**Proyecto Gestión Notas**

**NRC: 2120**

**Equipo No.** 10

**Integrantes:**

*Oscar Daniel Gómez Luna*

*Wrayan Alexis Reano Gomez*

*Juan Sebastian Bustamante Chaverra*

*Francisco De Paula Ossa Perez*

*Ana Maria Bendek Rojas*

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

|  |  |  |  |
| --- | --- | --- | --- |
| Rol | Integrante | Descripción | Tareas |
| Scrum Máster | Yurley Katherine Echeverria Leal | Se encargará de velar el cumplimiento de la metodología Scrum por parte del equipo, además guía a su equipo para que se mejore en el flujo de trabajo. | 1. Seguimiento de entregas. 2. Revisión de sprint. 3. Apoyar con ideas. 4. Correcciones. 5. Programar las reuniones |
| Developer Front-End | Oscar Gómez | Se encargará de implementar la parte de la interfaz de usuario, que la web sea atractiva para el usuario y al mismo tiempo, validar su funcionamiento. | 1. Implementar HTML. 2. Implementar CSS. 3. Implementar JavaScript. |
| Developer FullStack | Wrayan Reano | Se encargará de diseñar la parte tanto lógica como de interfaz de usuario. | 1. Apoyar en la implementación de HTML. 2. Apoyar en la implementación de la lógica. 3. Implementar conexión entre el backend y el frontend (controladores). |
| Developer Front End | Ana Bendek | se encargará de que la web sea atractiva para el usuario y, al mismo tiempo, funcione correctamente. | 1. Implementar HTML. 2. Implementar CSS. 3. Implementar JavaScript. |
| Developer Back End | Juan Bustamante | Se encargará de diseñar la lógica y las soluciones para que las acciones solicitadas en el sitio web se puedan llevar a cabo. | 1. Implementar codificación en Python. 2. Implementar framework Flask. 3. Implementar conexión con la base de datos (DB). |
| Full Stack | Francisco Ossa | Se encargará de manipular, limpiar y optimizar la información que se encuentra en una base de datos (DB). | 1. Crear el modelo relacional. 2. Crear la base de datos. 3. Brindar instrucciones en la conexión con el backend. 4. Validar el correcto funcionamiento de la base de datos. |
| Lider de Equipo | Francisco Ossa | Es la persona encargada de maximizar el valor del producto, tomando decisiones continuamente sobre qué construir y qué no construir en el producto. | 1. Determinar los requisitos generales y actividades iniciales del proyecto. 2. Analizar la viabilidad del emprendimiento. 3. Garantizar que el producto se entregue. 4. Definición de tiempos y cronograma. |

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

**Backlog Sprint 1**

|  |  |  |  |
| --- | --- | --- | --- |
| User story | Descripción | Estimación (Horas) | Responsable |
| Crear un documento con la definición y entrega de roles. | Documento donde se definen los roles y se asignan a cada integrante del equipo. | 2 horas | Francisco Osa/ Katherine Echeverria |
| Definición de historias de usuarios (Backlog producto) | Definir historias de usuario de forma general. | 4 horas | Ana Bendek /  Wrayan Reano /  Oscar Gomez / Sebastian Bustamante |
| Asociación de historias de usuarios a sus respectivos spring (Backlog por sprint) | Asignar historias de usuarios según corresponda a sus sprint. | 3 horas | Juan Bustamante/Francisco Ossa |
| Crear un diagrama de clases | Crear un diagrama UML de clases donde se traza claramente la estructura del sistema. | 3 horas | Wrayan Reano |
| Definición del cronograma | Determinar las fechas de inicio y fin planificadas por cada Backlog. | 3 horas | Oscar Gomez |

**Backlog Sprint 2**

|  |  |  |  |
| --- | --- | --- | --- |
| User story | Descripción | Estimación (Horas) | Responsable |
| Mapa de navegabilidad | Documento descriptivo del mapa de navegabilidad | 2 horas | Francisco Ossa |
| Vista de la aplicación | Diseño e implementación preliminar de las vistas | 4 horas | Ana Maria Bendek/Wrayan Reano |
| Vista de la aplicación | Documento con las imágenes de las vistas preliminares | 1 hora | Juan Bustamante/Oscar Gomez |
| Crear Usuarios | Superadministrador crea, visualiza, actualiza, elimina usuarios | 2 horas | Juan Bustamante/Francisco Ossa/Oscar Gomez |
| Estilos para las vistas | Selección e implementación de los estilos para las vistas | 2 horas | Ana Bendek/Oscar Gomez |
| Creación del Proyecto en GIT | Crear el proyecto en GIT | 1 hora | Francisco Ossa |
| Visualizar entorno estudiante | Estudiante visualizará:  Información personal  materias matriculadas  notas por materia  Actividades presentadas | 15 horas | Wrayan Reano/ Oscar Gomez/ Francisco Ossa |
| Visualizar entorno Profesor | Profesor visualizará:  Materias asignadas  Estudiantes Asignados | 15 horas | Juan Bustamante/Ana Bendek |

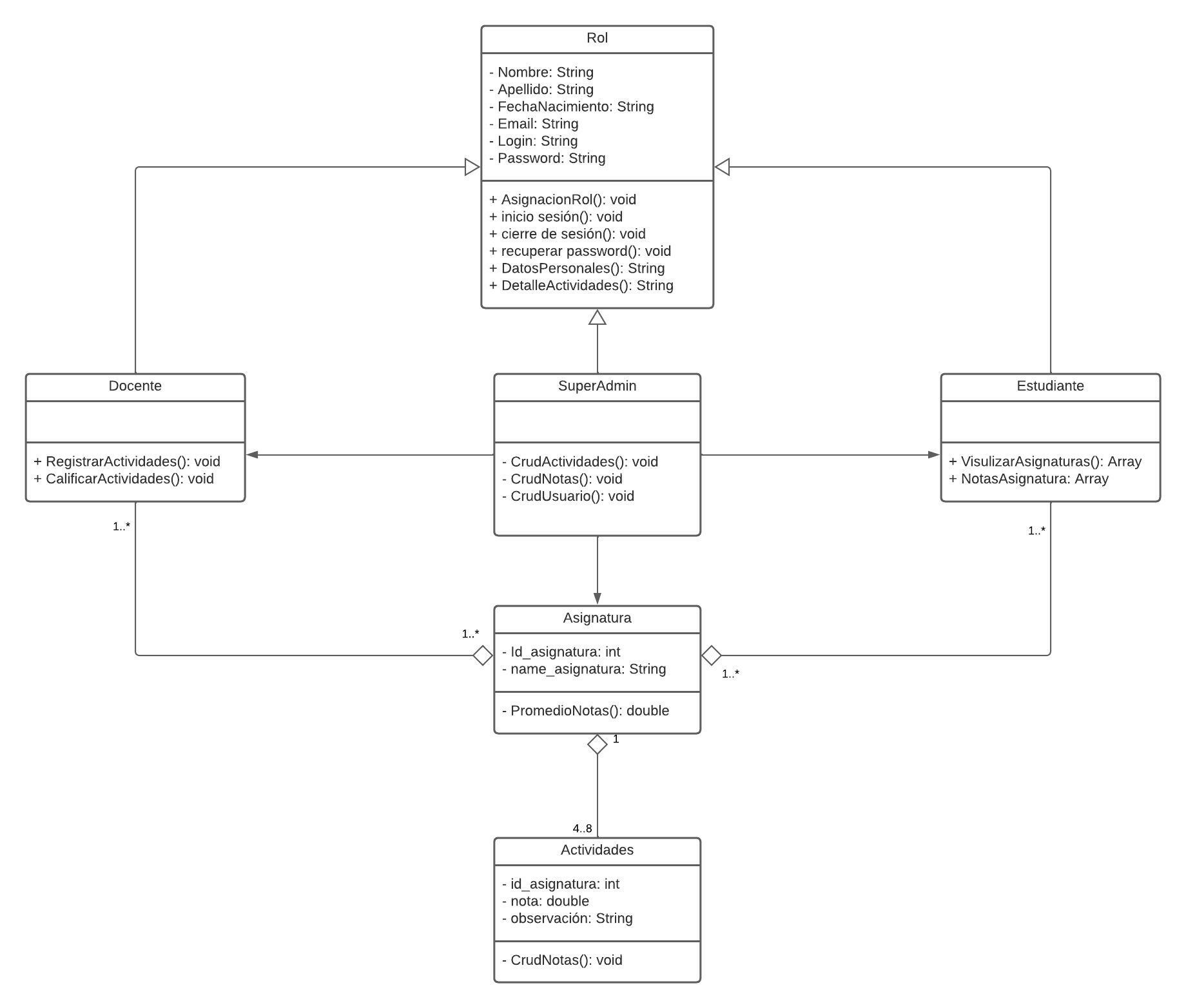
**Backlog Sprint 3**

|  |  |  |  |
| --- | --- | --- | --- |
| User story | Descripción | Estimación (Horas) | Responsable |
| Diseño e implementación de los controladores para formularios | Especificación de ruta | 4 horas | Wrayan Reano/Ana Bendek |
| Diseño e implementación de los controladores para formularios | Definición de métodos HTTP permitidos | 4 horas | Wrayan Reano/Ana Bendek |
| Diseño e implementación de los controladores para formularios | Lógica algorítmica | 6 horas | Juan Bustamante/Francisco Ossa/Oscar Gomez |
| Diseño e implementación de los controladores para formularios | Documento descriptivo del diseño y la especificación de los controladores definidos | 2 horas | Wrayan Reano/Ana Bendek |
| Diseño e implementación de base de datos | Diseño de diagrama relacional | 2 horas | Wrayan Reano/Ana Bendek |
| Diseño e implementación de base de datos | Diseño e implementación de las tablas de las bases de datos con SQLite | 6 horas | Juan Bustamante/Francisco Ossa/Oscar Gomez |
| Diseño e implementación de base de datos | Documento descriptivo de las tablas y las relaciones más importantes | 2 horas | Wrayan Reano/Ana Bendek |
| Desarrollo de integración de controladores y bases de datos | Validación de los datos de entrada | 6 horas | Juan Bustamante/Francisco Ossa/Oscar Gomez |
| Desarrollo de integración de controladores y bases de datos | Uso de librerías seguras (o prepared statements) para consultar/actualizar las bases de datos | 4 horas | Juan Bustamante/Francisco Ossa/Oscar Gomez |
| Desarrollo de integración de controladores y bases de datos | Diseño de quieres para consultar/actualizar las bases de datos | 4 horas | Juan Bustamante/Francisco Ossa/Oscar Gomez |
| Desarrollo de integración de controladores y bases de datos | Documento descriptivo de las buenas prácticas de programación segura para fortalecer su aplicación | 1 hora | Wrayan Reano/Ana Bendek |
| Diseño e implementación de portal de acceso usando método de autenticación basado en usuario y contraseña | Creación de sesiones | 2 horas | Wrayan Reano/Ana Bendek |
| Diseño e implementación de portal de acceso usando método de autenticación basado en usuario y contraseña | Uso de funciones hash criptográficas para almacenar contraseñas usando salts | 4 horas | Juan Bustamante/Francisco Ossa/Oscar Gomez |
| Login Superadministrador | superadministrador hace login | 4 horas | Wrayan Reano/Ana Bendek |
| CRUD roles | Superadministrador crea, visualiza, actualiza, elimina estudiantes, profesores | 3 horas | Juan Bustamante/Francisco Ossa |
| CRUD Materias | Superadministrador crea, visualiza, actualiza, elimina materias | 3 horas | Juan Bustamante/Francisco Ossa |
| Logout | Logout Superadministrador | 1 hora | Juan Bustamante/Oscar Gomez |
| Login | Estudiante hace login | 1 hora | Wrayan Reano/Oscar Gomez |
| Logout | Logout Estudiante | 1 hora | Wrayan Reano/Oscar Gomez |
| Login | Profesor hace login | 1 hora | Wrayan Reano/Ana Bendek |
| Logout | Logout Profesor | 1 hora | Wrayan Reano/Ana Bendek |

**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 | 2 horas | Ana Bendek |
| 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) | 4 horas | Francisco Ossa |
| Configuración, despliegue y verificación del funcionamiento de la aplicación en una plataforma como servicio (PaaS) | Verificación de la conectividad al dominio web https://mi\_dominio | 2 horas | Oscar Gomez |
| Configuración, despliegue y verificación del funcionamiento de la aplicación en una plataforma como servicio (PaaS) | Verificación del correcto funcionamiento de la aplicación | 2 horas | Wrayan Reano |
| Asignar permisos | Superadministrador crea, visualiza, actualiza, elimina permisos a usuarios, según rol | 2 horas | Wrayan Reano/ Ana Bendek |
| Asignar Materias | super administrador asigna materias a profesores y estudiantes | 3 horas | Wrayan Reano/ Francisco Ossa/Ana Bendek |
| CRUD actividades | Profesor crea, visualiza, actualiza, elimina actividades | 3 horas | Francisco Ossa/ Wrayan Reano/Oscar Gomez |
| Calificar actividades | Profesor califica actividades | 5 horas | Juan Bustamante/Ana Bendek |
| CRUD Retroalimentación actividades | Profesor realiza retroalimentación de actividades presentadas por estudiante, también podrá modificar, eliminar y actualizar dichas retroalimentaciones | 5 horas | Francisco Ossa/ Wrayan Reano/Oscar Gomez |
| Promediar notas | Sistema promedio notas (La nota final de una asignatura corresponde al promedio de las actividades desarrolladas) | 2 horas | Juan Bustamante/Ana Bendek |

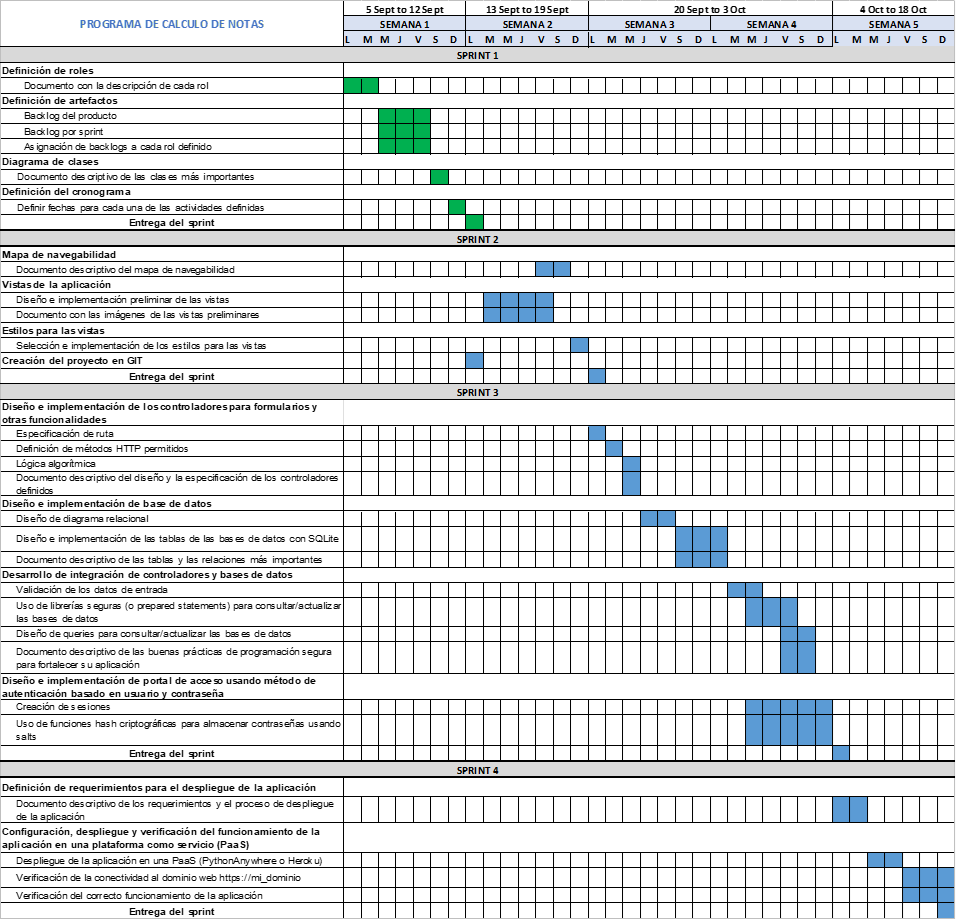
**Diagrama de clases** (*Sprint 1*)



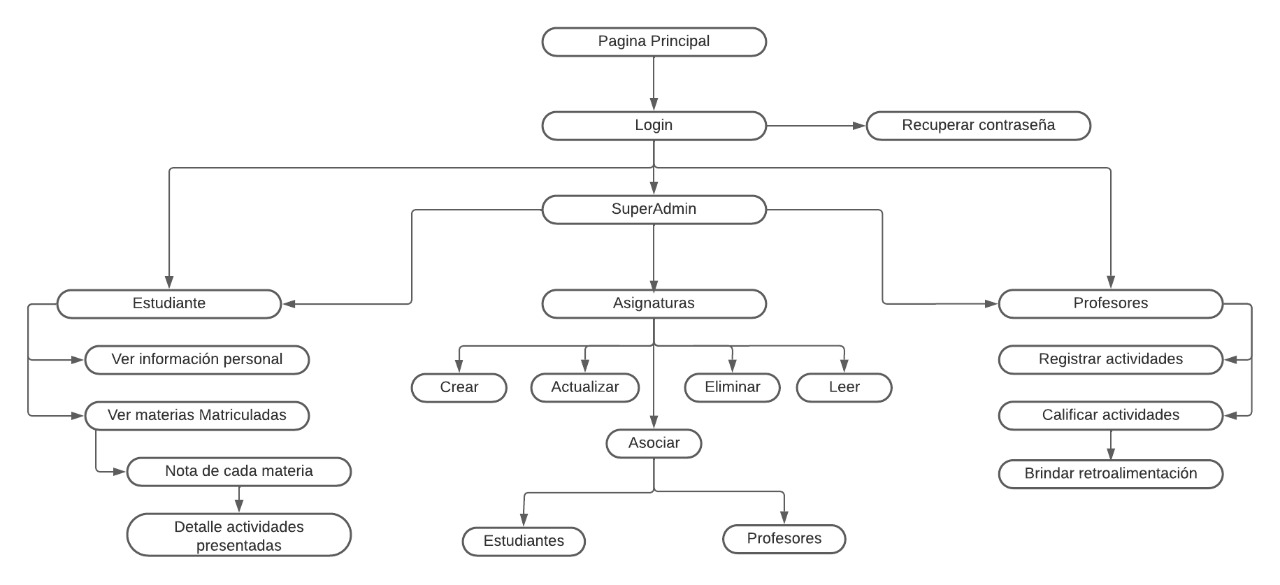
* **Clase Rol:** Es la Superclase donde se encuentran los atributos y métodos generales que utilizan las clases que extienden de ella.
* **Clases Docentes, Estudiantes y SuperAdmin:** Son subclases que extienden de la superclase (Rol), pero que también tienen unos métodos propios de su clase.
* **Clase SuperAdmin:** En el caso de la clase SuperAdmin tiene una asociación unidireccional con las clases Docente, Estudiantes y Asignaturas, es decir, que desde la clase SuperAdmin se puede acceder a las clases mencionadas, pero no en sentido contrario.

Tanto la clase Docente como Estudiante tienen una asociación de agregación con Asignaturas y una relación de uno a muchos, por ejemplo, la clase Docente puede tener una o muchas Asignaturas y Asignaturas puede tener uno o muchos Docentes, lo mismo sucede con Estudiantes.

La clase Actividades tiene una asociación de agregación con Asignatura y a su vez la relación es unitaria, es decir, que la clase Actividad solo puede estar asociada a una Asignatura, mientras la clase Asignatura puede tener entre cuatro y ocho Actividades.

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

**Mapa de navegabilidad** (Sprint 2)



Tenemos un mapa jerárquico, el cual se inicia con un formulario de ingreso y permite al usuario recuperar contraseña, en caso de no recordarla, enviándolo a otra visualización. Dependiendo del tipo de usuario que ingrese, puede llevarlo a tres pantallas con limitantes en las opciones como tal seria: super administrador, estudiantes, profesores.

**Pantallas según el perfil**

* Estudiantes: contaría con la visualización de información personal, ver materias matriculas, nota de cada materia y esta desplegara el detalle de las actividades.
* Profesores: podrá ver los estudiantes, registrar actividades, calificar actividades y retroalimentarlas.
* Super administrador: tendrá todas las opciones del (CRUD) en cuanto a materias, estudiantes, profesores.

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

* **Vista N°1 – Inicio de Sesión:**



* **Vista N°2 – Recuperación de Contraseña**

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* Vista N°3 – Perfil Estudiante:



* **Vista N°4 – Vista del Estudiante – Visualización de materias y calificaciones**

Interfaz de usuario gráfica

Descripción generada automáticamente

* **Vista N°5 – Perfil del Profesor:**

Interfaz de usuario gráfica

Descripción generada automáticamente

* **Vista N°6 – Vista del profesor-creación de actividades:**

Interfaz de usuario gráfica

Descripción generada automáticamente

* **Vista N°7 – Vista del profesor-Calificación de actividades:**

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

* **Vista N°8 – Super Admin- Creación de Roles**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

* **Vista N°9 – Super Admin- Asignación de materias a profesores y estudiantes**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

* **Vista N°10 – Super Admin- Listado de Materias Creadas:**

Interfaz de usuario gráfica

Descripción generada automáticamente

* **Vista N°11 – Super Admin- Listado de Estudiantes Activos:**

Interfaz de usuario gráfica

Descripción generada automáticamente

* **Vista N°12 – Super Admin- Listado de Profesores Activos**

Interfaz de usuario gráfica

Descripción generada automáticamente