**¿Cuál es el proyecto que se va a desarrollar?**

**Planteamiento de la situación problema que se va a desarrollar**

Se propone analizar, diseñar y construir una aplicación software que permita controlar los ingresos y los egresos de una empresa de manufactura y hacerles el correspondiente seguimiento.

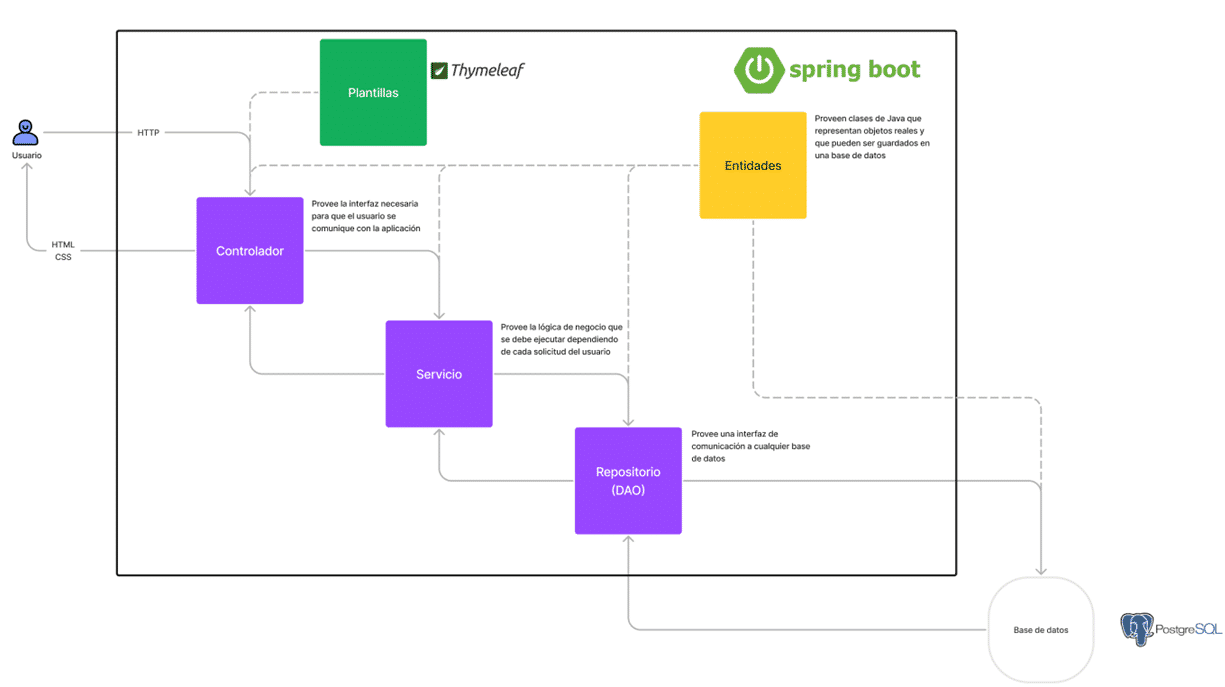
Para llevar a cabo este proyecto deberás tener en cuenta una serie de **indicaciones**, como verás a continuación.

**Primera indicación.** Debes reunirte con cinco compañeros del curso para conformar un equipo de trabajo y asignar un nombre al proyecto. Posteriormente se deben distribuir los roles y las funciones para cada uno de los integrantes (tabla 2). Todos ellos deben ser desarrolladores.

**Tabla 2.** Roles y funciones de los integrantes del equipo

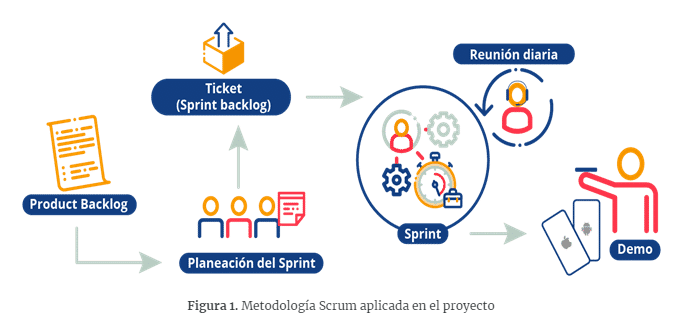
| **Rol** | **Función del rol** |
| --- | --- |
| 1. Desarrollador | Generar las fuentes para el frontend y el backend |
| 2. Analista | Levantar las necesidades del negocio para convertirlas en requerimientos del sistema |
| 3. Product Owner | Conocer el producto *software* que se va a elaborar en el proyecto |
| 4. Administrador de BD | Gestionar y disponer el motor de base de datos para el proyecto de desarrollo de *software* |
| 5. Scrum Master | Gestionar todas las ceremonias y actividades de la metodología Scrum y asignar los recursos a cada rol |

**Arquitectura de la solución propuesta para el proyecto final**



* **¿Cuál es la metodología que se utilizará para llevar a cabo el proyecto?**

El desarrollo del proyecto se abordará mediante metodologías ágiles, usando el marco de trabajo **Scrum** (figura 1), consistente en un proceso que se aplica de manera regular y frecuente a las buenas prácticas para trabajar colaborativamente en equipo y obtener el mejor resultado posible de un proyecto.



**Daily** (diligenciamiento del formulario), se debe establecer la ruta de ejecución de cada uno de los sprints. Un sprint normalmente tiene las siguientes ceremonias:

* **Sprint Planning**, que se realiza al comenzar el sprint.
* **Daily Scrum** (conocido comúnmente solo como la **Daily**), que es una reunión diaria de 15 minutos en la que deben participar todos los integrantes del equipo. En esta reunión, cada una de las personas del equipo responden las siguientes preguntas:
  + ¿Qué hice ayer para contribuir al Sprint Goal?
  + ¿Qué voy a hacer hoy para contribuir al Sprint Goal?
  + ¿Tengo algún impedimento que me impida entregar?
* **Sprint Review**, que es la reunión que ocurre al final del sprint. Es organizada por el producto **Owner** y se revisa cuál es la situación de todas las historias (**Backlog**) y sus nuevas condiciones para priorizar el trabajo posterior.

Para lograr llevar a cabo el proyecto es necesario contar con un grupo de herramientas que permitirán organizar, planear, estructurar y desarrollar la aplicación web.

**IDE**(interface development environment). En el ciclo 3 se utilizará IntelliJ para escribir todo el código en Java Spring Boot y Thymeleaf.

**Git.** Es un *software* de control de versiones de código distribuido, que se instala en la máquina local del desarrollador.

**Github (repositorio)**. Es una carpeta para alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente de programas de ordenador.

**Trello (administrador de proyecto)**. Es un *software* de administración de proyectos que permite llevar el registro de todas las metas determinadas para cada uno de los sprints, así como las actividades de su responsable.

**Heroku**. Es una plataforma en la nube que permite a las empresas construir, entregar y supervisar aplicaciones para luego poder alojarlas en la nube y posibilitar el acceso a los usuarios de las aplicaciones que se construyen.

**Stack tecnológico**: Thymeleaf con Bootstrap en el front end, con un servidor en Java Spring Boot con JPA como ORM y una base de datos de PostgreSQL. Todo el sistema está desplegado en Heroku, excepto la base de datos que se encuentra en Supabase.

Además, el lenguaje que se puede usar es Java, y el *stack* tecnológico a emplear será:

* HTML 5
* CSS3
* Bootstrap
* Java Spring Boot
* Thymeleaf
* PostgreSQL
* (JWT / Oatuh 2)
* (Heroku)
* (Git / GitHub)

**¿Qué debo alcanzar para desarrollar la aplicación web?**

Para que puedas desarrollar el proyecto que se te planteó, la aplicación web debe contar con los siguientes módulos:

**Módulo administrador de empresas.** Permite a un superadministrador crear una nueva empresa dentro del sistema, incluyendo datos de contacto y descripciones de la empresa.

**Módulo para registrar un empleado**. Permite a un administrador crear un empleado con rol operativo dentro de la empresa.

**Gestión de egresos**. Permite tanto a un administrador como a un usuario operativo registrar un egreso en la empresa, pudiendo especificar montos, descripciones y categorías.

**Gestión de ingresos**. Permite tanto a un administrador como a un usuario operativo registrar una nueva venta o un nuevo ingreso, pudiendo especificar montos, conceptos y fechas.

**Gestión de pérdidas y ganancias**. Permite a los usuarios visualizar una tabla con la información de los egresos y los ingresos, pudiendo obtener un total para saber si la empresa pierde dinero.

**Gestión de usuarios**. Este módulo permite a cada usuario iniciar sesión y visualizar diferentes partes de la aplicación dependiendo de su rol.

| **Semana de entrega** | **Sprint** | **¿Qué se desarrollará?** | **Porcentaje de evaluación** |
| --- | --- | --- | --- |
| Semana 2 | **Sprint 1:** Creación de un equipo de trabajo Definición de los roles de los integrantes Creación de un repositorio en Github | * Crear un equipo de trabajo y asignar los roles de cada uno de los integrantes según la metodología Scrum * Definir las ceremonias de Scrum y el calendario de cada una * Tener creado el repositorio de las fuentes en GitHub * Realizar pull request con cada cambio al proyecto * Tener creadas las ramas principales development, release, main * Dar permisos a todos los integrantes del equipo * Hacer comentarios sobre cada cambio que se sube al repositorio   Todos los integrantes realizan cambios a las fuentes, que se deben ver reflejados en el historial | 25 |
| Semana 3 | **Sprint 2:**Creación de entidades (Java POO) | * Modelar una empresa como una clase de Java * Modelar un empleado como una clase de Java * Modelar el movimiento de dinero como una clase de Java | 25 |
| Semanas 4 y 5 | **Sprint 3:**Backend | * Crear entidades, controladores y servicios para cada uno de los modelos * Implementar las reglas de negocio de la aplicación | 25 |
| Semanas 6 y 7 | **Sprint 4:**Front-end | * Implementar el frontend de la aplicación * Implementar autenticación en la aplicación * Desplegar la aplicación en Heroku | 25 |