**Documentación Proyecto**

**Plataforma de Gestión de Empleados**

**Concha y toro**

**NRC:** 2113

**Equipo No.** 12

**Integrantes:**

*JUAN PABLO MORENO*

*VICTOR ALFONSO PATIÑO*

*JUAN SEBASTIAN NOVOA*

*ANGELA MELISSA MERCADO*

*JULIANA PAOLA MARTINEZ*

**Descripción de roles del equipo** (*Sprint 1*)

| Rol | Integrante | Descripción | Tareas |
| --- | --- | --- | --- |
| Product Owner | **Concha y Toro.** | **Persona encargada de realizar requerimientos al proyecto y plantear el proyecto.** | * **Comunicar con tiempo los requerimientos del proyecto.** * **Establecer fechas de entregables de sprints.** |
| Scrum Master | **(Tutor)**  **Kevin Santiago Tocora** | **Persona encargada de hacer seguimiento y revisión del equipo.** | * **Revisar el avance de los sprints del equipo.** * **Aportar ideas para el desarrollo de los sprints.** |
| Líder del equipo | **Juan Pablo Moreno** | **Persona encargada de coordinar las entregas y que se cumplan con los entregables de cada sprint** | * **Mantener informados s los integrantes del equipo de las indicaciones del tutor.** * **Coordinar la comunicación entre integrantes, solicitar información pendiente y requerimientos.** * **Revisar que las entregas cumplan con los objetivos del enunciado.** * **Apoyar a los desarrolladores en tareas específicas de horas de ruta y codificación.** |
| Programador Full Stack | **Víctor Alfonso Patiño** | **Persona encargada de revisar, corregir y apoyar en tareas puntuales los componentes de parte visual, lógica y bases de datos.** | * **Creación del concepto de ruta del proyecto en Front End y de la lógica de implementación.** * **Creación del algoritmo principal para Back End.** * **Apoyar Front End y Back End.** * **Implementation de controladores.** |
| Programador Back End | **Juan Sebastián Novoa** | **Persona encargada de desarrollar el componente de lógica y bases de datos.** | * **Implementación de JAVA** * **Implementación de SQL** * **Administración de la base de datos** |
| Programador Back End | **Juliana Paola Martínez** | **Persona encargada de desarrollar el componente de lógica y bases de datos.** | * **Implementación de JAVA** * **Implementación de SQL** * **Administración de la base de datos** |
| Programador Front End | **Angela Melissa Mercado** | **Persona encargada de desarrollar el componente visual UX y ruta del proyecto del proyecto.** | * **Implementación de HTML** * **Implementación de CSS** * **Implementación de JavaScript** |

**Definición de artefactos** (*Sprint 1*)

**Backlog Sprint 1**

|  |  |  |  |
| --- | --- | --- | --- |
| User story | Descripción | Estimación (Horas) | Responsable |
| Definición de roles | * Documento con la descripción de cada rol | 5h | **Líder del equipo** |
| Definición de artefactos | * Backlog del producto * Backlog por sprint * Asignación de backlogs a cada rol definidoc | 3h | **Líder del equipo** |
| Diagrama de clases | * Diagrama descriptivo de las clases más importantes | 2h | **Programador FullStack – Back End** |
| Definición del cronograma | * Definir fechas para cada una de las actividades definidas | 3h | **Líder del equipo** |

**Backlog Sprint 2**

|  |  |  |  |
| --- | --- | --- | --- |
| User story | Descripción | Estimación (Horas) | Responsable |
| Mapa de navegabilidad | * Documento descriptivo del mapa de navegabilidad | 6h | **Líder del equipo** |
| Vistas de la aplicación | * Diseño e implementación preliminar de las vistas * Documento con las imágenes de las vistas preliminares | 15h | **Programador FullStack – Front End** |
| Estilos para las vistas | * Selección e implementación de los estilos para las vistas | 1h | **Programador FullStack – Front End – Lider de Equipo** |
| Creación del proyecto en GIT | * Creación del repositorio y los Push del primer avance del proyecto. | 1h | **Líder del equipo** |

**Backlog Sprint 3**

| User story | Descripción | Estimación (Horas) | Responsable |
| --- | --- | --- | --- |
| Diseño e implementación de los controladores para formularios y otras funcionalidades | * Especificación de ruta * Definición de métodos HTTP permitidos * Lógica algorítmica * Documento descriptivo del diseño y la especificación de los controladores definidos | 48h | **Programador FullStack – Back End – Lider de Equipo** |
| Diseño e implementación de base de datos | * Diseño de diagrama relacional * Diseño e implementación de las tablas de las bases de datos con SQLite * Documento descriptivo de las tablas y las relaciones más importantes | 48h | **Programador FullStack – Back End – Lider de Equipo** |
| Desarrollo de integración de controladores y bases de datos | * Validación de los datos de entrada * Uso de librerías seguras (o prepared statements) para consultar/actualizar las bases de datos * Diseño de queries para consultar/actualizar las bases de datos * Documento descriptivo de las buenas prácticas de programación segura para fortalecer su aplicación | 48h | **Programador FullStack – Back End – Lider de Equipo** |
| Diseño e implementación de portal de acceso usando método de autenticación basado en usuario y contraseña | * Creación de sesiones * Uso de funciones hash criptográficas para almacenar contraseñas usando salts | 48h | **Programador FullStack – Front End - Back End – Lider de Equipo** |

**Backlog Sprint 4**

|  |  |  |  |
| --- | --- | --- | --- |
| User story | Descripción | Estimación (Horas) | Responsable |
| Definición de requerimientos para el despliegue de la aplicación | * Documento descriptivo de los requerimientos y el proceso de despliegue de la aplicación | 5h | **Líder del equipo** |
| Configuración, despliegue y verificación del funcionamiento de la aplicación en una plataforma como servicio (PaaS) | * Despliegue de la aplicación en una PaaS (PythonAnywhere o Heroku) * Verificación de la conectividad al dominio web https://mi\_dominio * Verificación del correcto funcionamiento de la aplicación | 36h | **Programador FullStack – Front End - Back End – Lider de Equipo** |

**Diagrama de clases** (*Sprint 1*)

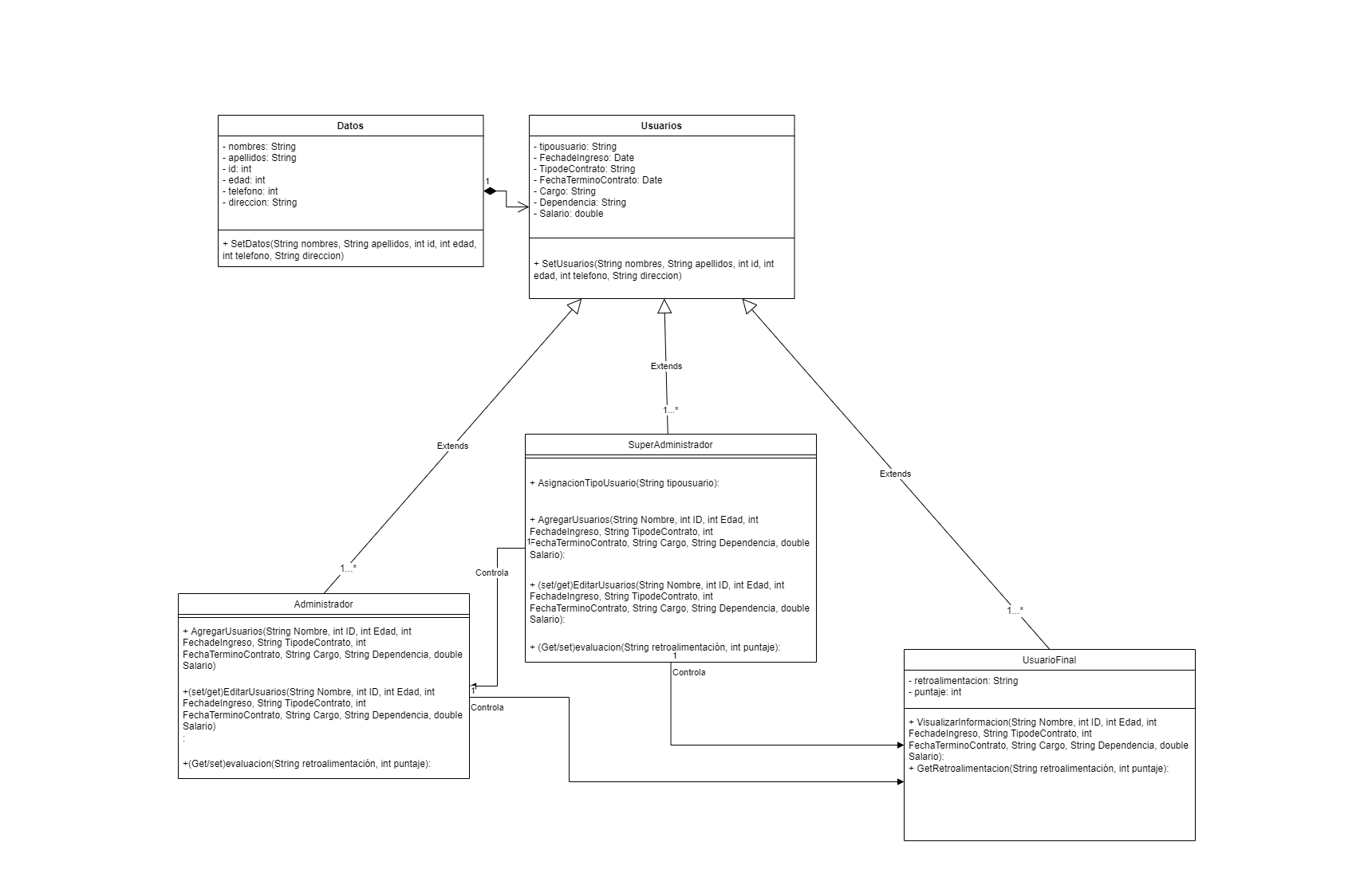


Figura . Diagrama de clases. Fuente. Elaboración propia

**Cronograma de tareas** (*Sprint 1*)

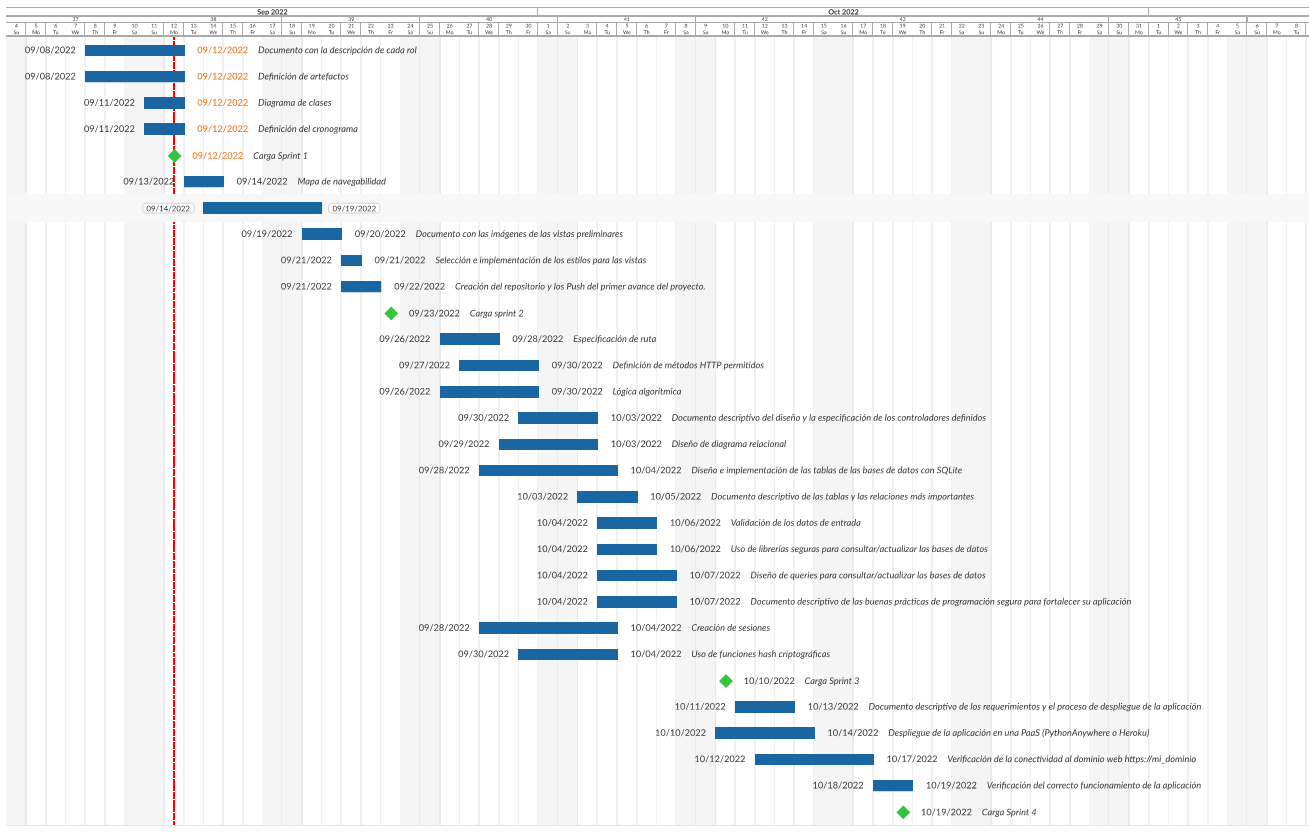


Figura . Cronograma del proyecto. Fuente. Elaboración propia

**Mapa de navegabilidad** (*Sprint 2*)

Diagram

Description automatically generated

**Vistas de la aplicación** (*Sprint 2*)

* LOGIN

****

El inicio de sesión identifica el tipo de usuario automáticamente entre (superadministrador/administrador/usuario final) a partir del email. También permite registrarse o restablecer la contraseña.

* INICIO / INDEX

**Graphical user interface, application, table

Description automatically generated**

El inicio o index del aplicativo permite consultar el informe o información de usuario según el tipo de login realizado, en este ejemplo se muestra el inicio del superadministrador que tiene adición sobre todos los usuarios y administradores de la plataforma.

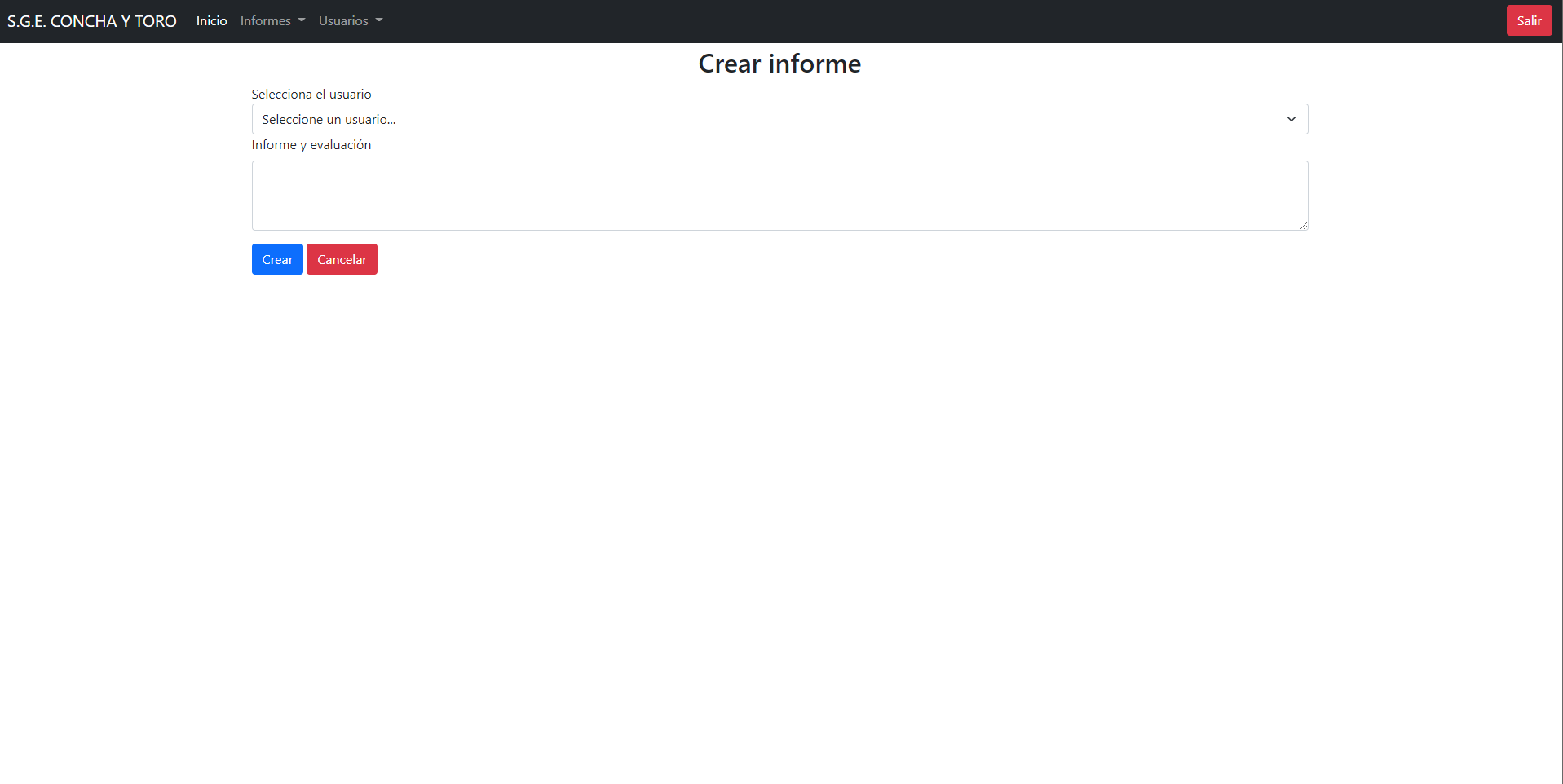
* CONSULTA DE INFORMES

**Graphical user interface, text, application, email

Description automatically generated**

La consulta de informes de empleados permite visualizar los informes creados según el tipo de usuario que ha accedido, en este caso se muestra un ejemplo de consulta de superadministrador o administrador donde aparecen todos los informes de cada empleado.

* CREACIÓN DE INFORMES

****

La vista de creación de informes permite crear un informe y evaluación asignado a un usuario final, este ejemplo corresponde tanto para administrador como para superadministrador.

* CREACION DE EMPLEADOS

**Graphical user interface, text, application, email

Description automatically generated**

La vista de creación de usuarios permite crear usuarios de cualquier de los 3 tipos permitidos (superadministrador/administrador/usuario final). Los tipos de datos están considerados según lo especificado por el cliente Concha y Toro para sus empleados. Este ejemplo corresponde tanto para administrador como para superadministrador.

* EDICION DE USUARIOS

**Graphical user interface, text, application, email

Description automatically generated**

La vista de edición de usuarios permite modificar la información de los usuarios de cualquier de los 3 tipos permitidos (superadministrador/administrador/usuario final). Los tipos de datos están considerados según lo especificado por el cliente Concha y Toro para sus empleados. Este ejemplo corresponde tanto para administrador como para superadministrador.

**LINK DE REPOSITORIO EN GITHUB** (*Sprint 2*)

[**https://github.com/jpml1771/Grupo12\_2113.git**](https://github.com/jpml1771/Grupo12_2113.git)