



UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería  
Informática**

**título del TFG  
Documentación Técnica**



Presentado por Alicia Olivares Gil  
en Universidad de Burgos — 17 de junio  
de 2019

Tutores: Álgvar Arnáiz González y José  
Franciso Díez Pastor



---

# Índice general

---

<b>Índice general</b>	<b>I</b>
<b>Índice de figuras</b>	<b>III</b>
<b>Índice de tablas</b>	<b>IV</b>
<b>Apéndice A Plan de Proyecto Software</b>	<b>1</b>
A.1. Introducción . . . . .	1
A.2. Planificación temporal . . . . .	1
A.3. Estudio de viabilidad . . . . .	9
<b>Apéndice B Especificación de Requisitos</b>	<b>11</b>
B.1. Introducción . . . . .	11
B.2. Objetivos generales . . . . .	11
B.3. Catalogo de requisitos . . . . .	11
B.4. Especificación de requisitos . . . . .	13
<b>Apéndice C Especificación de diseño</b>	<b>15</b>
C.1. Introducción . . . . .	15
C.2. Diseño de datos . . . . .	15
C.3. Diseño procedimental . . . . .	15
C.4. Diseño arquitectónico . . . . .	15
C.5. Diseño de interfaces . . . . .	15
<b>Apéndice D Documentación técnica de programación</b>	<b>17</b>
D.1. Introducción . . . . .	17
D.2. Estructura de directorios . . . . .	17

D.3. Manual del programador . . . . .	17
D.4. Compilación, instalación y ejecución del proyecto . . . . .	17
D.5. Pruebas del sistema . . . . .	17
<b>Apéndice E Documentación de usuario</b>	<b>19</b>
E.1. Introducción . . . . .	19
E.2. Requisitos de usuarios . . . . .	19
E.3. Instalación . . . . .	19
E.4. Manual del usuario . . . . .	19
<b>Bibliografía</b>	<b>21</b>

---

# Índice de figuras

---

C.1. Prototipos iniciales de las pantallas de: login, administración, visualización de camas y visualización de datos. . . . .	16
---	----

---

## Índice de tablas

---

A.1. Sprint 1 . . . . .	2
A.2. Sprint 2 . . . . .	2
A.3. Sprint 3 . . . . .	3
A.4. Sprint 4 . . . . .	3
A.5. Sprint 5 . . . . .	3
A.6. Sprint 6 . . . . .	4
A.7. Sprint 7 . . . . .	4
A.8. Sprint 8 . . . . .	5
A.9. Sprint 9 . . . . .	5
A.10.Sprint 10 . . . . .	6
A.11.Sprint 11 . . . . .	6
A.12.Sprint 12 . . . . .	6
A.13.Sprint 13 . . . . .	7
A.14.Sprint 14 . . . . .	7
A.15.Sprint 15 . . . . .	8
A.16.Sprint 16 . . . . .	8
A.17.Sprint 17 . . . . .	9
A.18.Sprint 18 . . . . .	9

## Apéndice A

---

# Plan de Proyecto Software

---

### A.1. Introducción

### A.2. Planificación temporal

La planificación temporal se ha realizado adaptando la metodología *Scrum* a un proyecto educativo, con los cambios que esto conlleva.

- El desarrollo se ha basado en iteraciones o *sprints* de una semana de duración aproximadamente.
- Cada uno de los *sprints* contiene las tareas o *issues* que se realizaron esa semana.
- Cada tarea tiene asociado un coste, que simboliza su dificultad en cuanto al esfuerzo que se estima invertir en ella.
- En caso de que la estimación del coste resultara inexacta al realizar el *issue*, este se modificó para reflejar el esfuerzo real empleado.
- Al finalizar cada *sprint* se realizaba una reunión de revisión con los tutores donde se exponían los progresos realizados y se planificaba el siguiente *sprint*.

### Sprint 1

Fecha: 19/12/2018 - 23/12/2018

El primer *sprint* consistió en realizar una exploración bibliográfica inicial sobre el estado del arte.

<i>Issue</i>	Estimado	Final
Crear y configurar repositorio	2	2
Exploración bibliográfica inicial	13	13

Tabla A.1: Sprint 1

## Sprint 2

Fecha: 23/12/2018 - 29/12/2018

Se continuó la exploración bibliográfica inicial, centrándose en artículos especialmente interesantes encontrados hasta el momento y se comenzó la exploración bibliográfica sobre otros métodos aplicables al problema.

<i>Issue</i>	Estimado	Final
Continuación de la exploración bibliográfica inicial	8	8
Exploración bibliográfica sobre otros métodos aplicables al problema	8	8
Lectura de "Automated Epileptic Seizure Detection Methods: A Review Study"	8	8
Instalar y configurar cliente VPN	2	2

Tabla A.2: Sprint 2

## Sprint 3

Fecha: 29/12/2018 - 11/01/2019

Se inició la documentación y se empezó a trabajar en la visualización de los datos en bruto y de algunos datos estadísticos.



<b><i>Issue</i></b>	<b>Estimado</b>	<b>Final</b>
Iniciar documentación	-	-
Instalar entorno y librerías de Python	5	5
Aprender a usar librerías	8	8
Procesar y mostrar datos	8	8

Tabla A.3: Sprint 3

## Sprint 4

Fecha: 11/01/2019 - 18/01/2019

Se configuró el acceso al computador del departamento para probar técnicas de reducción de la dimensionalidad de los datos y algunas opciones básicas de filtrado y suavizado de la señal.

<b><i>Issue</i></b>	<b>Estimado</b>	<b>Final</b>
Configurar acceso a gamma	5	5
Probar opciones filtrado y suavizado	8	8
Probar otras formas de proyección de datos	8	21

Tabla A.4: Sprint 4

## Sprint 5

Fecha: 18/01/2019 - 25/01/2019

Se hicieron cambios en el preprocesado, se probaron otras formas de filtrado de la señal y se estudiaron los puntos clave de las proyecciones del *sprint* anterior.

<b><i>Issue</i></b>	<b>Estimado</b>	<b>Final</b>
Leer apuntes de minería de datos	8	8
Modificar preprocesado	3	3
Representar señales en torno a la crisis epiléptica	5	5
Probar formas de filtrado de la señal	5	5
Estudiar los puntos clave de las proyecciones	5	8

Tabla A.5: Sprint 5

## Sprint 6

Fecha: 25/01/2019 - 31/02/2019

Se centraron las pruebas en las proyecciones con mejor rendimiento, concretamente en MDS [5], y se iniciaron la documentación de la planificación temporal y el cuaderno de investigación.

<i>Issue</i>	<b>Estimado</b>	<b>Final</b>
Cambiar a proyecciones con mejor rendimiento	13	13
Pasar cálculos estadísticos a funciones	5	5
Documentar 5 primeros Sprints en el Plan de Proyecto	8	8
Documentar investigación en overleaf [7]	5	5

Tabla A.6: Sprint 6

## Sprint 7

Fecha: 31/02/2019 - 07/02/2019

Se codificaron las transformaciones generadas en los *sprints* anteriores (normalización, filtros y estadísticas) como transformadores de sklearn [8].

<i>Issue</i>	<b>Estimado</b>	<b>Final</b>
Aprender sobre la clase sklearn.base.TransformerMixin [3]	3	3
Generar transformadores para las funciones usadas	21	13

Tabla A.7: Sprint 7

## Sprint 8

Fecha: 07/02/2019 - 14/02/2019

Se exploraron otras formas de proyección y se realizó una primera aproximación de clasificación mediante Random Forest [4] y detección de anomalías One-class [6].

<i>Issue</i>	<b>Estimado</b>	<b>Final</b>
Probar Kernel PCA	5	5
Acotar ataque a partir del aspecto de la señal y la salida de las proyecciones (MDS)	3	5
Probar MDS con el ataque reetiquetado	8	8
Probar clasificador Random Forest	8	8
Aplicar detección de anomalías one-class	13	13

Tabla A.8: Sprint 8

## Sprint 9

Fecha: 14/02/2019 - 21/02/2019

Se planteó la evaluación de los clasificadores mediante el área bajo la curva ROC y se terminaron de documentar las proyecciones en el cuaderno de investigación.

<i>Issue</i>	<b>Estimado</b>	<b>Final</b>
Valorar los resultados de Random Forest mediante el área bajo la curva	3	3
Incluir las proyecciones en la documentación	5	5
Preparar la visualización de las proyecciones para la documentación	5	8

Tabla A.9: Sprint 9

## Sprint 10

Fecha: 21/02/2019 - 28/02/2019

Una parte se invirtió en aprender sobre clasificación de conjuntos de datos desequilibrados mediante ensembles y por otro lado se realizó una exploración de ventanas para la aplicación de los datos al clasificador Random Forest.

<i><b>Issue</b></i>	<b>Estimado</b>	<b>Final</b>
Lectura de aprendizaje sobre datos desequilibrados	-	-
Aplicar Random Forest a datos estadísticos con distintas ventanas	8	8

Tabla A.10: Sprint 10

## Sprint 11

Fecha: 28/02/2019 - 07/03/2019

Se continuó con la lectura sobre desequilibrados y se inició el aprendizaje sobre la librería tsfresh [1] para extracción de características en series temporales.

<i><b>Issue</b></i>	<b>Estimado</b>	<b>Final</b>
Continuar con la lectura sobre uso de ensembles para conjuntos desequilibrados	5	5
Extracción de características en series temporales	8	8

Tabla A.11: Sprint 11

## Sprint 12

Fecha: 07/03/2019 - 14/03/2019

Se trataron de aplicar los resultados de la extracción de características de series temporales al clasificador Random Forest.

<i><b>Issue</b></i>	<b>Estimado</b>	<b>Final</b>
Random Forest con características de series temporales	5	5
Continuar extracción de características en series temporales	13	13

Tabla A.12: Sprint 12

## Sprint 13

Fecha: 14/03/2019 - 21/03/2019

Principalmente se exploraron formas de filtrar y combinar las mejores características de series temporales para ser aplicadas al clasificador.

<i>Issue</i>	<b>Estimado</b>	<b>Final</b>
Filtrado de características	13	5
Aplicar Random Forest a combinaciones de las mejores características	5	5
Documentación de sprints pasados y actualización del cuaderno de investigación	5	5

Tabla A.13: Sprint 13

## Sprint 14

Fecha: 21/03/2019 - 28/03/2019

Se planteó un filtrado de características mediante un algoritmo genético usando el framework deap [2] de python y se realizó una investigación inicial de técnicas para la implementación de servidores de *streaming*.

<i>Issue</i>	<b>Estimado</b>	<b>Final</b>
Investigar técnicas para implementar servidores de streaming	8	8
Algoritmo genético para la selección de características	13	13
Avanzar con la documentación en el cuaderno de investigación	5	13

Tabla A.14: Sprint 14

## Sprint 15

Fecha: 28/03/2019 - 04/04/2019

Se mejoró el algoritmo genético, se finalizó su ejecución con la ayuda de tmux [9] y se documentaron los resultados. Además, se inició el diseño de

los requisitos y los casos de uso de la aplicación y se plantearon los primeros prototipos.

<i><b>Issue</b></i>	<b>Estimado</b>	<b>Final</b>
Generar prototipo de la pantalla de visualización de datos	5	5
Aprender sobre tmux para la ejecución del genético	3	3
Mejorar algoritmo genético y documentar resultados	8	8
Plantear primeras cuestiones de diseño de la app	5	5

Tabla A.15: Sprint 15

## Sprint 16

Fecha: 04/04/2019 - 11/04/2019

Se ultimaron los detalles del cuaderno de investigación con la documentación generada hasta el momento, se finalizaron los prototipos y se documentó la parte de diseño y de las técnicas. Además, se instaló Android Studio para su uso en sprints posteriores.

<i><b>Issue</b></i>	<b>Estimado</b>	<b>Final</b>
Ultimar detalles del cuaderno e trabajo	-	-
Finalizar y documentar prototipos	5	5
Avanzar en la documentación temporal, de diseño y de las técnicas	13	13
Instalar Android Studio	2	2

Tabla A.16: Sprint 16

## Sprint 17

Fecha: 11/04/2019 - 18/04/2019

Se refactorizó el código de los experimentos para incluir el testeo mediante la métrica precision-recall, más adecuada para conjuntos de datos desequilibrados, y se volvieron a ejecutar los filtrados de características.

<i><b>Issue</b></i>	<b>Estimado</b>	<b>Final</b>
Refactorizar el código de Random Forest para incluir la métrica precision-recall	8	8
Volver a ejecutar los filtrados de características para la nueva métrica	8	8

Tabla A.17: Sprint 17

## Sprint 18

Fecha: 18/04/2019 - 02/05/2019

Se documentaron los resultados de los filtrados de características del sprint anterior y se comenzó la lectura sobre la documentación de Android Studio y la visualización del curso *Android Development for Beginners* de Google.

<i><b>Issue</b></i>	<b>Estimado</b>	<b>Final</b>
Documentar los resultados de las nuevas ejecuciones en el cuaderno de investigación	8	8
Aprender a usar Android Studio	13	21

Tabla A.18: Sprint 18

## A.3. Estudio de viabilidad

Viabilidad económica

Viabilidad legal





## Apéndice *B*

---

# Especificación de Requisitos

---

### B.1. Introducción

### B.2. Objetivos generales

### B.3. Catalogo de requisitos

Se indican los requisitos funcionales y no funcionales de la aplicación.

#### Requisitos funcionales

- **RF-1 Confidencialidad del sistema:** Solamente los usuarios autorizados podrán acceder al sistema.
  - **RF-1.1 Identificación de usuario:** los usuarios se identificarán con un *nickname* y una contraseña
  - **RF-1.2 Rol de administración:** existirá un usuario especial que podrá administrar el sistema completamente sin restricciones.
  - **RF-1.3 Visualización de una cama:** los usuarios validados deben poder observar los datos en tiempo real de las camas disponibles.
  - **RF-1.4 Restricción de acceso:** los usuarios solamente podrán tener acceso a los datos de las camas permitidas.
  - **RF-1.5 Acceso completo al administrador:** el administrador debe poder acceder a los datos de todas las camas existentes.

- **RF-2 Gestión de las camas:** El administrador debe poder gestionar las camas pudiendo añadir, modificar, borrar y dar acceso a un usuario a los datos de una cama determinada.
  - **RF-2.1 Añadir cama:** el administrador debe poder añadir una nueva cama al sistema.
  - **RF-2.2 Modificar cama:** el administrador debe poder modificar los datos una cama existente.
  - **RF-2.3 Borrar cama:** el administrador debe poder borrar una cama del sistema.
  - **RF-2.4 Asignar camas a usuarios:** el administrador se encarga de decidir qué usuario puede acceder a los datos de qué cama.
- **RF-3 Gestión de los usuarios:** el administrador debe poder gestionar los usuarios pudiendo añadir, modificar y borrar. El usuario debe poder gestionar su propia contraseña.
  - **RF-3.1 Añadir usuario:** el administrador debe poder añadir un nuevo usuario al sistema.
  - **RF-3.2 Modificar usuario:** el administrador debe poder modificar los datos un usuario existente. Igualmente el usuario debe poder modificar su propia contraseña.
  - **RF-3.3 Borrar usuario:** el administrador debe poder borrar un usuario del sistema.
- **RF-4 Visualización de los datos:** los usuarios deben poder ver, de las camas disponibles, el estado actual del paciente, la probabilidad de crisis epiléptica, sus constantes vitales y las presiones.

## Requisitos no funcionales

- **RNF-1 Usabilidad:** la aplicación debe cumplir estándares de usabilidad teniendo una curva de aprendizaje baja y un uso de metáforas adecuado.
- **RNF-2 Confidencialidad:** los datos de las camas, al ser en parte constantes vitales de pacientes, solamente han de ser accesibles por los usuarios permitidos.

- **RNF-3 Escalabilidad:** el sistema debe ser escalable para adaptarse de manera correcta a un incremento de carga del sistema.
- **RNF-4 Seguridad:** los usuarios deben poder identificarse sólidamente con el sistema sin que sus datos o sus credenciales (*tokens*) sean accesibles por terceros, incluso el administrador.

## B.4. Especificación de requisitos



## *Apéndice C*

---

# **Especificación de diseño**

---

### **C.1. Introducción**

### **C.2. Diseño de datos**

### **C.3. Diseño procedimental**

### **C.4. Diseño arquitectónico**

### **C.5. Diseño de interfaces**

Inicialmente se realizaron una serie de prototipos básicos en los que se plasmaron las principales funcionalidades de la aplicación, sin prestar especial atención a los aspectos estéticos de la misma. Para ello se usó el programa de prototipado Pencil.



Figura C.1: Prototipos iniciales de las pantallas de: login, administración, visualización de camas y visualización de datos.

## *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] Maximilian Christ et al. tsfresh, 2019. [Internet; accedido 09-mayo-2019].
- [2] Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, and Christian Gagné. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13:2171–2175, jul 2012.
- [3] Scikit learn developers. sklearn.base.transformermixin, 2018. [Internet; accedido 09-mayo-2019].
- [4] Scikit learn developers. sklearn.ensemble.randomforestclassifier, 2018. [Internet; accedido 09-mayo-2019].
- [5] Scikit learn developers. sklearn.manifold.mds, 2018. [Internet; accedido 09-mayo-2019].
- [6] Scikit learn developers. sklearn.svm.oneclasssvm, 2018. [Internet; accedido 09-mayo-2019].
- [7] Overleaf. Overleaf documentation, 2019. [Internet; accedido 09-mayo-2019].
- [8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [9] tmux. tmux - home, 2019. [Internet; accedido 09-mayo-2019].