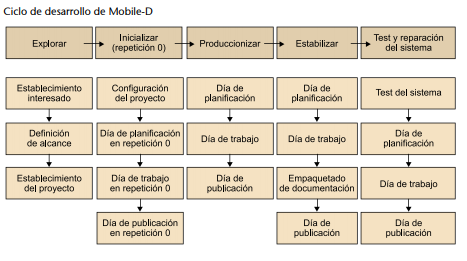
**Metodología Mobile D para el desarrollo del proyecto OTCA**

Es una metodología del tipo desarrollo ágil de Software, sin embargo esta más orientada al desarrollo de aplicaciones móviles. El objetivo es conseguir ciclos de desarrollos muy rápidos en equipos pequeños (es nuestro caso). Se basa método basado en soluciones conocidas y consolidadas: Extreme Programming (XP), Crystal Methodologies y Rational Unified Process (RUP), XP para las prácticas de desarrollo, Crystal para escalar los métodos y RUP como base en el diseño del ciclo de vida.

Se compone de varias fases: exploración, inicialización, fase de producto, fase de estabilización y la fase de pruebas; cada una posee un día de planificación y un día de entregas. Posee iteraciones en la fase de producto donde la entrada a la segunda iteración de la fase de producto es el resultado de la iteración 0 y todo está controlado bajo una versión para el proyecto.



Cada fase (excepto la inicial) tiene un día de planificación y otro de entrega.

Las fases son:

**• Exploración:** En esta fase se planificara y se conceptualizara el proyecto, esta es diferente del resto de fases, se establecen los alcances del proyecto y su relaciones con las funcionalidades de las metas propuestas en el modelo *Wireframing.*

Los objetivos de la fase de exploración son:

* Establecer los grupos de actores necesarios en la planificación y el seguimiento del proyecto de desarrollo de software.
* Definir los alcances y límites del proyecto de desarrollo de software de desarrollo.
* Planificar el proyecto respecto al entorno, el personal y los problemas del proceso.

Las entradas de la fase de exploración son:

* La propuesta del producto.
* Biblioteca de procesos de Mobile D.
* Contrato.
* Documento de requisitos iniciales.
* Normas y restricciones en caso de que existan.

Las salidas de esta fase son:

* El documento de requisitos iniciales donde se ha definido los requerimientos iniciales del desarrollo del producto.
* Plan de proyecto incluyendo línea de tiempo, el ritmo, las terminaciones, los recursos del proyecto, los actores y sus responsabilidades.
* Descripción base del proceso que incluye la línea de base, las actividades de seguimiento de calidad, documentación, puntos de integración el hardware a llegar las salidas.
* Plan de Medición y plan de Formación, descripción de la línea de la arquitectura.

**• Inicialización:** En esta fase se prepararán e identificarán todos los recursos necesarios, se establece el entorno técnico y sus responsabilidades.

Los objetivos de esta fase son:

* Obtener una buena compresión global del producto para el equipo de desarrollo del proyecto, sobre los requisitos iniciales y la línea de la arquitectura.
* Preparar los requisitos físicos, técnicos y humanos, así como la comunicación con el cliente, los planes del proyecto y todas las cuestiones fundamentales de desarrollo a fin de que todo esté en plena disposición para la implementación.

Las entradas de esta fase son:

* Documento de requisitos Iniciales.
* Plan de proyecto y descripción del proceso base.
* Plan de medición y plan de formación.
* Descripción de la línea de arquitectura

Las salidas de la fase son:

* Plan de proyecto actualizado
* La primera versión del diseño de software.
* Documento con descripción del diseño.
* Funcionalidad implementada.
* Documento de requisitos iniciales actualizados.
* Desarrollo de notas y la interfaz de usuario.
* Ilustración de cada requisito.
* Pruebas aceptadas de cada requisito

• **Producción o fase de producto:** Es la fase donde se repetirán iterativamente las sub-fases, con un lapso de tiempo breve de planificación, uno de trabajo y uno de entrega. Aquí se utilizarán técnicas como la del *test driven development,* para conseguir la mayor calidad posible.

Los objetivos de esta fase son:

* Implementar la funcionalidad del producto priorizando los requerimientos del cliente.
* Centrarse en la funcionalidad básica fundamental para permitir múltiples ciclos de mejora.

Las entradas de esta fase son:

* Actualizado del plan de proyecto y plan de la línea de la arquitectura.
* La primera versión de la arquitectura de software y descripción del diseño.
* Planes para la comprobación de los elementos críticos del desarrollo.
* Funcionalidad implementada
* Métrica de datos
* Experiencia del equipo de proyecto.
* Historia y tarjetas de tareas.
* Manuales, especificaciones API y material de apoyo.
* Pruebas unitarias.

Después de cada Iteración la entrada de la siguiente es:

* Los resultados de la iteración anterior.

Los elementos de salida de esta fase son:

* Funcionalidad Implementada.
* Documento de aceptación de pruebas.
* Ilustraciones de Interfaz de Usuario (maqueta).
* Actualización del plan del proyecto.
* Historia y tarjetas de tareas.
* Conocimiento de los requisitos del sistema y pruebas de aceptación.
* Lista de defectos.
* Documento de requisitos iniciales.
* Informe de estado diario.

• **Fase de estabilización:** En esta fase se llevarán a cabo las acciones de integración para asegurar que el sistema completo funciona correctamente.

Los objetivos de la fase de estabilización son:

* Finalizar la implementación del producto.
* Mejorar y garantizar la calidad del producto.
* Finalizar la documentación del proyecto.

Las entradas de la fase de estabilización son:

* La funcionalidad implementada del producto.
* Los artefactos de desarrollo relacionado.

Las salidas de esta fase son:

* La funcionalidad implementada de todo el proyecto de todo el software.
* La documentación del producto finalizado.

• **Fase de pruebas y reparación:** Tiene como meta la disponibilidad de una versión estable y plenamente funcional del sistema según los requisitos del *StackHolder*.

Los objetivos de la fase de pruebas son:

* Probar el sistema basado en la documentación producida en el proyecto.
* Proporcionar información de defectos encontrados.
* Planificar la solución a los defectos encontrados.
* Fijar los errores hallados.
* Producir un sistema libre de errores como sea posible.

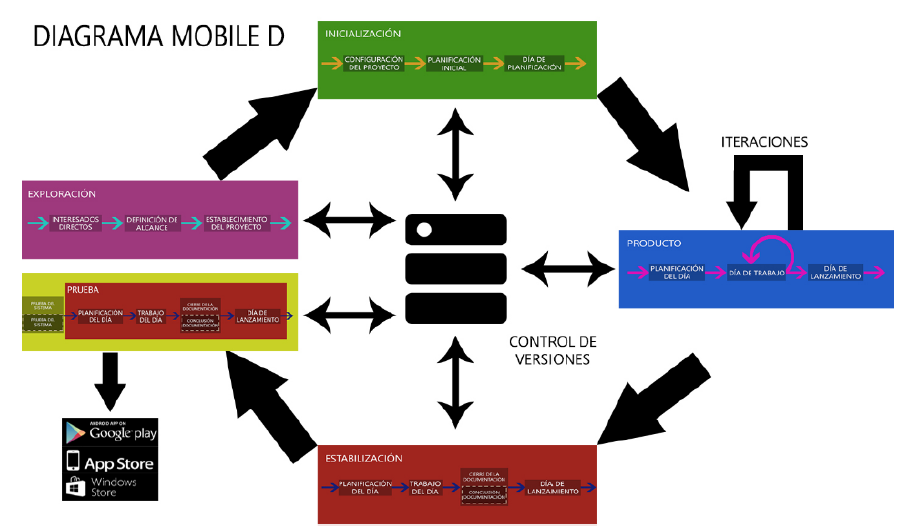
Las entradas de esta fase son las siguientes:

* La funcionalidad implementada.
* Documentación de aceptación de pruebas.
* Funcionalidad del usuario definida completamente.
* Descripción de la interfaz de usuario que se utiliza para crear casos de pruebas.

Las salidas de la fase de pruebas son:

* Un sistema testeado y corregido (versión final)
* Documentación de errores encontrados.
* Informe de pruebas del sistema descripción del proceso de pruebas y los errores y defectos encontrados en el software.
* Registro de pruebas realizados en el sistema y los resultados obtenidos al momento de ejecutar el testeo.

A continuación se muestra el esquema integrado de la metodología de desarrollo:



**Plan de Trabajo**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Meta del Plan de Trabajo | **Desarrollo de un sistema de trazabilidad para el comercio de productos y subproductos de lagarto** | | | | | | |
| Número | Objetivo a alcanzar | Acciones | Responsable | Fecha de Inicio | Fecha de Fin | Desviación en el tiempo (días) | Recursos |
| 1 | Documentación inicial y planificación | Investigación y análisis de tiempo | Developers | Semana 1 | Semana 2 | 1 | Office |
| 2 | Maquetado y aprobación | Consulta y diseño | Developers | Semana 2 | Semana 3 | 2 | Adobe XD |
| 3 | Diagrama de Clases | Análisis de actores | Grupo 1 | Semana 3 | Semana 3 | 1 | Enterprise Architect |
| 4 | Diagrama Entidad-Relación | Análisis BD | Grupo 2 | Semana 3 | Semana 3 | 1 | Enterprise Architect |
| 5 | Conf. Base de Datos | Creación de BD | Grupo 2 | Semana 4 | Semana 4 | 3 | Firebase |
| 6 | Frontend Login | Programación de pantalla de Login | Grupo 1 | Semana 4 | Semana 4 | 3 | Flutter |
| 7 | Frontend Pantallas Principales | Programación de pantallas | Grupo 1 | Semana 4 | Semana 6 | 4 | Flutter |
| 8 | Autentificación | Validación de usuarios | Grupo 2 | Semana 5 | Semana 5 | 1 | Firebase |
| 9 | Frontend Reportes | Programar presentación de reportes | Grupo 1 | Semana 5 | Semana 5 | 2 | Flutter |
| 10 | Generación de Reportes | Análisis de datos ingresados | Grupo 2 | Semana 6 | Semana 6 | 3 | Flutter |
| 11 | Administración de Archivos Multimedia | Programar manipulación de fotos | Grupo 2 | Semana 6 | Semana 6 | 4 | Flutter, Firebase |
| 12 | Conexión BD | Enlazar datos | Grupo 2 | Semana 6 | Semana 7 | 5 | Firebase |
| 13 | Pruebas de rendimiento | Revisar desempeño | Developers | Semana 7 | Semana 9 | 2 | Google Play console, Firebase Test Lab |
| 14 | Pruebas de seguridad | Verificar vulnerabilidad de datos | Developers | Semana 8 | Semana 9 | 3 | Ingeniería Inversa, Google Play console |
| 15 | Verificación de errores y retroalimentación | Revisión de errores | Developers | Semana 9 | Semana 10 | 3 | Firebase Test Lab |
| 16 | Testeo | Prueba de usuarios | Grupo 1 | Semana 10 | Semana 10 | 5 | Personal |

**Cronograma de Trabajo**

