

UN PROYECTO DE

UNIVERSIDAD DE MÁLAGA

SoftPro

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA



REPOSITORIO: <https://github.com/chemafdezuma/softpro.git>

ÁLVARO YUSTE MORENO
JOSE LUIS LÓPEZ RUIZ
PABLO ALARCÓN CARRIÓN
ADRIÁN CAMACHO FERREZUELO
MARTA GONZÁLEZ PALMERO

YUSTE@UMA.ES
PEPEFXLR@UMA.ES
PAPROKA@UMA.ES
ADRICAM@UMA.ES
MGONZALEZPALMERO@UMA.ES

PLANIFICACIÓN DEL PROYECTO

ÁLVARO VALENCIA VILLALÓN
RAFAEL CEBALLOS MARTÍNEZ
ROCÍO GÓMEZ MANCEBO
JOSE MARÍA FERNÁNDEZ CANTÓN

VALEAL@UMA.ES
RAFACEBAMAR@UMA.ES
ROCIOGOOMEZ27@UMA.ES
JOSEMARIAFCANTON@UMA.ES



ÍNDICE

Introducción	3
Roles	3-4
Uso de Scrum	4
Aplicación	4
Gestión del riesgo	6
Herramientas usadas	7



INTRODUCCIÓN

EXPLICACIÓN DEL PROBLEMA

Málaga es una ciudad que hoy en día recibe a miles de turistas, estudiantes, empresarios... Cada uno llega con un propósito específico, pero una vez aquí resulta inevitable querer conocer un poco más los sitios y oportunidades que puede ofrecer. Nuestro objetivo es llevar a cabo una aplicación que, a través de ciertas preguntas como número de personas para el plan, tipo de plan, dinero que se quiere gastar, el tiempo que hace, ubicación... Proponga al usuario el mejor plan que puede ofrecer la ciudad.

ROLES

DEL EQUIPO

Los diferentes roles que se han asignado son, básicamente, 6:

- PO (Product Owner): se encargará principalmente de guiar el diseño del producto, con una meta clara y concisa. Representa a los usuarios y clientes del producto en cuestión
- SM (Scrum Master): permite que los miembros del equipo tengan las herramientas necesarias para trabajar correctamente, y modera las reuniones que tenga el equipo. Además, hace de coach y de administrador del proyecto.
- Equipo de desarrollo
 - Code (Programador): como su nombre indica, se encarga de la programación front-end y back-end del proyecto en cuestión, así como todo lo complementario a código que sea necesario.
 - Testing (Pruebas): se encarga principalmente de poner a prueba las diferentes versiones que tenga el producto, para así encontrar posibles bugs y fallas del mismo. Este rol es esencial dentro de la calidad a presentar del producto final.
 - Graphic Desing: Se encarga de diseñar los elementos gráficos
 - App Design: Su función es la organización, percepción visual y elementos adicionales que formarán parte de la percepción visual de la aplicación.
 - Risks (Posibles riesgos): El atributo que se le asigna se basa en buscar los riesgos principales que tenga cada fase del desarrollo, y notificar acerca de posibles soluciones a los mismos, en caso de que ocurriesen.
 - Analyst (Analista): Identificar los requisitos que tendrá la aplicación coordinándose con el Product Owner.
 - Structural Designer: Idea y desarrolla los modelos necesarios para llevar a cabo el producto a bajo nivel.



Ya explicados, los miembros del grupo poseen los siguientes roles:

- Rocío Gómez Mancebo: PO / Graphic Design
- Alvaro Yuste Moreno: Analist / Structural Design
- Álvaro Valencia Villalón: Scrum / Graphic Design
- Marta González Palmero: Scrum / Structural Design
- Adrián Camacho Ferrezuelo: Testing / Structural Desing
- Pablo Alarcón Carrión: Code Specialist / Analist
- José María Fernández Cantón: Testing / Analist
- José Luis López Ruíz: Code Specialist / App Design
- Rafael Ceballos Martínez: Risks / App Design

Debido a la alta demanda de desarrolladores de código y teniendo en cuenta el equipo que tenemos, todos pertenecemos al departamento "code". También dándonos cuenta nuestra inexperiencia diseñando y que cada uno tiene ciertas habilidades al respecto, hemos decididos fraccionar el departamento de "design" como viene reflejado arriba.

APLICACIÓN

LOGO Y NOMBRE

El nombre 'where2go' se ha elegido teniendo en cuenta la funcionalidad principal de la aplicación, encontrar un plan acorde a tus necesidades.

El diseño del logotipo reúne todas las partes del nombre de la aplicación, en forma de símbolo abstracto y moderno.



'Where2go'

USO DE SCRUM

Y POR QUÉ LO HEMOS ELEGIDO

Creemos que este es el método más adecuado para este proyecto porque fomenta el trabajo en equipo, nos permite realizar cambios durante el proceso del proyecto y favorece el contacto con el cliente en todo momento.

Seguramente una vez terminado el proyecto habremos aprendido mucho sobre la metodología, las necesidades y habilidades de desarrollo de un proyecto y las capacidades necesarias para el trabajo en grupo, la colaboración y la planificación de actividades complejas.

Concretando, nuestra estrategia será la siguiente:

- Sprints de 1 semana de duración



- Sprint Planning (reunión grande) los lunes.
- Sprint Review Meeting (reunión de 15 minutos aproximadamente) los domingos

The screenshot shows a Trello workspace for 'SoftPro'. The left sidebar contains navigation options like 'Tableros', 'Miembros', 'Configuración', and 'Vistas del Espacio de trabajo'. The main area displays a 'Product Backlog' and three Sprints.

Product Backlog:

- (0) Preparar historias de usuario
- (0) Subir los elementos gráficos a git
- (0) Preparar la documentación de requisitos
- (Importancia: Muy alta) (RNF1) Gran variedad de lugares
- (Importancia: Medio) (RF1) Descripción del lugar
- (Importancia: Alta) (RF2) Sistema de filtros
- (Importancia: Muy alta) (RNF2) Buen sistema de filtro
- (Importancia: Alta) (RNF3) Diseño visual atractivo
- (Importancia: Medio) (RNF4) Fácil uso de la app
- (Importancia: Muy alta) (RNF5) Tiempo
- (Importancia: Muy alta) (RF3) Mapa

Sprint 1:

- TERMINADO Subir el documento actualizado
- TERMINADO Explicar por qué hemos usado SCRUM
- TERMINADO Crear la plantilla de word
- TERMINADO Hacer las capturas de Trello
- TERMINADO Asignar encargado de requisitos
- TERMINADO Añadir roles de analista y diseñador

Sprint 2:

- Requisitos beta
- Documento para escribir todos los requisitos y sus derivados y clasificarlo...
- Google Docs Vista previa
- EN PROGRESO INFORME SECRETO (INDIVIDUAL)
- TERMINADO Desarrollar los requisitos y clasificarlos en RF y RNF
- TERMINADO Captura de Pantalla de Trello
- TERMINADO Subir documento
- TERMINADO Corregir hiperenlace
- TERMINADO Arreglar archivo extraño de Git (modificación de la plantilla)
- 26 de mar.
- TERMINADO Incluir en el documento logo y

Sprint 3:

- POR HACER Actualizar el índice del documento
- EN PROGRESO Diseño pantalla de carga



GESTIÓN DEL RIESGO

TIPO, PROBABILIDAD, EFECTOS Y MITIGACIÓN

Tipo	Riesgos	Probabilidad	Efectos	Mitigación
Proyecto	Poca experiencia organizativa	Muy alta	Tolerable	Aprender lo máximo posible del profesor
Proyecto	Mala planificación del tiempo invertido en tareas	Alta	Serio	Tratar de evitar las máximas dependencias posibles y reducir el camino crítico en la planificación
Proyecto	Imposibilidad de implementación de las herramientas CASE	Alta	Tolerable	Tener varias herramientas disponibles por si una falla
Proyecto y producto	Cambio de diseño para implementar las funciones requeridas	Moderada	Serio	Optimizar lo máximo el código para que no suponga una pérdida de tiempo grande
Producto	La base de datos usada en el sistema no es capaz de procesar tantas transiciones	Moderada	Serio	Plantear la compra de una base de datos con mas capacidad y rendimiento
Proyecto y producto	Cambios repentinos en los requisitos	Moderada	Serio	Evaluar el impacto de los requisitos para ver si conviene o no
Negocio	Otra compañía lanza una aplicación con el mismo objetivo	Baja	Tolerable	Tratar de implementar más funciones llamativas sin que afecten al tiempo de entrega
Proyecto	Personal clave está enfermo o no esta disponible en momentos críticos	Moderada	Tolerable	Reorganizar el equipo para que haya más miembros que conzcan el trabajo del resto
Proyecto y producto	Componentes reutilizados limitan la funcionalidad	Muy baja	Insignificante	Modificar el código reutilizado para adaptarlo a las necesidades



HERRAMIENTAS USADAS

HASTA LA FECHA

- Métodos de comunicación
 - Discord (Reuniones)
 - WhatsApp
- Trabajo colaborativo
 - Git/GitHub
 - Trello
- Elaboración de documentos
 - Microsoft Word
 - Pages
- Diseño gráfico
 - Adobe Photoshop
 - Affinity Designer
 - Pencil 2D
 - Procreate
- Diagrama de requisitos
 - MagicDraw

REQUISITOS

FUNDAMENTALES PARA EL PROYECTO

Los requisitos son descripciones de los servicios que un sistema debe proporcionar y las restricciones a su modo de operación.

Un requisito es una necesidad documentada sobre el contenido, forma o funcionalidad de un producto o servicio. Se usa en un sentido formal en diversas ingenierías.

FUNCIONALES (RF)

Describen la funcionalidad o los servicios que se espera que el sistema suministre.

- **Sistemas de filtros:** Requisito funcional ya que hablamos de un servicio que ofrece el producto. Para ofrecer al cliente una experiencia más personal y satisfactoria debería de existir un mecanismo que mediante una interacción con el usuario se vaya profundizando en lo que busca para darle el lugar que más se ajuste a lo que busca.
 - Este sistema de filtros podrá tener en cuenta: la zona por la que quiere estar el usuario, el dinero que pretende gastarse, la calidad de la experiencia, el número de personas con las que irá (las personas con las que quedará), la hora a la que saldrá



el cliente (por el tema de los horarios de los locales), el tiempo que haga, ... Y habrá filtros específicos dependiendo de la actividad. Por ejemplo, si lo que quiere el cliente es comer en la calle podrá especificar si quiere comida italiana, mexicana, asiática, ... (esto último no es seguro).

- Incluso puede haber filtros que choquen entre sí. Por ejemplo, si se pide que la calidad sea alta y que sea económico puede llegar a ser una contradicción en algunos casos, por lo que podríamos añadir una opción que sea prioridad del filtro. Así podremos poner que nos ofrezca un local barato y de gran calidad, pero priorizando la calidad. Por lo que nos saldrían sitios menos económicos que si priorizamos el precio. Pero aun así nos saldrían sitios más baratos que si solo pusiéramos que sean de calidad. De esta manera con el uso de los filtros se mostrarán lugares acordes a lo que busca el usuario.
- **Fácil uso de la app:** La aplicación será implementada de tal forma que cualquier persona sea capaz de conseguir encontrar los lugares y utilizar todas sus herramientas de manera sencilla ya que todo será totalmente intuitivo.
 - El objetivo es maximizar las funcionalidades de la aplicación sin perder la esencia de sencillez que ofrece, así la experiencia de uso será muy cómoda intentando a su vez que haya una fluidez total del sistema para que el usuario pueda buscar los lugares de manera eficaz, sin pausas excesivas en los tiempos de carga.
- **Tiempo:** El apartado del tiempo será implementado en una esquina de la aplicación y aparecerá de forma constante indicando de manera actualizada la información del tiempo basándose en la ubicación del usuario, podríamos implementarlo mediante una api o al igual que el mapa, con un hiperenlace que te lleve a una pagina donde te muestre el tiempo, pero la implementación mediante una api, aunque necesitaría más trabajo y sería más compleja creo que para estos casos es la ideal.
 - Para llevar a cabo esta funcionalidad tendremos que precisar la ubicación del usuario mediante permisos o es posible precisar con ayuda del usuario proporcionando la misma e implementar la api (en el caso de que se use) para que esté constantemente mostrando actualizada, ya que facilita mucho al usuario el hecho de tener que estar buscando las condiciones meteorológicas en otras páginas.
- **Mapa:** La mecánica de mostrar sitios en el mapa será desarrollada en base a un botón dentro de cada ubicación determinada por las especificaciones que el usuario haya querido previamente. Por ende, al entrar en la pestaña de dicho lugar aparecerá, en la parte inferior de la ventana, un botón que tendrá escrito "MAPA", el cuál al pulsarlo se redirigirá (mediante un hiperenlace insertado en susodicho) a la aplicación o página web de Google Maps, con las coordenadas precisas del lugar y centrado en el mismo para la posterior navegación del usuario al lugar especificado en la página.
 - Esta mecánica hace mucho más simple el hecho de utilizar la herramienta proporcionada, puesto que la gran mayoría de usuarios están familiarizados con la herramienta proporcionada por Google, y simplificando la complejidad de nuestro proyecto, evitando futuros problemas (como hubiesen ocurrido si quisiéramos desarrollar nuestra propia aplicación de mapa integrada para la aplicación).



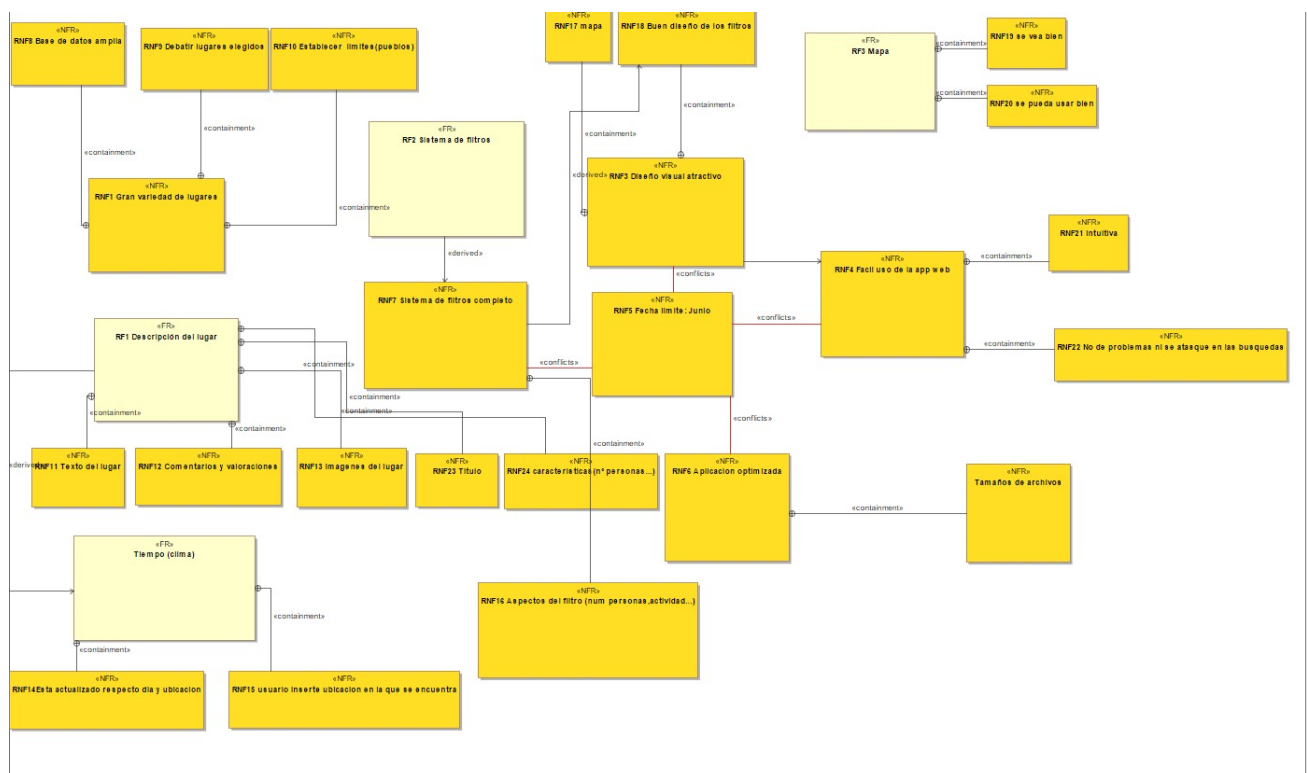
NO FUNCIONALES (NRF)

Restricciones sobre los servicios o funciones ofrecidas por el sistema.

- **Descripción del lugar:** Cada sugerencia que la app haga al usuario contará con cinco apartados principales.
 - Título: tiene que ser concreto, corto, incluir el tipo de lugar y el nombre de este.
 - Texto: describe el lugar en función de las oportunidades que presente y teniendo en cuenta algunas valoraciones de los usuarios. No será una descripción física del lugar ya que los elementos a destacar vendrán en las características del lugar.
 - Características: incluyen elementos físicos destacables como el parking, terraza o techado, posibilidad de fumar, bancos o lugares para sentarse, mesas para comer, barbacoas... Serán incluidas por los usuarios que vayan al lugar.
 - Comentarios y valoraciones: los usuarios que vayan a los sitios podrán comentar lo que les ha parecido el sitio y alguna sugerencia para el resto de usuarios. Los usuarios podrán valorar del 1 al 5 el lugar y las características del sitio añadidos por el resto de usuarios. (Esta valoración se hará marcando los dedos de un pie)
 - Imágenes: mínimo una imagen de cada sitio, para que los usuarios se hagan una idea de como es.
- **Sistema de filtros optimizado:** Este requisito deriva de "Sistema de filtros" ya que ahora estamos hablando de la calidad del sistema de filtros. Y por esto mismo es no funcional. El requisito entra en conflicto con el de la fecha límite, como la mayoría de los requisitos no funcionales.
 - Aquí se exige que todos los lugares que se muestren sean acordes a los filtros que se están especificando y que por supuesto, se muestren todos los posibles (aunque esto está relacionado con el requisito de "una gran variedad de lugares").
 - También deberíamos optimizar todo lo posible el aspecto de priorizar filtros.
 - Debemos minimizar en lo máximo posible la ralentización que el sistema de filtros puede causar en la web (esto va relacionado con el requisito de "aplicación optimizada").
 - El número de filtros que se pueden aplicar a la vez (en un principio) deberían ser todos los disponibles. Puede que por temas de implementación o de optimización esto se vea limitado.
- **Diseño visual atractivo:** Este requisito es no funcional porque estamos hablando de cómo va a ser el diseño.
 - Lo suyo sería disponer de un diseño simple, ya que esto ayuda al otro requisito de "fácil uso de la app web". Por esto mismo están relacionados ambos requisitos. Es decir, no saturar la pantalla de información ni de ilustraciones y hacer la navegación en la web lo más intuitiva posible. Pero que las ilustraciones sean de calidad, ilustrativas y elegantes.
 - Esto está relacionado con la idea de un diseño minimalista. Cómo por ejemplo el logo de la app, hemos buscado un logo que sea simple pero a la vez ilustrativo.



- A la hora de usar los colores buscamos una armonía visual. Es decir, usar una paleta de colores que permita que al estar usando la app durante mucho tiempo no incomode a la vista y que sea visualmente atractivo.
- **Tamaño:** El tamaño de la aplicación debería de ser el mínimo posible, para que pueda alcanzar la máxima cantidad de dispositivos usándola.
 - Para ello, las imágenes serán adaptadas y comprimidas, y el número de elementos, pestañas, efectos, etc, aparecerán en la cantidad justa mínima posible para la funcionalidad completa de la aplicación, y a su vez ocupando la menor cantidad posible de espacio en los discos duros de los dispositivos a usar.
- **Rendimiento:** La aplicación, ligada al requisito de tamaño, tendrá la mínima cantidad posible de elementos, animaciones y funcionalidades, de forma que pueda correr correctamente en la mayor cantidad de dispositivos.
 - Para ello, se limitará la cantidad de efectos y pestañas en la aplicación, ganando así velocidad de ejecución dentro de las diferentes funcionalidades de la aplicación, optimizando así la susodicha para un mejor rendimiento.



Esquema en MagicDraw