



# UNIVERSIDAD DE GRANADA

---

## CUADERNO DE PRÁCTICAS

### Primera entrega

*Planificación y Gestión de Proyectos Informáticos*

---

Máster Profesional en Ingeniería Informática

Curso académico 2022/2023

### **Autores**

Ramón García Verjaga ([rgarver@correo.ugr.es](mailto:rgarver@correo.ugr.es))

José Alberto Gómez García ([modej@correo.ugr.es](mailto:modej@correo.ugr.es))

## Versiones del documento

**v1:** [14-10-2022] Se inicia análisis de requisitos a partir del pliego de prescripciones técnicas proporcionado.

**v2:** [21-10-2022] Se tiene en consideración las respuestas ante “la entrevista que se tuvo” con la jefa de estudios del centro.

**v3:** [28-11-2022] Se detalla el manual de coordinación, basado en la experiencia en proyectos previos. Se comienza con la obtención de épicas asociadas a las funcionalidades más importantes de la aplicación. Se generan las primeras historias de usuario.

**v4:** [4-11-2022] Se realiza la planificación temporal del proyecto a partir de las historias de usuario y épicas definidas conforme se analizaron requisitos.

**v5:** [4-11-2022] Se inicia estimación de costos a partir del diagrama temporal realizado. Se comienza a detallar el plan de entregas en base a los productos mínimos viables que se van a generar a lo largo del proceso.

**v6:** [11-11-2022] Se realizan consideraciones adicionales y cambios menores en diversos apartados del documento. Se finaliza la definición del plan de entregas. Se asignan las épicas en relación al valor a generar. Se finaliza el documento.

**v7:** [16-11-2022] Revisiones finales del documento previas a la entrega del mismo. Se realiza la firma del documento.

## Firma del documento.

Tras finalizar el documento, y estando ambos miembros del equipo conformes con lo expuesto en el mismo, se procede a la firma de este cuaderno de prácticas.

José Alberto Gómez García	Ramon García Verjaga
	

# 1. Propuesta técnica resumida del proyecto

## ***Título***

**SPIVAD - SISTEMA PARA EL DESARROLLO PERSONAL E INTEGRACIÓN EN LA VIDA ADULTA DE ALUMNOS CON DIVERSIDAD FUNCIONAL.**

## ***Resumen***

El presente proyecto consiste en el desarrollo de una aplicación móvil que permita dotar a los alumnos del Colegio de Educación Especial San Rafael de una agenda individualizada con la que organizar las tareas que cada uno de los estudiantes debe realizar, de manera que dicho alumnado con necesidades especiales cuente con una mayor autonomía personal. Esta agenda estará supervisada por el personal del centro, el cual podrá consultar si cada alumno ha realizado las tareas asignadas e información sobre las mismas, por ejemplo, fecha de realización, valoración, etc.

## ***Lugar de ejecución***

En la ETSIIT de la Universidad de Granada, así como en las casas de los miembros de este grupo, se realizarán las tareas de planificación, gestión e hipotético desarrollo de la aplicación solicitada.

En el Colegio de Educación Especial San Rafael se realizará el despliegue de los servidores necesarios por la aplicación. Será también en este centro donde se realizarán las reuniones con el cliente y las diversas pruebas con prototipos. Idealmente, también será donde se realicen las pruebas de las versiones finales del software con los alumnos a los que está dirigido en última instancia.

## ***Tiempo de ejecución***

El proyecto se realizará durante el transcurso del cuatrimestre, desde el 17 de octubre de 2022 hasta el 27 de enero de 2023.

En este plazo de tiempo se fijarán diferentes fechas en las que se deberá entregar documentación, resultado de realizar las diferentes prácticas propuestas, correspondientes a tareas específicas de planificación y gestión del presente proyecto.

## ***Objetivos***

### *Objetivo general*

El objetivo general y principal del proyecto es dotar a los alumnos con diversidad funcional del Colegio de Educación Especial San Rafael de una mayor autonomía en su vida cotidiana, es decir, en su paso a la vida adulta.

Esto se pretende conseguir a través de una propuesta basada en la gestión intuitiva y sencilla de las tareas que deben realizar en su día a día. Así pues, también se ofrece una tranquilidad y seguridad adicional al personal del centro y a los familiares del alumnado.

### *Objetivos específicos*

- Diseño y desarrollo de un sistema de agenda de tareas interactivo basado en la sencillez y accesibilidad a través de pictogramas y elementos fácilmente identificables.
- Gestión de las tareas a realizar por el personal del centro.
- Envío de notificaciones como recordatorios de realización de las tareas.
- Inclusión de tutoriales y/o explicaciones en ciertos tipos de tareas.
- Chat entre personal del centro y el alumno, en el marco de una tarea.
- Localización del alumno en tiempo real o, en su defecto, en la última ubicación en la que estuvo.
- Análisis del estado emocional del estudiante gracias a los relojes inteligentes que dichos alumnos llevan a lo largo de la jornada escolar.
- Generación de estadísticas y gráficos respecto de las tareas realizadas y estado emocional del estudiante, prestando especial hincapié en la sencillez.

### ***Justificación***

Con el paso del tiempo, y especialmente hoy en día, ha ganado especial relevancia la inclusión en la sociedad de personas con diversidad funcional, una problemática que había sido ignorada y subestimada en el pasado, muy a nuestro pesar.

Para poder construir este producto social se requiere aunar esfuerzos muy importantes por parte de las propias personas afectadas, familiares y centros educativos, cuya tarea es crucial para garantizar el correcto crecimiento personal e integración social de este colectivo.

Compartimos la visión del colegio San Rafael de que, en el marco de los programas de transición a la vida adulta (PTVAL), una aplicación que ayude a las personas con discapacidad funcional a organizar su vida diaria, de forma sencilla y accesible, contribuiría al desarrollo de la autonomía personal y favorecería su integración en la sociedad de forma paulatina. Comprometidos con estos ideales, decidimos abordar el proyecto que dicha institución propone.

### ***Antecedentes***

#### *Historial del grupo*

Los miembros del grupo cuentan con la siguiente titulación y experiencia:

- **Ramón García Verjaga.** Graduado en Ingeniería Informática, con mención en Ingeniería de Software, por la Universidad de Granada. Cuenta con experiencia

profesional en el diseño y desarrollo de sistemas basados en microservicios con arquitecturas distribuidas escalables.

- **José Alberto Gómez García.** Graduado en Ingeniería Informática, con mención en Tecnologías de la Información, por la Universidad de Granada. Cuenta con experiencia en el desarrollo de aplicaciones móviles, gestión de servidores e infraestructura en la nube.
- **Mario Villar Sanz.** Graduado en Ingeniería Informática, con mención en Computación y Sistemas Inteligentes, por la Universidad de Granada. Cuenta con experiencia en el ámbito del aprendizaje automático, minería y análisis de datos.
- **Jesús Navarro Merino.** Graduado en Ingeniería Informática, con mención en Ingeniería de Software por la Universidad de Granada. Cuenta con amplia experiencia en el desarrollo de interfaces de usuario, prestando especial énfasis en la accesibilidad y la personalización de las mismas.

El grupo, con formación sólida y complementaria en varios ámbitos relacionados con el software a desarrollar, ya ha trabajado en proyectos de forma conjunta anteriormente, por lo que la planificación, organización y capacidad de resolver problemas internos son cualidades que se han trabajado a lo largo del tiempo, y consideramos una gran virtud de cara al desarrollo de este proyecto.

### *Trabajos similares*

Tras un proceso de búsqueda en la red, no encontramos aplicaciones que proporcionen la misma funcionalidad que la propuesta. La mayoría del software encontrado se centra en juegos adaptados a personas con diversidad funcional, o a que familiares de dichas personas puedan encontrar establecimientos y servicios especialmente adaptados; o comunicar emergencias médicas al servicio de salud correspondiente.

No encontramos ninguna propuesta que promueva la autonomía de las personas con diversidad funcional mediante un sistema que les permita gestionar más fácilmente su vida diaria. Tampoco se encuentra ese grado de conexión con el centro educativo que se espera de una aplicación de este estilo. Sin embargo, no se descarta que otros centros utilicen software desarrollado a medida similar al propuesto.

## **Innovación**

En la actualidad existen multitud de aplicaciones que permiten gestionar las actividades a realizar en el día a día. Algunas de ellas consisten en simples listas en las que los elementos se pueden marcar como completados; mientras que otras integran funciones de calendario que permiten establecer recordatorios y hacer uso de funciones algo más avanzadas. La gran mayoría de estas aplicaciones están orientadas al uso individual.

La aplicación propuesta se diferencia de las anteriores en el especial énfasis que se debe realizar en la accesibilidad, intuitividad y sencillez de uso. También, se incluyen funcionalidades adicionales de las que carecen otras aplicaciones, como la explicación de

las tareas a realizar mediante tutoriales por texto, imagen o vídeo; y la capacidad de proporcionar pruebas a terceros de que la tarea marcada como realizada efectivamente ha sido realizada.

En especial, destacamos como novedad la capacidad de gestión del equipo directivo del centro educativo encargado de velar por los alumnos, así como la capacidad de interactuar con el alumnado, para cada tarea, por medio de la aplicación. Como mencionamos anteriormente, la inmensa mayoría de aplicaciones para gestionar actividades tienen un enfoque individual orientado a la autogestión de las mismas; mientras que en nuestro caso se hace partícipe de este proceso a terceras partes que comprenden especialmente las necesidades individuales de cada uno de los alumnos bajo su cuidado.

### ***Lista de actividades a realizar alineadas con los objetivos***

Para cumplir con los deseos del cliente, se realizará una aplicación móvil que debe satisfacer los siguientes requisitos técnicos:

- ***Sistema de identificación de usuarios.*** Dada las distintas capacidades de los alumnos del centro, el sistema de identificación de los mismos dentro de la aplicación debe estar adaptado de manera individual para cada usuario. De esta manera, se deberá poder iniciar sesión haciendo uso de contraseñas alfanuméricas, pulsando en fotos concretas (siguiendo un determinado orden) o pulsando en la foto de uno mismo. Nótese la sencillez de los mecanismos utilizados, especialmente los dos últimos.
- ***Sistema de creación de tareas.*** El equipo directivo del centro debe poder especificar las tareas a realizar por cada estudiante, así como el momento y lugar donde debe realizarse, la fecha límite para la cual deberían estar completadas o la periodicidad con la que deben realizarse. Existen dos tipos de tareas:
  - Fijas, en las que el estudiante realiza una cierta tarea tras seguir una explicación detallada.
  - Comandas, en las que el estudiante debe introducir datos en un formulario además de realizar la tarea en sí misma.
- ***Sistema de gestión de tareas.*** El alumno debe poder visualizar las tareas que tiene asignadas (por ejemplo, en un calendario), así como las explicaciones y/o formularios asociados a las mismas, y marcarlas como completadas cuando corresponda. Esta confirmación debe ser visible para el equipo directivo del centro. También, el sistema debe recordar al usuario la realización de las llamadas “tareas fijas”, con una antelación mínima personalizable para cada caso por parte del personal que creó dicha tarea.

Para cierto tipo de tareas, el estudiante podrá opcionalmente subir fotos o vídeos que demuestren la correcta realización de la tarea, por lo que se deberán gestionar

estos recursos.

En caso de que una determinada tarea no se realice a tiempo, se deberá notificar tanto al alumno que no la realizó como al equipo del centro.

- **Sistema de historial.** Se debe implementar un historial semanal que permita visualizar rápidamente las tareas completadas por cada alumno, así como las tareas que han sufrido retrasos o no han sido completadas. Este historial es visible por el propio alumno y por el personal del centro. Adicionalmente, se deberán generar gráficas e infografías sencillas que reflejen esta información de manera sencilla con un carácter más visual.
- **Sistema de geolocalización.** El equipo directivo del centro debe poder saber dónde se encuentra cada estudiante en tiempo real, y de quién se encuentra acompañado, durante el horario escolar. Si se pierde la conexión, se debe poder consultar la última ubicación disponible. Si el alumno estuviera en una zona donde no debiera, el sistema podría notificar a los encargados para que realizarán las gestiones que consideren oportunas.
- **Sistema de análisis emocional.** Algunos estudiantes poseen relojes inteligentes con sistema operativo Android, de los que se deben poder recoger métricas que permitan la realización de un análisis cuyo objetivo es inferir el estado emocional y/o anímico del alumno durante la realización de las diversas tareas programadas en su agenda personal y en general, a lo largo del horario escolar.
- **Sistema de personalización.** Además de los ajustes generales a realizar por el personal del centro, descritos en puntos anteriores. El alumno debe poder configurar apartados de la interfaz como el tamaño de letra o los colores.

## ***Cronograma.***

Esta información podrá encontrarse en el tercer apartado de este documento, “Planificación temporal del proyecto”.

## ***Cauces de seguimiento por parte del cliente***

Inicialmente, se tendrá una reunión con el cliente en la que se detallará formalmente toda la funcionalidad y requisitos de la aplicación a desarrollar.

Para que el cliente sea consciente del estado del desarrollo de la aplicación se le remitirá cada dos semanas un documento con la funcionalidad desarrollada hasta el momento y qué será lo próximo en ser incorporado a la aplicación, para que se verifique que todo se desarrolla de acuerdo a la planificación inicial y se incluye la funcionalidad pactada.

Con fines de validación de diseño, se mostrarán diseños de alta fidelidad (mockups) con carácter mensual. Finalmente, en las últimas semanas, se realizará una prueba conjunta de la aplicación en su totalidad con los responsables del centro y se detallarán los últimos retoques a realizar de ser necesario. De ser posible, se realizarán también pruebas con usuarios finales.

## ***Garantía o mantenimiento***

De acuerdo a lo especificado en el pliego de prescripciones técnicas, la aplicación y los servidores asociados contarán con seis meses de garantía y mantenimiento incluidos en el precio. Adicionalmente, la empresa solicitante puede decidir extender dicho periodo de garantía y mantenimiento, hasta un máximo de 3 años en su totalidad, cuyo precio tendría que negociarse con el cliente a posteriori.

## ***Valor añadido***

La especificación de requisitos propuesta por el cliente es bastante amplia, detallando con mucha exhaustividad los puntos a los que debemos ceñirnos, teniendo siempre por bandera la accesibilidad y facilidad de uso.

Aun así, creemos que hay margen para la mejora. En la propuesta inicial solo se contemplan técnicas basadas en las habituales contraseñas, o “desafíos” basados en pictogramas. Creemos que sería conveniente disponer también de opciones para iniciar sesión mediante reconocimiento de voz, facial o de huella dactilar. Estas técnicas, en especial el reconocimiento facial y de voz, son simples de realizar por parte del alumnado, pudiendo realizarlas casi de manera inconsciente y proporcionan un alto grado de seguridad, el cual no es el objetivo principal de la aplicación, pero siempre es conveniente tenerlo en cuenta.

## ***Beneficios y beneficiarios***

Los principales beneficiarios de este proyecto son los alumnos con diversidad funcional del Colegio de Educación Especial San Rafael dado que dispondrán de un sistema software que les permitirá organizar su vida de una manera sencilla, de manera que progresivamente obtendrán un mayor nivel de autonomía y podrán integrarse de mejor manera en la sociedad. A parte de los beneficios expuestos, más orientados al largo plazo, la consecución de las tareas propuestas puede proporcionarles una satisfacción personal que mejore su estado emocional en el día a día, siendo este un punto de gran importancia.

Los beneficios anteriores también repercuten de forma indirecta en los familiares de los alumnos, que potencialmente tendrán que realizar un menor esfuerzo y sacrificio conforme dichos alumnos sean más autónomos e independientes.

Finalmente, consideramos que el personal del centro también resulta beneficiada de la aplicación, pues dispone de una manera centralizada de gestionar las tareas de los estudiantes que permitiría ahorrar tiempo y esfuerzo; los cuales podrían ser dedicados a la realización de otras tareas dentro del proyecto formativo con el que se comprometen en cuerpo y alma, y que sin duda hará crecer personalmente a estos estudiantes con diversidad funcional.



## 2. Manual de coordinación

<b>Metodología y ciclo de vida</b>
<i>Se decide utilizar la metodología de desarrollo ágil SCRUM. Se definirán historias de usuario, agrupadas en épicas, que se implementarán según su prioridad.</i>
<b>Recursos hardware para el desarrollo</b>
<i>Cada miembro del equipo hará uso de un portátil, cuyas especificaciones le permitan lanzar los recursos software especificados en el siguiente apartado. En particular, se usarán portátiles con procesadores Intel Core i7 de 7º generación o posterior, 16 GB de RAM, y unidades SSD. Es recomendable que se disponga de GPU dedicada.</i>
<b>Recursos software para el desarrollo</b>
<i>Se hará uso de los IDEs Android Studio y Visual Studio Code, ambos gratuitos. Como frameworks en los que basar el desarrollo se utilizará Flutter (móvil) y Firebase (backend).</i>
<b>Organización del equipo de trabajo (estructura, normas)</b>
<i>Ramón y Jesús trabajarán principalmente en el diseño de la aplicación, haciendo énfasis en la accesibilidad. José Alberto trabajará en lo relativo a los servidores y despliegue; mientras que Mario trabajará en el análisis de los datos recabados (especialmente los proporcionados por los relojes inteligentes) Los miembros del equipo deberán cumplir las tareas asignadas, en tiempo y forma, así como proporcionar feedback sobre el trabajo realizado por el resto del equipo. Se deberán cumplir las funciones asignadas durante cada iteración.</i>
<b>Herramientas para comunicaciones en el equipo de trabajo</b>
<i>Se utilizará Telegram para las comunicaciones informales, mientras para las reuniones y sesiones de revisión que deban hacerse online se hará uso de Microsoft Teams. En las horas de clase las reuniones y revisiones tendrán lugar de forma presencial en el aula. Para la gestión del proyecto y la información del desarrollo se hará uso de Jira.</i>
<b>Relaciones con el cliente (entrevistas, reuniones, revisiones)</b>
<i>Se realizarán entrevistas con el cliente, de manera que este pueda responder a preguntas preparadas por el equipo de desarrollo con anterioridad. Idealmente, tendrán lugar de forma presencial, pero si la situación lo requiere se podrán realizar online. Respecto de las reuniones y revisiones, se concertará por email con antelación suficiente, intentando que acudan el mayor número de miembros del equipo posibles. Se redactará un documento con el contenido tratado y las conclusiones alcanzadas.</i>
<b>Estándares de documentación</b>

La documentación del proyecto seguirá una guía de estilo marcada por una plantilla que los miembros del equipo tendrán. En lo relativo al código, se utilizará la herramienta del propio lenguaje, que en este caso es “dartdoc”.

### **Estándares de código**

Se hará uso del “linter” de “Flutter”, el cual se debe añadir al proyecto. Se velará por el cumplimiento por las recomendaciones del lenguaje Dart, que indica que se debe usar “UpperCamelCase” para los tipos, “lowercase\_with\_underscores” para los paquetes y archivos, y “lowerCamelCase” para el resto de elementos.

### **Medición del desempeño o progreso**

Con este objetivo, se realizarán las siguientes actividades:

- Estudio de la consecución de los objetivos en tiempo y forma, consultable en Jira.
- Historial del trabajo de desarrollo, consultable en Github.
- Revisión por pares y supervisión mutua entre los miembros del equipo.
- Establecimiento y revisión de criterios objetivos para valorar el cumplimiento de la planificación propuesta.

### **Plan de gestión de cambios**

Para realizar un cambio sustancial deberá generarse una solicitud, que deberá ser aprobada por todos los miembros del equipo tras analizar las posibles implicaciones técnicas y consecuencias de dicho cambio. El coordinador del proyecto autorizará el cambio e informará al equipo con el nuevo modo de proceder.

### **Control de versiones (método y herramientas)**

Se hará uso de Git, y el repositorio se encontrará alojado en Github. Adicionalmente, se utilizará Gitflow, de manera que se tendrán ramas de desarrollo, y subramas para las diferentes funcionalidades/características. Adicionalmente, se tendrán copias de seguridad de las diferentes publicaciones en Mega.

### **Gestión de calidad (proceso y herramientas)**

- En el margen del control de versiones, se hará uso de GitFlow, y se necesitará de la aprobación de todos los miembros del equipo para incluir aprobar un “pull request”.
- El proceso de desarrollo en su conjunto estará supervisado por el miembro del equipo que tenga asignadas las competencias de QA.
- En las fases finales del proyecto, se contactará al cliente para realizar demostraciones y obtener feedback respecto de posibles mejoras o correcciones a realizar.

### 3. Planificación temporal del proyecto

Se ha decidido seguir una metodología de desarrollo ágil similar a SCRUM, pero con pequeñas variaciones.

Se definen historias de usuario, agrupadas en épicas, que se planifican para su realización en un determinado sprint en función de su prioridad, dividiéndose en 6 sprints y 3 entregas. A continuación, se detallan las épicas, historias de usuario y las tareas a realizar:

## Épicas, historias de usuario y tareas

---

- **Sprints de 2 semanas**
- **Entregas de productos de valor cada 2 sprints (cada 4 semanas)**

***Entrega 1 (MVP: aplicaciones de usuario y administración, gestión de usuarios y gestión simple de tareas)***

*Sprint 1: 17 oct - 28 oct*

- **SPAVID-1: Especificación, documentación y planificación**
- **SPAVID-2: Configuración del repositorio y las herramientas**
- **SPAVID-3: Infraestructura inicial**

*Sprint 2: 31 oct - 11 nov*

- **SPAVID-4: Gestión de usuarios**
- **SPAVID-5: Gestión de tareas (parte I, hasta 5.5)**
- **SPAVID-10: Garantías de accesibilidad (parte I)**

***Entrega 2 (MVP: gestión compleja de tareas, gestión de notificaciones y gestión de informes)***

*Sprint 3: 14 nov - 2 dic*

- **SPAVID-5: Gestión de tareas (parte II)**
- **SPAVID-6: Gestión de notificaciones**

*Sprint 4: 5 dic - 17 dic*

- **SPAVID-8: Gestión de informes**
- **SPAVID-10: Garantías de accesibilidad (parte II)**

***Entrega 3 (MVP: gestión de comunicación en tiempo real, gestión de localización y mejoras en cuanto a accesibilidad y otros requisitos no funcionales)***

*Sprint 5: 19 dic - 6 ene (1 semana extra por las vacaciones de Navidad)*

- **SPAVID-7: Gestión de comunicaciones en tiempo real**
- **SPAVID-9: Gestión de localización**

*Sprint 6: 9 ene - 20 ene*

- **SPAVID-10: Garantías de accesibilidad (parte III)**
- **SPAVID-11: Garantías de otros requisitos no funcionales**

---

*\*\* La asignación de recursos a tareas, la especificación de la duración de las tareas, las fechas de realización de las tareas y las dependencias de las tareas se han reflejado y elaborado utilizando herramienta de gestión de proyectos ProjectLibre.*

---

## **SPAVID-1: Especificación, documentación y planificación**

*SPAVID-1.1: Propuesta técnica resumida*

*SPAVID-1.2: Ficha completa del manual de coordinación*

*SPAVID-1.3: Diagrama de Gantt*

*SPAVID-1.4: Diagrama de Actividades*

*SPAVID-1.5: Estimación del coste*

*SPAVID-1.6: Diseño de prototipos*

---

## **SPAVID-2: Configuración del repositorio y las herramientas**

*SPAVID-2.1: Como aplicación quiero estar alojada GitHub*

*SPAVID-2.1.1: Iniciar repositorio de Git*

*SPAVID-2.1.2: Subir repositorio a GitHub*

*SPAVID-2.1.3: Configurar permisos de acceso a los colaboradores*

*SPAVID-2.2: Como desarrollador quiero utilizar un IDE para simular dispositivos móviles virtuales*

*SPAVID-2.3: Como desarrollador quiero tener configurado el dispositivo móvil con el que se va a testear la aplicación*

---

## **SPAVID-3: Infraestructura inicial**

*SPAVID-3.1: Como aplicación quiero tener mi arquitectura definida*

*SPAVID-3.1.1: Definir y especificar la base de datos y el modelo básico a implementar*

*SPAVID-3.1.2: Definir y especificar la arquitectura de la API Rest*

*SPAVID-3.1.3: Definir y especificar la arquitectura del cliente de administración (acceso de administradores)*

*SPAVID-3.1.4: Definir y especificar la arquitectura del cliente móvil (acceso de profesores y alumnos)*

*SPAVID-3.2: Como aplicación quiero que mis componentes estén correctamente conectados*

SPAVID-3.2.1: Establecer conexión entre la API Rest (back-end) y la base de datos

SPAVID-3.2.2: Establecer conexión entre los diferentes clientes (front-ends) y la API Rest (back-end)

---

## **SPAVID-4: Gestión de usuarios**

*SPAVID-4.1: Como administrador quiero crear usuarios con un rol determinado (administrador, alumno, profesor, tutor legal)*

SPAVID-4.1.1: Diseño e implementación del modelo de usuario en la base de datos (usuarios + roles)

SPAVID-4.1.2: Diseño e implementación de la lógica de creación de usuarios (back-end)

SPAVID-4.1.3: Diseño e implementación del formulario de creación de usuarios (front-end)

*SPAVID-4.2: Como administrador quiero editar usuarios*

SPAVID-4.2.1: Diseño e implementación de la lógica de edición de usuarios (back-end)

SPAVID-4.2.2: Diseño e implementación del formulario de edición de usuarios (front-end)

*SPAVID-4.3: Como administrador quiero eliminar usuarios*

SPAVID-4.3.1: Diseño e implementación de la lógica de eliminación de usuarios (back-end)

SPAVID-4.3.2: Diseño e implementación del formulario de eliminación de usuarios (front-end)

*SPAVID-4.4: Como administrador quiero ver un listado de de los alumnos*

SPAVID-4.4.1: Diseño e implementación de la lógica de obtención de alumnos (back-end)

SPAVID-4.4.2: Diseño e implementación de la lista de alumnos (front-end)

*SPAVID-4.5: Como administrador quiero iniciar sesión*

SPAVID-4.5.1: Diseño e implementación de la lógica de autenticación con roles en la API (back-end)

SPAVID-4.5.2: Diseño e implementación de la lógica de inicio de sesión para el administrador (back-end)

SPAVID-4.5.3: Diseño e implementación de la pantalla de login en el cliente de administración (front-end)

*SPAVID-4.6: Como administrador quiero cerrar sesión*

SPAVID-4.6.1: Diseño e implementación de la lógica de cierre de sesión (back-end)

SPAVID-4.6.2: Diseño e implementación del botón de cierre de sesión en el cliente de administración (front-end)

*SPAVID-4.7: Como alumno quiero iniciar sesión*

SPAVID-4.7.1: Diseño e implementación de la lógica de inicio de sesión para el alumno (back-end)

SPAVID-4.7.2: Diseño e implementación de la pantalla de login en el cliente de alumnos (front-end)

*SPAVID-4.8: Como alumno quiero cerrar sesión*

SPAVID-4.8.1: Diseño e implementación del botón de cierre de sesión en el cliente de administración (front-end)

*SPAVID-4.9: Como alumno quiero ver una lista de los profesores*

SPAVID-4.8.1: Diseño e implementación de la pantalla de la lista en la aplicación móvil (front-end)

*SPAVID-4.10: Como administrador o profesor quiero ver una lista de los alumnos*

SPAVID-4.8.1: Diseño e implementación de la pantalla de la lista en la aplicación de administración (front-end)

---

## **SPAVID-5: Gestión de tareas**

*SPAVID-5.1: Como administrador o profesor quiero crear tareas*

SPAVID-5.1.1: Diseño e implementación del modelo de tareas en la base de datos

SPAVID-5.1.2: Diseño e implementación de la lógica de creación de tareas (back-end)

SPAVID-5.1.3: Diseño e implementación del formulario de creación de tareas (front-end)

*SPAVID-5.2: Como administrador quiero editar tareas*

SPAVID-5.2.1: Diseño e implementación de la lógica de edición de tareas (back-end)

SPAVID-5.2.2: Diseño e implementación del formulario de edición de tareas (front-end)

*SPAVID-5.3: Como administrador quiero eliminar tareas*

SPAVID-5.3.1: Diseño e implementación de la lógica de eliminación de tareas (back-end)

SPAVID-5.3.2: Diseño e implementación del formulario de eliminación de tareas (front-end)

*SPAVID-5.4: Como administrador o profesor quiero crear listas de acciones asociadas a cada tarea*

*SPAVID-5.5: Como administrador o profesor quiero ver las listas de acciones asociadas a cada tarea*

*SPAVID-5.6: Como administrador o profesor quiero asignar tareas de diferentes tipos*

*SPAVID-5.7: Como administrador o profesor quiero modificar el tiempo disponible para realizar la tarea*

*SPAVID-5.8: Como administrador o profesor quiero saber cuando un alumno completa su tarea*

*SPAVID-5.9: Como alumno quiero ver las tareas que tengo asignadas*

*SPAVID-5.10: Como alumno quiero marcar mis tareas como hechas*

*SPAVID-5.11: Como administrador quiero crear un nuevo menú*

*SPAVID-5.12: Como administrador quiero modificar un menú*

*SPAVID-5.13: Como administrador quiero eliminar un menú*

*SPAVID-5.14: Como administrador o profesor quiero determinar cuáles son los menús disponibles del día*

*SPAVID-5.15: Como alumno quiero rellenar los menús diarios*

*SPAVID-5.16: Como alumno quiero apuntar los menús diarios de mis compañeros*



## **SPAVID-6: Gestión de notificaciones**

*SPAVID-6.1: Como sistema quiero enviar una notificación al alumno cuando no haya completado su tarea*

SPAVID-6.1.1: Diseño e implementación del modelo de notificaciones en la base de datos

SPAVID-6.1.2: Diseño e implementación de envío de la notificación cuando suceda algún evento (back-end)

SPAVID-6.1.3: Diseño e implementación de la sección de notificaciones y alerta de recepción de la notificación (front-end)

---

## **SPAVID-7: Gestión de comunicaciones en tiempo real**

*SPAVID-7.1: Como profesor o alumno quiero comunicarme por el chat de una tarea tanto de forma escrita como de forma verbal*

---

## **SPAVID-8: Gestión de informes**

*SPAVID-8.1: Como administrador, profesor o tutor legal quiero generar un informe de tareas realizadas*

*SPAVID-8.2: Como administrador o profesor quiero enviar informe de tareas realizadas a los tutores legales*

*SPAVID-8.3: Como alumno quiero ver un resumen de las tareas realizadas y los logros conseguidos*

*SPAVID-8.4: Como administrador quiero conocer si los alumnos tienen estrés, ansiedad o depresión en base al uso de la aplicación a través de técnicas de machine learning*

---

## **SPAVID-9: Gestión de localización**

*SPAVID-9.1: Como administrador quiero saber la ubicación de un alumno en cada momento*

*SPAVID-9.2: Como administrador quiero saber la última ubicación disponible de un alumno*

*SPAVID-9.3: Como administrador quiero llamar a un alumno si no tengo información sobre la ubicación*

*SPAVID-9.4: Como alumno quiero informar al profesor de que he llegado a la localización*

---

## **SPAVID-10: Garantías de accesibilidad**

*SPAVID-10.1: Como alumno quiero iniciar sesión a través de pictogramas*

*SPAVID-10.2: Como alumno quiero acceder a una pantalla principal donde ver las secciones alcanzables representadas a través de pictogramas*

*SPAVID-10.3: Como alumno quiero acceder a las secciones mediante pulsaciones en los pictogramas*

*SPAVID-10.4: Como alumno quiero acceder a las secciones mediante comandos de voz en base al texto relacionado con los pictogramas*

*SPAVID-10.5: Como alumno quiero volver a la pantalla de inicio desde cualquier punto de la aplicación de forma sencilla*

*SPAVID-10.6: Como alumno quiero cambiar la vista de la aplicación para verla con diferentes elementos y/o de diferente tamaño*

*SPAVID-10.7: Como profesor quiero escoger el formato en que se le va a mostrar al alumno la información de la tarea (texto, imagen, sonido, vídeo, etc.)*

---

## **SPAVID-11: Garantías de otros requisitos no funcionales**

*SPAVID-11.1: Como usuario quiero sentir fluidez en el funcionamiento del producto*

*SPAVID-11.2: Como sistema quiero tener bajos tiempos de respuesta*

*SPAVID-11.3: Como sistema quiero ser tolerante a errores*

*SPAVID-11.4: Como sistema quiero tener los datos protegidos*

*SPAVID-11.5: Como producto quiero cumplir con los requisitos legales existentes*

---

Dada la gran extensión de los diagramas de Gantt y de actividad; así como las tablas de especificación de tareas y asignación de tareas a personas, estas no se incluyen en este documento. Por favor, consulten los ficheros PDF y vídeos que se pueden encontrar dentro del fichero comprimido adjunto en la entrega. En particular, para el diagrama de Gantt se recomienda hacer uso de la “vista de libro” del lector de PDF que se utilice.

Recomendamos encarecidamente hacer uso del propio software “Project Libre” para la visualización de la información.

## 4. Estimación del coste del proyecto

En base a la planificación temporal se ha obtenido un coste en tiempo de 1140 horas. Si suponemos que cada persona realiza al mes 160 horas, esto nos daría una cantidad de alrededor de 7 pm.

### ***Opción 1: Estimación por descomposición funcional***

En la siguiente tabla, se adjunta la descomposición en módulos funcionales que consideramos apropiada para el presente proyecto, así como el esfuerzo estimado para la realización de cada uno de estos módulos.

Módulo	Esfuerzo estimado
Bases de datos	2 pm
Estructura de clases	1 pm
Interfaz de usuario	3 pm
Comunicaciones	0.5 pm
Soporte a relojes inteligentes	0.5 pm
Algoritmos de análisis de datos	1 pm
<b>TOTAL (PM)</b>	8 pm

Asumiendo un coste por persona y mes de 3.000€, se estima un coste del proyecto de 24.000 €

### ***Opción 2: Estimación por descomposición funcional en 3 puntos***

Módulo	Pesimista	Más probable	Optimista
Bases de datos	3 pm	2 pm	1.33 pm
Estructura de clases	1.5 pm	1 pm	0.66 pm
Interfaz de usuario	4.5 pm	3 pm	2 pm
Comunicaciones	0.75 pm	0.5 pm	0.33 pm
Soporte a relojes inteligentes	0.25 pm	0.5 pm	0.33 pm
Algoritmos de análisis de datos	1.5 pm	1 pm	0.66 pm

<b>TOTAL (PM)</b>	11.5 pm	8 pm	5.33 pm
-------------------	---------	------	---------

Asumiendo un coste por persona y mes de 3.000€, se estima el rango en que se encontraría el coste del proyecto de la siguiente forma:

Duración más probable:  $(\text{pesimista} + 4 * \text{masProbable} + \text{optimista})/6$

$(11.5 + 4 * 8 + 5.33)/6 = 8,138$

Varianza:  $(\text{pesimista} - \text{optimista})/6$

$(11.5 - 5.33)/6 = 1,029$

Rango de costes:  $[\text{duración más probable} - \text{varianza}, \text{duración más probable} + \text{varianza}] * \text{coste por persona y mes}$ .

$[8.138 - 1.029, 8.138 + 1.029] * 3000 \text{ €} = [7.109, 9.167] * 3000 \text{ €} = [21327, 27501] \text{ €}$

De esta manera, obtenemos que el proyecto tendría un coste estimado que se encontraría en el intervalo  $[21327, 27501] \text{ €}$ .

### ***Opción 3: Estimación por descomposición de actividades***

<b>Módulo</b>	<b>Planning</b>	<b>Análisis</b>	<b>Diseño</b>	<b>Código</b>	<b>Test</b>	<b>Total</b>
Bases de datos	0.2 pm	0.4 pm	0.6 pm	0.6 pm	0.2 pm	<b>2 pm</b>
Estructura de clases	0.1 pm	0.1 pm	0.3 pm	0.2 pm	0.1 pm	<b>1 pm</b>
Interfaz de usuario	0.2 pm	0.5 pm	0.6 pm	1.2 pm	0.5 pm	<b>3 pm</b>
Comunicaciones	0.05 pm	0.1 pm	0.1 pm	0.15 pm	0.1pm	<b>0.5 pm</b>
Soporte a relojes inteligentes	0.05 pm	0.05 pm	0.1 pm	0.15 pm	0.5 pm	<b>0.5 pm</b>
Algoritmos de análisis de datos	0.1 pm	0.1 pm	0.3 pm	0.3 pm	0.2 pm	<b>1 pm</b>
<b>TOTAL (PM)</b>	0.7 pm	1.25 pm	2 pm	2.6 pm	1.6 pm	<b>8 pm</b>
<b>TOTAL (%)</b>	8.75 %	15.63 %	25 %	32.5 %	20 %	<b>100 %</b>

Al igual que en apartados anteriores, asumimos un coste por persona y mes de 3.000€. Por tanto, se estima un coste del proyecto de 24.000 €. Se puede observar que esta estimación coincide con la realizada en la opción 1.

## Opción 4: Estimación haciendo uso de “COCOMO II”

Por último, hemos decidido hacer uso de la herramienta de estimación de costes online “COCOMO II”.

Para realizar la estimación se ha utilizado el método “de puntos funcionales”, los cuales estimamos en 100 tras realizar el análisis de requisitos y planificación temporal del capítulo anterior. Como lenguaje de programación elegimos Java, el más similar de entre los disponibles a Flutter, que sería el que realmente utilizaremos.

Además, personalizamos otros parámetros. En particular, podemos destacar que el equipo se encuentra cohesionado y cuenta con experiencia en el ámbito en el que debe trabajar. Además, podemos mencionar que la base de datos no es compleja, pero el producto en su conjunto sí que podría llegar a serlo.

Todos los ajustes realizados se pueden consultar en la siguiente imagen:

The image shows the COCOMO II online estimation tool interface. It includes the following sections and inputs:

- Software Size:** Sizing Method (Function Points), Unadjusted Function Points (100), Language (Java).
- Software Scale Drivers:** Precedentedness (Nominal), Architecture / Risk Resolution (Nominal), Process Maturity (Nominal), Development Flexibility (High), Team Cohesion (Very High).
- Software Cost Drivers:**
  - Product:** Required Software Reliability (High), Data Base Size (Low), Product Complexity (High), Developed for Reusability (Nominal), Documentation Match to Lifecycle Needs (Nominal).
  - Personnel:** Analyst Capability (Nominal), Programmer Capability (High), Personnel Continuity (High), Application Experience (Nominal), Platform Experience (Nominal), Language and Toolset Experience (High).
  - Platform:** Time Constraint (Very High), Storage Constraint (Nominal), Platform Volatility (Nominal).
  - Project:** Use of Software Tools (Nominal), Multisite Development (Nominal), Required Development Schedule (Nominal).
- Maintenance:** Off.
- Software Labor Rates:** Cost per Person-Month (Dollars) (3000), Calculate button.

Con estos valores, obtenemos una estimación que nos dice que necesitaremos de 18.8 personas por mes, el proyecto debe desarrollarse durante unos 9 meses, y el coste sería cercano a los 56.350€. En la imagen adjunta en la siguiente página se pueden encontrar detalles adicionales, como el número de personas por mes que se necesita para la fase de comienzo, elaboración, construcción y transición.

## Results

### Software Development (Elaboration and Construction)

Effort = 18.8 Person-months

Schedule = 9.2 Months

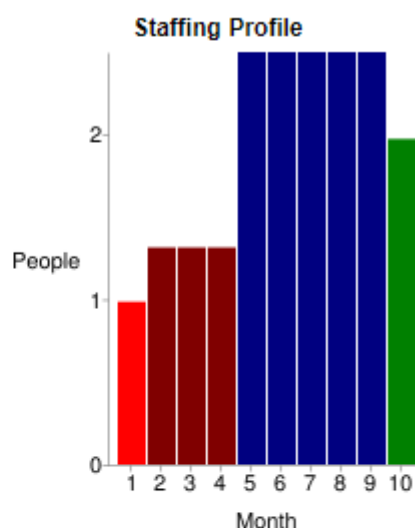
Cost = \$56357

Total Equivalent Size = 5300 SLOC

Effort Adjustment Factor (EAF) = 1.08

### Acquisition Phase Distribution

Phase	Effort (Person-months)	Schedule (Months)	Average Staff	Cost (Dollars)
Inception	1.1	1.1	1.0	\$3381
Elaboration	4.5	3.4	1.3	\$13526
Construction	14.3	5.7	2.5	\$42832
Transition	2.3	1.1	2.0	\$6763



### Software Effort Distribution for RUP/MBASE (Person-Months)

Phase/Activity	Inception	Elaboration	Construction	Transition
Management	0.2	0.5	1.4	0.3
Environment/CM	0.1	0.4	0.7	0.1
Requirements	0.4	0.8	1.1	0.1
Design	0.2	1.6	2.3	0.1
Implementation	0.1	0.6	4.9	0.4
Assessment	0.1	0.5	3.4	0.5
Deployment	0.0	0.1	0.4	0.7

## Conclusión

Como se ha podido observar, la opción 2 es una extensión de la opción 1. Cualquier método en el que se puedan tener en cuenta 3 puntos en lugar de uno solo para estimar va a ofrecer un visión global más flexible y ajustada a lo que sería la realidad. Hay más probabilidad de que el coste real se encuentre en el intervalo proporcionado por la estimación en 3 puntos que este sea justamente el coste concreto ofrecido por la estimación única de la primera opción.

La opción 3 ofrece una estimación basada en descomposición funcional. Si observamos el resultado, es el mismo que la opción 1, lo cual tiene sentido, porque la descomposición funcional se realiza por cada uno de los módulos o partes de la aplicación. A diferencia de la técnica de la opción 1, esta técnica nos ofrece una versión más refinada del coste de cada parte. Sin embargo, la opción 2 sigue arrojando un mejor resultado desde el punto de vista de los riesgos que se asumirán al ofrecerlos al cliente.

Finalmente, consideramos que las estimaciones realizadas por la aplicación online son demasiado diferentes respecto de las que hemos calculado en las otras opciones, por lo que las desestimamos por completo.