
MonkeyBits

Calle Dr. Severo Ochoa, s/n
18001 Granada (Granada)
(34) 958 723 645



MANUAL DE COORDINACIÓN

14 de octubre del 2018

Pablo Moreno Megías

ÍNDICE

1. Ciclo de vida	3
2. Metodología de desarrollo	3
3. Recursos software de desarrollo	4
4. Organización del equipo de trabajo	4
5. Herramientas de comunicación en el equipo de trabajo	5
6. Relaciones con el cliente	6
7. Estándares de documentación	6
8. Estándares de código	7
9. Control de versiones	8
10. Gestión de calidad	8

1. CICLO DE VIDA

Se trabajará a modo de espiral de Boehm, en la que en cada iteración se planificará los objetivos, se realizarán los diferentes diagramas de clases y secuencia y a partir de ahí se volverá a crear un prototipo con esta funcionalidad añadida.

En cada iteración se realizarán sprints de una semana y al finalizar cada sprint el equipo se reunirá para ver el progreso (cada viernes durante 10 minutos durante las horas de clase).

Vamos a realizar 3 iteraciones:

- En el primer sprint se realizarán las tareas de planificación global y de la propia iteración, el diseño de la aplicación y la implementación de algunas funcionalidades de la aplicación. Además, el equipo se documentará sobre la programación en **Android Studio** y sobre las características que la aplicación debe presentar para que sea accesible. Como resultado del sprint, obtendremos un diagrama de Gantt, un diagrama de clases, dos diagrama de secuencia y un prototipo de la aplicación que permitirá al cliente ver cómo se realiza un inicio de sesión y la gestión de perfil. Al finalizar el sprint, se deberá comprobar que los resultados cumplen lo establecido mediante las pruebas adecuadas.
- En la segunda iteración, se realizarán las tareas de la planificación de la iteración y la implementación de más funcionalidades de la aplicación.
- En la tercera iteración, se volverá a realizar la planificación de la iteración y se acabará de implementar la aplicación.

2. METODOLOGÍA DE DESARROLLO

Vamos a usar la metodología de desarrollo ágil SCRUM, para fomentar el trabajo en equipo. Para ello vamos a realizar 3 iteraciones de 4 semanas. En cada iteración realizaremos una búsqueda de requisitos funcionales y no funcionales y una planificación de la misma. Cada semana el equipo se reunirá entorno a 10 minutos para sincronizarse al principio de cada clase de prácticas (Viernes a las 11:30). Al final de cada iteración, el equipo deberá realizar una revisión de los documentos y del código que se ha hecho durante dicha iteración. Se trabajará repartiendo a los miembros del equipo en 4 grupos: Gestores, Desarrolladores, Testers y Encargado de accesibilidad.

En cada iteración, los miembros rotarán en a otro grupo, excepto el encargado de accesibilidad. El número de miembros de cada equipo es variable y se podrá ajustar en cada iteración en función de las necesidades del proyecto. Así, las diferentes historias se repartirán a cada grupo para fomentar la mayor productividad en el trabajo. Se podrá trabajar cualquier día, ya que lo verdaderamente importante es que se consigan los objetivos propuestos en cada sprint, cada miembro podrá organizarse como él crea oportuno.

3. RECURSOS SOFTWARE DE DESARROLLO

- A la hora de desarrollar, usaremos el entorno de programación **Android Studio**, ya que nos permitirá realizar una app **Android** de manera sencilla, al igual que el desarrollo de interfaces atractivas y de fácil uso para el usuario.
- Usaremos la base de datos de **Google** para guardar nuestros usuarios y todos sus datos. Sin embargo, usaremos una base de datos local para almacenar nuestros administradores.
- Para la obtención de los mapas de la ciudad que nos permitan crear rutas vamos a utilizar la API de **Google Maps**.
- Se usará un repositorio **GitHub** y la herramienta **GitKraken** para el almacenamiento y trabajo durante el desarrollo.

4. ORGANIZACIÓN DEL EQUIPO DE DESARROLLO

Para la realización del proyecto, los miembros del equipo se repartirán de la siguiente forma para la primera iteración:

- 2 miembros encargados de la gestión, planificación y la documentación.
- 3 miembros encargados de la especificación, el diseño y la implementación.
- 1 miembro encargado de la realización de pruebas.
- 1 miembro encargado de la usabilidad y la accesibilidad.

Si en algún momento del desarrollo del proyecto un departamento necesitase ayuda para la correcta realización del mismo, los miembros de los demás departamentos deberán colaborar para que la consecución de los objetivos sea posible.

La toma de decisiones la tomarán los encargados de la gestión, siempre consultando y teniendo en cuenta las opiniones de los demás miembros del equipo. El reparto de

trabajo dentro de cada departamento será realizado de manera interna y equitativa en el mismo, buscando asignar a cada persona las tareas que le resulten más sencillas de realizar para cada trabajador.

5. HERRAMIENTAS DE COMUNICACIÓN DEL EQUIPO DE DESARROLLO

Como herramienta principal para la comunicación entre los miembros del equipo de desarrollo se usará **Telegram**. Así, los miembros del equipo se podrán comunicar de manera instantánea con los demás. Cada miembro deberá revisar los mensajes de dicho grupo al menos una vez al día (a las 14:00). Además, se usará la herramienta **Trello** para comunicar al equipo que tareas están todavía pendientes de su realización y qué tareas ya se han realizado.

6. RELACIONES CON EL CLIENTE

Cada viernes se realizará una reunión presencial con el cliente para la realización de una entrevista. El cliente podrá revisar el avance del proyecto y comprobar que todo se está realizando de la forma adecuada. Además, se utilizará el correo electrónico para cualquier duda ocasional que surja durante el desarrollo para contactar con el cliente.

Al final de cada iteración, la documentación se entregará de manera impresa al cliente.

7. ESTÁNDARES DE DOCUMENTACIÓN

Todos los documentos de este equipo se realizarán siguiendo las siguientes normas:

- A. Se usará la plantilla de **Google Drive Propuesta de proyecto (Menta)**.
- B. Al principio de todos los documentos, se pondrá el logotipo de **MonkeyBits**.
- C. La primera página contendrá una portada.
- D. La segunda página contendrá un índice.
- E. Se usará el tipo de letra **Arial**.
- F. El tamaño de letra de los títulos será 18.
- G. El tamaño de letra de los subtítulos será 14.
- H. El tamaño de letra del texto será el 12.

-
- I. En caso de que se quiera usar algún color, se usará el azul claro o el verde claro.
 - J. Al finalizar cada apartado, se realizarán un salto de línea.
 - K. Al finalizar la escritura de un título, se realizarán dos saltos de línea.
 - L. Se usará el formato de lista y enumeración predeterminado de la plantilla.
 - M. El nombre de empresas, herramientas de desarrollo, autores y similares se escribirá en letra negrita.
 - N. Los títulos se escribirán en letra mayúscula y negrita.
 - O. Al comienzo de cada hoja se dejará un espacio acordado.

8. ESTÁNDARES DE CÓDIGO

Todo el código desarrollado por este equipo deberá seguir las siguientes normas:

- A. Todos los nombres de las variables deberán comenzar por letra minúscula. Si tuviera más palabras, cada palabra empezará en mayúscula.
- B. Todos los nombres de funciones deberán comenzar por letra mayúscula. Si tuviera más palabras, cada palabra empezará en mayúscula.
- C. Al utilizar las llaves, se deberán poner en la misma línea que se necesita y a continuación realizar un salto de línea y una tabulación. Al cerrar las llaves, se realizará un salto de línea y después se escribirá la llave. Como ejemplo:

```
void MiEjemplo() {  
  
    miejemplo miejemplo;  
  
}
```

- D. Se usará el inglés a la hora de codificar.
- E. Nunca se devolverán comparaciones booleanas, sino que se devolverán variables booleanas que las contengan.

9. CONTROL DE VERSIONES

Para la realización de diferentes versiones usaremos **GitHub**. Se mantendrán almacenadas las diferentes versiones funcionales y se nombrarán de la siguiente manera:

RoutabilityVX siendo X el número de la versión (de 01 a 99).

10. GESTIÓN DE CALIDAD

Para que un documento de **MonkeyBits** se considere de calidad deberá cumplir todos los estándares de documentación. De la misma manera, el código deberá cumplir su estándar acordado.

Los documentos deberán estar redactados de una manera clara y explicar su contenido de manera precisa.

El prototipo deberá cumplir los requisitos funcionales y no funcionales establecidos en el pliego.

Finalmente, un miembro del equipo comprobará que estas condiciones se cumplen para que el proyecto sea de calidad.