

Ambiente de Ingeniería de Software

**Recordatorio del proyecto:** Este proyecto de software consiste en una aplicación que entrega información sobre recorridos de medios de transporte.

# Metodología de desarrollo:

Para este proyecto la metodología de desarrollo que utilizaremos será el desarrollo ágil a través de Feature Driven Development y Kanban. Estas metodologías nos permiten llevar un control de las funcionalidades que se deben implementar de una manera visual y ordenada, ayudándonos en la distribución de las tareas a programar, probar o pasar a producción. Kanban en conjunto con FDD, suponen igualmente un acierto dado el contexto de nuestro proyecto, donde no poseemos un cliente como tal, sino que al ser un proyecto de software comercial, poseemos un público objetivo del cual recibiremos retroalimentación para el desarrollo, lo que significa una gran cantidad de puntos de vistas e ideas que para ser desarrolladas requerirán de una metodología flexible y ordenada.

* **Feature Driven Development** es una metodología iterativa e incremental, que centra su funcionamiento en las funcionalidades importantes para el cliente a desarrollar. FDD posee 5 procesos para su funcionamiento:

1. **Diseñando un modelo general:** Esta metodología necesita primeramente de un modelo general que permita a todos los miembros del equipo el conocer la problemática o el contexto del problema a resolver, para ello **todos** los miembros del equipo colaboran en la construcción de un modelo general que describa la problemática.
2. **Armando una lista de funcionalidades:** Dado que las **Funcionalidades** o **Features** resultan vitales para esta metodología, es necesario el construir una lista con las funcionalidades que deberá poseer la aplicación para satisfacer las necesidades de los usuarios.
3. **Orden y planificación por funcionalidad:** Continuando con el paso anterior, se deberán ordenar las funcionalidades dependiendo del valor agregado a la aplicación, considerando riesgos asociados, complejidad y pre o post requisitos de las mismas. Luego de tener ordenadas las funcionales, se elegirá un número acotado de estas para implementar.
4. **Diseño por funcionalidad:** Para cada funcionalidad se deberá realizar un diseño que permita la comprensión total de los miembros del equipo para su implementación.
5. **Desarrollo por funcionalidad:** Finalmente se realizará el desarrollo de las funcionalidades por parte del equipo.

* **Kanban** es una metodología visual que permite la construcción de un producto de software mediante un tablero con etiquetas o tarjetas (Funcionalidades o Actividades dentro del proceso de desarrollo), que deberán alcanzar diferentes estados para conceder un resultado óptimo y esperado por el usuario final. Kanban se centra en el trabajo colaborativo, sin dar importancia a una jerarquía dentro del equipo de desarrollo, permitiendo a todos los miembros la colaboración en una determinada tarea y así cumplir con los requerimientos del usuario de una manera amigable con los desarrolladores, esto dado que su trabajo se ve dividido y apoyado entre todos los miembros del equipo.

FDD y Kanban trabajan muy bien juntos, debido a que esta modalidad de tableros permite ordenar y priorizar las actividades y/o funcionalidades de una manera cómoda, rápida y visualmente ordenada, distribuyendo los esfuerzos entre los miembros del equipo de trabajo, lo que para nuestro proyecto supone una ventaja debido al número reducido de personal.

# Herramientas:

Para el desarrollo de esta aplicación, serán utilizados las siguientes tecnologías y lenguajes de programación:

# Leguajes:

* + **JavaScript:** Lenguaje de programación ligero e interpretado open source. Utilizado en desarrollo web para interfaces de usuario e igualmente para desarrollo en ambientes externos al navegador como lo es Node.js.
  + **CSS:** Cascading Style Sheets es un lenguaje de estilos utilizado para describir la presentación de documentos HTML o XML. Este se encarga de describir como debe ser renderizado un elemento.

# Framework:

* + **React Native:** Bajo el slogan de React Native “learn once, write everywhere” este framework permite la creación de aplicaciones móviles para la mayoría de las plataformas con un único código. Basado en JavaScript con un enfoque orientado a componentes, este permite una programación rápida y simple de aplicaciones móviles utilizando componentes nativos ya sean estos de iOS o Android.

# Base de Datos:

* + **Firebase:** Basede datos no relacionalorganizada en forma deárbol JSON. Esta base de datos creada por Google permite subscribirse en tiempo real, devolviendo información de la BDD si esta posee cambios.

# Entorno de desarrollo:

* + **Visual Studio Code:** Editor de código multilenguaje desarrollado por Microsoft. Este editor de código posee integración con Git lo que permite mantener un control más rápido e interactivo de las versiones del producto.
  + **Git:** Software que permite llevar un control de versiones. Utilizado para llevar un registro de los cambios en archivos presentes en un equipo, con el objetivo de coordinar los esfuerzos que varias personas realizan sobre estos archivos.