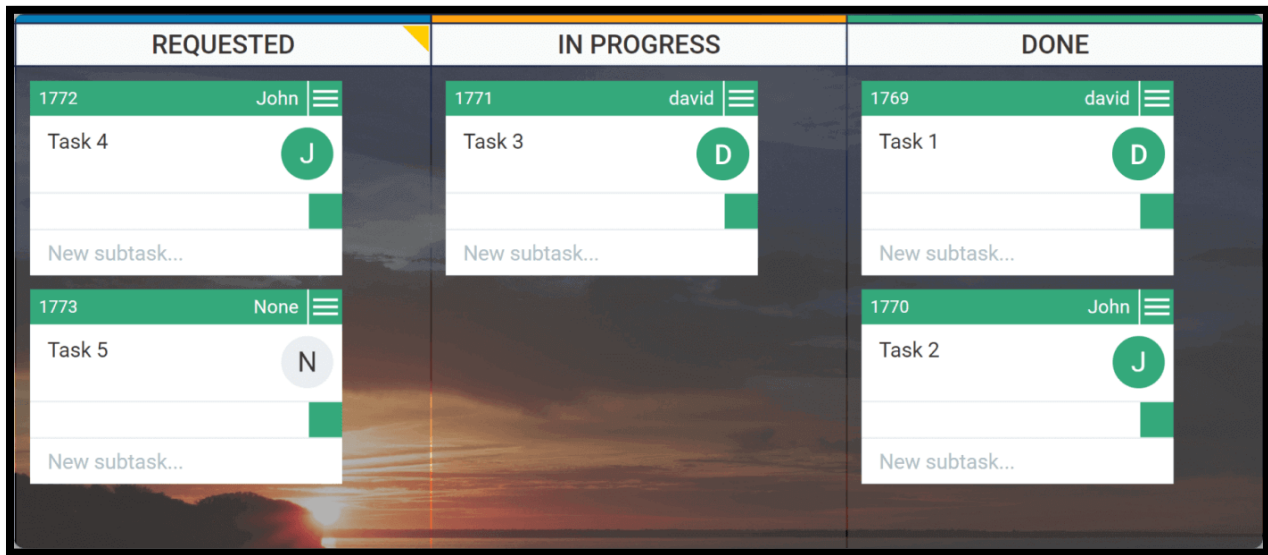


Ilustración 3 - Tablero Básico de Kanban

Fuente: (Kanbanize, s.f.)

Kanban es ideal en cualquier flujo de trabajo, no se limita a una industria. Su estrategia se basa en optimizar el flujo de trabajo, encontrando el equilibrio adecuado entre la efectividad, la eficiencia y la precisión de la previsión. También permite efectuar las preguntas adecuadas como parte de un esfuerzo de mejora para completar los objetivos.

Roger Pressman (2010) menciona que las aplicaciones web son poco más que un conjunto de archivos de hipertexto vinculados que presentan información con uso de texto y gráficas limitadas. Sin embargo, las aplicaciones web están evolucionando hacia ambientes de cómputo sofisticados que no sólo proveen características aisladas, funciones de cómputo y contenido para el usuario final, sino que también están integradas con bases de datos corporativas y aplicaciones de negocios.

En diseño de software el front-end es la parte que interactúa con los usuarios. De una manera práctica, el front-end son todos esos formularios que gestionan el ingreso por medio de los navegadores a los sitios y aplicaciones web. (Murcia Perez & Melendez Martínez, 2013)

El back-end es el encargado del procesamiento de los datos de entrada desde el front-end, siendo este el controlador de la vista y realizando la conexión con la base de datos. (Nieto, 2016)

El trabajo del back-end se basa en procesar la entrada de datos realizada por los usuarios desde el front-end, de esta manera el front-end y el back-end interactúan en una aplicación para resolver las necesidades de los usuarios. En general, la idea es mantener por separado las diferentes partes de una aplicación con el fin de tener un mejor control y rendimiento.

La Ilustración 4 muestra una estructura básica del front-end y back-end.

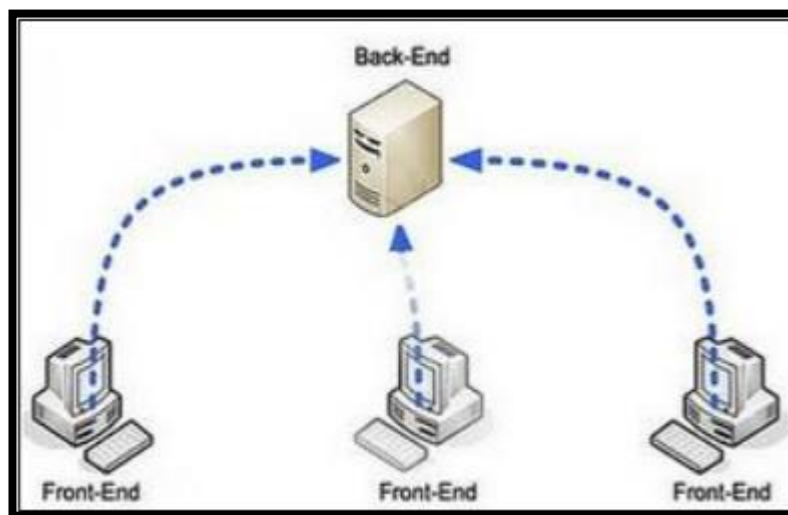


Ilustración 4 - Estructura Básica Front-end y Back-end

Fuente: (Murcia Perez & Melendez Martínez, 2013)

React es una biblioteca de Javascript para construir interfaces de usuario interactivas de forma sencilla. Diseña vistas simples para cada estado en tu aplicación, se encarga de actualizar y renderizar de manera eficiente los componentes correctos cuando los datos cambien. (React, s.f.)

Esta librería es la que se utiliza actualmente como front-end, su uso brinda diversas ventajas como lo pueden ser:

- Un DOM (Modelo de Objetos del Documento) mucho más ligero y eficiente: no interactúa con el DOM generado por el navegador, pero reacciona al modelo de objeto de documento almacenado en la memoria. (Aggarwal, 2018)
- JSX: es un lenguaje similar a XML. No es obligatorio utilizarlo, pero es bastante popular entre los programadores. Aggarwal (2018) menciona que facilita el desarrollo, ya que ayuda a optar por técnicas menos complejas.
- Rendimiento: ofrece un alto rendimiento debido a una característica en particular llamada virtual DOM. Lo que sucede es que mantiene un DOM virtual en memoria. Cuando un cambio es reflejado en la página en vez de actualizar el DOM del navegador, primero cambia el DOM virtual. (Aggarwal, 2018)

Gracias a las distintas ventajas que brinda React se logra comprender el potencial para cumplir con los requisitos de las tendencias actuales.

NestJS es un marco para crear aplicaciones de servidor Node.js escalables y eficientes. Utiliza JavaScript progresivo, está construido y es totalmente compatible con Typescript (pero aún permite a los desarrolladores codificar en JavaScript puro) y combina elementos de Programación Orientada a Objetos (OOP), Programación Funcional (FP) y Programación Reactiva Funcional (FRP). (nestjs, s.f.)

Para el desarrollo del back-end se usó el marco NestJS. Utilizar este marco brinda la facilidad a los desarrolladores de entender mejor la estructura. Esto es debido a que tiene un orden en la arquitectura.

Typescript es un lenguaje de código abierto que se basa en JavaScript, agregándole tipos estáticos y objetos basados en clases. (Typescript, 2020)

La plataforma utiliza este lenguaje de programación como front-end y back-end. Gracias a que Typescript cuenta con un sistema de módulos, clases, interfaces y un tipado estático, proporciona una ayuda ligera en cuanto a la resolución de problemas.

PostgreSQL es un potente sistema de base de datos relacional de objetos de código abierto que utiliza y amplía el lenguaje SQL combinado con muchas características que almacenan y escalan

de forma segura las cargas de trabajo de datos más complicadas. (The PostgreSQL Global Development Group, 2020)

La plataforma utiliza PostgreSQL como base de datos, es de código abierto y soporta una larga parte de los estándares de SQL como:

- Consultas complejas
- Llaves foráneas
- Triggers
- Integridad transaccional
- Control de concurrencia

Gracias a la licencia permisiva PostgreSQL puede ser utilizado, modificado y distribuido por cualquier persona sin importar el propósito, ya sea académico, comercial o privado.

Zapier es una herramienta de automatización en línea que conecta aplicaciones. Se pueden conectar dos o más aplicaciones para automatizar tareas repetitivas sin codificar ni depender de los desarrolladores. Dentro de Zapier se pueden crear Zaps, que es un flujo de trabajo automatizado entre sus aplicaciones. Por ejemplo, es posible que tenga un Zap para guardar archivos adjuntos de Gmail en Dropbox y otro Zap que guarde los correos electrónicos de Gmail en un archivo de texto. (Zapier | ¿Qué es Zapier?, s.f.)

Un trigger es el evento en una aplicación que inicia el Zap. Una vez que configure un Zap, Zapier controlará la aplicación para ese evento. Por ejemplo, puede recibir muchos correos electrónicos a través de su cuenta de Gmail, pero el Zap no se activa hasta que un correo electrónico contiene un archivo adjunto. (Zapier | ¿Qué es Zapier?, s.f.)

La Ilustración 5 muestra la creación de un trigger, donde se solicita la aplicación y el evento que inicia el Zap.

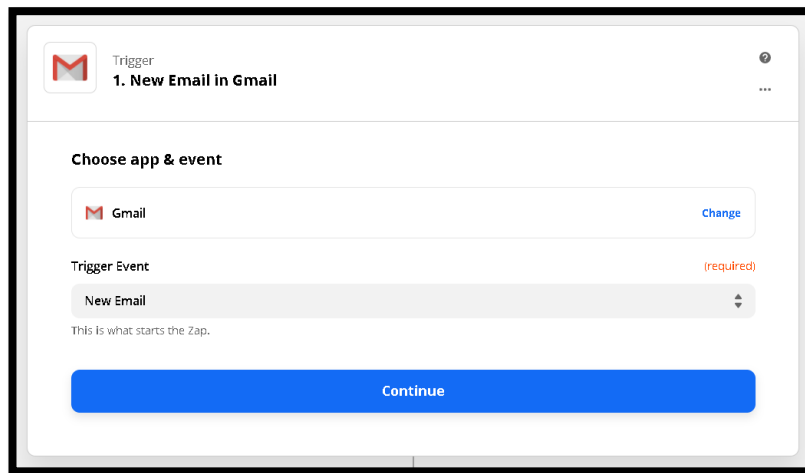


Ilustración 5 - Creación de un trigger

Fuente: (Zapier, s.f)

Una acción es el evento que completa el Zap. Para el ejemplo de guardar archivos adjuntos de Gmail en Dropbox, la acción es cargar el archivo adjunto desde su correo electrónico a Dropbox. (Zapier | ¿Qué es Zapier?, s.f.)

La Ilustración 6 muestra la creación de una acción, donde se solicita la aplicación y el evento que llevará a cabo el Zap.

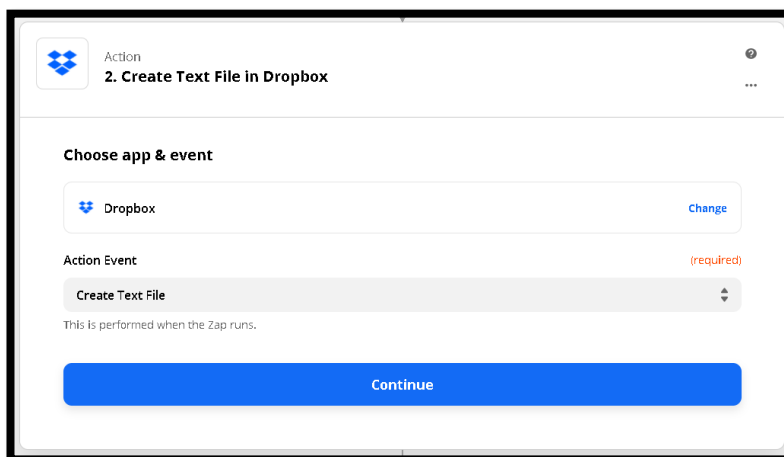


Ilustración 6 - Creación de una acción

Fuente: (Zapier, s.f)

IV. DESARROLLO

4.1 DESCRIPCIÓN DEL TRABAJO REALIZADO

4.1.1 INDUCCIÓN

Durante las primeras semanas de la práctica profesional, se trabajó en un proceso de inducción creado por la empresa. El proceso consta de realizar pequeñas actividades, donde se da una introducción de la empresa, explican las herramientas utilizadas, la metodología de trabajo y también se da la oportunidad de conocer a las personas que forman parte de la empresa. Una vez terminadas las actividades se reclama una insignia.

Dentro de las herramientas utilizadas se encuentran Trello, Slack, Zoom, Google Meet, Google Calendar y Peeplo. Siendo cada una de ellas una pieza importante para el éxito del desarrollo de proyectos. Para el desarrollo y gestión de proyectos se utiliza una combinación del marco de trabajo Scrum y Kanban.

En la Ilustración 7 se pueden observar las insignias obtenidas durante el proceso de inducción.

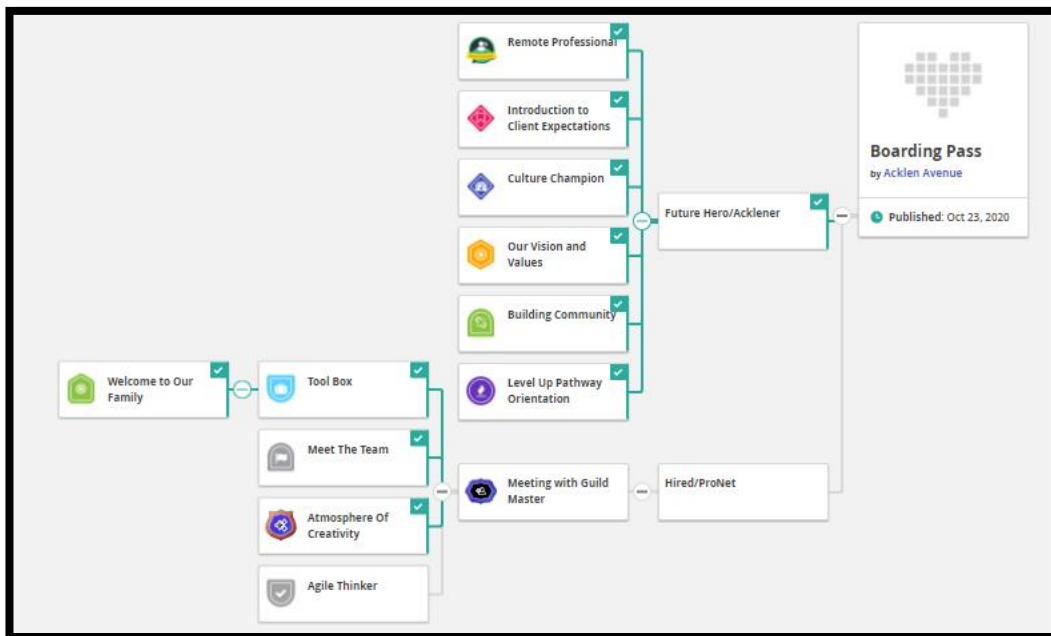


Ilustración 7 - Insignias ganadas en el proceso de inducción

Fuente: (Badgr Pathway, s.f.)

4.1.2 ENTRENAMIENTO CON LAS HERRAMIENTAS

El practicante, al no tener experiencia previa con las herramientas utilizadas en el proyecto Peeplo, se le asignó el trabajo de preparación. El entrenamiento consistió en aprender TypeScript, React y NodeJS en la plataforma Pluralsight. Cada curso brindaba una breve explicación de las características más importantes de cada herramienta.

El curso de TypeScript Fundamentals brindaba una guía de los conceptos y características clave que se necesitan conocer para impulsar el proceso de aprendizaje.

La Ilustración 8 muestra la tabla de contenido del curso TypeScript Fundamentals.

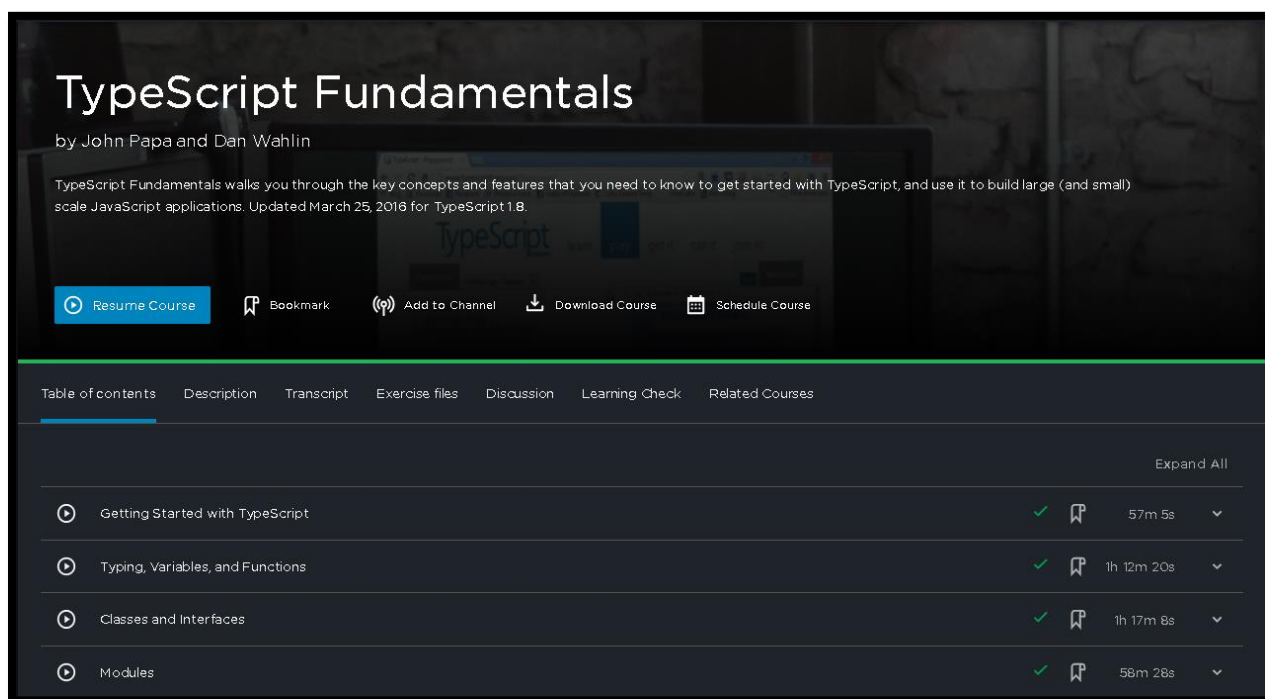
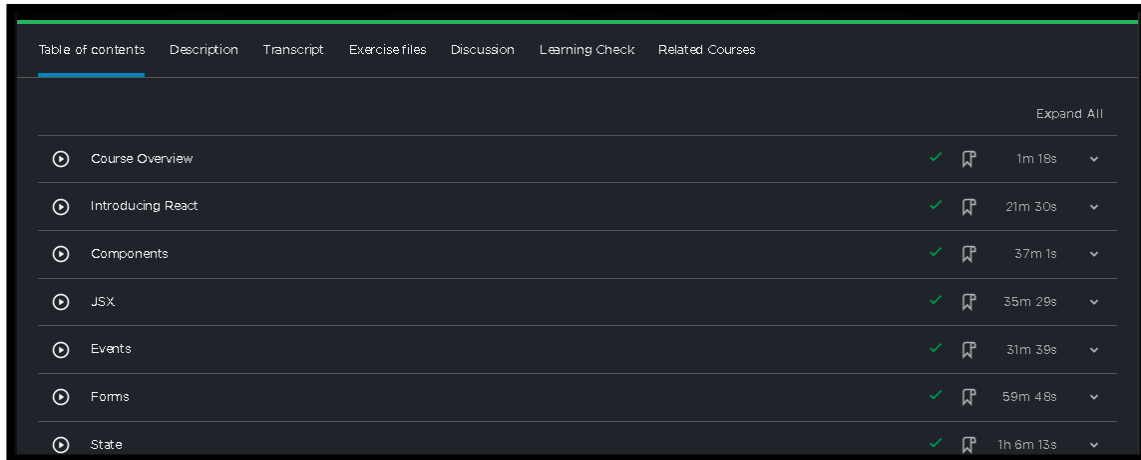


Ilustración 8 - Tabla de contenido para el curso de TypeScript Fundamentals

Fuente: (TypeScript Fundamentals | Pluralsight, s.f.)

El curso de React Fundamentals brindó una guía de cómo se estructura una aplicación a partir de componentes y como construirlos. También, la sintaxis de JSX y la administración de estados utilizando Redux.

La Ilustración 9 muestra la tabla de contenido del curso React Fundamentals.



The screenshot shows a dark-themed interface with a navigation bar at the top containing tabs: 'Table of contents', 'Description', 'Transcript', 'Exercise files', 'Discussion', 'Learning Check', and 'Related Courses'. Below the navigation bar, there is a list of course sections. Each section includes a play button icon, the section title, a green checkmark, a bookmark icon, the duration, and a dropdown arrow. An 'Expand All' link is located in the top right corner of the list.

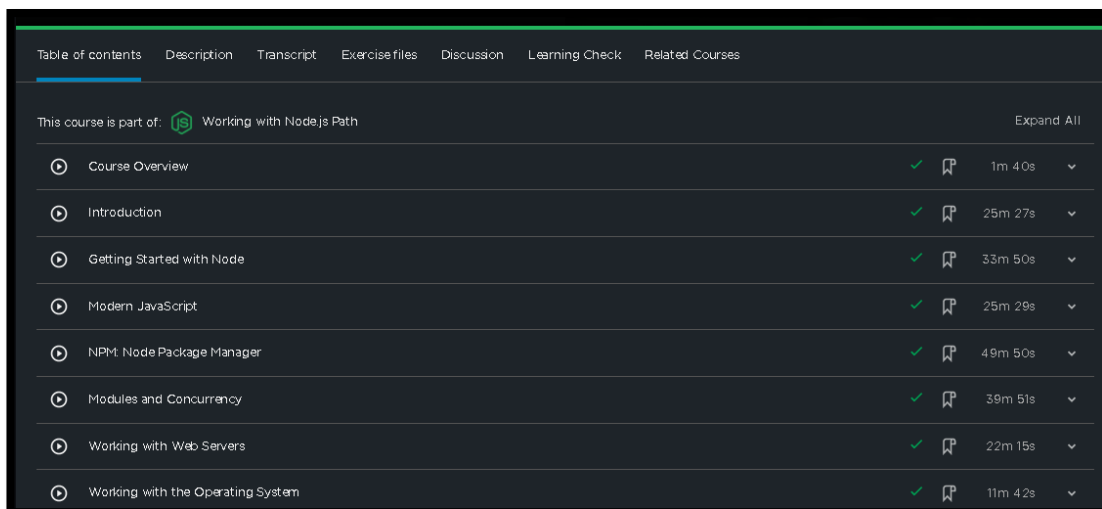
Section	Duration
Course Overview	1m 18s
Introducing React	21m 30s
Components	37m 1s
JSX	35m 29s
Events	31m 39s
Forms	59m 48s
State	1h 8m 13s

Ilustración 9 - Tabla de contenido para el curso de React Fundamentals

Fuente: (React Fundamentals | Pluralsight, s.f.)

El curso de Node.js: Getting Started brindaba una explicación de las herramientas integradas en Node.js. Se comprendió la gestión de paquetes, las dependencias, las estrategias de concurrencia y cómo escribir código para interactuar con los sistemas operativos en los servidores.

La Ilustración 10 muestra el contenido del curso Node.js: Getting Started.



The screenshot shows a dark-themed interface with a navigation bar at the top containing tabs: 'Table of contents', 'Description', 'Transcript', 'Exercise files', 'Discussion', 'Learning Check', and 'Related Courses'. Below the navigation bar, there is a list of course sections. Each section includes a play button icon, the section title, a green checkmark, a bookmark icon, the duration, and a dropdown arrow. An 'Expand All' link is located in the top right corner of the list. Above the list, it says 'This course is part of: Working with Node.js Path'.

Section	Duration
Course Overview	1m 40s
Introduction	25m 27s
Getting Started with Node	33m 50s
Modern JavaScript	25m 29s
NPM: Node Package Manager	49m 50s
Modules and Concurrency	39m 51s
Working with Web Servers	22m 15s
Working with the Operating System	11m 42s

Ilustración 10 - Tabla de contenido para el curso de Node.js: Getting Started

Fuente: (Node.js: Getting Started | Pluralsight, s.f.)

4.1.3 INTRODUCCIÓN AL PROYECTO PEEPLO

Para introducir al practicante al proyecto se le dio una breve explicación del funcionamiento del tablero de Trello. El tablero cuenta con diversas columnas donde se puede observar el seguimiento del proyecto. Cada columna cuenta con un propósito en específico y en ellas se observan diferentes viñetas que explican la actividad a realizar dentro del proyecto y los encargados en llevarla a cabo

Se brindó un repaso del código fuente, donde se explicaron las herramientas utilizadas y se otorgaron las credenciales del repositorio donde se encuentra alojado el proyecto.

4.1.4 CORRECCIÓN DE REDIRECCIONAMIENTO

Los administradores o los gerentes poseen el permiso de agregar empleados a una organización. Esto se realiza mediante un enlace de invitación que provee la aplicación web cuando se crea un empleado. El problema que se encontró con esta implementación se debía a que cuando un enlace de invitación no era válido la página redireccionaba a la vista de iniciar sesión, cuando lo correcto era mostrar un mensaje que indicara que el enlace era inválido. Se encontró que este problema era debido a una validación en el código. Por lo tanto, la validación fue modificada para que en esas circunstancias no realizara dicha acción.

En la Ilustración 11 se puede observar la modificación realizada. La sección de color rojo indica el código original y la sección verde el código modificado. En este caso se agregó una variable booleana llamada `redirect` en la condición del `if` que su propósito consistió en evitar la redirección a la página de inicio cuando el enlace de invitación no era válido.

```
if (!this.authService.isAuthenticated()) {  
  if (shouldRedirect) this.authService.logout();  
  if (!this.authService.isAuthenticated() && redirect) {  
    this.authService.logout();  
  }  
}
```

Ilustración 11 - Modificación de una condición

Fuente: (GitLab, 2020).

4.1.5 ELIMINAR AL PRIMER EMPLEADO CUANDO SE CREA UNA NUEVA ORGANIZACIÓN

El objetivo de esta tarea consistía en eliminar al primer empleado cuando se creaba una nueva organización en la aplicación. Cuando un usuario creaba una organización, este era agregado automáticamente como empleado y se le otorgaban los permisos de administrador. Debido a que los permisos eran brindados a los empleados y no a los usuarios, por lo que se pidió realizar este cambio.

En un principio se creyó que la tarea resultaría ser sencilla, debido a que lo único que se tendría que hacer era no agregar al usuario en primera instancia a la tabla de empleados y otorgarle los permisos de administrador a ese usuario. Al realizar estos cambios la aplicación no permitía iniciar sesión. Se encontró que esto era debido a que la aplicación estaba esperando que un empleado realizara las acciones y no un usuario. Por lo tanto, se tuvo que cambiar toda referencia relacionada con un empleado a un usuario. Esto trajo consigo muchas complicaciones, ya que se tuvieron que realizar modificaciones en varios archivos donde existían referencias a un empleado.

La Ilustración 12 muestra la documentación realizada, donde el desarrollador debe especificar todos los cambios que conllevaría esta tarea.

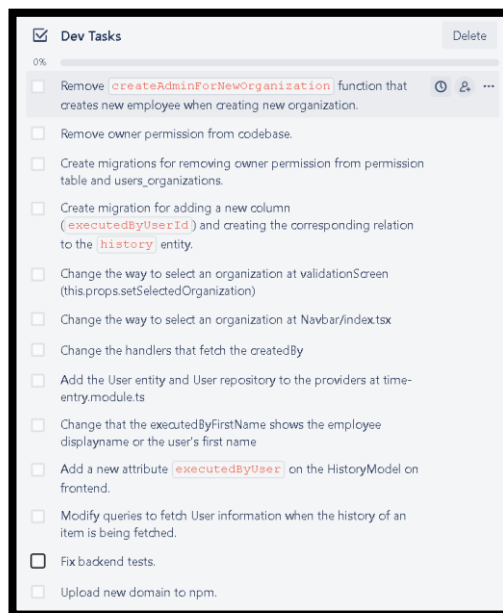


Ilustración 12 - Documentación de cambios para eliminar al primer empleado

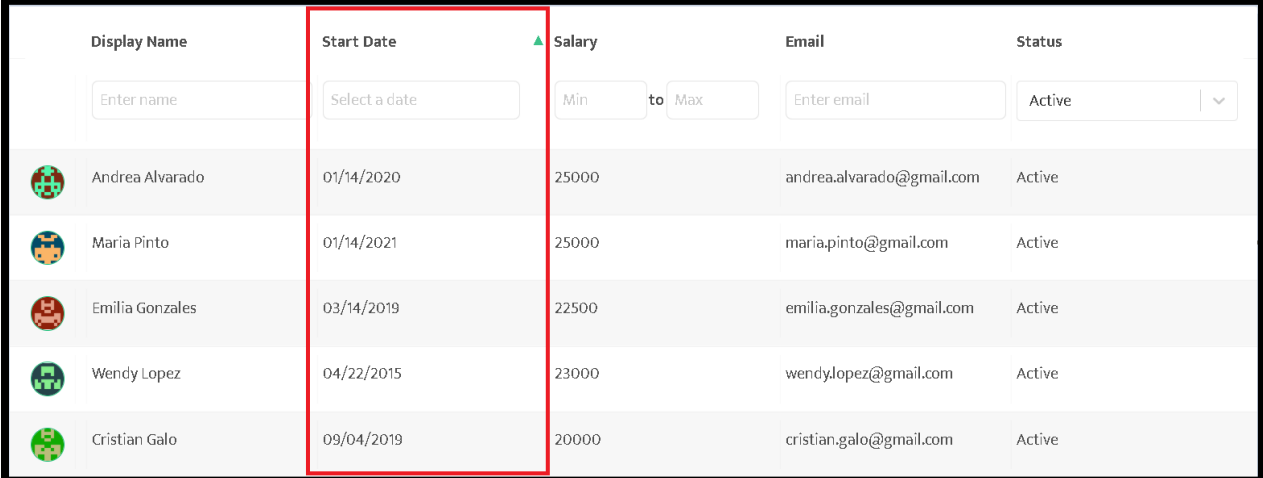
Fuente: (Trello, s.f)

4.1.6 CORRECCIÓN DE ERROR AL ORDENAR FECHAS

Los empleados cuentan con diferentes atributos, entre ellos se encuentran los relacionados con las fechas. Como lo son la fecha de inicio, fecha de término y fecha de nacimiento.

La tabla en la que eran listados contaba con un encabezado, donde se mostraba el nombre del atributo y un botón que permitía ordenar las fechas de mayor a menor o viceversa; sin embargo, el comportamiento de este botón no era el correcto, por esa razón se agregó un algoritmo personalizado en el front-end, para ordenar correctamente las fechas.

La Ilustración 13 muestra el comportamiento original al ordenar las fechas ascendentemente.








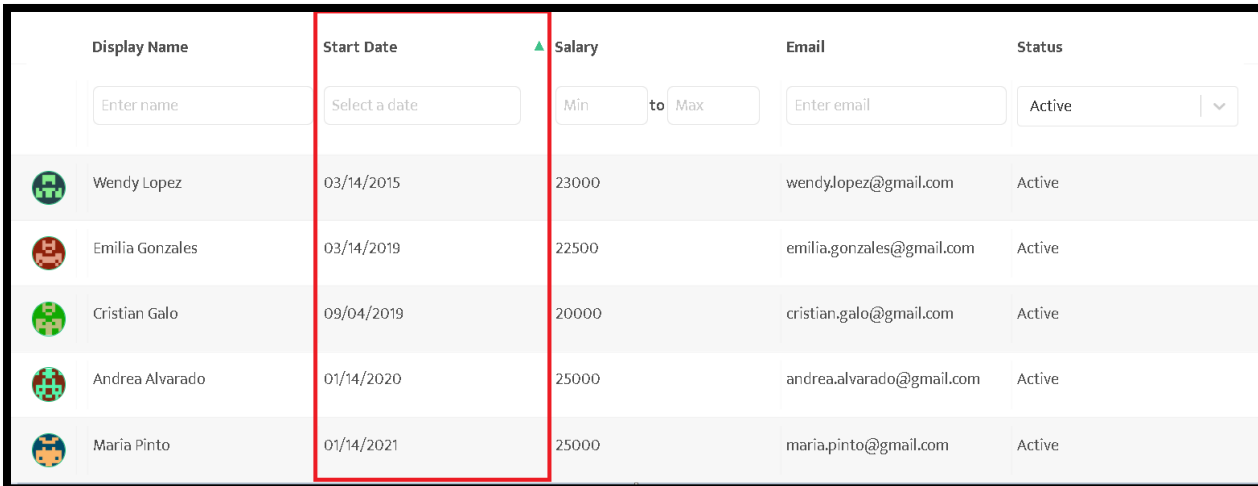
Display Name	Start Date	Salary	Email	Status
<input type="text" value="Enter name"/>	<input type="text" value="Select a date"/>	<input type="text" value="Min"/> to <input type="text" value="Max"/>	<input type="text" value="Enter email"/>	<input type="text" value="Active"/>
 Andrea Alvarado	01/14/2020	25000	andrea.alvarado@gmail.com	Active
 Maria Pinto	01/14/2021	25000	maria.pinto@gmail.com	Active
 Emilia Gonzales	03/14/2019	22500	emilia.gonzales@gmail.com	Active
 Wendy Lopez	04/22/2015	23000	wendy.lopez@gmail.com	Active
 Cristian Galo	09/04/2019	20000	cristian.galo@gmail.com	Active

Ilustración 13 - Comportamiento original al ordenar las fechas

Fuente: (Peeplo, s.f)

Se encontró que este comportamiento era debido a que el algoritmo ordenaba tipos de datos numéricos y no de tipo fecha.

La Ilustración 14 muestra el comportamiento luego de implementar el algoritmo personalizado.



Display Name	Start Date	Salary	Email	Status
<input type="text" value="Enter name"/>	<input type="text" value="Select a date"/>	<input type="text" value="Min"/> to <input type="text" value="Max"/>	<input type="text" value="Enter email"/>	<input type="text" value="Active"/>
Wendy Lopez	03/14/2015	23000	wendy.lopez@gmail.com	Active
Emilia Gonzales	03/14/2019	22500	emilia.gonzales@gmail.com	Active
Cristian Galo	09/04/2019	20000	cristian.galo@gmail.com	Active
Andrea Alvarado	01/14/2020	25000	andrea.alvarado@gmail.com	Active
Maria Pinto	01/14/2021	25000	maria.pinto@gmail.com	Active

Ilustración 14 - Comportamiento al implementar el algoritmo customizado

Fuente: (Peeplo, s.f)

4.1.7 IMPLEMENTACIÓN DE UN CHAT DE SOPORTE

Peeplo es una aplicación de uso interno en la empresa, debido a que desean brindar esos servicios a otras empresas se creó la idea de implementar un chat de soporte para resolver cualquier inquietud de los nuevos usuarios. Para ello, se utilizó la aplicación Smallchat, que funciona como un bot. Esta aplicación permite publicar los mensajes de los usuarios a un canal que se crea en Slack.

Para agregar esta aplicación se deben realizar una serie de pasos. Primero se debe brindar autorización a Smallchat para acceder a algunos permisos básicos de Slack, luego Smallchat redirige a su panel para nombrar el canal que se creará en Slack y realizar las configuraciones necesarias del chat. Por último, Smallchat brinda una línea de código que debe ser copiada en el archivo global index.html.

En la Ilustración 15 se aprecia cómo aparece el chat de soporte en la aplicación Peeplo.

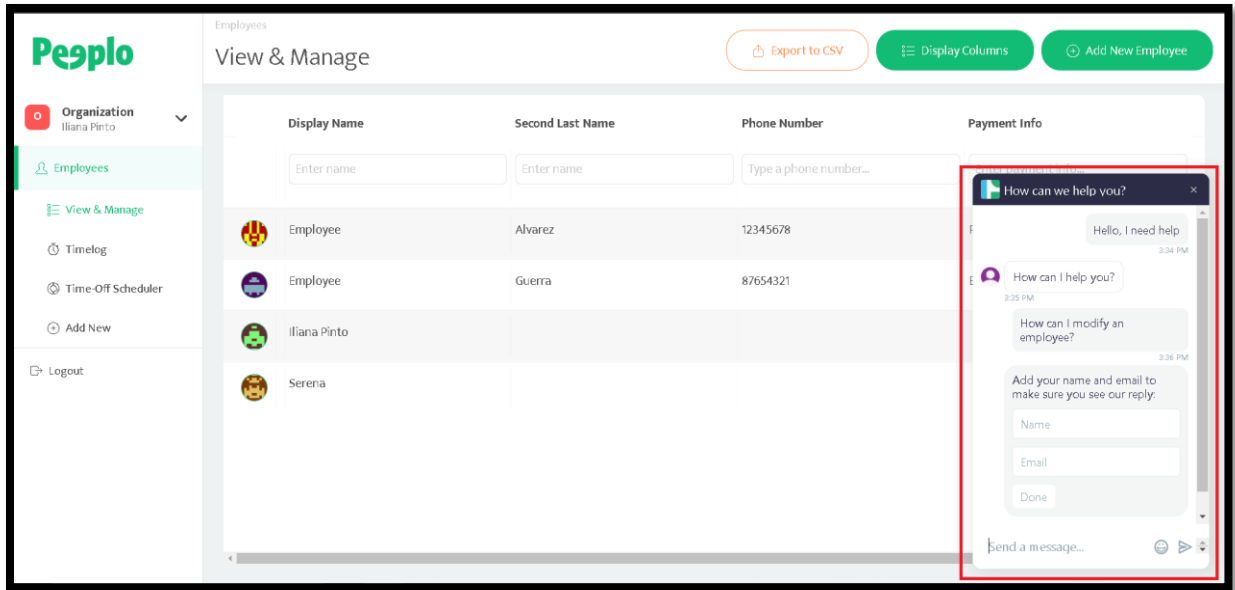


Ilustración 15 - Implementación de un chat de soporte

Fuente: (Peeplo, s.f)

4.1.8 IMPLEMENTACIÓN DE UN INDICADOR DE CARGA

Al momento de cargar la información de un empleado, en ocasiones, mostraba una pantalla en blanco, denotando que un error estaba ocurriendo. Se encontró que esto sucedía debido a que las peticiones a la base de datos no se realizaban a tiempo. La solución a ello fue agregar un indicador de carga, señalando que los datos se están recuperando.

La Ilustración 16 muestra una validación que tiene como propósito mostrar un indicador de carga en caso de que la información necesaria aún no haya sido recuperada. La variable `isFetching` es de tipo booleano e indica si los datos están recuperados o no.

```
{this.props.isFetching && this.state.activeTabId !== 3 ? (  
  <div className="center-spinner">  
    <ReactLoading  
      width={"5%"}  
      height={"5%"}  
      type="spin"  
      color="#FF9051"  
    />  
  </div>  
}
```

Ilustración 16 - Validación para un indicador de carga

Fuente: Elaboración propia

4.1.9 IMPLEMENTACIÓN DEL CAMPO CONTACTO DE EMERGENCIA

La base de datos tiene la tabla empleado que contaba con los atributos necesarios para poder contactarlo. Sin embargo, no incluía el atributo para contactar a un familiar en caso de emergencias. Esta petición tenía como requisito tener un máximo de 500 caracteres.

En el frontend se agregó un campo llamado contacto de emergencia y se incluyó la validación necesaria para cumplir con el requisito.

Debido a que los navegadores permiten modificar líneas de código se consideró agregar una validación en el back-end referente a los 500 caracteres. De esta forma si un usuario considera cambiar la validación hecha en el frontend con el navegador, igualmente la aplicación no permitiría guardar los cambios ya que esta lanzaría una alerta.

La Ilustración 17 muestra la modificación realizada a la base de datos.

```
await queryRunner.query(  
  'ALTER TABLE "employee" ADD COLUMN "emergencyContact" character varying(500) DEFAULT ''',  
);
```

Ilustración 17 – Cambios realizados a la base de datos

Fuente: Elaboración propia

4.1.10 IMPLEMENTACIÓN DE LA HERRAMIENTA ZAPIER

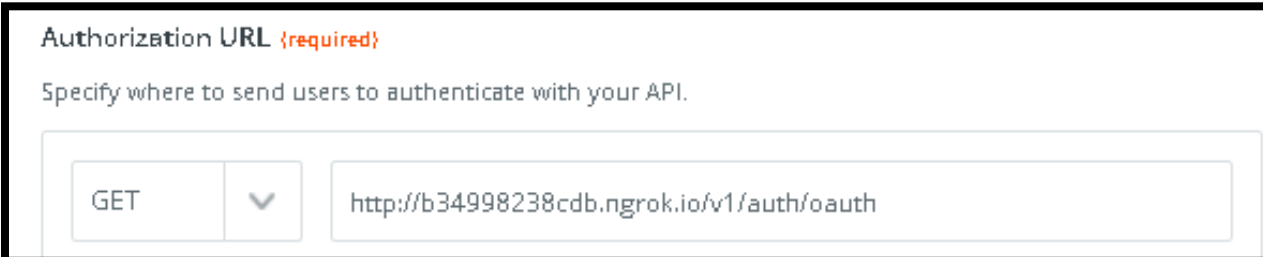
Zapier es una plataforma web que permite conectar diferentes aplicaciones, con el propósito de compartir datos. Debido a la automatización de trabajo que brinda esta aplicación se pidió integrar Peeplo dentro del universo de aplicaciones de Zapier. De esta forma, se podrán crear Zaps para automatizar el flujo de trabajo. Como el envío automático de correos electrónicos a un cliente externo o guardar información en una hoja de cálculo.

Zapier cuenta con una plataforma para facilitar el proceso de integración. Esta plataforma permite habilitar la autenticación, la creación de triggers y acciones. En ella se especifica toda la información necesaria para llevar a cabo cada tarea y lograr publicar la aplicación.

El primer paso consistía en la autenticación. Su propósito era verificar la identidad del usuario para luego otorgarle autorización a Zapier de acceder a sus datos. La plataforma brindaba la libertad de seleccionar la autenticación más conveniente dependiendo de la aplicación. En el caso de Peeplo la más conveniente era la autenticación con OAuth2. Este tipo de autenticación permitía redirigir al usuario a un sitio donde pueda ser autenticado.

Para lograr autenticar al usuario, se crearon tres endpoints en el backend. Cada uno se probó utilizando ngrok. Esta herramienta permite exponer un servidor alojado localmente a estar alojado en un subdominio.

La Ilustración 18 muestra el endpoint creado para que el usuario pueda autenticarse en la página de Peeplo.

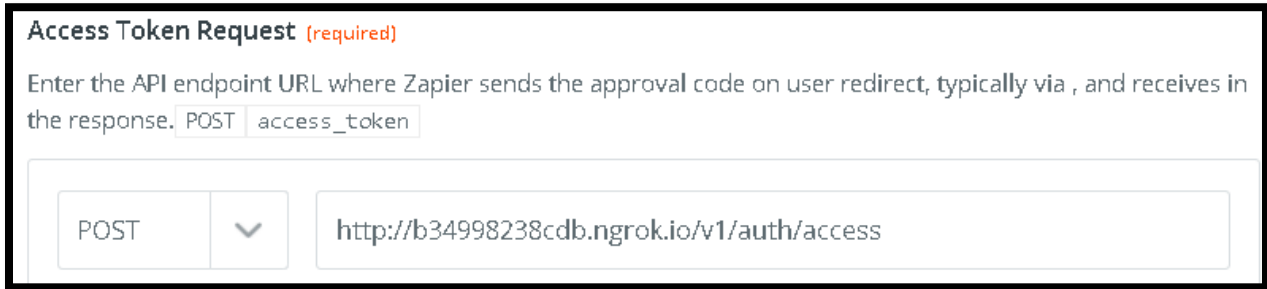


The screenshot shows a configuration box for an API endpoint. At the top, it says "Authorization URL {required}" in red. Below that, it says "Specify where to send users to authenticate with your API." There is a dropdown menu with "GET" selected and a downward arrow. To the right of the dropdown is a text input field containing the URL "http://b34998238cdb.ngrok.io/v1/auth/oauth".

Ilustración 18 - Endpoint para autenticar al usuario

Fuente: (Zapier, s.f)

La Ilustración 19 muestra el endpoint creado para enviar un token de acceso a Zapier.



Access Token Request (required)

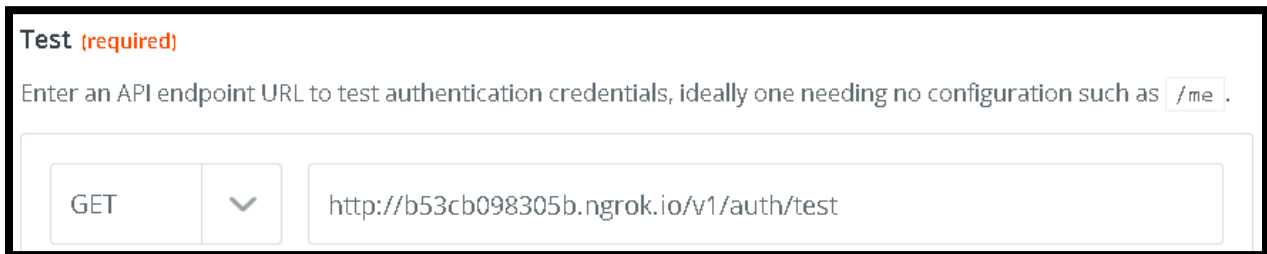
Enter the API endpoint URL where Zapier sends the approval code on user redirect, typically via `POST`, and receives in the response. `POST` `access_token`

`POST` `▼` `http://b34998238cdb.ngrok.io/v1/auth/access`

Ilustración 19 - Endpoint para enviar un código de aprobación

Fuente: (Zapier, s.f)

La Ilustración 20 muestra el endpoint creado para verificar la autenticación.



Test (required)

Enter an API endpoint URL to test authentication credentials, ideally one needing no configuration such as `/me`.

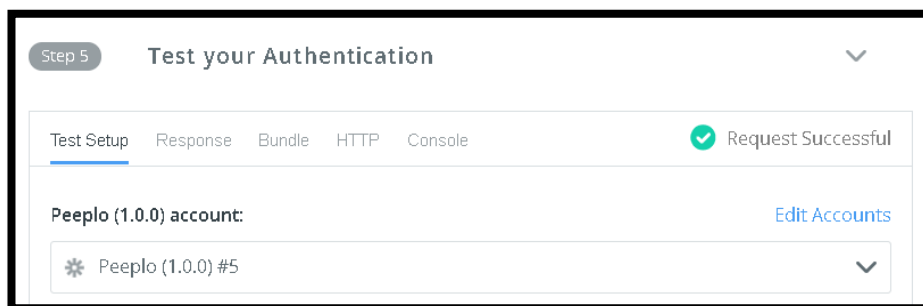
`GET` `▼` `http://b53cb098305b.ngrok.io/v1/auth/test`

Ilustración 20 - Endpoint para verificar credenciales

Fuente: (Zapier, s.f)

Una vez se hayan ingresado las URL, se procede a guardar los cambios y probar la autenticación.

La Ilustración 21 muestra el último paso a seguir, donde se verifica que la configuración para la autenticación se realizó con éxito.



Step 5 Test your Authentication `▼`

Test Setup Response Bundle HTTP Console ✔ Request Successful

Peeplo (1.0.0) account: [Edit Accounts](#)

`✳` Peeplo (1.0.0) #5 `▼`

Ilustración 21 - Prueba de autenticación en Zapier

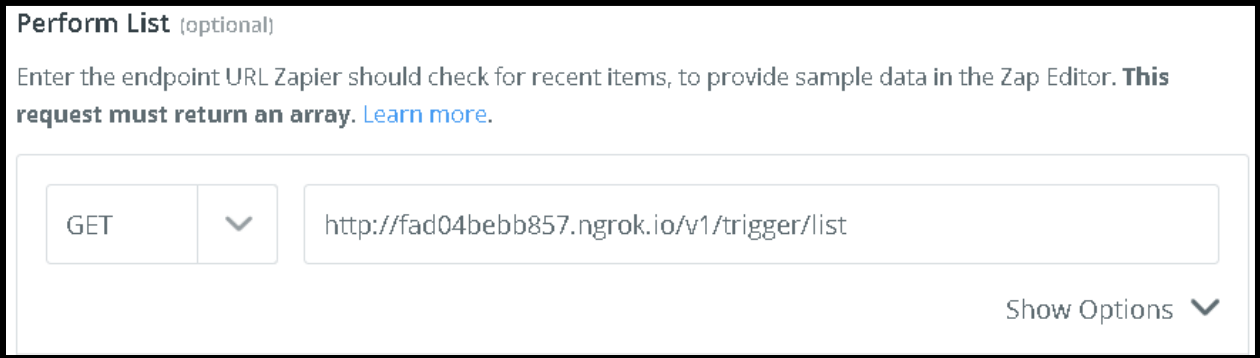
Fuente: (Zapier, s.f)

4.1.10.1 Creación de Triggers en Zapier

El propósito de esta tarea consiste en implementar dos triggers. Un trigger cuando se crea un registro de tiempo o cuando se actualice. Para llevar a cabo esta tarea se crearon tres endpoints en el backend. Cada uno debe ser especificado en la plataforma de Zapier.

El primer endpoint tenía como propósito enviar datos de prueba para que el usuario pueda utilizarlos al momento de crear un Zap. Cuando se crea un Zap el usuario puede utilizar esos datos para especificar a qué aplicación desea enviarlos y de qué forma serán ordenados o almacenados en ella.

La Ilustración 22 muestra el endpoint creado para enviar los datos de prueba.



Perform List (optional)

Enter the endpoint URL Zapier should check for recent items, to provide sample data in the Zap Editor. **This request must return an array.** [Learn more.](#)

GET Show Options ▾

Ilustración 22 - Endpoint para enviar datos de prueba.

Fuente: (Zapier, s.f)

Para verificar la funcionalidad del endpoint se debe crear un zap. Al momento de crear el zap se debe seleccionar la aplicación Peeplo, luego proceder a la autenticación y por último seleccionar el tipo de trigger.

En la Ilustración 23 se aprecia cómo Zapier recibe los datos de prueba.

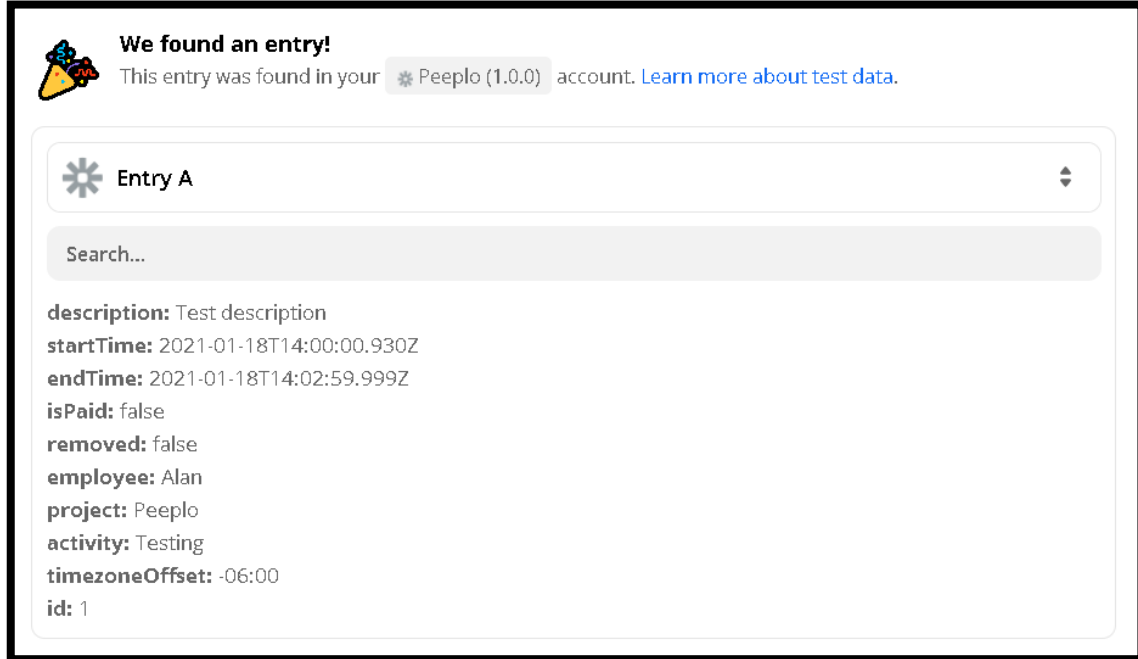
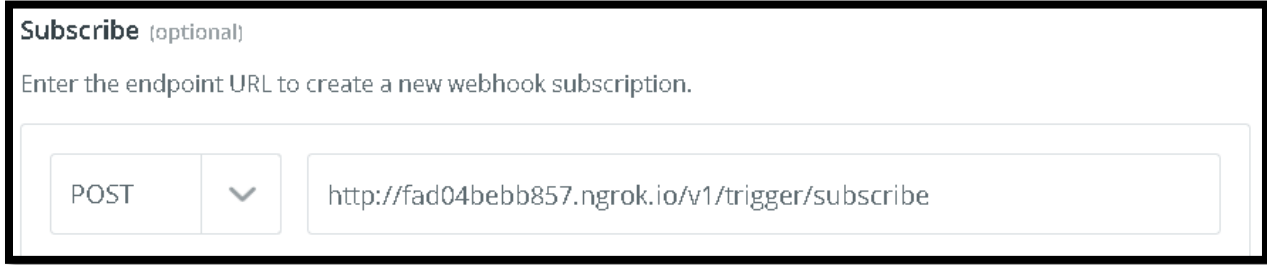


Ilustración 23 - Datos de prueba enviados a Zapier

Fuente: (Zapier, s.f)

Cuando un usuario finaliza con la creación de un zap, tiene la oportunidad de activarlo. Al realizar esta acción un webhook URL era enviado al segundo endpoint. El propósito de este webhook era realizar una función POST para enviar los datos necesarios y de esta forma Zapier entienda cuándo el zap debe ser ejecutado. Para llevar un control de los webhooks, se creó una entidad llamada hook en la base de datos. Esta entidad contenía los campos: id, hookUrl, userId y organizationId. De esta forma se logró filtrar los webhooks por usuario y realizar los POST correspondientes.

La Ilustración 24 muestra el endpoint creado para indicar cuando un zap ha sido activado



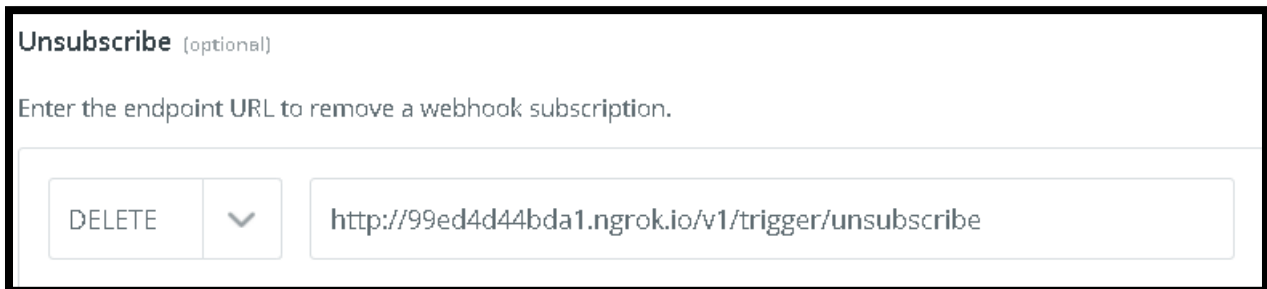
The screenshot shows a form titled "Subscribe (optional)". Below the title is the instruction "Enter the endpoint URL to create a new webhook subscription." The form contains a dropdown menu with "POST" selected, a downward arrow icon, and a text input field containing the URL "http://fad04bebb857.ngrok.io/v1/trigger/subscribe".

Ilustración 24 - Endpoint para obtener un webhook

Fuente: (Zapier, s.f)

El tercer endpoint tenía como propósito indicar cuándo un zap era desactivado. Al desactivar un zap el webhook URL queda deshabilitado y por lo tanto debe ser eliminado en la base de datos.

La Ilustración 25 muestra el endpoint creado para eliminar el webhook.



The screenshot shows a form titled "Unsubscribe (optional)". Below the title is the instruction "Enter the endpoint URL to remove a webhook subscription." The form contains a dropdown menu with "DELETE" selected, a downward arrow icon, and a text input field containing the URL "http://99ed4d44bda1.ngrok.io/v1/trigger/unsubscribe".

Ilustración 25 - Endpoint para eliminar un webhook

Fuente: (Zapier, s.f)

4.1.10.2 Configuración de un Trigger

Esta tarea consistía en agregar un menú desplegable para seleccionar la organización en la que el usuario poseía permisos de administrador o de gerente. La integración de Zapier permite utilizar tres tipos de campos.

- Campos de entrada: permite al usuario ingresar texto.
- Campos dinámicos: realiza llamados a la aplicación para mostrar los datos retornados en un menú desplegable para que los usuarios puedan seleccionar el elemento necesario.

- Grupo de artículos: permite ingresar artículos separados por comas para crear un artículo nuevo en la aplicación. A menudo se utiliza para aplicaciones de facturación o contabilidad.

Para completar esta tarea se utilizó el de campos dinámicos debido a que la aplicación debe enviar los datos de las organizaciones donde el usuario tenga permisos de administrador o gerente. Para ello se creó un endpoint que retorna los datos necesarios de una organización.

La Ilustración 26 muestra un llamado al endpoint que retorna los datos de una organización.

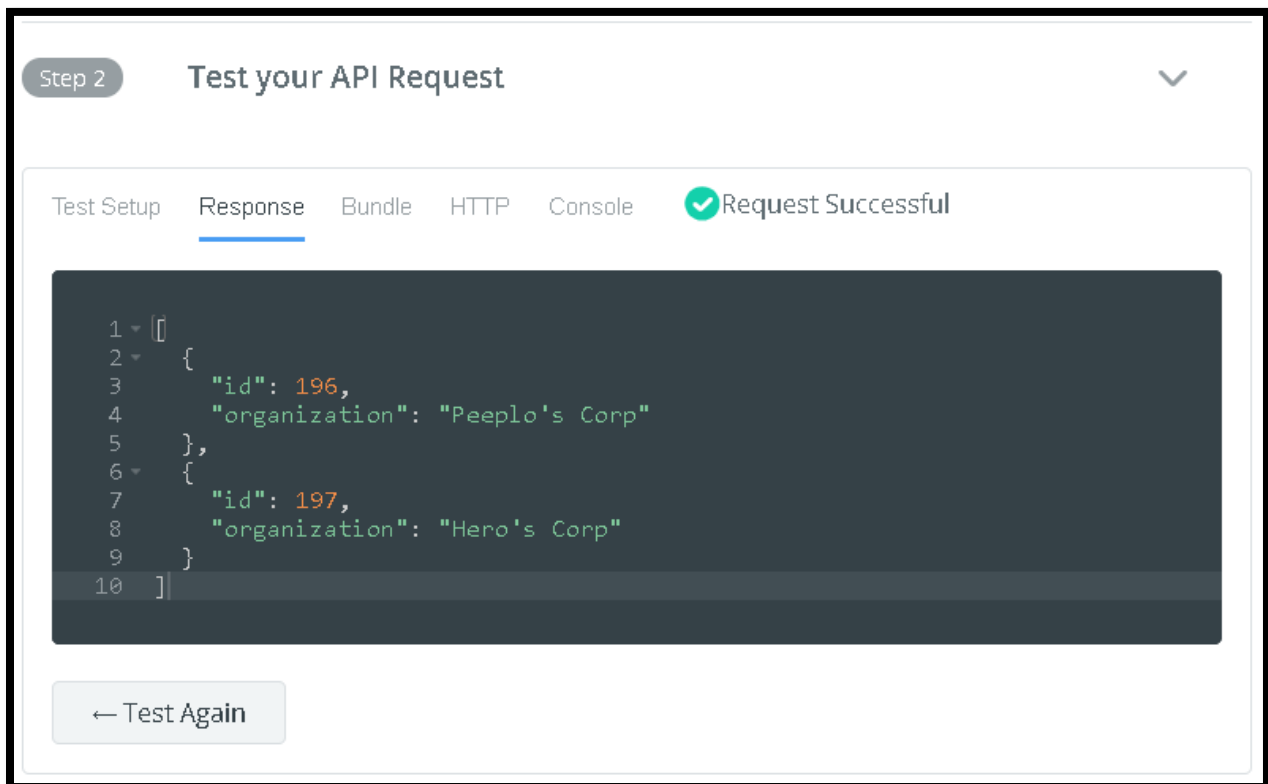


Ilustración 26 - Llamado al endpoint de obtención de organizaciones

Fuente: (Zapier, s.f)

4.1.11 CORRECCIÓN DE CÁLCULOS EN LA DURACIÓN DE HORAS

Esta tarea consistía en corregir los cálculos de duración al momento de registrar horas. Para completar esta tarea se revisó cómo se estaba almacenando la hora de inicio y la hora de término en la base de datos y cómo la librería moment estaba obteniendo estos datos. Se encontró que esto era debido a que la librería moment estaba agregando 59 segundos extra, entonces al

momento de realizar la suma del total de horas o al momento de sacar la diferencia de tiempo este se veía afectado.

La Ilustración 27 muestra el comportamiento original, donde el cálculo es de 2:49 h pero el correcto debe ser 2:47 h.

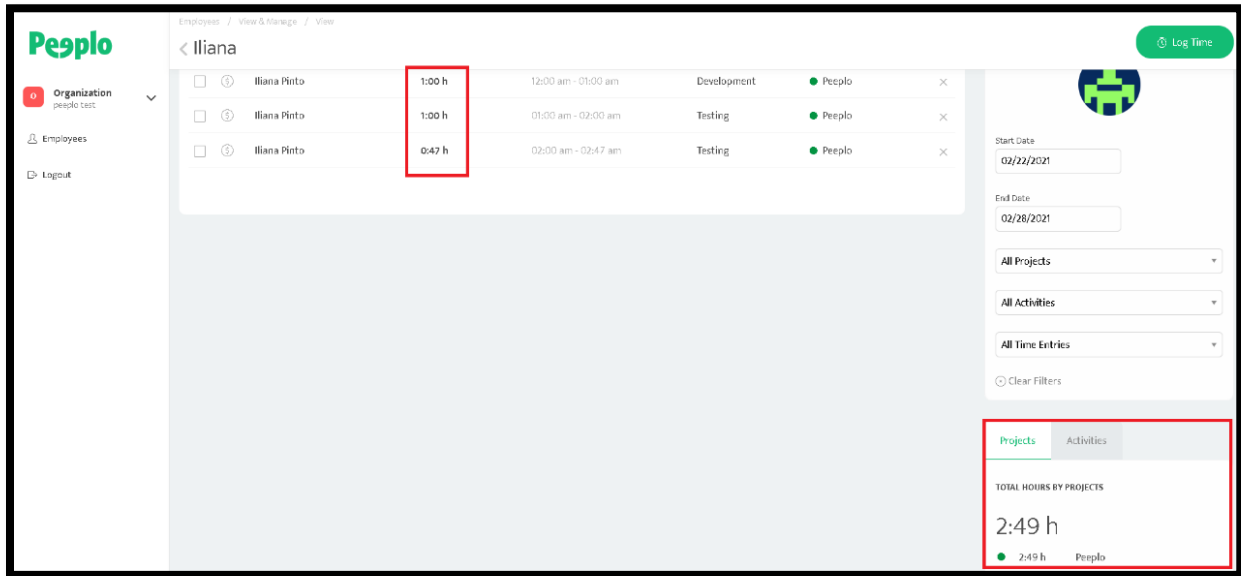


Ilustración 27 - Cálculos erróneos en la suma del total de horas

Fuente: (Peeplo, s.f)

Para solucionarlo se utilizó una función de la librería moment, que permite establecer los segundos en cero.

La Ilustración 28 muestra cómo se ve el total de horas una vez corregido el error.

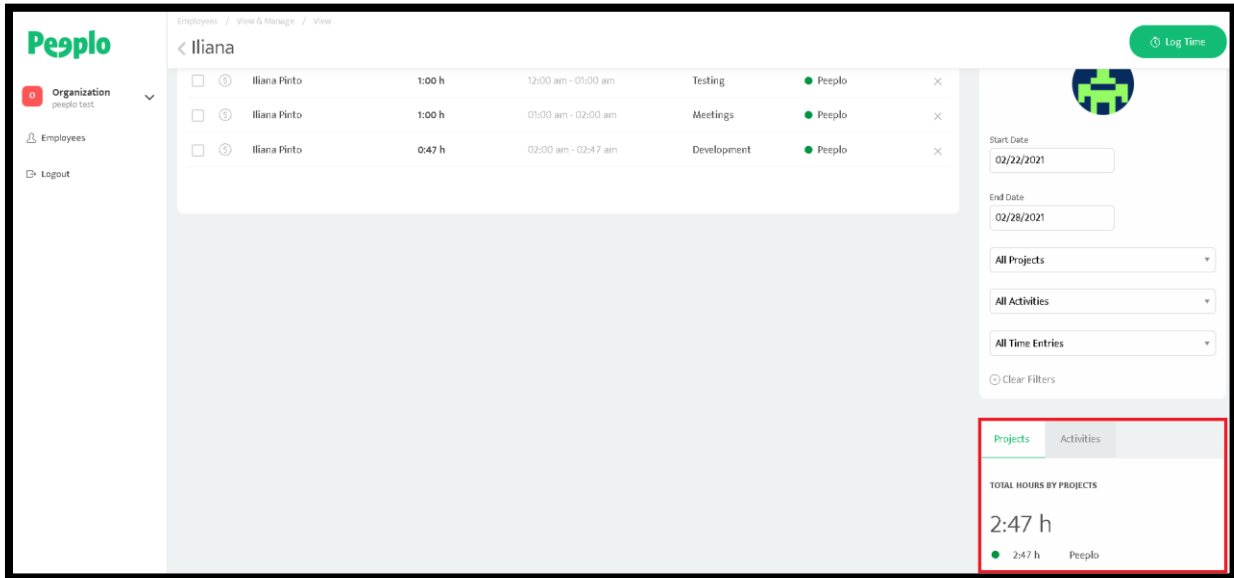


Ilustración 28 - Corrección en la suma del total de horas

Fuente: (Peeplo, s.f)

4.1.12 MOSTRAR LA DURACIÓN DE HORAS EN FORMATO DECIMAL

Esta tarea consistía en cambiar el formato de duración h:mm a formato decimal. Debido a que el equipo de RRHH se encarga de calcular el pago de cada empleado, al realizar la conversión a formato decimal facilitarían su proceso. Debido a que la base de datos almacena la duración en segundos, solamente se utilizó una función de la librería moment que convierte los segundos a decimal, igualmente se agregó una función para redondear a la centésima más cercana.

En la Ilustración 29 se aprecia la duración en formato decimal.

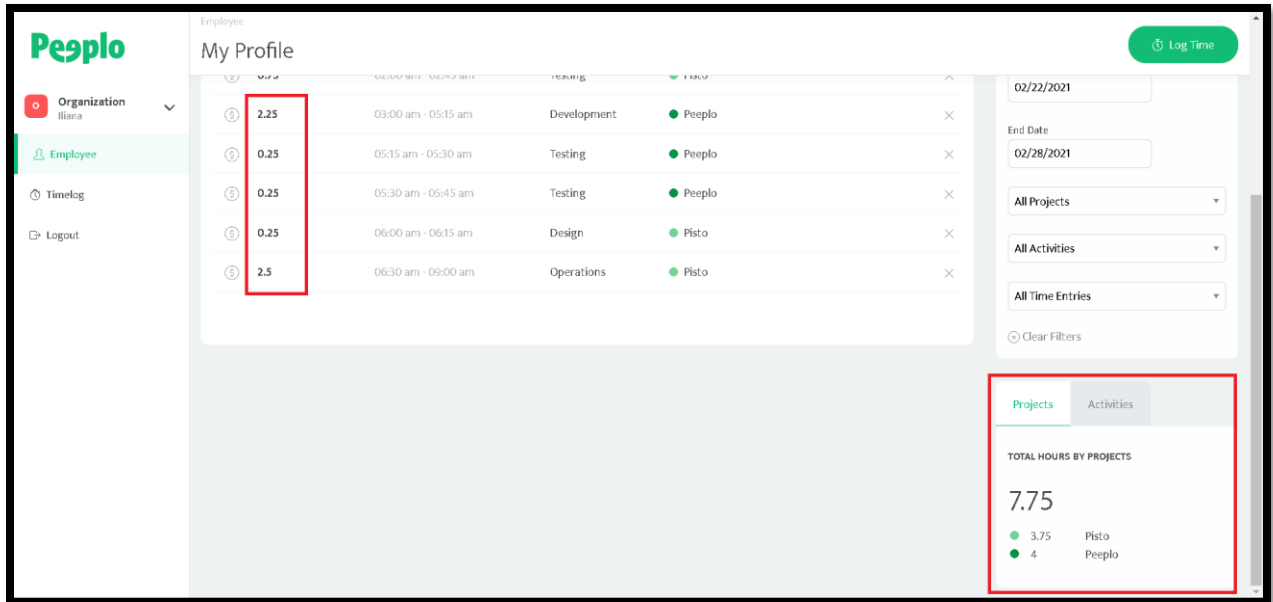


Ilustración 29 - Cálculos en formato decimal

Fuente: (Peeplo, s.f)

4.2 CRONOGRAMA DE ACTIVIDADES

La Ilustración 30 muestra las actividades realizadas durante la práctica profesional.

Actividades	Octubre	Noviembre	Diciembre	Enero	Febrero	Marzo
Inducción	■					
Entrenamiento con las Herramientas		■				
Introducción al Proyecto Peeplo		■	■			
Corrección de Redireccionamiento			■			
Eliminar al Primer Empleado Cuando se Crea una Organización			■			
Corrección de Error al Ordenar Fechas			■			
Implementación de un Chat de Soporte			■			
Implementación de un Indicador de Carga			■			
Implementación del Campo Contacto de Emergencia			■			
Implementación de la Herramienta Zapier			■	■	■	
Creación de Triggers				■	■	
Configuración de un Trigger					■	
Corrección de Calculos en la Duración de Horas						■
Mostrar la Duración de Horas en Formato Decimal						■

Ilustración 30 - Diagrama de Gantt

Fuente: Elaboración propia

V. CONCLUSIONES

1. Se lograron mostrar las soluciones de software implementadas, cumpliendo con los requisitos establecidos por el dueño del producto. Cada funcionalidad desarrollada pasó por un largo proceso de evaluación entre el Scrum máster, el desarrollador senior, el QA y el dueño del producto. De esta forma queda demostrado que las actividades realizadas se completaron correctamente.
2. Se demostró que cada cambio realizado es documentado en una tarjeta de Trello. Esto con el propósito de facilitar las pruebas a realizar por parte del QA.
3. Se evidenció que los errores encontrados fueron corregidos exitosamente satisfaciendo los requisitos solicitados. Cada tarea fue verificada por el QA y aceptada por todo el equipo.
4. Cada actividad desarrollada se trabajó utilizando los frameworks React y NestJS, demostrando la capacidad de trabajo como developer.

VI. RECOMENDACIONES

1. Se deben tener reuniones con el dueño del producto para obtener un enfoque claro de lo que se debe realizar.
2. Se debe evitar la creación de tarjetas en Trello repetitivas o la mezcla entre ellas. A pesar de estar relacionadas, al mezclarlas puede resultar confuso para el equipo.
3. Se debe priorizar el mantenimiento de las librerías utilizadas en el frontend para estar actualizados en el uso de nuevas tecnologías.

BIBLIOGRAFÍA

1. *Acklen Avenue*. (s.f.). Recuperado el 23 de octubre de 2020, de Acklen Avenue: acklenavenue.com
2. Aggarwal, S. (2018). *Modern Web-Development using ReactJS*. Obtenido de <http://ijrra.net/Vol5issue1/IJRR-05-01-27.pdf>
3. Aggarwal, S. (2018). *Modern Web-Development using ReactJS*.
4. *Badgr Pathway*. (s.f.). Recuperado el 24 de octubre de 2020, de Badgr: <https://badgr.com/>
5. *GitLab*. (s.f.). Recuperado el 15 de noviembre de 2020, de <https://gitlab.com/gitlab-org/gitlab>
6. *Google Calendar*. (s.f.). Recuperado el 25 de octubre de 2020, de Google Calendar: <https://www.google.com/calendar/about/>
7. *Google Meet*. (s.f.). Recuperado el 25 de octubre de 2020, de Google Meet: <https://apps.google.com/intl/es-419/meet/>
8. Kanbanize. (s.f.). *Kanbanize*. Recuperado el 12 de noviembre de 2020, de Kanbanize: <https://kanbanize.com/es/recursos-de-kanban/primeros-pasos/que-es-kanban>
9. *Merriam-Webster*. (s.f.). Recuperado el 25 de octubre de 2020, de Merriam-Webster: <https://www.merriam-webster.com/dictionary/software>
10. Murcia Perez, E. S., & Melendez Martínez, J. C. (2013). *Modulo Web Front-end para el Desarrollo de Simulación a Partir de Weibull, JI Cuadrado y Beta*.
11. *nestjs*. (s.f.). Recuperado el 2020 de noviembre de 14, de nestjs: <https://docs.nestjs.com/>
12. *Ngrok*. (s.f.). Obtenido de <https://ngrok.com/docs>
13. Nieto, J. (2016). *Desarrollo de una aplicación web, con Front-end y Back-end, para compraventa de segunda mano*.

14. *Node.js: Getting Started* | Pluralsight. (s.f.). Recuperado el 29 de noviembre de 2020, de <https://app.pluralsight.com/library/courses/nodejs-getting-started/table-of-contents>
15. Peeplo. (2020). *Peeplo*. Recuperado el 7 de 11 de 2020, de Peeplo: <https://peeplo.io/>
16. Pérez, J. E. (2017). *Introducción a JavaScript*.
17. Pluralsight. (s.f.). *Sobre* | Pluralsight. Recuperado el 7 de noviembre de 2020, de Pluralsight: <https://investors.pluralsight.com/>
18. Pressman, R. (2010). *Ingeniería de Software*. Mc Graw Hill.
19. *React*. (s.f.). Recuperado el 14 de noviembre de 2020, de React: <https://es.reactjs.org/>
20. React. (2020). *React*. Recuperado el 12 de noviembre de 2020, de React: <https://es.reactjs.org/>
21. *React Fundamentals* | Pluralsight. (s.f.). Recuperado el 29 de noviembre de 2020, de <https://app.pluralsight.com/library/courses/react-fundamentals-update/table-of-contents>
22. S. Rubin, K. (2012). *Essential Scrum*. Pearson Education.
23. *Slack*. (s.f.). Recuperado el 25 de octubre de 2020, de Slack: <https://slack.com/intl/es-hn/help/articles/115004071768-%C2%BFQu%C3%A9-es-Slack->
24. *Sobre* | Acklen Avenue. (s.f.). Recuperado el 22 de octubre de 2020, de LinkedIn: <https://www.linkedin.com/company/acklen-avenue/>
25. Sutherland, J., & Schwaber, K. (noviembre de 2017). *La Guía de Scrum*. Recuperado el 25 de octubre de 2020, de <https://scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Spanish-SouthAmerican.pdf>
26. *The Kanban Guide*. (julio de 2020). Recuperado el 8 de noviembre de 2020, de The Kanban Guide: <https://kanbanguides.org/wp-content/uploads/2020/07/Kanban-Guide-2020-07.pdf>

27. *The PostgreSQL Global Development Group*. (2020). Recuperado el 14 de noviembre de 2020, de <https://www.postgresql.org/files/documentation/pdf/10/postgresql-10-A4.pdf>
28. *Trello*. (s.f.). Recuperado el 25 de octubre de 2020, de Trello: <https://trello.com/es/about>
29. *Typescript*. (2020). Recuperado el 14 de noviembre de 2020, de Typescript: <https://www.typescriptlang.org/>
30. *TypeScript Fundamentals | Pluralsight*. (s.f.). Recuperado el 29 de noviembre de 2020, de Pluralsight: <https://app.pluralsight.com/library/courses/typescript/table-of-contents>
31. *Zapier*. (s.f.). Recuperado el 15 de noviembre de 2020, de <https://zapier.com/engineering/automating-billions-of-tasks/>
32. *Zapier | ¿Qué es Zapier?* (s.f.). Recuperado el 10 de enero de 2021, de <https://zapier.com/learn/getting-started-guide/what-is-zapier/>
33. *Zoom*. (s.f.). Recuperado el 25 de octubre de 2020, de Zoom: <https://zoom.us/about>