



RESCUE EATS

Proyecto FCT

ALUMNO: ENRIQUE GARCÍA PINO - 2º DAM

TUTOR: ABEL MORILLO SÁNCHEZ

CENTRO: SAN PABLO ANDALUCÍA CEU

FECHA: 02 de junio del 2023



CEU

| Centro de Estudios
Profesionales

Índice

1. Introducción y objetivos	3
1.1. Introducción del proyecto.....	3
1.2. Objetivos del proyecto.....	4
- Concienciación:	4
- Beneficio monetario de la aplicación:	4
- Mejora en los comercios locales:	4
2. Contexto del proyecto	5
2.1. Estado actual.....	5
2.2. Justificación	7
3. Metodología	9
3.1. Fases de la metodología	9
4. Planificación y costes.....	11
Costes	11
Planificación temporal.....	12
5. Análisis de requisitos	14
6. Diseño de solución	16
7. Plan de despliegue.....	20
8. Conclusiones y propuestas de mejora.....	24
Conclusiones.....	24
Propuestas de mejora.....	25
9. Bibliografía.....	26

1. Introducción y objetivos

1.1. Introducción del proyecto

El trabajo que se va a presentar se basa en una aplicación para móviles cuyo nombre sería “*Rescue Eats*”. El objetivo principal de la aplicación “*Rescue Eats*” se centraría en reducir el desperdicio general de comida que hay actualmente a nivel mundial y, con esto, concienciar sobre el problema que genera el desecho de alimentos diario.

Para poder llevarlo a cabo, los negocios de restauración o comida para llevar ofrecerán a los usuarios excedentes de comida a un precio bastante más bajo y económico del habitual.

De esta manera, favoreceríamos a los distintos comercios, ya que, contarían con un número mayor de clientes a través del uso de nuestra aplicación; por tanto, lograrían beneficios extra de los excedentes de comida que anteriormente hubieran desechado a la basura, perdiendo así dinero, productos y alimentos.

Finalmente, los usuarios podrían tener acceso a comida de distintas zonas de su región a menor precio; por tanto, una mayor accesibilidad para mayor cantidad de personas y conocerían lugares nuevos cerca de ellos donde, en el futuro, podrían volver a visitar.

1.2. Objetivos del proyecto

Por parte del desarrollo, los principales objetivos que se pretenden conseguir con la salida de esta aplicación al mercado son los siguientes:

- **Concienciación:**

Con la salida de este proyecto, queremos causar un cambio total en uno de los puntos más críticos y menos tratados en la sociedad de hoy en día, el desperdicio en cantidades masivas de alimentos que se produce diariamente.

Queremos concienciar a la sociedad sobre el gran desperdicio de alimentos que generamos, debido a la compra masiva de alimentos y comidas, que finalmente, acabamos desechando.

Según un estudio realizado por la ONU¹, más de la quinta parte de los alimentos de cada compra los terminamos desechando o dejando en la nevera hasta que llega la fecha de caducidad y ya no hay vuelta atrás.

Tenemos que ser conscientes de las compras que realizamos, ya que, muchas veces compramos con los ojos o “por si acaso”, cuando finalmente, solemos terminar utilizando y comiendo lo mismo.

Por último, el desperdicio no solo viene en las compras masivas, sino en que no llegamos a aprovechar al 100% cada alimento que compramos, además de saber qué y cuánta cantidad es la que necesitamos comprar, hay que saber gestionar esos alimentos de manera eficiente.

- **Reducción del desperdicio de alimentos:**

Gracias al gran ejercicio de concienciación con respecto al desperdicio de alimentos se conseguiría un impacto significativo en la producción de alimentos; puesto que, se reduciría el consumo excesivo y, por tanto, no sería necesario generar alimentos en cantidades tan grandes y, por consiguiente, conseguiríamos un menor impacto en el medio ambiente y una disminución en su explotación.

- **Beneficio monetario de la aplicación:**

Gracias a la salida y expansión de este proyecto, los creadores de este, también podrían conseguir rédito de él. El dinero que se recibe sería únicamente del pequeño porcentaje que se lleva la App en cada venta y de los anunciantes que se quisieran unir a la publicación de la aplicación.

- **Mejora en los comercios locales:**

El beneficio propio no es el único campo donde fluya el dinero, ni siquiera el más importante. En cuanto al proyecto, su crecimiento se expande a nivel mundial y centra su objetivo primordial en potenciar todos aquellos mercados y comercios locales que tenemos a nuestro alrededor, esos pequeños puestos de alimentos que aún conservan ese aroma a familia.

¹ Fuente: Noticias ONU: *El desperdicio de comida, una oportunidad para acabar con el hambre.*

(2018). Naciones Unidas.

2. Contexto del proyecto

2.1. Estado actual

Hoy en día, el desperdicio de alimentos es un problema a nivel global con importantes consecuencias en ámbitos económicos, sociales y ambientales.

Para evitar el desperdicio de esta inmensa cantidad de alimentos, es necesario que cada uno de nosotros se implique en adoptar tanto prácticas como hábitos positivos respecto al uso que le damos a la comida para que luego, durante todas las diferentes etapas del suministro de alimentos (desde la recogida del producto y abastecimiento hasta que tras varios pasos ese alimento consigue llegar a nuestras manos a través de la distribución), podamos sumar todos en conjunto. Por tanto, si un eslabón de la cadena fallara, todo terminaría volviendo al punto en el que nos encontramos hoy.

En términos de producción, es esencial promover una agricultura sostenible y eficiente, para así, reducir las pérdidas que suelen producirse durante la cosecha, y sobre todo el transporte y almacenamiento, pues el transporte y el almacenamiento son el segundo y tercer lugar donde más se desecha comida, después del hogar. El primero por largos recorridos o nefastas conservaciones, y la segunda porque en la selección se suelen desechar, sobre todo frutas y vegetales, simplemente por su forma, apariencia o color.

Cada año se tira a la basura $1/3^2$ de los alimentos que han sido adquiridos en los supermercados, un dato devastador sabiendo que gran parte de los países del mundo sufren de deficiencia alimentaria o no tienen la capacidad de obtener alimentos.

Además, es importante fomentar la diversificación de cultivos y la gestión adecuada de la calidad y la cantidad de alimentos producidos. Con la masiva cantidad en producción y la constante explotación de las tierras los productos cada vez tienen más químicos y sustancias extra como colorantes y edulcorantes. Los sabores cada vez se alejan más a los reales y nuestro gusto se ve constantemente alterado hasta que nos acostumbramos a estos sabores explosivos.

Respecto a las soluciones que hay hoy en día, la mejor que existe, antes de tirarlos a la basura, es donar estos alimentos que no vamos a utilizar a personas que realmente lo necesiten y vayan a agradecer que ese alimento llegue a su plato mediante una ONG o los bancos de alimentos que haya en tu ciudad. A veces, también existe la opción de donar directamente a escuelas y comunidades de tu zona.

² Fuente: ONU: *Pérdidas y desperdicio de alimentos en el mundo*. (2011).

Pero si para cambiar como sociedad y a nivel global, debemos empezar por nosotros mismos, la pregunta que nos tendríamos que hacer sería, “¿cómo podría mejorar y así reducir la cantidad de alimentos que desechamos cada uno?”

Para disminuir la cantidad de alimentos que cada día terminamos tirando por diversos motivos, existen los siguientes trucos que, con constancia, terminaríamos ayudando mucho al planeta y a nosotros mismos.

Empecemos por el principio, el supermercado. Cuando estamos en el supermercado, muchas veces no logramos recordar de memoria qué hace falta realmente en casa y terminamos guiándonos por lo primero que nos entra por los ojos o por aquello que más nos gusta. ¿El problema? Al llegar a casa nos damos cuenta que la mitad de alimentos los hemos comprado por capricho y que lo que realmente necesitábamos ni siquiera se nos había ocurrido. Pero por suerte, esto tiene fácil solución, un simple papel y boli.

Puede parecer una tontería, pero muchas personas, hoy en día, aún existiendo una gran cantidad de aplicaciones móviles para crear una lista de la compra, no se organizan y, finalmente, cuando llega la hora de comprar cometen el error de no saber qué hace falta.

Existen algunas aplicaciones móviles como “*TooGoodToGo*” que acerca a empresas la posibilidad de aportar los excedentes de alimentos a un precio menor aplicaciones móviles también pueden desempeñar un papel importante en la reducción del desperdicio de alimentos.

Nuestro proyecto marcaría la diferencia, ya que, contaríamos con la colaboración de las empresas de restauración y de alimentación, para que se comprometiesen, mensualmente, a destinar una parte de su excedente de alimentos a una ONG. Además, ofreceríamos la oportunidad de realizar repartos a domicilio, atendiendo así, a las necesidades de nuestros clientes.

En resumen, para poder evitar el desperdicio de alimentos necesitamos implicación a la hora de adoptar prácticas sostenibles en todas las etapas de la cadena de suministro, desde la producción hasta el consumo y debemos ser conscientes de que la educación y concienciación son fundamentales para combatir el desperdicio de alimentos.

2.2. Justificación

La creación de “*Rescue Eats*”, dedicado a evitar el desperdicio de alimentos, tiene justificaciones tanto a nivel social como económico. A continuación, se presentan las principales justificaciones para cada uno de estos aspectos:

A nivel social:

- **Responsabilidad social:** “*Rescue Eats*” demuestra un compromiso social por parte de los fundadores y colaboradores. El desperdicio de alimentos es un problema ético y moral, y abordarlo de manera eficaz, muestra preocupación por el bienestar de la sociedad y el medio ambiente.
- **Sostenibilidad ambiental:** el desperdicio de alimentos tiene un impacto significativo en el medio ambiente. La producción de alimentos requiere recursos naturales como tierra, agua y energía, y cuando se desperdician, también se desperdician estos recursos. Además, los alimentos en descomposición en los vertederos generan emisiones de gases de efecto invernadero. Al evitar el desperdicio de alimentos, se contribuye a la conservación de los recursos naturales y se reduce la contaminación a nivel medioambiental.
- **Lucha contra el hambre:** a nivel mundial, hay millones de personas que sufren de insuficiencia alimentaria y no tienen acceso suficiente a alimentos nutritivos. Al reducir el desperdicio de alimentos, se pueden recuperar grandes cantidades de comida que de otro modo se perderían y ponerlas a disposición de aquellos que más lo necesitan a un precio razonable.
Finalmente, acordaríamos con las empresas que, de manera mensual, el excedente de los alimentos, lo donasen a una ONG.

A nivel económico:

- **Ahorro de costos:** el desperdicio de alimentos representa una pérdida económica significativa para los negocios de alimentos, como “*RescueEats*”, ya sean productores, distribuidores o minoristas. Al evitar este desperdicio, se puede reducir la cantidad de alimentos que se pierden y, por lo tanto, ahorrar costos para las empresas.
Además, al vender los excedentes de alimentos a precios más bajos, las empresas pueden obtener ingresos adicionales que de otra manera no habrían obtenido.
- **Generación de empleo:** la creación de “*RescueEats*” puede generar oportunidades de empleo. Se requerirían trabajadores para gestionar la recolección, clasificación, redistribución y promoción de los alimentos rescatados. Esto no solo beneficia a la economía local, sino que también brinda oportunidades laborales a personas que podrían estar desempleadas o subempleadas.

- **Estímulo a la economía local:** al evitar el desperdicio de alimentos y fomentar la redistribución de los mismos, se promueve el comercio local. Las empresas locales de alimentos podrían participar en la iniciativa, lo que les permitiría generar más ventas y atraer a nuevos clientes. Esto puede tener un impacto positivo en la economía local, fortaleciendo el tejido empresarial y contribuyendo al crecimiento económico.
- **Beneficio para los clientes:** además de obtener los alimentos a menor precio, los clientes podrán obtener descuentos al emplear la app de “*RescueEats*”, consiguiendo, aproximadamente, un 15% de descuento por compra realizada en el negocio. Esto puede atraer a más clientes con el fin de ahorrar en las compras de sus casas.

En resumen, la creación de “*RescueEats*” para evitar el desperdicio de alimentos tiene una justificación sólida tanto a nivel social como económico. Contribuye a la lucha contra el hambre, promueve la responsabilidad social y la sostenibilidad ambiental, ahorra costos a las empresas, genera empleo, beneficia a nuestros clientes y estimula la economía local.

3. Metodología

La metodología ágil *Scrum* es un marco de trabajo que se basa en principios incrementales.

El trabajo se divide en ciclos cortos llamados “*sprints*”, cada uno con una determinación concreta y fija; al final de cada “*sprint*”, se entrega un incremento del producto; a su vez, *Scrum* la componen tres roles principales, que son los siguientes:

- **Product Owner:** se encarga de definir y priorizar los elementos del producto; trabaja de la mano del *Equipo de Desarrollo*, para asegurar que se entienden los requisitos y se prioricen de manera adecuada.
- **Scrum Máster:** facilita el proceso y su función principal es asegurar que el equipo siga los principios de *Scrum* y, al mismo tiempo, ayuda a resolver problemas, elimina obstáculos y fomenta la buena comunicación en el equipo.
- **Equipo de Desarrollo:** se encarga de realizar el trabajo necesario para entregar el producto; al tratarse de un equipo autoorganizado y disciplinar, determina cómo ha de llevarse a cabo el trabajo de la manera más eficiente posible.

He escogido esta metodología porque se caracteriza por su adaptabilidad, ya que, se adapta rápidamente a los cambios del proyecto; porque garantiza una comunicación constante y transparente al cliente; por su eficacia y rapidez y por la colaboración y responsabilidad del equipo y la obtención de resultados de alta calidad.

3.1. Fases de la metodología

Las tareas que se deben desarrollar para este proyecto se dividen en las tareas de desarrollo, test y marketing.

- **Estructuración del proyecto:** la primera tarea sería dejar todo claro y ordenado. Para ello se crearían documentos donde se definirían todos los requisitos que tiene el proyecto por zonas. Por ejemplo, en la BBDD se diría qué datos se quieren tener y el lenguaje, en el documento de la app todas las pantallas y lugares a los que el usuario puede acceder y el lenguaje con el que se trabajaría en Android e IOS, y para la API qué datos son los que queremos obtener, devolver y manejar y cuándo ocurriría esto, a parte de la herramienta con la que se trabajaría.
- **Definición de la BBDD:** una de las tareas más importantes es definir y crear una BBDD amplia, bien estructurada y suficientemente clara para facilitar posteriormente la obtención de datos. Esta tarea sería realizada por varios programadores expertos en BBDD. La BBDD sería una base de datos MySQL. Aquí se crearían las tablas, las conexiones entre ellas y se insertarían todos los datos de configuración.

- **Creación del Servicio REST API:** Tras ya tener definida y creada la BBDD, y estar trabajando en la funcionalidad y capas de presentación de la aplicación para móviles, necesitamos un mediador, aquella herramienta que utilizemos para comunicar los dos flacos. Para ello utilizaremos un servicio de REST API creado en el lenguaje de programación Java.
- **Creación de la APP para Móviles:** Junto a la BBDD esta es una de las tareas más pesadas del proyecto. En esta tarea los programadores crearían la herramienta que todos los usuarios llegarían a utilizar. Se crearían dos aplicaciones a la vez, una para los dispositivos Android y otra para los dispositivos con sistema operativo IOS. Las 2 tendrían las mismas pantallas y mismas funcionalidades, únicamente habría dos apps porque los lenguajes para cada uno difieren.
- **Test para la BBDD:** Tras terminar la creación de la BBDD, se harían repastos a los datos y tablas y se testearían mediante pruebas de funcionalidad para verificar que la base de datos almacena lo que se requiere, pruebas de integración de datos para verificar que los datos que están almacenados son los correctos, pruebas de seguridad y el uso de respaldos de información para no perder todo en caso de fallos.
- **Test para la API:** Una vez ya testada la base de datos y los dispositivos móviles, lo que faltaría sería comprobar si la API interconecta ambas herramientas y si las llamadas de datos desde la App y la recogida de la BBDD funcionan correctamente mediante pruebas de rendimiento y solución a posibles errores.
- **Test para la APP móvil:** Esta tarea tiene la misma función que el testing en la BBDD, verificar que todo está en correcto funcionamiento. En este caso, se harían pruebas de funcionalidad, pruebas de usabilidad con usuarios ajenos a la producción, de rendimiento, seguridad, compatibilidad con distintas versiones de Android e IOS y distintos tipos de teléfonos.
- **Marketing:** Una vez todo está listo, solo nos queda el darnos a ver en el mercado. Nuestra expansión sería, sobre todo, gracias a las redes sociales y a la aparición en los anuncios de distintas páginas web e incluso producción de anuncios vía televisión. Utilizaríamos los medios tecnológicos pues es donde más jóvenes hay y donde la expansión puede ser tanto más rápida como lejana.

4. Planificación y costes

Nuestro negocio se encontraría en un piso situado en la calle Dilar nº 8.



Costes

Al principio de este proyecto, el local estaba en construcción y una vez acabadas las obras, invertimos alrededor de unos 35.000€ en total, aproximadamente unos 32.600€ el coste del local y alrededor de 2.400€ en la reforma.

Una vez finalizada la compra y la reforma del local, llegó el momento de la selección del personal.

Fueron casi 100 candidatos los que se presentaron a los distintos puestos de trabajo y, tras una larga selección, siete fueron los candidatos a trabajar en “RescueEats”.

Personal:

Contamos con siete empleados contratados, dos de los cuales trabajan como programadores de BBDD; otros dos en el lenguaje de *Python*; dos más ubicados en el lenguaje de *Swift* y, finalmente, contaríamos con un programador de *API REST*.

Salario:

Los programadores de BBDD obtendrán un salario medio de 2083€/mes; seguidamente, los trabajadores de *Python* contarán con un salario medio de 2687€/mes; 3000€/mes será el salario de los empleados de *Swift* y para finalizar, el trabajador de *API REST* contarán con un salario de 2500€/mes.

Plazas	Puesto de trabajo	Sueldo medio
2	Programador de BBDD	2.083 €
2	Programador de Python	2.687 €
2	Programador de swift	3.000 €
1	Programador de API Rest	2.500 €

Materiales:

Cada uno de nuestros empleados tendrá a su disposición material para poder llevar a cabo su trabajo.

El material con el que contaremos será: seis PC de *Windows 10* cuyo precio es de 399€/ud; el trabajador de *API REST* contará con un *iMac* cuyo coste es de 1200€/ud.

Además, contarán con la licencia de Microsoft, para que puedan trabajar libre y cómodamente en sus proyectos, dichas licencias tienen un precio de 18,90€/ud.

Material	Cantidad	Coste/ud
iMac	2	1.200 €
PC Windows 10	5	399 €
Licencia	7	18,90 €

Planificación temporal

La creación de este proyecto ocupará aproximadamente unos nueve meses.

A continuación, se detallan todos los procesos que hemos realizado para conseguir alcanzar nuestro objetivo, la creación de “*Rescue Eats*”.

En primer lugar, realizaremos la “*Estructuración y definición del proyecto*”, a esta fase le dedicaremos quince días aproximadamente, dando comienzo el 22/05/2023 y finalizando el 05/06/2023. En dichas sesiones, marcaremos los objetivos de “*Rescue Eats*”, detallaremos las fases del proyecto, estableceremos estrategias de mercado, identificaremos las necesidades de los clientes y analizaremos la competencia que nos puedan generar otras empresas, y una vez detectados los puntos fuertes de dichas empresas, pondremos el foco en marcar la diferencia para así atraer a más usuarios.

Una vez finalizada la primera fase, pasaremos a la “*Creación de BBDD*” que abarcará un mes de trabajo finalizando el 06/07/2023. En esta fase nuestros expertos en BBDD, realizarán las tablas, índices y restricciones. Para ello, contarán con un equipo electrónico al completo.

Posteriormente, se realizará el “*Test de BBDD*”, cuya fecha de inicio es 07/07/2023 y durará una semana aproximadamente. En estas sesiones, se llevarán a cabo pruebas al detalle, para garantizar que la BBDD funcione correctamente y se cumplan todos los requisitos establecidos, al mismo tiempo, se verifica la eficacia y la seguridad de la BBDD.

Damos paso a la fase de “*Creación servicio API REST*”, donde contaremos con un solo trabajador. En dicha fase se crea la conexión entre la BBDD con la App de móvil a través de la transmisión de datos. Esta fase dará comienzo el 17/07/2023 y finalizará la primera semana de agosto.

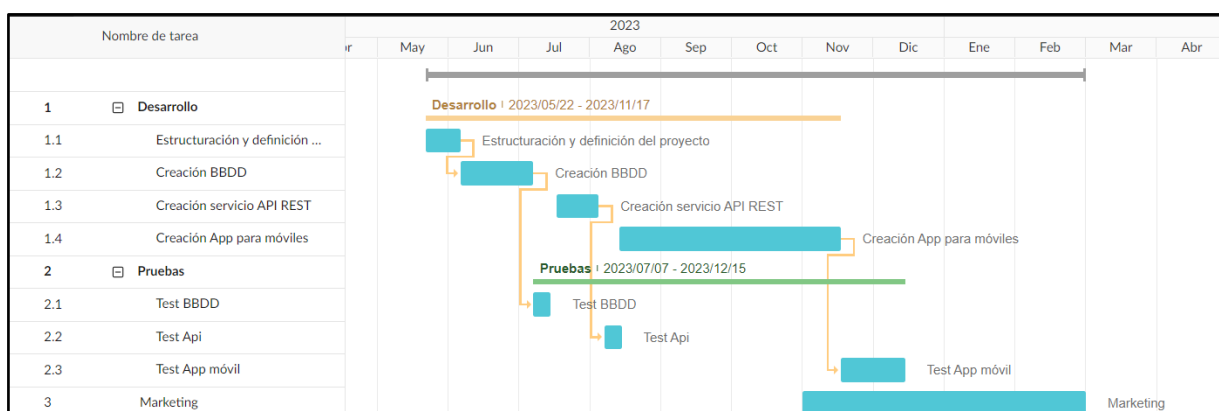
Seguidamente, se realizará el “*Test API*”, cuya fecha del inicio será 07/08/2023 y terminará el 14/08/2023. En esta fase se llevará a cabo una revisión de lo que se ha hecho en la fase anterior, garantizando la perfección en nuestros servicios.

Pasamos a la siguiente fase, la “*Creación App para móviles*”, cuya duración será desde la segunda semana del mes de agosto hasta el 17/11/2023. En dichas sesiones se diseña la App, se estructura de manera ordenada, se establecen los objetivos que se quieren cumplir, se cumplen los requisitos propuestos para dicha sesión y se desarrolla un servicio de calidad.

Ya finalizada la App para móviles, realizaremos el “*Test App móvil*”, que abarcará desde la segunda semana de noviembre hasta aproximadamente la primera quincena de diciembre.

En dicha sesión revisaremos todos los posibles fallos y errores que pueda tener la App, así como, su funcionamiento, su rapidez, sus distintas funciones, su eficacia, etc.

–Para finalizar, destinaremos aproximadamente cuatro meses a la fase de “*Marketing*”. En estos cuatro meses, nos anunciaremos por distintas redes sociales, carteles publicitarios, anuncios digitales, etc. A su vez, realizaremos un estudio del mercado, ya que, podremos encontrarnos con empresas que nos hagan competencia, generaremos demanda y, por último, daremos visibilidad a nuestra marca.



5. Análisis de requisitos

Para llevar a cabo el análisis de requisitos del proyecto, he decidido recoger la lista de requisitos mediante el método MoSCoW para así tener una correcta gestión de prioridades.

El método MoSCoW es una técnica desarrollada por Dai Clegg que ayuda a priorizar características y requisitos en proyectos con limitaciones de tiempo. Este enfoque se centra en dar mayor valor y prioridad a las características del producto que tienen un mayor valor comercial.

Los requisitos en el método MoSCoW se dividen en cuatro categorías, lo que facilita la comprensión y la priorización estratégica. Estas categorías son las siguientes:

- **Must have** (Debe tener): Estas características son absolutamente críticas para el éxito del proyecto. Sin ellas, el proyecto se consideraría un fracaso. Estos requisitos son esenciales y no pueden ser comprometidos.
- **Should have** (Debería incluir): Estos aspectos del proyecto son también críticos, pero no son absolutamente indispensables. Son requisitos importantes que se deben cumplir para lograr los objetivos del proyecto, pero pueden ser negociados o adaptados en función de las limitaciones de tiempo y recursos.
- **Could have** (Podría incluir): Estas iniciativas son deseables y añadirían valor al proyecto, pero no son críticas. Son requisitos que se considerarían como mejoras o características adicionales que podrían implementarse si hay recursos y tiempo disponibles después de cumplir con los requisitos anteriores.
- **Won't have** (No se van a hacer): Estas características no se incluirán en el proyecto en este momento. No se considera que valga la pena invertir recursos en ellas, ya que no aportarían beneficios significativos en este momento. Sin embargo, podrían ser consideradas en futuras iteraciones o versiones del producto.

A. MUST HAVE

+ Proyecto y pruebas:

- Que el proyecto tenga una BBDD relacional que sea consistente, rápida y esté bien estructurada.
- Que el proyecto tenga un servicio API REST que conecte la BBDD y la aplicación móvil para el traspaso de datos entre ellas dos (entrada y salida de datos).
- Que el proyecto tenga una aplicación para móviles tanto en el market place de Android como en el de iOS.
- Que la BBDD, el API y la aplicación para móviles tengan sus respectivas pruebas antes de salir al mercado.

+ App:

- La app móvil será multilinguaje.
- La app de móvil tendrá un uso diferente dependiendo de si el usuario es un negocio o un consumidor.

- La app de móvil utilizará la ubicación para realizar la búsqueda de negocios cercanos para el usuario.
- La app tendrá servicio a domicilio.
- El lanzamiento de la app tendrá su difusión mediante redes sociales y anuncios en internet y televisión.
- La app debe de ser visual, manejable y con una correcta y fluida usabilidad a nivel de usuario.
- Registro de usuario
- La app debe de ser segura y debe tener la política de privacidad.

+ Usuario negocio:

- Los negocios tendrán una valoración basada en las opiniones de sus clientes.
- El usuario con negocio tendrá la app dividida en 2 interfaces; muro del negocio y perfil del negocio.
- El usuario con negocio podrá subir sus productos a su muro con fotos, títulos, comentarios, ingredientes y precios.

+ Usuario consumidor:

- El usuario consumidor, cuando uno de los negocios cercanos, o de los que sigue, suba a su muro un producto nuevo, este recibirá una notificación.
- El usuario consumidor tendrá la app dividida en 3 interfaces; negocios cercanos, negocios a los que sigue y su perfil.
- El usuario consumidor podrá dejar comentarios y una nota a los respectivos pedidos que haya hecho.

B. SHOULD HAVE

- La app tendrá actualizaciones eventuales.
- La aplicación tendrá un apartado dedicado a la atención al cliente 24h que deberá de ser eficiente y resolutivo.
- La interfaz de la app para móviles tendrá el código de colores de la empresa.
- La app debe de ser responsive.
- Integración con las redes sociales.
- Notificaciones push.
- Los usuarios tendrán la posibilidad de personalizar y ajustar su perfil a su gusto.

C. COULD HAVE

- La app móvil tendrá modo oscuro.
- Chat en tiempo real.
- Integración de pagos móviles/online.

D. WON'T HAVE

- Animaciones excesivas
- Integración con tecnologías obsoletas (versiones de Android e iOS antiguas).

6. Diseño de solución

En cuanto al proyecto de 'Rescue Eats', se va a dividir en tres grandes partes, el servidor, el servicio y el cliente.

Para tener una buena estructura de proyecto y poder dividir el sistema en tres partes hemos decidido utilizar el modelo Cliente/Servidor³.

La arquitectura Cliente/Servidor es un tipo de desarrollo de *software*, que se basa en la asignación de tareas concretas entre dos elementos esenciales, el servidor y los clientes. En este modelo, los clientes son los demandantes de la petición al programa y el servidor es quien da la respuesta.

Una de las principales características de este modelo, es que el cliente da comienzo a las peticiones o solicitudes y, por consiguiente, tiene un papel activo en la comunicación. El cliente espera la respuesta del servidor y, de esta manera, pueda conectarse desde varios servidores; en numerosos casos, interactúa a través de una interfaz.

Por último, el servidor es quien recibe las peticiones de los clientes; y, a diferencia de ellos, se trata de un papel pasivo en la comunicación, ya que, espera la llegada de las peticiones y solicitudes de los clientes; posteriormente las procesa y envía las respuestas adecuadas. El servidor está preparado para atender a múltiples clientes al mismo tiempo sin perder su eficacia.

- Servidor

El servidor es la parte de *BackEnd* del proyecto. Dicha parte está formada por la BBDD y los servidores donde se almacenarán todos los datos que se necesiten recoger.

Para la gestión y el almacenamiento de la información, utilizaremos una base de datos relacional de tipo MySQL.

Se ha decidido escoger y trabajar con una base de datos de tipo relacional principalmente por los siguientes aspectos:

- **Sencillez:** las bases de datos de tipo relacional tienen un modelo que, a la hora de gestionar la BBDD, permite manejar grandes masas de datos e información. Además, este tipo tiene un modelo más sencillo que, por ejemplo, las bases de datos 'NoSQL'.
- **Conocimiento:** al haber sido el modelo más común e utilizado durante los últimos años, los programadores profesionales tienen un mayor conocimiento y control sobre este lenguaje, pues las BBDD SQL, por suerte, tienen ya definidos unos estándares para mantener una uniformidad y un criterio correcto.

³ Fuente: *¿Qué significa el término cliente-servidor?* (2023). Distritok.

- **Organización:** las bases de datos de tipo relacional están estructuradas por tablas con filas, columnas e índices. Esto facilita la organización, gestión y manipulación de los datos. Y el punto fuerte es que esta estructura permite establecer relaciones entre todas las tablas y permite definir restricciones para mantener la integridad y consistencia de los datos.
- **Flexibilidad:** las bases de datos relacionales se manejan mediante un lenguaje “*Structured Query Language*” (SQL) que nos permite realizar consultas de un nivel complejo para, teniendo la información específica de los datos, poder lograr una extracción. Esto nos hace más fácil la recuperación de información de manera eficiente y precisa.

La decisión de escoger MySQL como gestor para nuestra base de datos está basada en tres conceptos:

- **Es de código abierto:** MySQL es un sistema de gestión de bases de datos de código abierto, es decir, gratuito. Esto lo convierte en una opción asequible y accesible para nuestro proyecto.
- **Rendimiento y escalabilidad:** MySQL está diseñado para ofrecer un rendimiento eficiente y escalable. Puede manejar grandes volúmenes de datos y es capaz de procesar consultas y transacciones de manera rápida y eficiente. Que MySQL sea veloz con el procesamiento de datos es perfecto para la fluidez que necesita una aplicación para móviles como la nuestra.
- **Escalabilidad:** MySQL es escalable y puede manejar grandes conjuntos de datos y aplicaciones en crecimiento. Esta característica nos proporciona la opción de aumentar la base de datos si el producto sale como lo previsto, y así, no teniendo que hacer una emigración en caso de hacerse más grande por limitación.

En resumen, la elección de utilizar una base de datos relacional como MySQL se basa en la sencillez del modelo, el conocimiento y dominio existente, la organización de los datos, la flexibilidad en las consultas y los beneficios adicionales que ofrece MySQL en términos de código abierto, rendimiento y escalabilidad.

- Estructura de la BBDD

- **Servicio**

La parte del servicio estará formado por *API REST*. Este “*API REST SERVICE*” estará codificado en el lenguaje de programación *JAVA* por tres razones; la seguridad que este lenguaje aporta, la cantidad de librerías y ayudas que existen a día de hoy que se pueden utilizar con y la facilidad que este aporta a la hora de querer conectarte con otras tecnologías.

El “*API REST*” trabajará como nexo entre los dos bandos, el *BackEnd* y el *FrontEnd*. Para ello, se comunicará con ambos mediante peticiones y llamadas a la base de datos y a la aplicación para móviles.

A la hora de querer llamar a nuestra BBDD desde *API*, haremos uso del protocolo HTTP y de sus principales operaciones, que son las siguientes:

- **GET (Obtención de datos):** esta operación se utiliza para obtener o leer información de nuestra base de datos. Por ejemplo, cuando un usuario quiera ingresar en la aplicación mediante su perfil, deberemos comprobar si el usuario y la contraseña introducidos son los correctos.
- **POST (Inserción de nuevos datos):** dicha operación se emplea para crear o insertar un nuevo registro en nuestro servidor. El POST se puede utilizar cuando una persona quiera utilizar la aplicación, y para ello, se cree un nuevo usuario.
- **PUT (Actualizar datos):** esta operación se utiliza para actualizar o reemplazar un registro ya existente en nuestra BBDD. Un ejemplo de una situación donde el PUT sería usado es cuando un usuario actualiza un comentario, pues puede cambiar tanto el título, el cuerpo como la valoración o cuando un usuario acepta el envío de notificaciones push a su dispositivo.
- **PATCH (Actualización parcial de datos):** Se utiliza para realizar modificaciones parciales a un dato existente. La operación PATCH solo envía los cambios o actualizaciones específicos que se deban realizar en la base de datos sin la necesidad de reemplazarlo por completo.

Dado el caso de que un usuario quiera actualizar su dirección de correo a otra, primero se hará una comprobación de si ese correo ya está ligado a otro usuario, y si no es el caso, se actualizará el campo del correo de su usuario en la base de datos.

- **DELETE (Borrado de datos):** se utiliza para eliminar un recurso existente en el servidor. La solicitud DELETE elimina el recurso especificado. Por ejemplo, si un usuario quiere borrar un comentario que ha hecho a un negocio, se utilizaría esta operación.

- Cliente

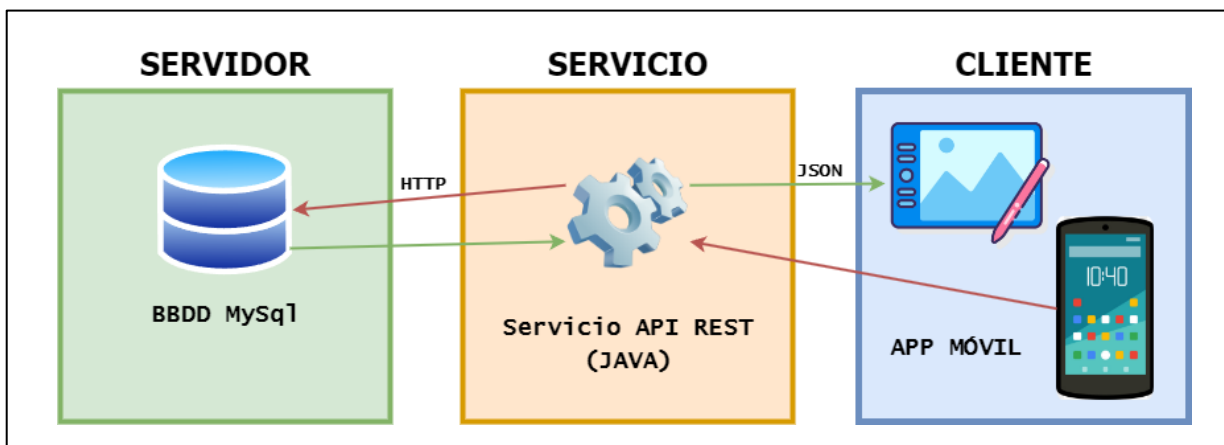
“*Rescue Eats*” es una App para móviles tanto Android como iOS. La aplicación estará hecha mediante el lenguaje *Python* en el caso de la app para Android y con el lenguaje *Swift* para los dispositivos iOS.

Para su creación, en primer lugar, se desarrollará la aplicación en *Python*, y cuando esta esté terminada, se procederá a hacer una copia con *Swift* para el resto de los dispositivos.

La aplicación para móviles será la interfaz gráfica desde la que cualquier usuario podrá utilizar la App y poder comprar los productos de los negocios de sus cercanías.

Para que la aplicación pueda utilizar la información guardada en la base de datos y la pueda mostrar en pantalla, se utilizará la *API REST* y se le harán diversas peticiones que dependiendo de la pantalla podrán ser de creación, inserción, actualización o borrado de datos. En dichas peticiones, encontraremos varias opciones, como, por ejemplo, crearse un usuario nuevo, crear un nuevo producto que voy a poner a la venta, escribir una reseña, eliminar algún producto que se ofrecía a los consumidores o el cambio de estado de una oferta de un producto ‘*express*’ a “Agotado” porque ya no queden más unidades disponibles.

Para la comunicación con *API REST* nos hemos inclinado por utilizar el formato de documento *JSON* pues es más legible y su sintaxis es más sencilla que la que utiliza *XML*.



7. Plan de despliegue

- Plan de pruebas

El plan de pruebas se llevará a cabo individualmente en sus respectivas sesiones como se explicó en la planificación temporal. En el momento donde se termine el desarrollo de la BBDD, del API REST o de la aplicación móvil se pasará a la tarea de 'testing'.

BBDD

Para testear la BBDD se utilizarán diferentes tipos de pruebas que no solo pondrán a prueba a la base de datos con su información sino también testeará el nivel de seguridad que esta tiene.

Lo primero que se hará será una prueba de integridad de datos donde se verificarán que los datos almacenados en la base de datos sean correctos y coherentes.

También se llevarán a cabo pruebas de seguridad contra la base de datos para comprobar la integridad y la robustez del servidor, y pruebas de capacidad y rendimiento bajo cargas de trabajo bajas, medias y altas.

Por último, se crearán respaldos de la base de datos para, en caso de que ocurriera cualquier error o problema, no desaparezca todo el trabajo.

API REST

Para la tarea de poner a prueba el servicio *API REST* se realizarán paso por paso desde una manera más individual a más colectiva.

Para empezar, se harán pruebas de lo más simples con cada llamada que se le puede hacer al API REST, que en este caso son: GET, POST, PUT, PATCH y DELETE.

Posteriormente nos centraríamos en sus capacidades, pues siendo la herramienta que conecta el servidor con el cliente es muy importante que en ningún momento falle. Para ello, se harían pruebas tales como pruebas de inyección de datos, manipulación de parámetros, autenticación y autorización.

También se realizarían pruebas de carga y rendimiento donde se evaluarían el rendimiento y la capacidad de respuesta del API bajo cargas de trabajo reales.

Como prueba final se llevaría el trabajo de la API a su extremo mediante pruebas de estrés. Las pruebas de estrés se realizan para evaluar el comportamiento del API bajo condiciones extremas o cargas de trabajo muy altas. Y, al obtener los resultados podemos conocer el límite de capacidad, la memoria y estabilidad, en cuanto a carga de trabajo, que nuestra API podrá soportar una vez la aplicación salga al mercado.

Aplicación para móviles

Una vez los programadores hayan probado por ellos mismos la app que han creado, las pruebas que se realizarán en el caso de la aplicación para móviles se dividirán en dos bandos.

Por una parte, serán los mismos programadores los que pongan a prueba el *software* mediante pruebas de seguridad desde la misma app (por ejemplo, con inyecciones también se podría probar la base de datos), pruebas de compatibilidad pues en nuestro caso la aplicación saldrá en dos mercados (Play Store y Apple Store) y pruebas de estabilidad.

Por otro lado, se proporcionarán a distintas personas una copia de nuestra aplicación para que hagan pruebas con ella a nivel de usuario y del uso que, una vez salga al mercado, se le dará.

Gracias a esto último se podrían realizar pruebas que afectarían a tres bandas. Pues al hacer tareas como crear un perfil, añadir un producto, seguir a un negocio, etc. Se podrá comprobar, al mismo tiempo, que la app funciona correctamente, que la *API REST* recoge y lleva datos de un lado a otro cuando debe y que los datos introducidos por los usuarios se guardan correctamente en la base de datos.

- Manuales

En nuestro caso, al ser una aplicación que está en los dos mercados de aplicaciones más conocidos y usados para móviles, no vemos necesario crear un manual de instalación.

Pero en el caso de un manual de usuario, pues va dirigido a los usuarios que van a utilizar el *software* una vez salga al mercado. En dicho *software*, se proporcionarían las instrucciones detalladas sobre cómo utilizar la aplicación, cómo acceder a sus funciones o distintas pantallas y cómo solucionar los problemas comunes.

Este manual incluiría:

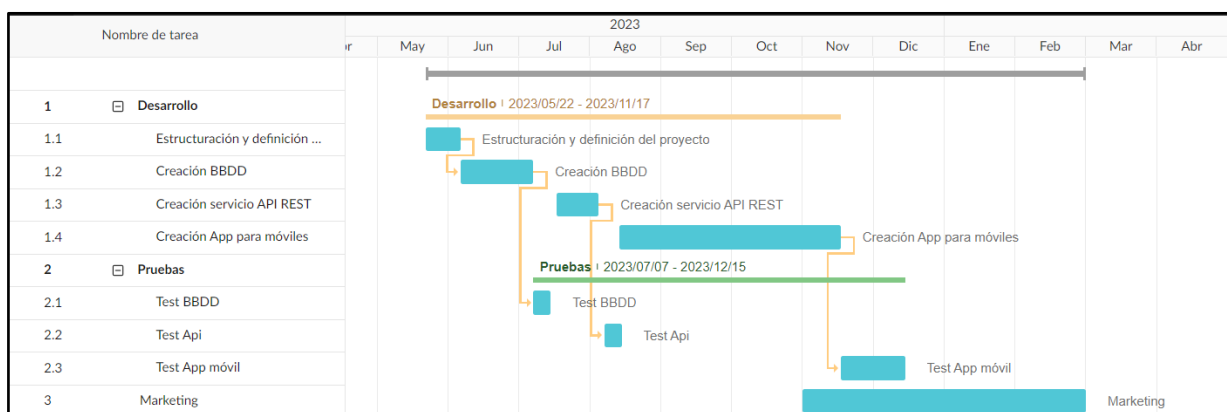
- Un índice donde se desglosaría cómo funciona la App pantalla por pantalla empezando por iniciar sesión o registrarse en nuestro sistema.
- Dentro de cada punto, se hablaría sobre esa pantalla explicando a detalle cada botón o espacio de la pantalla y cuál es su función.
- A su vez, cada apartado iría explicado con sus respectivas capturas de pantalla, tutoriales paso a paso y ejemplos de casos de uso.
- Y por último se crearía un apartado de preguntas frecuentes o FAQ donde se abordarían los problemas más comunes que le pueden ocurrir al usuario.
- Al ser un manual escrito en una página web, también se podría incluir soporte de atención al cliente (también se incluiría una ventana en la propia app desde donde se podría llamar al hacer clic al número de teléfono).

Las actualizaciones del manual, además de aplicarse el cambio en la propia web, por redes sociales se notificaría del cambio mediante un comunicado.

A parte del manual de usuario también se crearían manuales para los programadores como un manual de desarrollo donde se explicaría qué requisitos pide la aplicación, un manual de pruebas donde se enumerarán y describirán las pruebas que se realizarán y un manual de mantenimiento para el continuo desarrollo de la aplicación una vez se haya sacado al mercado.

- Plan de despliegue

El tiempo durante el cual se daría a conocer la aplicación es un poco antes de su salida al mercado como se detalló en la planificación temporal en la tarea de Marketing.



Una de las principales acciones sería la creación de perfiles en las redes sociales más relevantes, donde se detallaría el propósito del proyecto y se anunciaría la próxima salida de la aplicación para dispositivos móviles.

A partir de la creación de estos perfiles, se realizaría una constante actualización de noticias, cambios, novedades y comunicados relacionados con el proyecto en general. El objetivo sería expandir la red de usuarios y hacer crecer la comunidad en torno a la aplicación; para lograr esto, se utilizarían diversas estrategias publicitarias.

Con respecto a la publicidad, se emplearían diferentes formatos y canales para maximizar el alcance. Esto incluiría la colocación de *banners* publicitarios en sitios web relevantes, la publicación de anuncios en las redes sociales donde se tenga presencia y la inclusión de anuncios en televisión.

Cada una de estas estrategias tienen sus beneficios, por ejemplo, la publicidad a nivel de *banners* en sitios web permitiría llegar a más usuarios que visitan esas plataformas. Los anuncios en redes sociales, además de dar a conocer la aplicación a través de las publicaciones en nuestro perfil, podrían ser amplificados mediante campañas publicitarias para llegar a un tipo de usuarios específico según intereses, necesidades...

Con respecto a los anuncios en televisión, podrían ser una estrategia menos económica, pero, a su vez, permitiría llegar a una audiencia más amplia y diversa. Dependiendo del presupuesto disponible, se podría optar por anuncios en televisión local, regional o incluso a nivel nacional, para generar un mayor impacto y aumentar la visibilidad de la aplicación.

En resumen, el plan de despliegue se basa en la creación de perfiles en redes sociales, la constante actualización de contenido relevante y la utilización de diferentes formatos de publicidad, como banners en sitios web, anuncios en redes sociales y anuncios en televisión; para así, dar a conocer la aplicación antes de su lanzamiento y conseguir un gran número de usuarios, generando interés y anticipación en el mercado.

8. Conclusiones y propuestas de mejora

Conclusiones

A modo de conclusión, “*Rescue Eats*” es una aplicación que puede servir para que las personas sean conscientes y se involucren en la lucha contra el desperdicio de alimentos.

“*Rescue Eats*” promueve la reducción del desperdicio alimenticio, lo que trae consigo una serie de beneficios a nivel económico y a nivel medioambiental; a nivel económico, las empresas de alimentación y los restaurantes, obtendrán un beneficio de la comida que, generalmente, desecharían; a nivel medioambiental, se obtienen beneficios en las cadenas de producción y distribución de los productos.

Esta App ofrece a los usuarios un servicio rápido, sencillo, eficiente y de calidad, para que, a la hora de generar pedidos, bien recogiénolos en tienda o bien a domicilio, no les causa ninguna dificultad, consiguiendo así, buenas opiniones para “*Rescue Eats*”.

Al mismo tiempo, los restaurantes y empresas de alimentación que se acogen a la filosofía de “*Rescue Eats*”, donarían el excedente de sus alimentos a un banco de alimentos u ONG, de manera que, estaría atendiendo a los más necesitados.

A pesar de todos los beneficios que tiene nuestra App, debemos ser críticos y conscientes de nuestras debilidades y las posibles amenazas externas que pueden hacernos competencia; por ello, hemos realizado un DAFO, cuya finalidad es ayudarnos a combatir y trabajar esos aspectos.



Propuestas de mejora

Para lograr marcar la diferencia con respecto a la competencia, hemos creado una propuesta o plan de mejora, que consiste en:

- **Ampliar la red de restaurantes y empresas de alimentación:** para aumentar la variedad de alimentos y platos, es importante expandir la base de restaurantes y comercios asociados; de esta manera, el usuario tendrá miles opciones de comida para escoger.
- **Mejorar la facilidad de uso de la aplicación:** la aplicación “*Rescue Eats*” tiene que ser fácil de usar y navegar tanto para los usuarios cuando realizan sus pedidos como para los restaurantes, ya que, deben de saber emplear la App para poder realizar de manera adecuada los pedidos.
- **Implementar un sistema de seguimiento:** establecer un mecanismo para recopilar y analizar cuántos usuarios emplean nuestra app y recopilar los datos sobre la satisfacción de los usuarios, los restaurantes más demandados... Esto permitirá evaluar el impacto de “*Rescue Eats*” y realizar mejoras basadas en los resultados obtenidos.
- **Mejorar la visibilidad y promoción:** utilizar estrategias de marketing y medios de comunicación social para aumentar la visibilidad de “*Rescue Eats*”. Se pueden incluir estrategias como, trabajar con distintos *influencers*, anuncios en las distintas redes sociales, anuncios publicitarios, en revistas, carteleros...

En resumen, con ayuda de esta propuesta de mejora, “*Rescue Eats*” crecerá y se fortalecerá; además, será imagen de la lucha contra el desperdicio de alimentos. “*Rescue Eats*” trabajará constantemente en la mejora de su aplicación, a través de la innovación tecnológica, nuevas estrategias de marketing, nuevas ideas para la App; ofreciendo así, un servicio de calidad y eficiente.

9. Bibliografía

- *Noticias ONU. El desperdicio de comida, una oportunidad para acabar con el hambre. (2018). Noticias ONU.*
- *ONU. Pérdidas y desperdicio de alimentos en el mundo. (2011). ONU.*
- *¿Qué significa el término cliente-servidor? (2023). Distritok.*
- *Innova, L (2017). Datos- Desperdicio alimentario. Desperdicio alimentario.*
- *Aprobada la nueva ley de desperdicio alimentario: el gobierno multará a las empresas que tiren comida. (2022). Hogarmania.*
- *Del Caño, G. (2022). Desperdicio alimentario, mucho más que tirar comida. ¿Cómo evitarlo? Tiempo.com / Meteored.*
- *Alonso, A. L. (2020). No tirar comida. Consejos y trucos para aprovechar restos. De Rechupete.*
- *Acosta, M. B. (2023). Desperdicio de alimentos: causas, consecuencias y cómo evitarlo. ecologiaverde.com*
- *Aquae. (2021). El desperdicio de alimentos y su efecto en el planeta - Fundación Aquae.*
- *El desperdicio masivo de alimentos, un problema no solo de los países ricos. (2021). Noticias ONU.*
- *BBC News Mundo. (2021). Las impactantes cifras que deja el desperdicio de comida en el mundo (y cuáles son sus efectos). BBC News Mundo.*
- *De Sevilla, D. (2022). Estas apps te ayudan a cocinar con los ingredientes que tienes en casa. Diario de Sevilla.*
- *Consejos para evitar el desperdicio de alimentos. (2019).*
- *Fecha de caducidad vs fecha de consumo preferente - Google Search. (2020).*
- *Lenguajes de programación para móvil. (2021). Immune Technology Institute.*
- *¿Cuánto tiempo se tarda el desarrollo una aplicación móvil para iOS o Android? – Zoom. (2021).*

- Portal, T. (2022). *Análisis de requisitos de software: ¿Cómo saber qué se necesita y a qué darle prioridad?* TIC Portal.
- Osman, J. (2022). *Ejemplos de la API REST. AppMaster - ultimate all-in no-code platform.*
- Stsepanets, A., & Stsepanets, A. (2023). *Modelo cascada, qué es y cuándo conviene usarlo. Gantt Chart GanttPRO Blog.*
- ¿Cuánta comida tiras? (2023). www.ocu.org.
- Robert Clark, National Geographic. (2017). *De las granjas a las tiendas o a nuestras mesas, el 30 por ciento de la comida que cultivamos, se pierde. Pero debemos ponerle remedio. National Geographic.*
- Atlassian. (s. f.). *Scrum: qué es, cómo funciona y por qué es excelente.*