Projecte Final de Cicle

Aplicació per a l'organització i seguiment àgil de tasques per a projectes I+D

**Alumne**: Alejandro Rubio Rico

**DNI**: 48646684-M

**Tutor Individual:** Alicia Gonzalez Canet

**Tutor del grup:** Sebastian Ciscar Breitzler

# Dades del Projecte

### Dades de l’alumne

|  |  |
| --- | --- |
| **Nom i cognoms** | Alejandro Rubio Rico |
| **NIF/NIE** | 48646684M |
| **Curs i CF** | 2n DAM |

### Dades del projecte

|  |  |
| --- | --- |
| **Títol del projecte** | Aplicació per a l'organització i seguiment àgil de tasques per a projectes I+D |
| **Nom del tutor individual** | Alicia González Canet |
| **Nom del tutor del grup** | Sebastian Ciscar Breitzler |
| **Resum** | *El proyecto consta de una aplicación multiplataforma desarrollada en flutter con backend programado mediante springboot, que tiene como objetivo mejorar el actual sistema de planificación y seguimiento de tareas diarias y semanales bajo premisas de un sistema de organización àgil en un entorno de desarrollo de proyectos I+D en el ámbito de la energia. Tecnologías utilizadas: Flutter, MySQL/PostgreSQL, SpringBoot, Java, Dart* |
| **Abstract** | *The project consists of a multiplatform application developed in flutter with backend programmed using springboot, which aims to improve the current system of planning and monitoring of daily and weekly tasks under the premises of a system of agile organization in an environment of development of R & D projects in the field of energy. Technologies used: Flutter, MySQL/PostgreSQL, SpringBoot, Java, Dart.* |
| **Mòduls implicats** | Bases de dades, Desenvolupament d’interfícies, Programació Multimèdia i Dispositius Mòbils, Accés a dades |
| **Data de presentació** | xx de juny de 2024 |

# Índex

[Dades del Projecte 1](#__RefHeading___Toc572_3431357138)

[Dades de l’alumne 1](#__RefHeading___Toc574_3431357138)

[Dades del projecte 1](#__RefHeading___Toc576_3431357138)

[Índex 2](#__RefHeading___Toc578_3431357138)

[1. Introducció/Marc del projecte 3](#__RefHeading___Toc580_3431357138)

[1.1. Descripció del projecte 3](#__RefHeading___Toc582_3431357138)

[1.2. Objectius 3](#__RefHeading___Toc584_3431357138)

[1.3. Tipus de projecte 3](#__RefHeading___Toc586_3431357138)

[1.4. Orientacions per al desenvolupament i recursos 3](#__RefHeading___Toc588_3431357138)

[2. Anàlisi/Estudi de l'estat actual 4](#__RefHeading___Toc590_3431357138)

[2.1. Descripció del sistema actual 4](#__RefHeading___Toc592_3431357138)

[2.2. Viabilitat del sistema actual 4](#__RefHeading___Toc594_3431357138)

[2.3. Requeriments del nou sistema 4](#__RefHeading___Toc596_3431357138)

[3. Anàlisi de la solució 5](#__RefHeading___Toc598_3431357138)

[3.1. Anàlisi de les possibles solucions 5](#__RefHeading___Toc600_3431357138)

[3.2. Avaluació de les possibles solucions 5](#__RefHeading___Toc602_3431357138)

[3.3. Descripció de la solució escollida 5](#__RefHeading___Toc604_3431357138)

[4. Disseny de la solució. 6](#__RefHeading___Toc606_3431357138)

[4.1. Requisits. 6](#__RefHeading___Toc608_3431357138)

[4.2. Anàlisi de riscos. 6](#__RefHeading___Toc638_3431357138)

[4.3. Casos d’us. 6](#__RefHeading___Toc640_3431357138)

[4.4. Diagrames 6](#__RefHeading___Toc642_3431357138)

[5. Desenvolupament de la solució 7](#__RefHeading___Toc610_3431357138)

[5.1. Estimació de cost temporal 7](#__RefHeading___Toc612_3431357138)

[5.2. Dotació de recursos 7](#__RefHeading___Toc614_3431357138)

[5.3. Configuració i/o desenvolupament del sistema 7](#__RefHeading___Toc616_3431357138)

[5.4. Avaluació del sistema 7](#__RefHeading___Toc618_3431357138)

[6. Implantació de la solució 8](#__RefHeading___Toc620_3431357138)

[6.1. Definir les fases d'implantació/migració 8](#__RefHeading___Toc622_3431357138)

[6.2. Inventari de maquinari i programari 8](#__RefHeading___Toc624_3431357138)

[6.3. Diagrames de xarxa, estructura, 8](#__RefHeading___Toc626_3431357138)

[6.4. Migració/Implantació de serveis 8](#__RefHeading___Toc628_3431357138)

[6.5. Formació, comunicació i suport a l'usuari 8](#__RefHeading___Toc630_3431357138)

[6.6. Calendari d'execució 8](#__RefHeading___Toc632_3431357138)

[7. Conclusions i treballs futurs 9](#__RefHeading___Toc634_3431357138)

[8. Bibliografia 10](#__RefHeading___Toc636_3431357138)

# 1. Introducción al proyecto

## 1.1. Descripción del proyecto

El proyecto busca diseñar, desarrollar y desplegar una aplicación para la gestión de tareas diarias y semanales de un área de trabajadores en una empresa especializada en proyectos I+D en el ámbito de la energía. Para ello, el proyecto parte de la situación actual en la empresa, identificando y caracterizando la necesidad organizativa y de uso de la aplicación, con el fin de traducir esa necesidad en unos requisitos de usuario y, por ende, de funcionalidades que debe presentar dicha aplicación para la función para la que es concebida.

La empresa organiza su trabajo por proyectos I+D mediante una estructura matricial de recursos técnicos que se caracteriza por la polivalencia de éstos, existiendo personas con grados diferentes de responsabilidad dependiendo del proyecto. Además, la variedad e incertidumbre de las tareas a realizar requiere enfocar la planificación y el seguimiento del trabajo mediante un enfoque bajo el paraguas, hasta cierto punto, de metodologías ágiles que permitan una respuesta rápida y práctica a la resolución de problemas combinada con un seguimiento incremental de los objetivos de cada proyecto.

Se busca una herramienta multiplataforma para maximizar las facilidades de acceso a cada una de las personas del área en que se pretende implantar en un inicio, inicialmente pensando en terminales móviles y ordenadores de sobremesa o portátil de la empresa. La aplicación se diseñará teniendo en cuenta los diferentes perfiles de usuario que harán uso de ella, de manera que sirva tanto para la organización del trabajo personal de cada uno como para el seguimiento y detección de desviaciones de los objetivos del proyecto. Se deja fuera de este alcance el seguimiento de alto nivel del proyecto (A nivel de milestones

## 1.2. Objetivos

Se pretende conseguir los siguientes objetivos:

* Identificar y definir correctamente las necesidades actuales en la empresa
* Describir unos requisitos de la aplicación acorde a dichas necesidades
* Desarrollar una serie de funcionalidades que cubran las citadas necesidades en base a los requisitos
* Diseñar la aplicación a nivel tanto funcional como estético (Interfaces)
* Definir una arquitectura de la aplicación y herramientas (Lenguajes, frameworks) a emplear
* Diseñar y desarrollar la base de datos necesaria
* Desarrollar las operaciones CRUD
* Desarrollar las interfaces previamente diseñadas, la navegación, etc hasta el punto en que las limitaciones de tiempo y recursos lo permitan

## 1.3. Tipo de proyecto

El proyecto es de tipo interno, ya que se lleva a cabo por la propia empresa para su aprovechamiento directo y sin ningún tipo de explotación prevista, y es un proyecto de programación (Desarrollo de software). Dicho desarrollo abarca tanto la parte de cliente (Frontend) como al de servidor (Backend). La aplicación final a obtener se genera de cero, como alternativa a un método organizativo compuesto por varias aplicaciones que cubren solo parcialmente de los requisitos y necesidades y que, en todo caso, se consideran insuficientes para la realización óptima del trabajo que se prevé, tal y como se detallará más adelante.

## 1.4. Orientaciones para el desarrollo de recursos

Como fuentes documentales principales, se han empleado:

* Los apuntes proporcionados por los profesores de los siguientes módulos del CFGS DAM:
  + Bases de datos (1DAM)
  + Programación (1DAM)
  + Programación de interfaces (2DAM)
  + Programación multimedia (2DAM)
  + Programación de servicios y procesos (2DAM)
* Spring® Framework. Notes for Professionals
* Página oficial documentación Flutter <https://docs.flutter.dev/>

# 2. Análisis del sistema actual

## 2.1. Descripción del sistema actual

En este punto se realiza una revisión del sistema actual con el fin de contextualizar la necesidad de una aplicación para la mejora organizativa, y la estructura que ésta debe seguir en base al método organizativo actual.

Lo primero conviene describir la actividad de la empresa y el sistema actual de organización. La actividad principal de la empresa se da en la ejecución de proyectos I+D en el ámbito de la digitalización y la energía. Para ello, la empresa trabaja en un enfoque a proyectos de períodos variantes entre el corto plazo (semanas o meses) y el largo plazo (Hasta 5 años). El área para la que se diseña la aplicación tendrá previsiblemente en el momento del despliegue 8 empleados, que se pueden clasificar en los siguientes rangos:

* Responsable de área: 1 persona
* Coordinadores de proyecto: 2 personas
* Recursos técnicos: 5 personas

La atribución de tareas y responsabilidades se realiza acorde al siguiente diagrama:

Diagrama

Descripción generada automáticamente

Como se aprecia, a efectos prácticos, el coordinador del área es la persona con mayores atribuciones seguida de los coordinadores de proyecto. Los recursos técnicos se limitan a ejecutar las tareas de proyecto. No obstante, los coordinadores de proyecto también pueden ejecutar tareas en los proyectos, ya sea en los suyos propios o en los de otro coordinador, al igual que el responsable de área. Sin embargo, se considerará a efectos prácticos en las atribuciones de usuario y permisos que el responsable de área tiene las mismas atribuciones que los responsables de proyecto, ya que al nivel organizativo para el que se plantea la aplicación no se considera necesaria una diferenciación entre esos dos niveles de usuario.

Por otro lado, se ha de considerar que el enfoque a Proyectos implica diferentes niveles de organización, tal y como se recoge en la siguiente tabla:

|  |  |  |  |
| --- | --- | --- | --- |
| **Horizonte temporal** | **Acción organizativa** | **Recursos involucrados** | **Herramienta(s) empleada(s) actualmente** |
| Diario | Planificación individual de acciones de cada recurso | Todos | Excel organizativa |
| Semanal / Bisemanal | Planificación y seguimiento de objetivos a corto plazo del proyecto | Todos, en especial Coordinador de proyecto | Microsoft planner, metodología SCRUM, Excels de seguimiento de plazos y costes (GANTT) |
| Mensual | Planificación estratégica y reporte del proyecto a mandos superiores | Coordinador de proyecto y responsable de área | Excel general de seguimiento de costes y plazos |

Es importante recalcar que la aplicación a desarrollar en este proyecto busca suplir enteramente la aplicación actualmente empleada para la organización diaria (que no deja de ser una hoja Excel online), y complementar el uso de aplicaciones de seguimiento semanal y bisemanal, ya que esta aplicación pretende proporcionar seguimiento del cumplimiento de tareas y tangibles asociados (como se explicará más adelante).

A continuación, se adjunta una captura de pantalla de la Excel organizativa a modo de ejemplo, donde por cuestiones de confidencialidad se difuminan las tareas específicas de los proyectos.

Captura de pantalla de computadora

Descripción generada automáticamente

Para más claridad, se expone a continuación el detalle de una casilla diaria de tareas de un trabajador concreto vacía:

Subtotales

Interfaz de usuario gráfica, Aplicación, Tabla, Excel

Descripción generada automáticamente

Los campos son los siguientes:

* **Capacidad diaria**: horas diarias efectivas de trabajo para ese recurso. Puede oscilar entre 7 y 9 (Según la modalidad de horario que siga el trabajador y el día de la semana). En ocasiones puede ser incluso menor, si el trabajador hace menos horas de las que teóricamente debería, sea por razones personales, sea por motivos justificados (Como puede ser, por ejemplo, una visita al médico).
* **Tarea:** En la columna de la izquierda se irá registrando una descripción breve cada tarea.
* **Horas previstas:** horas que la persona estima que dedicará a la tarea ese día. En esta Excel, para tareas no previstas al inicio del día, el trabajador debe registrar un 0 en esta casilla, que la Excel automáticamente resaltará en rojo.
* **Horas imputadas:** horas que la persona ha dedicado finalmente a la tarea ese día. Para tareas previstas, pero no ejecutadas, la persona deberá ingresar en este campo un 0.
* **GAP o desviación:** es una métrica que realiza la diferencia (con signo) entre planificación y dedicación efectiva. Proporciona información de si la tarea ha sido sobredimensionada, o si se ha realizado en más tiempo del previsto.
* Filas de **subtotales:** realizan una suma de cada una de las columnas con los siguientes objetivos:
  + **Horas previstas:** en esta casilla se suma el total de horas indicadas en la columna para obtener el total de horas previstas de trabajo para ese día
  + **Horas imputadas:** en esta casilla se suma el total de horas que se ha indicado que se imputa en la columna para obtener el total de horas que se han realizado de trabajo efectivo ese día. Este total debería coincidir con el de horas previstas, salvo excepciones muy contadas.
  + **Capacidad vs plan:** resta el total de horas previstas a la capacidad diaria indicada. Sirve como indicador de ajuste de la planificación.
* **Tipo**: en esta columna se activa un desplegable de proyectos o actividades generales en las que encuadrar cada tarea, tal y como se muestra a continuación (Los proyectos se han censurado por cuestiones de confidencialidad):

Imagen que contiene Diagrama

Descripción generada automáticamente

Por otro lado, el otro software de referencia que se emplea actualmente es el Microsoft Planner. Para el área para la que se diseña la herramienta de la empresa, esta herramienta se emplea fundamentalmente con un enfoque de planificación y revisión de sprints de trabajo, tomando el concepto de “Sprint” de la metodología SCRUM, si bien no se aplica más que una adaptación simplificada de dicha metodología:

* Sin los roles de SCRUM como tal, siendo únicamente el coordinador de cada proyecto quien ejerce de “Product owner” y de “Scrum master”. El coordinador de proyecto dinamiza y orienta cada reunión de revisión de sprint (Y posterior reunión de planificación del sprint siguiente) para, junto al resto de recursos (Trabajadores técnicos participantes en el proyecto) definir las tareas que pasan a formar parte de la pila de sprint o nuevas tareas a considerar en la pila de producto.
* Empleando pilas de producto únicas o desagregadas, según la necesidad
* Empleando pilas de sprint para cada semana o cada dos semanas, según el proyecto
* Indicando mayor o menor grado de detalle, poniendo o no etiquetas, designando uno o varios responsables para cada tarea, según el caso

A continuación se muestra una captura de pantalla (Parcialmente censurada para nombres de empresa, conceptos del proyecto y personal de la empresa, por cuestiones de confidencialidad= de la citada herramienta para un proyecto (Plan) concreto:

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Como se ha justificado antes, no se pretende sustituir con la herramienta a desarrollar en este proyecto el uso del Planner, que además está implantado de manera más o menos extendida a toda la empresa, sino que el diseño de ésta se orientará a un uso complementario del Planner, tal y como se describirá más adelante.

A continuación, se muestran un par de diagramas de las diferentes herramientas software que se emplean para la gestión de la actividad operativa de la empresa junto a la jerarquía a nivel de gestión de datos en que se ubican:

Diagrama

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

## 2.2. Viabilidad del sistema actual

El actual sistema tiene una serie de limitaciones que se detallan en la siguiente tabla:

* Cada nuevo mes se debe generar una nueva hoja Excel manualmente adaptando el número de días y fechas concretas, y eliminando los fines de semana
* Cada persona tiene acceso a toda la información del resto sin ninguna limitación
* No existen restricciones a la hora de indicar horas o tareas
* La detección de tareas no previstas es muy básica, debiendo la Excel detectar las casillas de “Horas previstas” puestas a 0.
* La detección de desviaciones puede ser enmascarada por tareas planificadas, pero no realizadas, que tienen igualmente una desviación o gap positivo. En consecuencia, no se puede distinguir cuando una tarea no se ha realizado y cuando se ha realizado una tarea en menos tiempo del previsto. Se tiene un problema igual en el caso contrario; cada tarea no prevista pero realizada genera una desviación negativa que la hace indistinguible de una tarea que se realice en más tiempo del previsto.
* La explotación de los datos y cálculo de indicadores es ardua y todavía no se ha implementado en la Excel por falta de tiempo, sobre todo a la hora de cuantificar tareas previstas, pero no realizadas, tareas no previstas, porcentaje de tareas por proyecto, etc…
* Solo se indican tareas de cada proyecto, pero no se indican de qué manera generan incremento en el valor del proyecto. En otras palabras; las tareas se presentan de una manera aislada, sin contextualizar su importancia en el proyecto y como contribuyen a mejorar los objetivos marcados por éste.

Por otro lado, la propia herramienta es la hoja Excel que hace de base de datos, de manera que ante cualquier problema con ésta (Borrado accidental, baja de la empresa de la persona que está alojando dicha hoja Excel en su OneDrive, corrupción del archivo...) se perdería la herramienta y todo el consecuente registro de tareas. Es uno de los principales aspectos a mejorar la persistencia de esos datos.

## 2.3. Requerimientos del nuevo sistema

Los requisitos que debe cumplir el sistema son los siguientes:

* Debe ser **multiplataforma**, para facilitar el acceso desde el ordenador o cualquier terminal móvil, dado que en ocasiones el personal técnico se encuentra desplazado fuera del centro de trabajo y no tiene acceso al portátil (Puestas en marcha, visitas a planta…), de manera que el hecho de que se pueda acceder también desde el móvil es una ventaja
* Debe permitirse un **acceso autentificado** con diferentes niveles de usuario y proporcionando funcionalidades específicas según el nivel de perfil de acceso. Se consideran 3 niveles: coordinador de área, coordinador de proyecto y personal técnico general.
* Los datos se **almacenarán** en una base de datos única y consistente, de manera que la definición de tangibles o proyectos / conceptos sea única. Ciertos campos, como este tipo de conceptos, solo deben poder ser definidos por el coordinador de área, que actúa a modo de superadmin
* Debe **replicarse** el **sistema de registro de tareas** que ya presenta el sistema actual añadiendo campos como el tangible asociado o una casilla para indicar si se trata de un evento o reunión
* Los **colores** principalmente empleados en la aplicación serán los corporativos de la empresa, que son los siguientes:

Imagen que contiene Texto

Descripción generada automáticamente

Imagen que contiene Texto

Descripción generada automáticamente

* La aplicación debe incluir un espacio para mostrar **indicadores** tanto estadísticos como de rendimiento de cada persona. Dado que la aplicación empleada actualmente no dispone de esos indicadores, es tarea también de este trabajo diseñar los indicadores óptimos para el seguimiento efectivo del grado de consecución de las tareas, desviaciones, carga de trabajo, etc.
* El **acceso y uso de los datos** es fundamental, y la herramienta debe implementar una base de datos relacional consistente tipo Mysql (o similar),por la siguientes razones:
  + Estructura de datos clara, con entidades como los tangibles que pueden ser compartidos por varias personas, y atributos, como las horas de realización de una tarea, etc
  + Se necesita una integridad de los datos para que todas las tareas se almacenen correctamente
  + La estructura debe ser fácilmente escalable si finalmente se decide extender su uso a otras áreas o recursos
* La interfaz gráfica de la aplicación debe ser atractiva para fomentar la adopción en toda la empresa y garantizar la usabilidad

# 3. Análisis de la solución

## 3.1. Análisis de las posibles soluciones

Se van a listar las posibles soluciones que resolverían los requerimientos de la aplicación, para lo cual se ha tenido en cuenta la experiencia en distintos lenguajes del alumno que desarrolla e implementa la aplicación:

1. Desarrollo de toda la aplicación en Python puro, empleando como librerías frontend Flet (Librería más reciente pero sencilla y muy declarativa), PyQt / Pyside o incluso Tkinter (Aunque en ésta el alumno no tiene experiencia). Si quisiéramos que la aplicación fuera web (Que, a fin de cuentas, sería multiplataforma en el sentido que de podría accederse desde un terminal móvil sin problema), se podría combinar un entorno de desarrollo como Django o Flask con el uso de lenguajes frontend tipo javascript, combinado con HTML y CSS.
2. Desarrollo de backend en java y frontend en Python: aunque es poco común, combinando librerías de interfaz gráfica de Python con la gestión de las solicitudes llevadas a cabo por el propio frontend, se harían las peticiones a la API de Springboot, siendo éste último quien gestione las peticiones a la base de datos.
3. Combinando java para el backend y javascript para el frontend. Uno de los enfoques más comunes, empleando Springboot para el backend y frameworks de referencia como React o Angular para el frontend. La aplicación podría incluso desarrollarse fullstack en javascript mediante Nodejs.
4. Desarrollo backend en java y frontend con flutter(Dart). Esta opción también es cada vez más común dado la capacidad que tiene el framework Flutter para compilar a iOS y Android, así como para ejecutarlo en navegadores web, asegurando una aplicación realmente multiplataforma y aprovechando lo completo que es el propio Flutter.

## 

## 3.2. Evaluación de las posibles soluciones

A continuación, se muestra el DAFO de cada una de las cuatro soluciones propuestas:



## 

## 3.3. Descripción de la solución escogida

Finalmente se decide escoger la Opción 4, por la experiencia del alumno (Que ya ha realizado algunas aplicaciones sencillas tanto con Springboot como con Flutter por separado durante el curso) así como por la oportunidad de profundizar en dos utilidades de alta demanda actualmente, tanto para programadores frontend, como backend y fullstack.

Como fortalezas de **Springboot**; el uso de lombook, la facilidad de programar las operaciones CRUD de interacción con la base de datos o la gestión de usuarios son algunos de los aspectos diferenciadores.

Como fortalezas de **Flutter**; su carácter multiplataforma, la potencia de Dart como lenguaje, la disponibilidad de widgets y de la biblioteca Material Design de Google (Implementada mediante material.dart), y

Como resumen, la solución se compondrá de los siguientes elementos:

* **Base de datos SQL**, donde se almacenará toda la información sobre usuarios, tareas, estado de las tareas, indicadores, tangibles, proyectos, etc. Para desarrollarla se empleará MySQL workbench y para testearla se empleará previsiblemente Docker.
* **Backend Java mediante Springboot,** creando así un proyecto tipo Spring donde gestionaremos los accesos a los datos, la seguridad, los usuarios y la API REST. Para desarrollarlo y testearlo se empleará Spring Tool suite.
* **Frontend Dart mediante Flutter,** implementando toda la interfaz de usuario, navegación, y las peticiones a la API REST de Java.

# 4. Diseño de la solución.

## 4.1. Requisitos.

Como ya se introducía previamente, la aplicación tiene dos requisitos funcionales principales:

* Permitir un seguimiento efectivo de las **tareas** diarias que cada recurso se propone realizar en un contexto de ejecución dentro de **proyectos** con **tangibles** (resultados, objetivos materializados) específicos
* Realizar un análisis de **rendimiento**, **desviaciones** y otros **indicadores** de interés, tanto a nivel individual (Cada técnico sobre sus propias tareas) como a nivel agregado (Cada coordinador de proyecto de sus propios proyectos y cada coordinador de área de todos los recursos de su área)

Estos requisitos se traducen, a su vez, en una serie de requisitos de detalle que la herramienta debe cumplir:

***Requisito 1****. Permitir un seguimiento efectivo de las* ***tareas*** *diarias que cada recurso se propone realizar en un contexto de ejecución dentro de* ***proyectos*** *con* ***tangibles*** *(resultados, objetivos materializados) específicos*

* **Registrar o eliminar tareas** diarias a realizar. Cada tarea será única de cada usuario con el fin de evitar registros de tareas demasiado extensos sobre los que seleccionar
* Asociar a cada tarea una cantidad de **horas prevista** y una cantidad de **horas dedicadas**, y en base a la planificación global del día, proporcionar información sobre la sobre o infraasignación del recurso según la **capacidad diaria** que éste indique
* Permitir, en la definición de cada tarea, que ésta sea asociada a un **proyecto** (P.ej. Proyecto LAMBDA[[1]](#footnote-1), Proyecto NEMESIS [[2]](#footnote-2), Oportunidades) y a un **tangible** (P.ej. Informe de Inicio, modelo de correlación). Ambas entidades (Proyecto y tangible) deben tener instancias compartidas, es decir, tanto a cada proyecto como a cada tangible, los usuarios deberán seleccionar sobre una En este sentido, por seguridad, se definen un par de aclaraciones:
  + Los proyectos serán dados de alta en el sistema por el coordinador de área. Cuando un proyecto sea dado de alta no podrá ser eliminado
  + Los tangibles serán dados de alta por cada uno de los recursos, pero no podrán ser eliminador por ellos, sino por el coordinador de área
* Durante el día en curso, el usuario podrá **añadir o borrar cuantas tareas** **desee** sin problema. Cuando se dé de alta una tarea se ha de indicar el tiempo previsto, el proyecto y el tangible al que se asocia. Se pueden prever otros campos, como por ejemplo indicar si la tarea es una reunión en sí misma, con el fin de poder hacer posteriores análisis del porcentaje de tiempo dedicado a reuniones, por poner un ejemplo.
* Al finalizar cada día, a las 23:59, **las tareas de dicho día quedarán bloqueadas** sin poder modificarse, de manera que solo podrá acceder en “modo visualización” para consultar los datos que dio de alta cada día pasado
* Para cada tarea, se generará un **valor de desviación** entre el tiempo que inicialmente se tenía previsto para la tarea y el tiempo finalmente dedicado. Se debe valorar incluir asimismo campos que permitan afinar mejor el seguimiento y la representatividad de las tareas que se registran, como por ejemplo un campo de “tarea no prevista” o un campo de “tarea finalizada”, que sean simplemente indicativos pero permitan hacer un mejor análisis de los datos.

***Requisito 2.*** *Realizar un análisis de* ***rendimiento****,* ***desviaciones*** *y otros* ***indicadores*** *de interés, tanto a nivel individual (Cada técnico sobre sus propias tareas) como a nivel agregado (Cada coordinador de proyecto de sus propios proyectos y cada coordinador de área de todos los recursos de su área)*

Parte de ese análisis del rendimiento ya se lleva a cabo con el valor de desviación que se indica para cada tarea, indicador que ya proporciona información sobre lo bien que planifica cada usuario sus tareas. Partiendo de ese punto, se debe contemplar que el sistema cumpla los siguientes requerimientos de detalle:

* Proporcionar un **análisis básico de grado de rendimiento individual** a cada usuario de la aplicación en base a sus tareas como individuo. Para ello, se deben realizar algunos cálculos estadísticos básicos (Ya que los framework escogidos tampoco son especialmente fuertes en análisis y representación avanzada de datos). Algunos de los indicadores que se propone calcular son:
  + Cantidad de tareas mensuales completadas
  + % Mensual de cumplimiento de tareas
  + Cantidad de tareas no previstas
  + % de tiempo mensual de tareas no previstas

Estos indicadores podrán ser, no obstante, revisados según la complejidad de implementación y dificultad de obtención de datos, e incluso ampliados si se detectan mejoras o métricas específicas de mayor interés. No obstante, todos los indicadores se calcularán a partir del marco de datos (Cantidad y tipo de datos) especificados en el el requisito 1.

* Proporcionar un **análisis general de rendimiento por recurso**, con indicadores de seguimiento general tanto a nivel de todos los recursos (Datos agregados) como a nivel de cada usuario de la aplicación (Rendimiento individual). Estos datos serán de interés, sobre todo, para el Coordinador de área, que será el único que tenga acceso a ellos. Se podrán designar varios coordinadores de área.
* Proporcionar un **análisis general de rendimiento por recurso y proyecto**. De manera similar al anterior pero acotado a todos los recursos de un proyecto dado, siendo información accesible solo para el Coordinador de proyecto. Se podrán designar varios coordinadores de proyecto.

## 4.2. Análisis de riesgos.

A continuación, se exponen los principales riesgos detectados, clasificados por nivel de importancia, desde alto riesgo hasta bajo riesgo:

|  |  |  |
| --- | --- | --- |
| Riesgo | Descripción | Nivel de importancia |
| Complejidad de la lógica de cálculo | En lo que vendría a definirse como “lógica de negocio” para calcular algunos indicadores es probable que haya que hacer análisis con orientación ligeramente estadística de un gran número de tareas. La complejidad de esta lógica puede determinar la correcta funcionalidad de la aplicación. | **Medio**. La lógica debe ser definida de manera sencilla incluso en detrimento de no poder calcular algunos indicadores, ya que gran parte de la consistencia de la aplicación está asociada a la capacidad de registro y seguimiento y no tanto a la explotación de los datos de rendimiento. |
| Desbordamiento de tareas del sistema | Dado que cada usuario tiene un número potencialmente infinito de tareas que poder registrar (Las tareas se las genera cada usuario según su criterio propio), se pueden dar casos en que el sistema tenga una gran cantidad de tareas registradas si se emplea durante mucho tiempo por parte de muchos usuarios. | **Bajo**. La cantidad de usuarios prevista inicialmente (9) no hace pensar que en el medio plazo vaya a haber una cantidad excesiva de tareas registradas en sistema |
| Diseño y estructuración incorrecta de la base de datos | Un diseño incorrecto de la base de datos, que contiene gran parte del valor añadido de la aplicación (Cuyas funcionalidades principales se orientan a registrar correctamente y dar seguimiento a esos datos) puede llevar a que la aplicación no funcione todo lo bien que se espera. | **Medio**. El diseño debe considerar el uso final de la aplicación y la base de datos, relaciones entre entidades (Usuarios, proyectos, tangibles, tareas…) deben ser consistentes y coherentes a la necesidad |
| Desarrollo excesivo | Es posible que el tiempo de desarrollo sea mayor del previsto inicialmente, tanto por la falta de práctica del alumno en el desarrollo fullstack de aplicaciones, como por la falta de experiencia en la puesta en producción de estas. | **Alto**. Debe considerarse como escenario realista implementar sobre todo la primera funcionalidad prevista para la aplicación (Registro y visualización de tareas). Según el grado de avance, se ajustará el alcance de implementación de la segunda (KPIs y seguimiento mensual) |

## 4.3. Casos de uso.

A continuación, se muestra el diagrama de casos de uso:

Diagrama

Descripción generada automáticamente

## 4.4. Diagramas

A continuación, se muestran algunos diagramas y documentación gráfica de ayuda. El primero de ellos es el prototipo de baja-media fidelidad sobre el que se orienta el desarrollo de la interfaz de usuario:

**Pantalla 1: Login**

*Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente*

**Pantalla 2: Tareas**

*Interfaz de usuario gráfica

Descripción generada automáticamente*

*Calendario

Descripción generada automáticamente*

*Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente*

*Interfaz de usuario gráfica

Descripción generada automáticamente*

*Interfaz de usuario gráfica

Descripción generada automáticamente*

*Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente*

# 5. Desarrollo de la solució

Ara ens centrem ja en la part grossa del projecte, el desenvolupament de la solució. Una vegada tenim clar el què volem, anem a detallar les diferents fases i què anem fent en cadascuna d’elles.

## 5.1. Estimación de coste temporal

Una vegada tingueu clara la solució que aneu a implantar, caldrà veure quines són les diferents tasques i subtasques necessàries per aplicar aquesta solució, establireu el cost temporal de cadascuna (quina durada tindrà). En aquest apartat, podeu elaborar un diagrama de Gantt on apareguen relacionades aquestes tasques, i/o un calendari per al desenvolupament de les diferents fases. Finalment, també caldrà fer, si és el cas, una estimació del cost econòmic que aquest tindrà, mitjançant un pressupost o estimació de despeses.

## 5.2. Dotación de recursos

Primerament, caldrà veure de quins recursos disposem inicialment per a la realització del projecte: Equips de desenvolupament, maquinari, propotips, etc.

## 5.3. Configuración i/o desarrollo del sistema

La part central del projecte és aquesta, on establireu com s’ha desenvolupat i configurat el nou sistema: automatització de processos, scripts, empaquetat, configuracions, etc.

## 5.4. Evaluación del sistema

Una vegada finalitzat el desenvolupament, es realitzarà una avaluació per veure si compleix amb els requisits que ens haviem plantejat i millora el sistema anterior.

# 6. Implantació de la solució

Una vegada tenim la solució, passem a implantar aquesta en l’organització o entitat per a la què s’haja realitzat. En cas que es tracte d’un projecte propi, cadria indicar com podríen posar-lo en marxa en qualsevol organització, mitjançant guíes d’instal·lació i manuals d’usuari.

## 6.1. Definir les fases d'implantació/migració

Indicarem en quines fases es dividirà la implantació del sistema nou. Generalment, en organitzacions mitjanes i grans, no s’implanta tot el sistema d’una, sinò que es fa per departaments o seccions, de manera que cal planificar aquesta migració.

## 6.2. Inventari de maquinari i programari

Indicarem el maquinari i programari de què disposarem.

## 6.3. Diagrames de xarxa, estructura,

Com queda la xarxa estructurada després de la posta en funcionament de la solució.

## 6.4. Migració/Implantació de serveis

Indicarem els passos per tal de realitzar la migració o implantació dels diferents serveis.

## 6.5. Formació, comunicació i suport a l'usuari

Caldrà pensar en la formació que es dóna als usuaris i el suport que aquests tindran.

## 6.6. Calendari d'execució

Finalment, caldrà indicar quin serà el calendari d’implantació de les diferents fases de la solució.

# 7. Conclusions i treballs futurs

Com a últim apartat, deixarem un apartat per indicar, a mode de resum què s’ha fet en el projecte, així com possibilitats de millora o futures ampliacions que aquest podrà tindre.

# 8. Bibliografia

1. Nombre ficticio [↑](#footnote-ref-1)
2. Nombre ficticio [↑](#footnote-ref-2)