



UNIVERSIDAD DE GRANADA

GRADO EN INGENIERÍA INFORMÁTICA

**Trabajo Fin de Grado — Aplicación de gestión de
rutinas semanales basada en tecnologías web**

Juan Ocaña Valenzuela

15 de agosto de 2021

Índice general

1. Introducción	2
1.1. Resumen	2
1.2. <i>Abstract</i>	3
1.3. Integración de conocimientos previos	4
2. Estado del arte	5
2.1. Crítica al estado del arte	5
2.2. Propuesta	7
2.2.1. Objetivos	7
3. Metodología	10
3.1. Metodología de desarrollo	10
3.2. Análisis de requisitos	10
3.3. Planificación temporal	11
3.3.1. Planificación inicial	11
3.3.2. Planificación final	11
3.4. Presupuesto	12
4. Desarrollo del proyecto	14
4.1. Primera iteración	14
4.1.1. Análisis	14
4.1.2. Diseño	14
4.1.3. Implementación	14
4.1.4. Pruebas	14
5. Conclusiones y vías futuras	15
5.1. Conclusiones del proyecto	15
5.2. Posibles modificaciones futuras	15
Bibliografía	16

Capítulo 1:

Introducción

Palabras clave: palabras clave

§1.1: Resumen

NOTA: provisional

Existe una gran oferta en cuanto a herramientas de organización personal. Desde activas comunidades que apuestan por formatos tradicionales como agendas y calendarios personalizados hasta numerosos servicios preparados para ajustarse a cualquier tipo de persona, escoger un método de organización puede suponer una odisea para alguien interesado en la materia.

Muchas de las herramientas más populares en este campo optan por ofrecer un servicio de suscripción —a menudo con versión gratuita—, que aporta ventajas tales como la sincronización de los datos del usuario en la nube o integración con servicios adicionales. Sin embargo, a veces estas opciones son demasiado complejas para algunas personas, que simplemente buscan algo más sencillo o acorde a sus necesidades.

Se plantea una propuesta de aplicación con una funcionalidad simple, orientada a asistir en la creación de rutinas y hábitos, utilizando como soporte un sistema de objetivos y una estructura de horario semanal. La idea principal es brindar una funcionalidad simple a la vez que intuitiva, haciendo énfasis en la facilidad de uso, optando por no incluir herramientas complejas, tales como un sistema de calendario o un bloc de notas —opciones que muchos de los servicios mencionados anteriormente implementan—.

La aplicación propuesta se ha desarrollado utilizando tecnologías web extendidas y con gran soporte en la actualidad, y una arquitectura cliente-servidor basada en el patrón *REST* [5].

§1.2: *Abstract*

Una vez esté aprobado el resumen, traducir.

§1.3: Integración de conocimientos previos

Para la realización de este proyecto, han sido útiles los conocimientos adquiridos en las siguientes asignaturas del grado:

- Fundamentos de Bases de Datos
- Fundamentos de Ingeniería del Software
- Ingeniería de Servidores
- Sistemas de Información Basados en Web
- Dirección y Gestión de Proyectos
- Metodologías de Desarrollo Ágil

Capítulo 2:

Estado del arte

§2.1: Crítica al estado del arte

El mercado de aplicaciones de organización personal está repleto de opciones, con algunas de ellas muy populares. Las aplicaciones estudiadas pueden dividirse en las siguientes categorías, según su principal enfoque en cuanto a la organización personal

Aplicaciones de agenda y calendario

Las aplicaciones de agenda y calendario son las más populares, debido a su enfoque principalmente profesional. Se trata de herramientas completas y potentes, con capacidad para gestionar diferentes calendarios simultáneos, recordatorios, exportación a distintos formatos e integración con otros servicios —como por ejemplo es el caso de Google Calendar, en el que se pueden planificar citas a través de *Gmail*, o adjuntar salas de videoconferencia de *Google Meet*—.

Una de las principales ventajas de los principales servicios de este tipo es su gratuidad. Al estar mantenidos en su mayoría por grandes empresas tecnológicas —Google, Microsoft, Apple, etc.—, no requieren de un esfuerzo adicional para el usuario del resto de sus respectivas aplicaciones.

Estos servicios, pese a estar preparados para un uso intensivo y profesional, también se adaptan al usuario medio en mayor o menor medida. No obstante, su complejidad puede resultar abrumadora, y ante todo, suplen necesidades específicas: una agenda y un calendario tradicionales.

Algunas de las aplicaciones de calendario más populares o influyentes son las siguientes:

- Google Calendar
- Microsoft Outlook
- Apple Calendar (anteriormente llamado *iCal*)

Aplicaciones de tareas

Otro enfoque popular a la hora de ofrecer herramientas de organización personal es el de las aplicaciones de tareas. Estos servicios ofrecen la posibilidad de planificarse a través de listas de tareas, a menudo con añadidos como notificaciones y recordatorios, similares a los presentes en las aplicaciones más tradicionales anteriormente mencionadas.

Estas aplicaciones utilizan la **tarea** como elemento base, construyendo todo su funcionamiento en torno a ella, y poniendo el foco en facilitar la interacción del usuario con la misma. Pese a estar presentes en entornos profesionales, muchas de estas aplicaciones se orientan hacia el usuario medio en primer lugar, ofreciendo una experiencia más sencilla e intuitiva.

No obstante, muchos de estos servicios cuentan con una funcionalidad limitada, a menudo como expositor de una suscripción; y aquellos que no disponen de esta opción suelen requerir de un esfuerzo adicional por parte del usuario —por ejemplo, *Tasks.org* [tas]—.

Algunos ejemplos de aplicaciones de tareas son las siguientes:

- any.do
- Todoist

- Tasks.org

Aplicaciones de espacio de trabajo

Existen aplicaciones cuya premisa es ofrecer al usuario diversas herramientas para crear su método de organización, con todas las ventajas de un soporte digital. Estas aplicaciones suelen brindar al usuario sistemas de bases de datos sencillos y fáciles de manipular, con tal de organizar notas y documentos compuestos de distintos módulos preestablecidos. No existe un estándar para las aplicaciones de este tipo, y cada servicio define su propio sistema —por ejemplo, *Notion* centra sus esfuerzos en las bases de datos, mientras que *Evernote* lo hace en un completo sistema de notas—.

Pese a que este tipo de aplicaciones ofrecen una gran versatilidad, requieren un esfuerzo adicional por parte del usuario para comprender y aprender a utilizar las herramientas a su disposición.

Algunas aplicaciones de este tipo son las siguientes:

- Notion
- Evernote
- Microsoft OneNote

Aplicaciones de seguimiento de hábitos

Por último, existe una categoría de aplicaciones de organización enfocadas al seguimiento de diferentes objetivos, ya sean definidos por el usuario o preestablecidos en base a unas pautas concretas.

Dichas aplicaciones, mayoritariamente publicadas para dispositivos móviles, permiten registrar los avances del usuario en diferentes tareas, contabilizar el progreso y presentarlo en distintos formatos. Muchas de estas aplicaciones implementan sistemas de ludificación [4] —también llamada *gamificación*—, utilizando logros, puntos y otros mecanismos similares para mantener el interés del usuario.

Un aspecto relevante de estos servicios es que suelen centrarse en temáticas específicas, como el deporte, hábitos saludables, productividad o registro del estado de ánimo, por poner algunos ejemplos. Casi todos ellos siguen orientados a construir rutinas, ya sean diarias, como mantener un horario de sueño constante, hidratarse o regar las plantas, o con diferentes plazos —semanales, mensuales, personalizados, etc.—, pero la forma de presentar la información al usuario es muy diferente.

Debido a las grandes diferencias entre las aplicaciones de seguimiento de hábitos estudiadas, es difícil definir un usuario objetivo claro. Están orientadas a un perfil personal, pero muchas de ellas tratan de implementar soluciones a problemas demasiado específicos y abusan de la ludificación, volviendo su uso tedioso.

Algunas aplicaciones populares en este ámbito son las siguientes:

- Rabbit
- Dailyo
- Habit (*Leap Fitness Group*)
- Habitica

Crítica

No faltan soluciones de este tipo en el mercado. Desde un ámbito más profesional hasta el usuario más ajeno a la tecnología, existe una gran variedad de aplicaciones para organizarse, con diversos enfoques y objetivos.

Pese a lo completas y maduras que son las aplicaciones más tradicionales, su enfoque profesional y multitud de funcionalidades pueden suponer un obstáculo para un perfil de usuario que no sólo no va a beneficiarse de todas las integraciones y ventajas que ofrecen, sino que no las necesita.

Las aplicaciones de tareas suplen este exceso de complejidad con un planteamiento básico y suficiente, añadiendo funciones complementarias, a menudo similares a las de servicios tradicionales, pero adaptadas al flujo de la lista de tareas. No obstante, muchas de estas funciones requieren de una suscripción, estando las aplicaciones muy limitadas en su versión gratuita al no disponer de una infraestructura tan grande.

Las aplicaciones de espacio de trabajo son herramientas muy potentes, que en manos de usuarios experimentados pueden resultar muy útiles y servir multitud de propósitos. Muchas de ellas ofrecen plantillas predefinidas para usos comunes, y permiten establecer sistemas de organización a medida, pero suelen centrarse en colecciones de datos en lugar de eventos y tareas, y aunque puedan definirse fechas, notificaciones y, en algunos casos, calendarios, suelen ser atributos secundarios, que necesitan trabajo previo por parte del usuario para funcionar.

Por último, las aplicaciones de seguimiento de hábitos siguen un propósito específico, alejado del resto de sistemas. Su especificidad hace que este tipo de aplicaciones sean fáciles de entender con respecto a otras soluciones. No requieren de gran configuración previa, su función es limitada pero concreta, y sirven como apoyo al usuario para conseguir sus objetivos y ver su progreso de forma sencilla, independientemente de los medios que utilicen para llevarlos a cabo. No pretenden ser un sustituto de otros servicios como las aplicaciones de calendario o espacios de trabajo, y pueden llegar a complementarse muy bien.

No obstante, pese a que gozan de gran popularidad y variedad, muchas de ellas ofrecen incentivos y mecanismos tan distintos al usuario que pueden resultar desde poco interesantes a abrumadoras. Además, el soporte de muchas de estas aplicaciones comparadas con el resto de casos estudiados suele ser escaso, o depender de servicios externos a configurar por el usuario. Esto último no es necesariamente algo negativo, pero puede inclinar al usuario a explorar otras opciones.

§2.2: Propuesta

2.2.1: Objetivos

Se plantean los siguientes objetivos para la aplicación propuesta. Pueden dividirse en varias secciones, cada una con sus propios objetivos anidados de diferente prioridad:

- La aplicación debe permitir registrar, modificar y eliminar objetivos definidos por el usuario, permitiendo una visualización clara y detallada de su progreso, e incentivar el seguimiento de los mismos.
- La aplicación debe permitir registrar, modificar y eliminar actividades habituales que realiza el usuario, y mostrar un horario semanal con las mismas, de forma que el usuario pueda visualizar de forma clara su tiempo.
- La aplicación debe permitir registrar, modificar y eliminar categorías definidas por el usuario, para que pueda organizar sus objetivos y actividades de forma temática, así como visualizar su progreso de la misma manera.
- La aplicación debe mostrar un panel de navegación entre las diferentes vistas, con una lista de categorías definidas por el usuario, un indicador de fecha y hora, y acceso a las opciones de configuración, así como sugerencias de objetivos a completar.
- La aplicación debe permitir configurar aspectos como el día de la semana en el que se reinician los objetivos definidos por el usuario, la vista por defecto y el tema de la aplicación, y la zona horaria.
- La aplicación debe cumplir con los niveles de conformidad WCAG... (investigar)

Sección de objetivos

Fundamentales

1. Se debe poder crear, consultar, editar y eliminar objetivos.
2. Se debe poder registrar una entrada de un objetivo, y ver reflejado el progreso del mismo.
3. Se debe poder ver una lista de objetivos, que permita ser filtrada por estado de finalización.
4. El progreso de los objetivos debe reiniciarse en el día de la semana establecido.

5. Se debe mostrar una racha de finalización para cada objetivo, que indique el número de veces consecutivas en las que se ha completado. Tras completar un objetivo, se ha de incrementar dicha racha.
6. Se debe guardar la máxima racha alcanzada para cada objetivo. En caso de ser superada, se actualizará debidamente.
7. Se debe mostrar una vista con información general sobre los objetivos definidos, que incluya una lista de los mismos, detalles sobre las rachas, y un historial de entradas.

Opcionales

8. Se debe permitir pausar un objetivo indefinidamente. Esto lo excluirá de la lista de objetivos activos, y no se podrá introducir nuevas entradas del mismo hasta que no sea reanudado. La racha actual se guardará.
9. Se debe poder definir diferentes frecuencias al crear objetivos, que determinarán el reinicio del progreso de los mismos. Se podrá seleccionar entre frecuencia semanal y mensual.

Sección de actividades

Fundamentales

10. Se debe poder crear, consultar, editar y eliminar actividades.
11. Se debe mostrar una vista con el horario de la semana en formato de lista de actividades.

Opcionales

12. Se debe poder ver una lista de las actividades pertenecientes a una categoría en la vista de resumen de la misma.

Sección de categorías

Fundamentales

13. Se debe poder crear, consultar, editar y eliminar categorías.
14. Se debe poder asignar una categoría a objetivos y actividades.
15. Se debe mostrar una vista resumen de cada categoría, similar a la vista general de objetivos, con información relevante sobre los objetivos de la misma.

Sección de panel de navegación o *Vista Rápida*

Fundamentales

16. Se debe mostrar la fecha y hora actuales.
17. Se deben mostrar varios objetivos pendientes de finalización como sugerencias al usuario, para incentivarlo a completarlos en caso de no saber qué hacer.
18. Se debe mostrar un menú de navegación con las diferentes vistas principales de la aplicación para poder acceder a ellas.
19. Se debe mostrar un acceso a la configuración de la aplicación.

Opcionales

20. Se debe poder ocultar y expandir el panel de navegación.

Sección de configuración

Fundamentales

21. Se debe poder ajustar el día de la semana y la hora en la que se reinicia el progreso de los objetivos.
22. Se debe poder definir la zona horaria de la aplicación.
23. Se debe poder seleccionar la vista por defecto al entrar en la aplicación.

Opcionales

24. Se debe poder seleccionar el tema de la aplicación.
25. Se debe poder acceder a la configuración de las categorías desde los ajustes generales.

Sección de accesibilidad

Fundamentales

26. Se debe cumplir el estándar WCAG nivel ... (investigar)

Capítulo 3:

Metodología

§3.1: Metodología de desarrollo

Para la elaboración del proyecto se ha planteado utilizar aspectos sólidos de diversas metodologías ágiles, como *SCRUM* [6] o *Kanban* [kan]. Al estar planteadas originalmente para equipos —generalmente pequeños—, ha sido necesario adaptarlas a un modelo de desarrollo individual. Se han seguido los principios del Manifiesto Ágil [3].

Los elementos a destacar de la metodología desarrollada son los siguientes:

Tablero *Kanban*

Utilizado para representar los posibles estados de cada tarea de forma visual.

Al ser *Kanban* un marco flexible, se han definido estados o columnas acordes a las necesidades del proyecto. Dado que el desarrollo ha sido llevado a cabo por un individuo, se puede prescindir de estados intermedios, como por ejemplo «pendiente de aprobación por el resto del equipo».

Los estados definidos para este proyecto son los siguientes:

- **PENDIENTE**
- **DISEÑO**
- **ELABORACIÓN DE TESTS**
- **IMPLEMENTACIÓN**
- **HECHO**

Desarrollo iterativo e incremental

Presente en metodologías como *SCRUM* o *eXtreme Programming*, el desarrollo iterativo permite tener una visión temprana y real del proyecto, así como una mayor flexibilidad a la hora de detectar e implementar cambios y mejoras.

Al tratarse de un proyecto centrado en el aprendizaje de nuevas tecnologías, esta flexibilidad ha permitido una mayor creatividad y un enfoque centrado en la implementación de la aplicación, utilizando documentación auxiliar en los casos necesarios.

Historias de Usuario

Las historias de usuario

§3.2: Análisis de requisitos

Listado de las historias de usuario

§3.3: Planificación temporal

3.3.1: Planificación inicial

La planificación que se hubiese seguido en condiciones ideales

3.3.2: Planificación final

Debido a diversos factores, principalmente externos al proyecto, la planificación inicial no pudo llevarse a cabo. Se analizó el estado del proyecto, y se decidió reiniciar la fase de desarrollo con una nueva planificación, ajustada esta vez a una fecha de entrega posterior.

Distribución de etapas del proyecto en el tiempo

Con tal de evitar los posibles imprevistos —algo que, como ya se ha podido observar, ha afectado con severidad al proyecto anteriormente—, la nueva planificación se ha realizado de forma exhaustiva, presentando un mayor número de iteraciones, ajustadas de antemano con un listado de funcionalidades concretas a implementar.

Las etapas del proyecto quedaron distribuidas de la siguiente manera:

1. Investigación previa: elección de tecnologías a utilizar, revisión del estado actual del proyecto y estimación del coste de los cambios necesarios.

Plazo: del 1 al 31 de mayo.

2. Configuración inicial del proyecto: preparación de las tecnologías elegidas y todo lo necesario para comenzar a implementar la aplicación.

Plazo: del 22 de mayo al 7 de junio.

3. Iteraciones de desarrollo o *sprints*:

- 3.1. Iteración 1: del 8 al 22 de junio.
- 3.2. Iteración 2: del 23 de junio al 7 de julio.
- 3.3. Iteración 3: del 7 al 21 de julio.
- 3.4. Iteración 4: del 22 de julio al 5 de agosto.
- 3.5. Iteración 5: del 6 al 20 de agosto.
- 3.6. Iteración 6: del 21 al 31 de agosto.

La distribución de las etapas en el tiempo se puede consultar en el siguiente diagrama de Gantt:

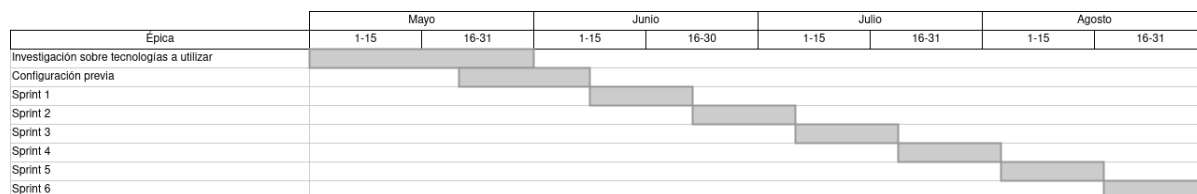


Figura 3.1: Diagrama de Gantt de las etapas del proyecto

Distribución de las tareas a realizar en las etapas propuestas

A la hora de distribuir las tareas en las distintas iteraciones, se realizó una estimación de la complejidad y relevancia de cada una de las épicas del proyecto, y se planteó su implementación de forma mayormente secuencial. No obstante, la dependencia entre algunas funcionalidades obligó a adelantar el desarrollo de historias de usuario pertenecientes a épicas menos prioritarias. La última iteración se ha dedicado a realizar ajustes menores, por lo que presenta tareas de todas las épicas.

La distribución aproximada de las épicas en las iteraciones de desarrollo se puede ver en este diagrama de Gantt:

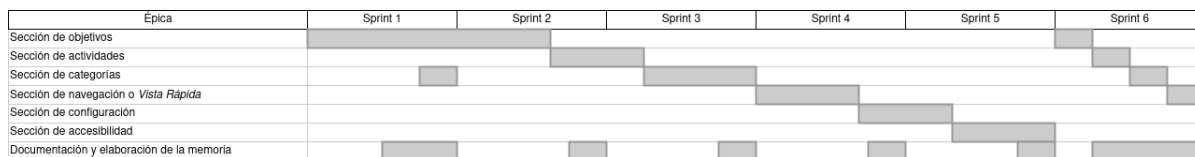


Figura 3.2: Diagrama de Gantt de las épicas del proyecto

Distribución final de las historias de usuario en las iteraciones de desarrollo

Iteración 1

- [TFG-1] Ejemplo

Iteración 2

- Ejemplo

Iteración 3

- Ejemplo

Iteración 4

- Ejemplo

Iteración 5

- Ejemplo

Iteración 6

- Ejemplo

§3.4: Presupuesto

Material requerido

Material	Cantidad	Precio/unidad	Total

Recursos humanos

Servicio	Tarifa (€/mes)	Duración (meses)	Total
Salario de ingeniero informático junior a jornada completa	1450	3	4.350€

Se ofrece además un presupuesto aproximado del coste de un año de mantenimiento de la aplicación por parte del equipo de desarrollo trabajando a media jornada (veinte horas semanales).

Servicio	Tarifa (€/mes)	Duración (meses)	Total
Salario de ingeniero informático junior a media jornada	725	12	8.700€

Servicios opcionales de despliegue y mantenimiento

Se ofrece además un presupuesto orientativo sobre el coste de desplegar y mantener la aplicación en *Amazon Web Services*, utilizando instancias de coste moderado.

Servicios opcionales	Tarifa (€/hora)	Coste/año
Despliegue de la base de datos en Amazon Aurora con una instancia de tipo db.t2.medium, modelo Reserved, arrendamiento de un año, No Upfront	0,055	481,80€
Despliegue de la aplicación en un servidor virtual Amazon EC2 de tipo r4.xlarge	0,053	464,28€

Total

Total	4.350€
Total, con servicios opcionales durante un año	9.646,08€

Capítulo 4:

Desarrollo del proyecto

§4.1: Primera iteración

4.1.1: Análisis

4.1.2: Diseño

4.1.3: Implementación

4.1.4: Pruebas

Capítulo 5:

Conclusiones y vías futuras

§5.1: Conclusiones del proyecto

§5.2: Posibles modificaciones futuras

Bibliografía

[kan] Qué es kanban: definición, características y ventajas.

[tas] Tasks.org | Tasks.org.

[3] Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. (2001). Manifesto for agile software development.

[4] Deterding, S., Khaled, R., Nacke, L., and Dixon, D. (2011). Gamification: Toward a definition. pages 12–15.

[5] Fielding, R. T. (2000). *Architectural styles and the design of network-based software architectures*. University of California, Irvine.

[6] Schwaber, K. and Sutherland, J. (2017). The scrum guide. 2017. *No. November*, pages 6–17.