



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería
Informática**

**UBUDiabetes 2.0
Documentación Técnica**



Presentado por Luis Miguel Inapanta Oyana
en Universidad de Burgos — 15 de junio
de 2018

Tutor: Raúl Marticorena Sánchez

Índice general

Índice general	I
Índice de figuras	III
Índice de tablas	IV
Apéndice A Plan de Proyecto Software	1
A.1. Introducción	1
A.2. Planificación temporal	1
A.3. Estudio de viabilidad	13
Apéndice B Especificación de Requisitos	17
B.1. Introducción	17
B.2. Objetivos generales	17
B.3. Catalogo de requisitos	17
B.4. Especificación de requisitos	20
Apéndice C Especificación de diseño	29
C.1. Introducción	29
C.2. Diseño de datos	29
C.3. Diseño procedimental	29
C.4. Diseño arquitectónico	29
Apéndice D Documentación técnica de programación	31
D.1. Introducción	31
D.2. Estructura de directorios	31
D.3. Manual del programador	31

D.4. Compilación, instalación y ejecución del proyecto	31
D.5. Pruebas del sistema	31
Apéndice E Documentación de usuario	33
E.1. Introducción	33
E.2. Requisitos de usuarios	33
E.3. Instalación	33
E.4. Manual del usuario	33
Bibliografía	35

Índice de figuras

A.1. Progreso en el <i>sprint</i> 0.	2
A.2. Progreso en el <i>sprint</i> 1.	3
A.3. Progreso en el <i>sprint</i> 2.	4
A.4. Progreso en el <i>sprint</i> 3.	5
A.5. Progreso en el <i>sprint</i> 4.	6
A.6. Progreso en el <i>sprint</i> 5.	7
A.7. Progreso en el <i>sprint</i> 6.	8
A.8. Progreso en el <i>sprint</i> 7.	9
A.9. Progreso en el <i>sprint</i> 8.	10
A.10. Progreso en el <i>sprint</i> 9.	11
A.11. Progreso en el <i>sprint</i> 10.	12
A.12. Progreso en el <i>sprint</i> 11.	13
B.1. Diagrama general de casos de uso	21

Índice de tablas

A.1. Costes totales	15
B.1. CU-1 Registrar datos de perfil	22
B.2. CU-2 Registrar niveles de glucemia	23
B.3. CU-3 Calcular Bolo Corrector	24
B.4. CU-4 Consultar historial de glucemias	25
B.5. CU-5 Consultar registros diarios	26
B.6. CU-6 Ajustes	27

Apéndice A

Plan de Proyecto Software

A.1. Introducción

En este apartado se pretende estimar el trabajo, el tiempo y los costes necesarios para la realización del proyecto. Cabe mencionar que se trabajaron con 3 repositorios:

- UbuDiabetes2.0_GII_14b: Repositorio inicial de partida. En él se realizaron las primeras configuraciones y ediciones para el nuevo proyecto.
- UBUDiabtes2.0_GII17.0J: Repositorio nuevo creado para continuar con el proyecto. Se puede consultar más detalles del motivo de la creación de un nuevo repositorio para el proyecto en la sección **Aspecto relevantes del desarrollo del proyecto**.
- Documentation_UBUDiabtes2.0_GII17.0J: Repositorio creado únicamente para la realización de la documentación.

A.2. Planificación temporal

A continuación, se detallarán los diferentes objetivos que se han establecido para cada *sprint* y, a su vez, el progreso obtenido en los mismos. Para ello se ha utilizado una metodología ágil denominada *SCRUM* [1]. Se empleará a su vez el gestor de tareas provisto por GitHub y se generarán gráficos *burndown* para el seguimiento de los *sprint*, los cuales son provistos por la extensión *ZenHub*.

Sprint 0 - Febrero 19-2018 - Marzo 01-2018

En este sprint se pretenden instalar las herramientas necesarias para la realización del proyecto, así como la lectura y comprensión de las memorias y anexos de los proyectos previos relacionados con el mismo. A su vez se pretende ejecutar la aplicación en un emulador para comprobar su correcto funcionamiento. En este sprint se crearon algunas tareas que, como se puede observar quedaron sin finalizar ya que, como se mencionó en la sección **Aspectos relevantes del desarrollo del proyecto**, se creó un nuevo proyecto en el que se continúa con la realización del proyecto. Podemos observar el progreso del *sprint* en la figura A.1

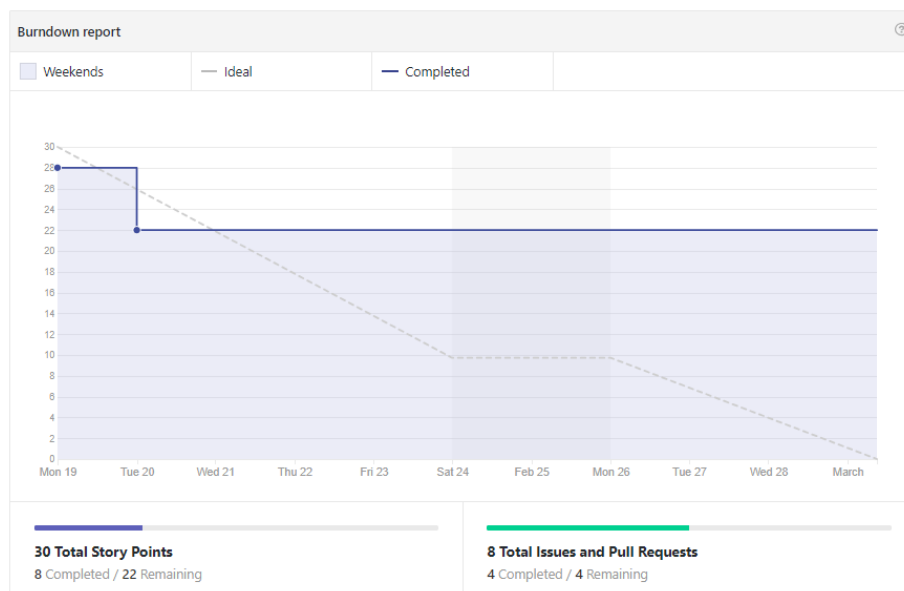
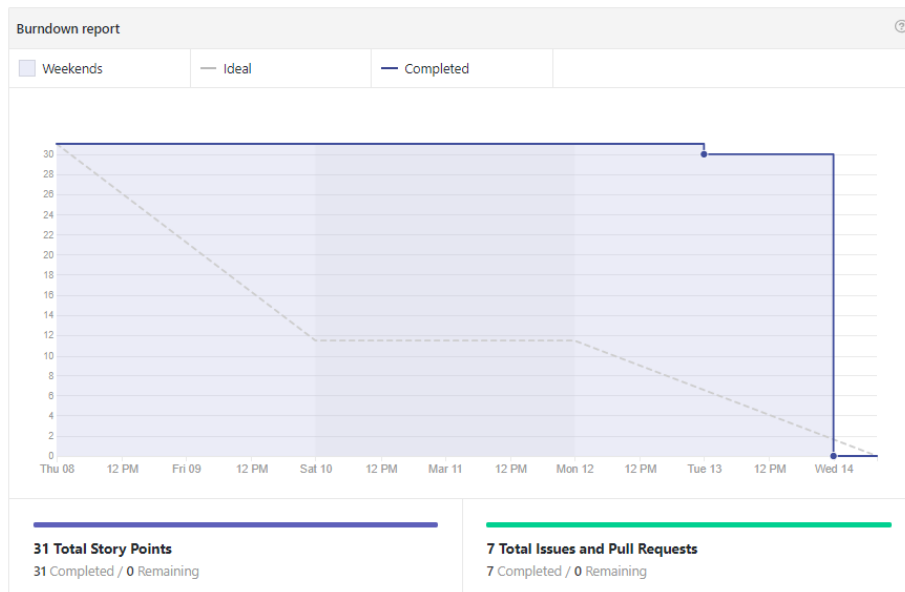


Figura A.1: Progreso en el *sprint* 0.

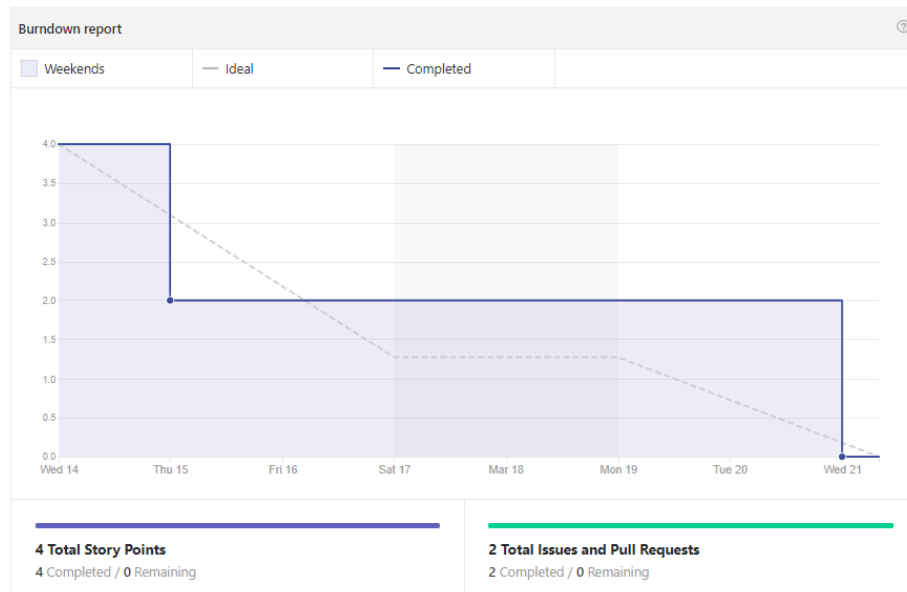
Sprint 1 - Marzo 8-2018 - Marzo 14-2018

En este *sprint* se pretende leer la documentación relacionada con el *testing* con *Espresso* y empezar con las configuraciones necesarias para la creación de los primeros test instrumentados. Podemos observar el progreso del *sprint* en la figura A.2

Figura A.2: Progreso en el *sprint* 1.

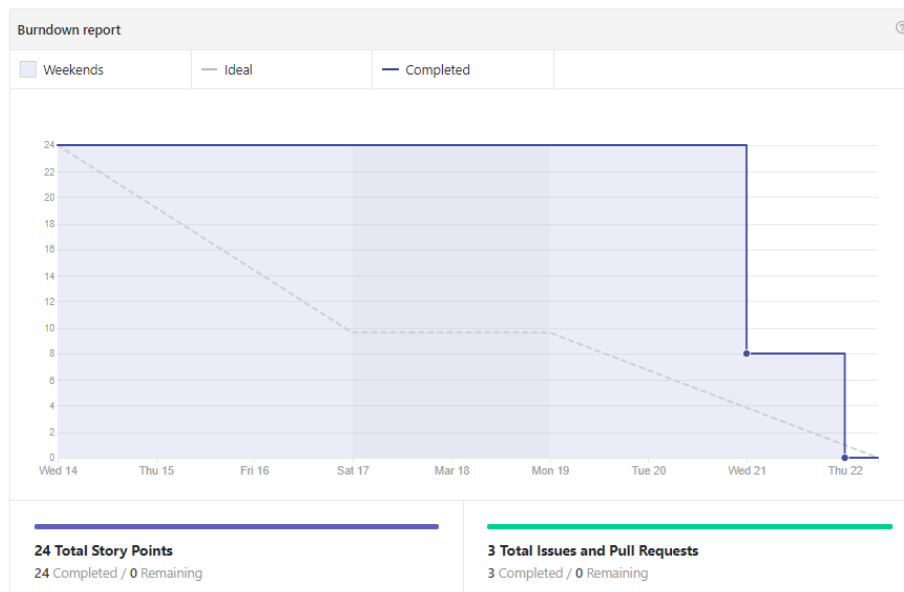
Sprint 2 - Marzo 14-2018 - Marzo 21-2018

En este *sprint* se pretende entender el funcionamiento de la herramienta *Texmaker* para poder empezar a realizar las primeras modificaciones para la documentación del proyecto. Así mismo, se pretende crear un nuevo proyecto en Android Studio, con su repositorio en *GitHub*, con las configuraciones necesarias para poder continuar con el proyecto. Podemos observar el progreso del *sprint* en la figura [A.3](#)

Figura A.3: Progreso en el *sprint* 2.

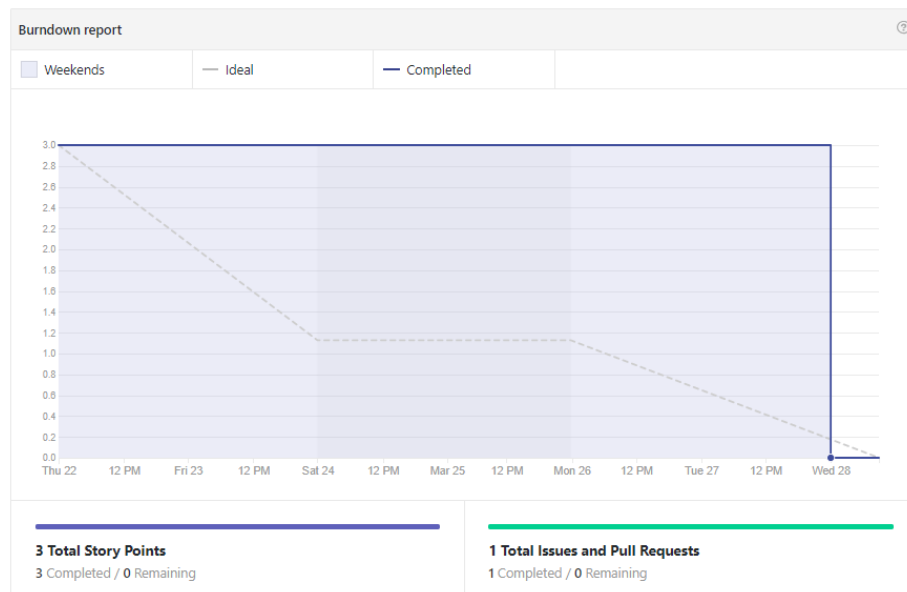
Sprint 3 - Marzo 15-2018 - Marzo 22-2018

Este *sprint* se realizó en la misma semana del *sprint* anterior, pero en el nuevo repositorio creado. En él se pretende migrar todos los datos de UBUDiabetes 1.0 al nuevo proyecto creado, realizando las configuraciones y modificaciones necesarias para la correcta compilación y ejecución del proyecto ya que este nuevo proyecto se realizó en una nueva versión de Android Studio, lo que requería configuraciones diferentes a las del proyecto anterior, por ejemplo, nuevas librerías, , diferentes dependencias de Android, contenedores diferentes en los diseños, etc. Así mismo se pretende seguir con la lectura de la documentación relacionada al *testing*. Podemos observar el progreso del *sprint* en la figura [A.4](#)

Figura A.4: Progreso en el *sprint* 3.

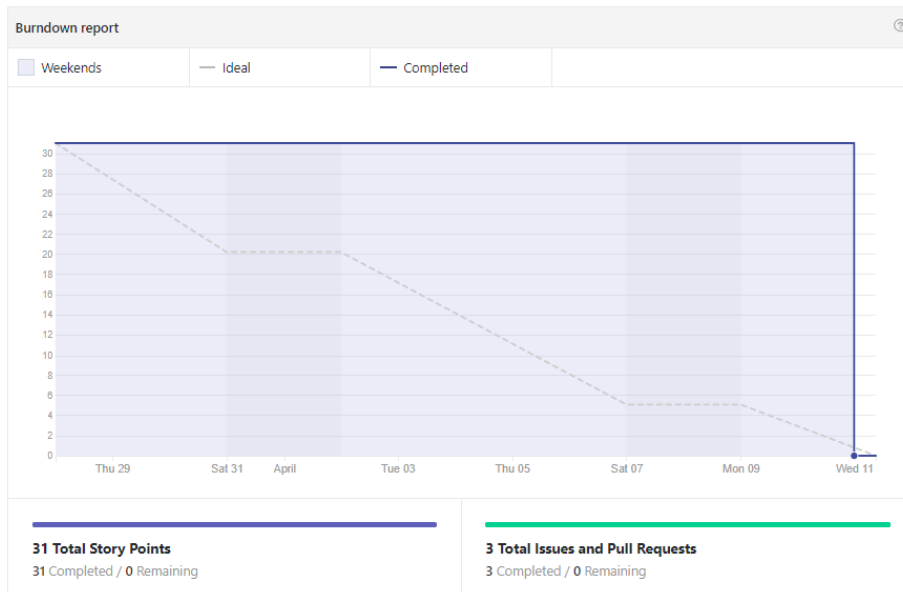
Sprint 4 - Marzo 22-2018 - Marzo 28-2018

En este *sprint* se pretende realizar y ejecutar los primeros test instrumentados con *Espresso*. Podemos observar el progreso del *sprint* en la figura [A.5](#)

Figura A.5: Progreso en el *sprint* 4.

Sprint 5 - Marzo 28-2018 - Abril 11-2018

En este *sprint* se pretende crear las primeras líneas para la documentación del proyecto, crear un nuevo diseño para la pantalla “Perfil-Registro” y crear un buscador de alimentos para la pantalla “Carbohidratos”. Podemos observar el progreso del *sprint* en la figura [A.6](#)

Figura A.6: Progreso en el *sprint* 5.

Sprint 6 - Abril 12-2018 - Abril 19-2018

En este *sprint* se realizaron varias tareas, la mayoría relacionadas con modificaciones en la interfaz gráfica del usuario:

- Empezar con la configuración y ejecución para *sonarQube*.
- Introducir en la tabla “alimentos” el índice glucémico de cada alimento.
- Mostrar en la lista de alimentos ingeridos por el usuario, la categoría según el índice glucémico de cada alimento.
- Buscar e implementar una manera de guardar en un registro los alimentos ingeridos por el usuario con su Bolo corrector final.
- Añadir opción de configuración (En el Perfil) de número de decimales a devolver en el cálculo del bolo corrector.
- Implementar la selección y eliminación de los *items* de la lista de ingesta.
- Cambiar la forma de almacenamiento de las raciones en gramos para cada alimento.

- Implementar la selección de una imagen de galería para el Perfil del usuario.

Podemos observar el progreso del *sprint* en la figura A.7

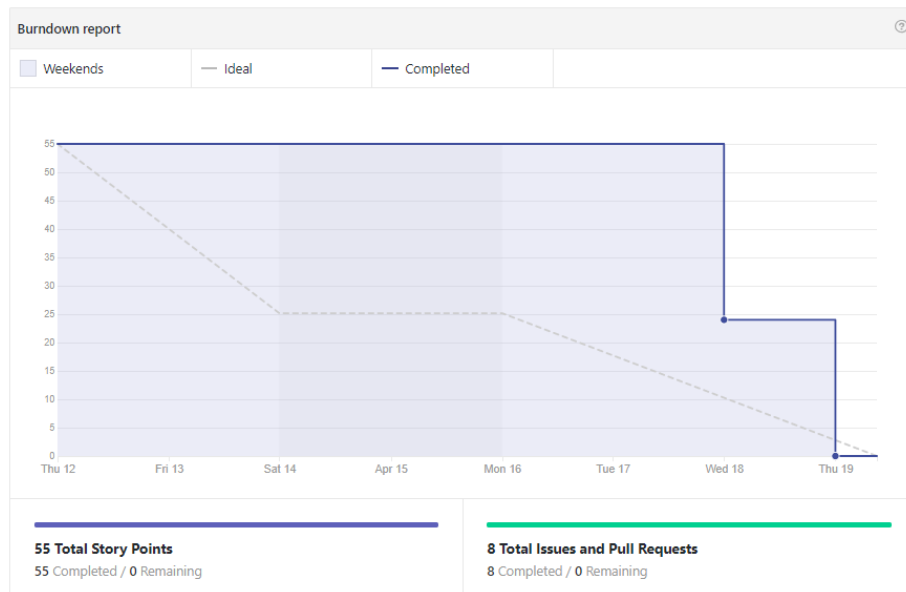


Figura A.7: Progreso en el *sprint* 6.

Sprint 7 - Abril 19-2018 - Abril 26-2018

En este *sprint* se pretende continuar con la memoria del proyecto y realizar nuevas modificaciones para la interfaz gráfica del usuario. Las tareas para este *sprint* fueron:

- Asignar permisos de uso de la cámara y permisos de escritura en la tarjeta SD para Android 6.0 y superior.
- Cambiar la carga de los datos de la tabla “Alimentos” de la pantalla “Carbohidratos” a la pantalla “SplashScreen”.
- Implementar orden inverso para las listas.
- Mostrar el sumatorio de los Hidratos de Carbono y del Bolo corrector en la pantalla “Carbohidratos”.
- Continuar con la documentación para la memoria del proyecto.

Podemos observar el progreso del *sprint* en la figura A.8

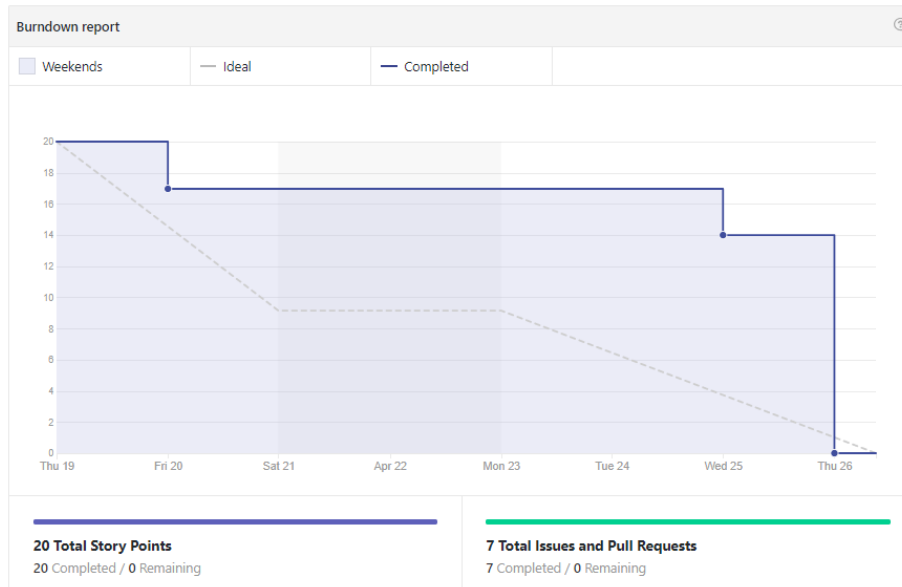
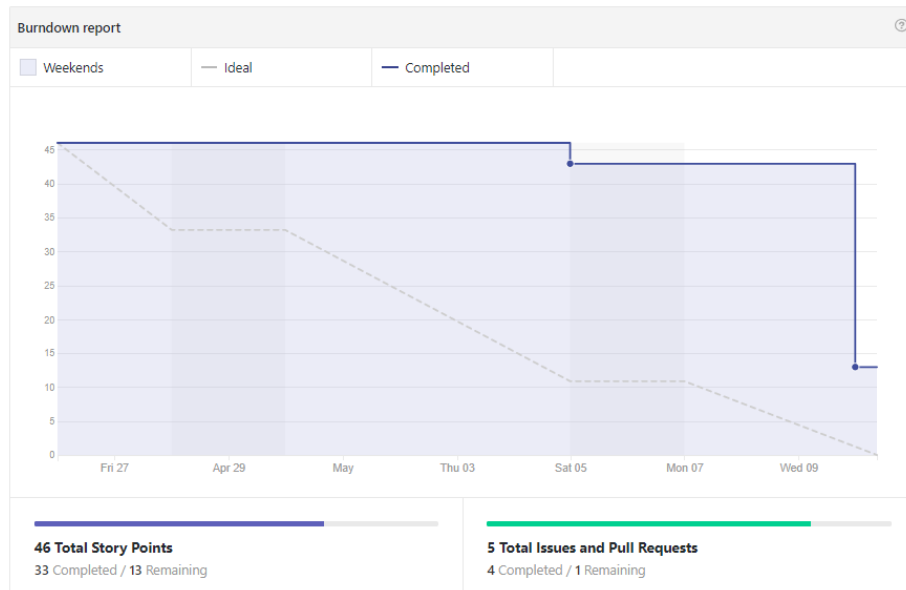


Figura A.8: Progreso en el *sprint* 7.

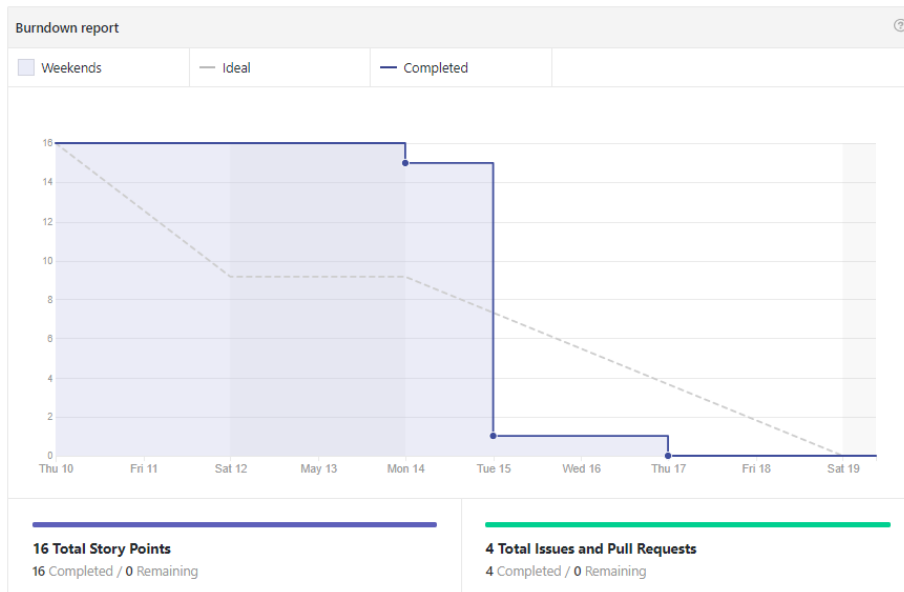
Sprint 8 - Abril 26-2018 - Mayo 10-2018

En este *sprint* se pretende continuar con la realización y/o modificación de nuevos test tanto unitarios como con *Espresso*. A su vez, se pretende seguir con la documentación, pero en un nuevo repositorio creado solo para tal. Por otro lado, se continuó también con la modificación de la interfaz gráfica del usuario. En este *sprint* se creó la primera **Apk** del proyecto en una *release*. Podemos observar el progreso del *sprint* en la figura A.9

Figura A.9: Progreso en el *sprint* 8.

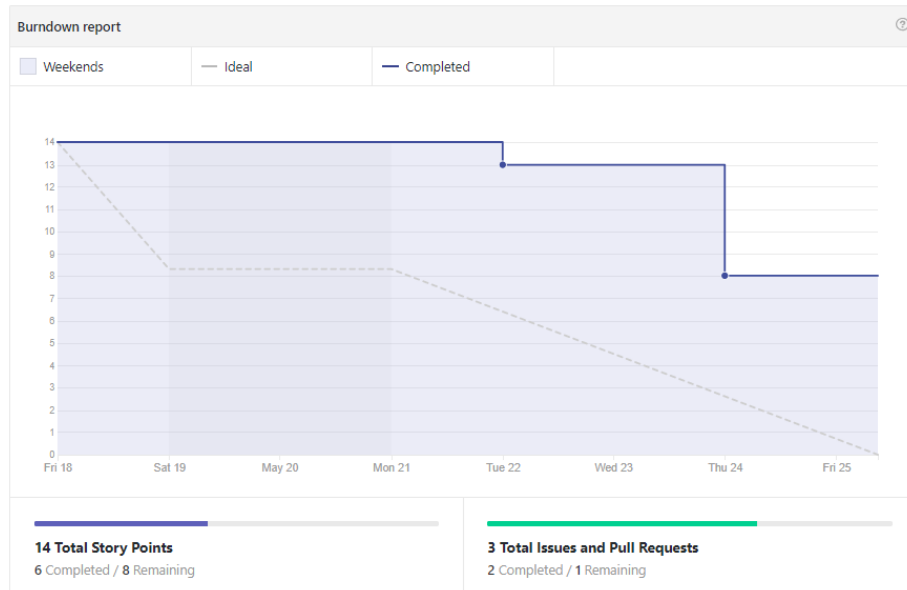
Sprint 9 - Mayo 10-2018 - Mayo 18-2018

En este *sprint* se pretende corregir pequeños detalles que se observaron que fallaban tras instalar la primera Apk creada en el *sprint* anterior en un dispositivo móvil real. Tras corregir esos detalles y añadir nuevas modificaciones, se crearon dos nuevas versiones **Apk** para la aplicación. Por otro lado, se corrigieron pequeños fallos en la memoria del proyecto. Podemos observar el progreso del *sprint* en la figura [A.10](#)

Figura A.10: Progreso en el *sprint* 9.

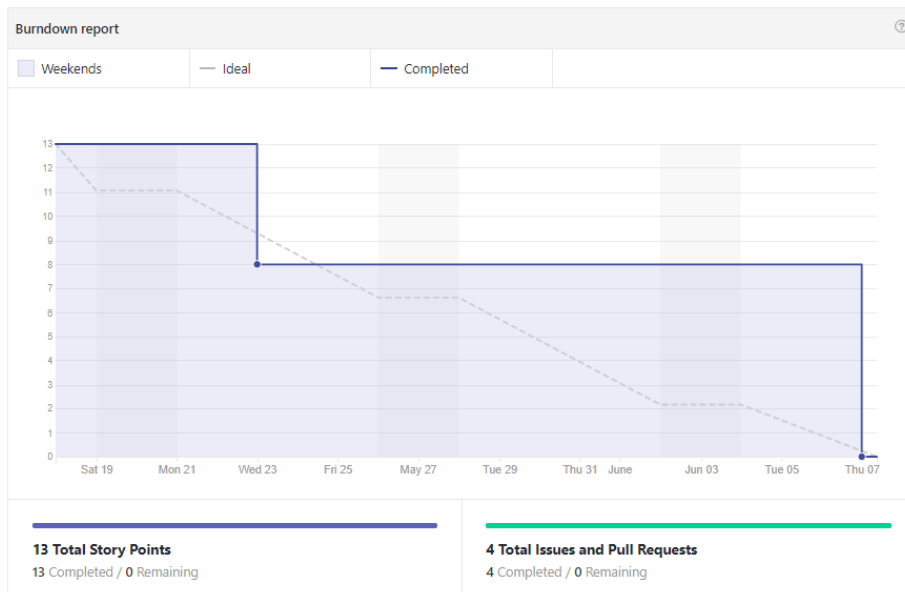
Sprint 10 - Mayo 18-2018 - Mayo 25-2018

En este sprint se pretende seguir con la modificación de la interfaz grafica del usuario. A su vez, se pretende crear una nueva estructura para los test unitarios. Se pretende separar en distintas clases los test que son precios/exactos, de los que tienen cierto margen de error. Podemos observar el progreso del *sprint* en la figura [A.11](#)

Figura A.11: Progreso en el *sprint* 10.

Sprint 10 - Mayo 18-2018 - Junio 07-2018

Este *sprint* se realizó desde el repositorio creado para la documentación. En el se pretende continuar con la documentación del proyecto. Cabe mencionar que no he realizado una dedicación a tiempo completo a este *sprint*, sino que ha sido una dedicación parcial. Esto es debido a que me encuentro cursando dos asignaturas al mismo tiempo, por lo tanto, este *sprint* ha tenido una duración superior a los primeros ya que por estas fechas los alumnos tenemos cerca los exámenes finales. Podemos observar el progreso del *sprint* en la figura [A.12](#)

Figura A.12: Progreso en el *sprint* 11.

CONTINUAR CON EL RESTO DE SPRINT

A.3. Estudio de viabilidad

En esta sección vamos a realizar un estudio de viabilidad económica de nuestro proyecto y su viabilidad legal, dos aspectos importantes y relevantes.

Viabilidad económica

A continuación, analizaremos la viabilidad económica de nuestro proyecto detallando los costes que hubiera supuesto en un caso real, es decir, en los diferentes niveles de una empresa.

Coste de personal

Los recursos humanos empleados en este proyecto han sido 2: el propio alumno quien realizó el proyecto y el tutor del mismo. Se puede estimar que el alumno ha empleado xxx horas (aun por estimar) a 10€ la hora:

$$10 \text{ €/Hora} * xxx\text{Horas} = xxx \text{ €}$$

Para calcular el coste de las reuniones con el tutor se tendrá en cuenta el número de reuniones a lo largo del desarrollo del proyecto. En total fueron xxx reuniones de 1 hora cada una aproximadamente, lo que supondría:

$$10 \text{ €/Hora} * xxx \text{ Horas} = xxx \text{ €}$$

A esta cuantía debemos añadirle el valor que debería llevarse la Seguridad Social, lo cual ha sido obtenido a partir de¹:

- 1,55 %: Desempleo.
- 0,03 %: Formación profesional.
- 4,7 %: Contingencias comunes
- Lo que hace un total de: 6,28

Por lo que el total de costes de personal sería de:

$$(xxx \text{ Alumno} + xxx \text{ reuniones}) \text{ €} * 0,0628 = xxx \text{ €}$$

Coste de hardware

Cabe mencionar que no se necesitó de ningún equipo informático adicional para la realización del proyecto. No obstante, se deberá calcular el coste del *hardware* utilizado en el proyecto. En este caso al tratarse de una aplicación móvil, únicamente se necesitó un ordenador portátil cuyo valor se estima en 600€ y suponiendo una vida útil de 7 años y xxx semanas de proyecto, se tiene un coste de amortización de xxx€ aproximadamente.

$$\frac{600\text{€}}{7 \text{ años}} * \frac{1 \text{ año}}{52 \text{ semanas}} * xxx \text{ semanas} \approx xxx\text{€}$$

Coste de software

Este proyecto es una segunda versión y se ha desarrollado empleando plataformas *software* libre para todo lo imprescindible, por lo que el coste *software* es nulo.

Coste total

Así, el coste total del proyecto ha sido:

¹http://www.seg-social.es/Internet_1/Trabajadores/CotizacionRecaudaci10777/Basesytiposdecotiza36537/index.htm

Tabla A.1: Costes totales

Costes	Importe
Personal	xxx €
Seguridad Social	xxx €
Material	xxx €
Totales	xxx €

Viabilidad legal

En este apartado pasamos a analizar la viabilidad legal del proyecto. Para ello vamos a describir las licencias de elementos o librerías que se hayan podido utilizar.

El primer caso sujeto a estudio es el de la librería **Hitomis/CircleMenu**. Esta librería se distribuye bajo la licencia Apache License v2.0.

El segundo caso sujeto a estudio es el de la librería **(nueva librería para las gráficas)**. Esta librería se distribuye bajo la licencia xxx.

Apache License V2.0

La licencia *Apache* (Apache License o Apache Software License para versiones anteriores a 2.0) es una licencia de software libre permisiva creada por la *Apache Software Foundation (ASF)*. La licencia *Apache* (con versiones 1.0, 1.1 y 2.0) requiere la conservación del aviso de derecho de autor y el descargo de responsabilidad, pero no es una licencia copyleft, ya que no requiere la redistribución del código fuente cuando se distribuyen versiones modificadas. Al igual que otras licencias de software libre, todo el software producido por la *ASF* o cualquiera de sus proyectos está desarrollado bajo los términos de esta licencia, es decir, la licencia permite al usuario del software de la libertad de usar el software para cualquier propósito, para distribuirlo, modificarlo y distribuir versiones modificadas del software, bajo los términos de la licencia, sin preocuparse de las regalías [2].

Apéndice B

Especificación de Requisitos

B.1. Introducción

En esta sección se expondrán los distintos objetivos que la aplicación debe lograr, así como los requisitos que debe cumplir.

B.2. Objetivos generales

Como objetivos generales en este proyecto tenemos:

- Crear y/o mejorar la interfaz gráfica del usuario.
- Mostrar todos los registros diarios del usuario.
- Facilitar la navegabilidad durante el uso de la aplicación.
- Corregir los métodos utilizados para el cálculo del bolo corrector.
- Incluir pruebas unitarias e instrumentales.

B.3. Catalogo de requisitos

A continuación, listaremos el conjunto de requisitos funcionales extraídos a partir de los objetivos generales del proyecto.

Requisitos funcionales

Los requisitos funcionales se han obtenido a partir de un diagrama de casos de uso hecho desde cero, aunque ciertos requisitos coinciden con los de la versión anterior, por lo que serán iguales.

- **RF-1 Gestión del perfil de usuario:** la aplicación debe ser capaz de registrar y modificar los datos personales y médicos del usuario.
 - **RF-1.1 Comprobar datos:** El sistema deberá comprobar que se han rellenado todos los campos de registro.
 - **RF-1.2 Validar valores de Glucemia:** El sistema debe comprobar que se han introducido los valores de glucemia máximo y mínimo dentro de unos márgenes establecidos.
 - **RF-1.3 Registrar datos del perfil:** El sistema deberá registrar los datos del perfil del usuario.
 - **RF-1.4 Modificar datos del perfil:** El sistema deberá permitir la modificación de los datos del perfil del usuario.
- **RF-2 Cálculo del bolo corrector:** El usuario podrá calcular las unidades de insulina que necesita a partir de su nivel de glucemia.
 - **RF-2.1:** El sistema deberá registrar el nivel de glucemia haciendo uso de *Gestión de glucemias*.
 - **RF-2.2 Cantidad de alimento:** El sistema deberá permitir al usuario introducir y/o eliminar los gramos de uno o más alimentos a consumir. Así mismo, el sistema deberá registrar dichos alimentos con sus respectivas cantidades.
 - **RF-2.3 Bolo corrector:** El sistema deberá mostrar por pantalla el resultado del bolo corrector obtenido a partir de los datos introducidos.
- **RF-3 Gestión de glucemias:** El usuario podrá llevar un registro de sus niveles de glucemia en diferentes etapas del día, así como registrar incidencias en los casos en que corresponda.
 - **RF-3.1 Registrar niveles de glucemia:** El sistema deberá registrar en una base de datos local los valores de glucemia introducidos por el usuario.
 - **RF-3.2 Comprobar límites:** El sistema deberá comprobar si el valor introducido se encuentra dentro de los límites mínimo-máximo introducidos por el usuario en su perfil.

- **RF-3.2.1 Registrar incidencia:** Si el valor introducido esta fuera de los límites establecidos, el sistema permitirá al usuario registrar una incidencia que podrá tener opcionalmente una observación por parte del usuario.
- **RF-4 Gestión del historial:** El usuario podrá comprobar de manera gráfica sus niveles de glucemia registrados.
- **RF-5 Gestión de registros diarios:** El usuario podrá comprobar todos sus registros de manera simplificada o detallada.
 - **RF-5.1 Resumen registros:** El sistema deberá mostrar por pantalla los registros diarios del usuario:
 - Id: Identificador del registro.
 - Fecha: Momento del registro.
 - HC: Sumatorio de hidratos de carbono en el cálculo del bolo corrector.
 - Bolo corrector: Unidades de insulina obtenidas en el cálculo del bolo corrector.
 - **RF-5.2 Detalle registros:** El sistema deberá mostrar por pantalla los detalles de un registro previamente seleccionado por el usuario:
 - Tipo: Categoría del alimento.
 - Alimento: Nombre del alimento consumido.
 - Cantidad: Gramos consumidos del alimento.
 - IG: Índice glucemico.
- **RF-6 Gestión de ajustes:** El usuario podrá realizar pequeños ajustes en la aplicación.
 - RF-6.1: El sistema deberá permitir configurar los datos del perfil del usuario haciendo uso de *Gestión del perfil de usuario*.
 - **RF-6.1 Copia de seguridad:** El sistema deberá permitir realizar copias de seguridad de los datos registrados por el usuario.
 - **RF-6.1 Liberar espacio:** El sistema deberá permitir eliminar los datos registrados por el usuario.

Requisitos no funcionales

Alguno de estos requisitos coinciden con los de la versión anterior. Aun así, mencionaremos todos para que quede más claro.

- **RNF-1 Apariencia:** La representación de los datos deberá ser atractiva para el usuario.
- **RNF-2 Usabilidad:** El conjunto de elementos visuales de la interfaz deben ser claros e intuitivos, permitiendo un rápido aprendizaje.
- **RNF-3 Mantenibilidad, Portabilidad y Escalabilidad:** La aplicación debe desarrollarse siguiendo alguna técnica que permita facilidad de mantenimiento e incorporación de nuevas características, así como corrección de errores.
- **RNF-4 Eficiencia:** La aplicación debe ser capaz operar adecuadamente con los datos introducidos por el usuario.
- **RNF-5 Seguridad:** La aplicación debe ser capaz de realizar copias de seguridad de los datos registrados por el usuario.
- **RNF-6 Rendimiento:** La aplicación debe ser capaz de soportar el manejo de gran cantidad de datos durante su proceso.
- **RNF-7 Testabilidad:** Capacidad de que la aplicación puede ser probada.

B.4. Especificación de requisitos

Actores

La aplicación **UBU**Diabetes está orientada a usuarios diabéticos del tipo I. En este caso los clientes de la aplicación son miembros del Área de enfermería (Grado de Enfermería) de la facultad de Ciencias de la Salud, como Diego Serrano Gómez y Jesús Puente Alcaraz, y Raúl Marticorena Sánchez, tutor del proyecto. Por lo tanto:

- **Desarrollador:** Luis Miguel Inapanta Oyana.
- **Clientes:** Diego Serrano Gómez, Jesús Puente Alcaraz, y Raúl Marticorena Sánchez.

Diagrama de casos de uso

A continuación se mostrara el diagrama de casos de uso del proyecto.

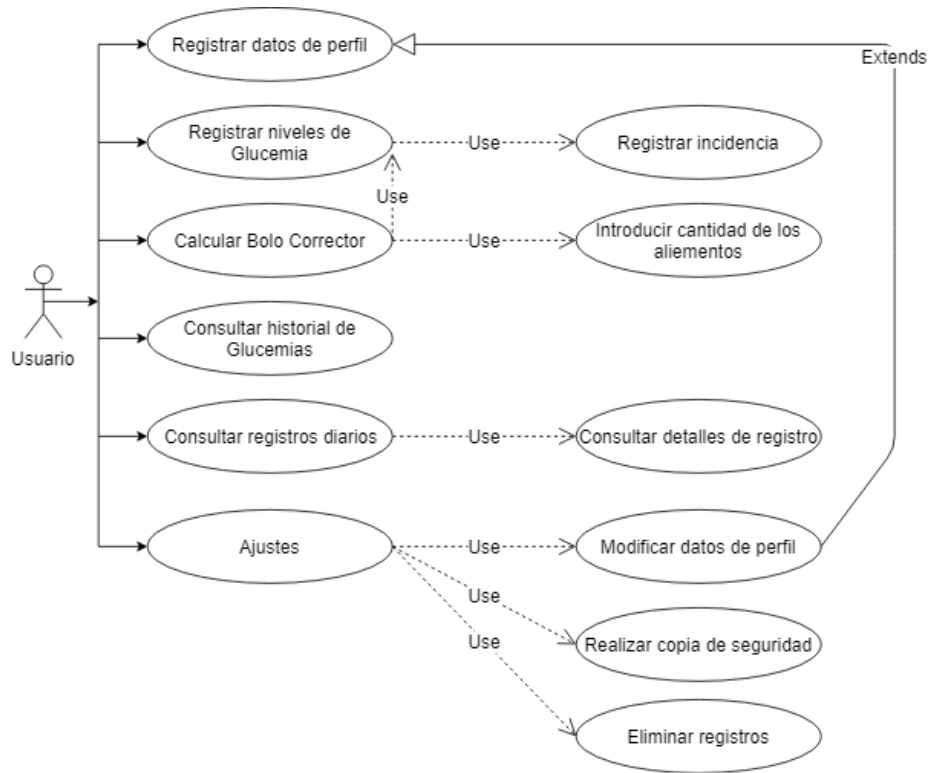


Figura B.1: Diagrama general de casos de uso

CU-1	Registrar datos de perfil
Versión	2.0
Autor	Luis Miguel Inapanta Oyana
Requisitos asociados	RF-1.1, RF-1.2, RF-1.3, RF-1.4
Descripción	Permite al usuario registrar sus datos, tanto personales como médicos.
Precondiciones	<ul style="list-style-type: none"> • Consultar los datos relacionados con la diabetes con su médico.
Acciones	<ol style="list-style-type: none"> 1. El usuario rellena todos los campos del registro. 2. Se cargan los datos de la tabla de alimentos de la federación española de la diabetes. 3. Se visualiza el menú principal.
Postcondiciones	Visualizar menú principal.
Excepciones	No rellenar algún campo del registro
Importancia	Muy Alta

Tabla B.1: CU-1 Registrar datos de perfil

CU-2	Registrar niveles de glucemia
Versión	2.0
Autor	Luis Miguel Inapanta Oyana
Requisitos asociados	RF-3.1, RF-3.2, RF-3.2.1
Descripción	Permite al usuario registrar sus niveles de glucemia en diferentes etapas del día, así como registrar incidencias en los casos en que corresponda.
Precondiciones	<ul style="list-style-type: none"> • Estar registrado. • Seleccionar la opción Registrar glucemia del menú principal - icono de lápiz.
Acciones	<ol style="list-style-type: none"> 1. El usuario introduce su nivel de glucemia. 2. El usuario introduce una incidencia si corresponde.
Postcondiciones	Se registran ambos datos en la Base de Datos.
Excepciones	No introducir nivel de glucemia o incidencia
Importancia	Alta

Tabla B.2: CU-2 Registrar niveles de glucemia

CU-3	Calcular Bolo Corrector
Versión	2.0
Autor	Luis Miguel Inapanta Oyana
Requisitos asociados	RF-2.1, RF-2.2, RF-2.3
Descripción	Permite al usuario calcular las unidades de insulina que necesita a partir de su nivel de glucemia.
Precondiciones	<ul style="list-style-type: none"> • Estar registrado. • Seleccionar la opción Calcular Bolo del menú principal - icono de calculadora.
Acciones	<ol style="list-style-type: none"> 1. El usuario introduce su nivel de glucemia. 2. El usuario introduce una incidencia si corresponde. 3. El usuario introduce introduce o elimina uno o varios alimentos a consumir con sus respectivas cantidades (gramos).
Postcondiciones	<ul style="list-style-type: none"> • Se muestra la usuario el bolo corrector calculado por el sistema. • Se registran los alimentos con sus respectivas cantidades, el sumatorio de los hidratos de carbono consumidos y el bolo corrector calculado en la Base de Datos.
Excepciones	<ol style="list-style-type: none"> 1. No introducir nivel de glucemia o incidencia. 2. No introducir cantidad del alimento.
Importancia	Muy Alta

Tabla B.3: CU-3 Calcular Bolo Corrector

CU-4	Consultar historial de glucemias
Versión	2.0
Autor	Luis Miguel Inapanta Oyana
Requisitos asociados	RF-4
Descripción	Permite al usuario consultar sus niveles de glucemia de manera gráfica.
Precondiciones	<ul style="list-style-type: none"> • Estar registrado. • Seleccionar la opción Historial de glucemias del menú principal - icono de gráfica.
Acciones	1. pendiente aún.
Postcondiciones	• —Pendiente aún.
Excepciones	1. pendiente aún.
Importancia	Media

Tabla B.4: CU-4 Consultar historial de glucemias

CU-5	Consultar registros diarios
Versión	2.0
Autor	Luis Miguel Inapanta Oyana
Requisitos asociados	RF-5.1, RF-5.2
Descripción	Permite al usuario consultar sus registros de manera simplificada o detallada
Precondiciones	<ul style="list-style-type: none"> • Estar registrado. • Seleccionar la opción Consultar registros del menú principal - icono de lupa.
Acciones	<ol style="list-style-type: none"> 1. Seleccionar un <i>item</i> de la lista de registros. 2. Pulsar el botón consultar.
Postcondiciones	• Se muestran los detalles del registro
Excepciones	1. No seleccionar ningún <i>item</i> de la lista.
Importancia	Alta

Tabla B.5: CU-5 Consultar registros diarios

CU-6	Ajustes
Versión	2.0
Autor	Luis Miguel Inapanta Oyana
Requisitos asociados	RF-6.1, RF-6.2, RF-6.3
Descripción	Permite al usuario realizar pequeños ajustes en la aplicación.
Precondiciones	<ul style="list-style-type: none"> • Estar registrado. • Seleccionar la opción Ajustes del sub-menú del menú principal.
Acciones	1. Seleccionar un <i>ítem</i> de la lista de ajustes.
Postcondiciones	<ul style="list-style-type: none"> • Permite al usuario modificar sus datos de perfil. • Permite al usuario realizar una copia de seguridad de sus registros. • Permite al usuario eliminar sus registros.
Excepciones	<ol style="list-style-type: none"> 1. No seleccionar ningún <i>ítem</i> de la lista. 2. Intentar eliminar registros.
Importancia	Media

Tabla B.6: CU-6 Ajustes

Apéndice C

Especificación de diseño

- C.1. Introducción
- C.2. Diseño de datos
- C.3. Diseño procedimental
- C.4. Diseño arquitectónico

Apéndice D

Documentación técnica de programación

- D.1. Introducción
- D.2. Estructura de directorios
- D.3. Manual del programador
- D.4. Compilación, instalación y ejecución del proyecto
- D.5. Pruebas del sistema

Apéndice E

Documentación de usuario

- E.1. Introducción
- E.2. Requisitos de usuarios
- E.3. Instalación
- E.4. Manual del usuario

Bibliografía

- [1] Ken Schwaber and Mike Beedle. *Agile software development with Scrum*, volume 1. Prentice Hall Upper Saddle River, 2002.
- [2] Wikipedia. Apache license — wikipedia, la enciclopedia libre, 2017. [Internet; descargado 13-junio-2018].