

AñadaMaster: K-VATA

Memoria trabajo final

Antonio Cabrera, Alejandro Jiménez y Antonio Pérez

Trabajo para el doble grado de
Ingeniería del Software y Matemática Computacional



CENTRO UNIVERSITARIO
DE TECNOLOGÍA Y ARTE DIGITAL

Ingeniería Software

U-tad

España

Diciembre 2024

Contents

1	Metodología de la empresa	3
1.1	Investigación previa	3
1.1.1	Metodologías Ágiles	3
1.1.2	Metodologías de empresas	5
1.2	Metodología de nuestra empresa	6
2	Actas de entrevistas	8
2.1	Acta 16/10/2024	8
2.2	Acta 06/11/2024	10
3	Análisis de riesgos	11
4	Épicas y User Stories	14
4.1	Épicas	14
4.2	User Stories	14
5	Backlog	16
6	User Story Mapping	19
7	Casos de uso	21
7.1	CoreBodegas	21
7.2	CoreReservas	21
8	Diagrama de clases	23
9	Diagramas de secuencia	24
9.1	Buscar, añadir y modificar elementos	24
9.2	Inicio de sesión	25
9.3	Personal secuencia	26
10	Diagrama de Gantt	27
11	Acta de entrevista post mortem	29

List of Figures

5.1	Primera parte del Backlog	16
5.2	Segunda parte del Backlog	17
5.3	Tercera parte del backlog	17
5.4	Cuarta parte del Backlog	18
5.5	Quinta parte del Backlog	18
6.1	Pimera parte del mapa de las historias de usuario	19
6.2	Segunda parte del mapa de las historias de usuario	20
7.1	Diagrama de casos de uso de CoreBodega	21
7.2	Diagrama de casos de uso de CoreReservas	22
8.1	Diagrama de clases UML	23
9.1	Diagrama de secuencia de elementos	24
9.2	Diagrama de secuencia de inicio de sesión	25
9.3	Diagrama de secuencia del personal	26
10.1	Primera parte del diagrama de Gantt	27
10.2	Segunda parte del diagrama de Gantt	28

Chapter 1

Metodología de la empresa

1.1 Investigación previa

1.1.1 Metodologías Ágiles

Scrum

Es una forma de organizar el trabajo en ciclos cortos denominados sprints, estos ciclos suelen durar de entre 2 a 4 semanas, con la idea de aumentar la calidad del producto en cada entrega o añadir funcionalidades nuevas.

Dentro de esta metodología existen varios roles importantes Scrum Master (facilita los procesos y elimina obstáculos), Product Owner (gestiona el backlog y prioriza tareas) y Equipo de Desarrollo (Realiza el trabajo técnico).

Además, tiene ciertas características que lo distingue, como los distintos tipos de reuniones Daily Scrum (), Sprint Planning (para planificar el próximo ciclo de Sprint) y Sprint Review (tras terminar el sprint).

Funciona muy bien en proyectos con requisitos cambiantes.

Kanban

Un método visual de gestión de flujo de trabajo que utiliza un tablero Kanban para rastrear las tareas desde “Por Hacer” hasta “Terminado” (como un trello). Este método tiene el objetivo de limitar el trabajo en proceso, evitando la sobrecarga de este, y mejorar continuamente los procesos. Por otro lado, no hacen falta iteraciones fijas. Funciona muy bien en equipos con tareas recurrentes o impredecibles

Safe

Una forma de implementar las metodologías ágiles en empresas grandes, consiguiendo una coordinación entre varios equipos. Este método implementa el Scrum o/y el Kanban, ya que une varios equipos

ágiles que usan estas metodologías, aplicando dentro de estos las mismas características que se han destacado antes.

Con este método se agrupan varios equipos en un Agile Release Train, que se define como un “equipo de equipos” con el objetivo de entregar incrementos de gran valor. Se trabaja en Program Increments, que son ciclos de entre 8 a 12 semanas con múltiples sprints dentro de este.

Funciona muy bien con empresas grandes de miles de personas.

DevOps

Combina el desarrollo con las operaciones para automatizar y optimizar el ciclo de vida del software.

Cuenta con unos principios a seguir Integración continua (Automatiza la integración de código), Entrega continua (Automatiza el despliegue) y Colaboración (Incita la comunicación entre equipos). Esta práctica se lleva a cabo con el uso de herramientas como Jenkins, Docker y Kubernetes, necesitando una monitorización constante y feedback rápido.

Funciona muy bien en equipos con tiempo limitado.

Lean Development

Diseñada para optimizar el desarrollo de software. Tiene un enfoque en eliminar cualquier desperdicio, maximizar el valor del producto entregado al cliente y mejorar el proceso.

Con esta metodología se busca reducir el número de tareas innecesarias, el tiempo perdido y la duplicación de los esfuerzos; Previene errores y el diseño es óptimo desde un primer momento; busca el aprendizaje a lo largo del desarrollo; valora la entrega del producto rápido (ciclos cortos); Fomenta la colaboración entre el equipo y la autonomía de este; Mejorar el flujo general, no se centra en partes individuales del producto.

Funciona muy bien en equipos que buscan adaptabilidad y eficiencia.

Extreme Programming (XP)

Prioriza las necesidades del cliente, centrándose en la calidad de software.

Promueve la colaboración cliente-equipo, llevando a toma de decisiones rápidas. Incluyendo prácticas técnicas como integración continua y programación en pareja. Además de buscar un ambiente de trabajo respetuoso y sin miedo de tomar decisiones difíciles.

Prácticas de esta metodología:

- Desarrollo guiado por pruebas
- Programación en parejas
- Integración continua

- Pequeñas entregas
- Refactorización constante (Limpiar el código)
- Retroalimentación frecuente del cliente
- Simplicidad del diseño
- Propiedad colectiva del código
- Ritmo sostenible
- Pruebas funcionales
- Coding standards

Funciona muy bien con equipos pequeños.

1.1.2 Metodologías de empresas

En este apartado, estudiaremos algunas de las metodologías ágiles que utilizan empresas líderes en el ámbito tecnológico.

Amazon

Trabajan con un sistema de Working Backwards, comienzan pensando en la experiencia del cliente, por lo que hacen un comunicado de prensa y un FAQ antes del desarrollo. Una vez hecho el comunicado se basan en todos los datos que tienen para tomar una decisión, por lo que usan herramientas como A/B Testing.

Sus equipos siguen el concepto de 2 pizza teams, es decir, equipos pequeños, autónomos y multidisciplinarios, son equipos que pueden ser alimentados por dos pizzas. Al ser tan pequeños se tiene una toma de decisiones mucho más rápida.

A la hora de las memorias de funcionalidades, rechazan todo lo que son presentaciones y presentan documentos de seis páginas, para fomentar la claridad y profundidad.

Spotify

Trabajan con una metodología propia a la que denominan, Spotify Model. Este modelo organiza equipos de forma descentralizada, para ello se asignan Squads (equipos pequeños) que forman lo que llaman Tribes (grupos de squads que tienen un objetivo relacionado), y los Guilds (comunidades informales). También existen los chapters (grupos de personas con habilidades similares en diferentes squads).

Además, se usan los principios de Lean Startup y DevOps.

1.2 Metodología de nuestra empresa

Viendo todas las metodologías más usadas actualmente, nos hemos dado cuenta que no existe ninguna que junte metodologías administrativas con las de implementación práctica, por lo que vamos a crear una que cumpla con esto y que además tenga en cuenta el bienestar de nuestros empleados.

Lo primero es definir la estructura de los equipos, ya que son la parte más fundamental de un proyecto, sin los equipos sería muy complicado organizar a grandes grupos de personas, por ello hemos decidido usar dos tipos:

- Grupos: Son equipos formados por pocas personas (4-10) que están encargados de una pequeña parte del proyecto.
- Clanes: Un conjunto de grupos con un mismo objetivo conjunto, es decir, trabajan en una misma parte del proyecto.

El método a seguir para la planificación del proyecto está dividida en varias partes, cada una de estas partes es importante para un seguimiento acertado del proyecto y poder estimar el tiempo necesario para realizar próximas partes:

- Sprints: Son periodos de entre 1 a 3 meses, durante este tiempo se intentará completar en su totalidad un objetivo definido en la reunión de comienzo de sprint.
- Sub-Sprints: Son periodos de entre 1 a 2 semanas y conforman un sprint, al comienzo de este se especificarán las user stories que queremos completar, para ello nos reuniremos con el cliente quien priorizará cada una de las US seleccionadas y el equipo de desarrollo estimará el esfuerzo que llevará dicha tarea.
- Daily chat: Es una reunión diaria de unos 20-30 mins donde se discutirán problemas que hayan surgido a lo largo de ese día y lo que se va a avanzar el próximo.
- Emergencia: En caso de que se haya infravalorado una US y requiera más esfuerzo del establecido en la reunión del sub-sprint, se llamará a una reunión de emergencia donde junto al cliente discutiremos cómo solventar el obstáculo.
- Evaluation: Al finalizar un sub-sprint se hará una reunión con todo el equipo de desarrollo para exponer todos los avances realizados y prepararse para el siguiente sub-sprint.
- End: Una vez finalizado el sprint se establecerá una reunión con el cliente para enseñarle los avances conseguidos, verificar su valor y discutir qué camino tomar en el próximo sprint.

Una vez explicada la forma en la que se va a planificar cada uno de los grupos, expondremos las implementaciones prácticas para mejorar la calidad y el valor del producto entregado, además de mantener al cliente cerca del desarrollo, evitando futuras discusiones:

- En las reuniones de los sub-sprints contaremos con la presencia del cliente, el se encargará de asignar el valor que tiene cada una de US, de esta forma estaremos completamente seguros de la importancia de cada una de ellas, además de contar con la estimación de esfuerzo de los desarrolladores, esto facilitará el no asignar US inviables o demasiado complejas.

- Antes de comenzar la implementación de código, los desarrolladores definirán una serie de tests automatizados que el software deberá completar correctamente al finalizar su desarrollo, esto ayudará a conseguir una mejor funcionalidad de la aplicación y es una forma sencilla de poder comprobar el código.
- A la hora del diseño, se mantendrá un diseño lo más simple posible a lo largo del desarrollo para centrarnos principalmente en la calidad del código y su funcionalidad, una vez terminado el software se modificará el diseño de este para tener más estética.
- Al implementar código todos los desarrolladores pueden editar todas las partes del software, por lo que todos pueden editar cualquier zona, esto fomenta una responsabilidad compartida hacia el código, haciendo más eficaz la producción de este.

Otra parte importante será la transparencia dentro del proyecto, todas las actividades estarán visibles para todo el mundo en un único tablero, dentro de este se dispondrá de toda la información necesaria para llevar a cabo la tarea y quien la comprenderá, para poder ver todo de una forma más visual existen varios iconos que se les asignará a cada US para transmitir rápidamente información a los miembros del grupo.

Por último, queremos que cada miembro de cada grupo mantenga su bienestar, para esto se tendrá en cuenta cómo se siente cada uno de los integrantes emocionalmente. Se fomentan las pausas activas, es decir, descansos del trabajo a lo largo del día en unas zonas designadas para ello donde se contará con varios juegos o zonas para socializar con otros grupos, de esta forma se alivia el estrés y el malestar. Por otro lado, los miembros de la empresa contarán con la posibilidad de proponer actividades grupales o participar en ellas, donde sin importar tu grupo o clan podrás desarrollar todavía más tu carrera, aprendiendo nuevas tecnologías, estrategias, etc. Para terminar, se dispondrá de un psicólogo para mantener la salud mental de los empleados, estos podrán establecer citas cuando las necesiten.

Chapter 2

Actas de entrevistas

2.1 Acta 16/10/2024

ACTA DE LA ENTREVISTA REALIZADA AL CLIENTE PARA RESOLVER DUDAS SOBRE SU PETICIÓN. K-VATA, PARA EL DESARROLLO DE UN PROYECTO DE GESTIÓN Y VENTA DE RECURSOS.

En U-tad, a dieciséis de octubre de dos mil veinticuatro, siendo las 18:20 horas, se reúnen los siguientes integrantes de la empresa K-VATA para realizar la entrevista al cliente de la bodega.

- Antonio Pérez Márquez
- Antonio Cabrera Landín
- Alejandro Jiménez García

En la entrevista se realizaron las siguientes preguntas con la respuesta correspondiente:

1. ¿Si hay algún error en el inicio de sesión se bloquea o se realiza alguna otra acción?
 - (a) Al iniciar sesión cada uno de los empleados introduce su usuario y contraseña. Además, si un empleado se equivoca al introducir sus credenciales no habrá restricciones, es decir, los intentos serán ilimitados.
2. ¿Cómo acceden los administradores a CoreBodegas?
 - (a) Los administradores de la bodega (gerente de la bodega) accederán como todos los usuarios, sin embargo, serán los únicos que puedan acceder a la parte de personal.
3. ¿Existen productos predeterminados o se introducen todos los datos?
 - (a) Todos los productos deben ser introducidos por lo menos una vez.
4. ¿Qué datos hay que añadir para ingresar productos?

- (a) Cuando se agregue un producto se debe introducir un ID, una descripción, el precio (tanto en euros como en dólares), el formato y la cosecha.
- 5. ¿Los productos se pueden eliminar?
 - (a) Los datos se podrán ajustar, es decir, se podrán añadir o reducir la cantidad de producto. Pero no se podrá eliminar el producto (su cantidad mínima es 0).
- 6. ¿Se necesita algún tipo de validación o confirmación para ingresar datos?
 - (a) Tras ingresar los datos se podrán guardar a través de un botón que lo permite y a su vez lo valide.
- 7. ¿Qué datos del cliente/empresa se necesitan aparte del domicilio?
 - (a) El cliente consta de un ID, nombre, apellido, NIF, dirección, teléfono y fecha de cumpleaños.
 - (b) Las empresas constan de nombre, CIF, dirección, teléfono, dirección de entrega y descuento.
- 8. ¿Se requiere de una confirmación para eliminar o modificar algún elemento?
 - (a) Todos los datos que se agreguen o eliminen tiene un botón para guardar los cambios.
- 9. ¿Cuáles son los datos introducidos en el pedido?
 - (a) Los datos necesarios para un pedido son los siguientes: ID, ID del cliente, estado (trámite, cerrado, entregado y liquidado) y los detalles.
- 10. ¿Al realizar las modificaciones que tipo de confirmación es necesaria?
 - (a) Para cualquier tipo de modificación se requiere de un botón para aceptar los cambios y validarlos
- 11. ¿Qué tipos de datos se añaden en la factura a parte de la cantidad y tipo de producto?
 - (a) Los datos que deben ser introducidos en la factura serán los mismos que en el pedido más la dirección de entrega y nombre del cliente.
- 12. ¿Se necesitan variables extras?
 - (a) Se requiere de secciones que diferencien entre los productos que ha pedido el cliente.
- 13. ¿Qué datos del pedido se necesitan?
 - (a) Se necesitan los detalles del pedido. Que consta de cantidad de botellas, el tipo de producto, precio por producto, importe total del pedido con IVA y sin IVA
- 14. ¿Se imprime la factura digital y se facilita al cliente?
 - (a) La factura siempre se enviará al cliente. Además, dentro de la sección de productos existirá un botón para imprimir la factura.
- 15. ¿Se registran las facturas a los clientes del pedido?
 - (a) La factura se deberá guardar asociado al ID del cliente.

2.2 Acta 06/11/2024

ACTA DE LA ENTREVISTA REALIZADA AL CLIENTE PARA RESOLVER DUDAS SOBRE SU PETICIÓN. K-VATA, PARA EL DESARROLLO DE UN PROYECTO DE GESTIÓN Y VENTA DE RECURSOS.

En U-Tad, a seis de noviembre de dos mil veinticuatro, siendo las 17:30 horas, se reúnen los siguientes integrantes de la empresa K-VATA para realizar la entrevista al cliente de la bodega.

- Alejandro Jiménez García

En la entrevista se realizaron las siguientes preguntas con la respuesta correspondiente:

1. ¿Con respecto a la última vez se ha pensado algún cambio sobre el proyecto?
 - (a) Los clientes deben estar incluidos en la información del pedido.
2. ¿Las reservas de productos deben realizarse de alguna manera en específica?
 - (a) Al realizar la compra se guarda el pedido y se reduce la cantidad del producto pedido.
3. ¿Existe la prioridad en los pedidos?
 - (a) Los pedidos se tramitarán en el orden en que se hayan recibido.
4. Para usted cuál es el orden de prioridad de las tareas.
 - (a) 1. Gestión de Empleados/Materia 2. Cliente 3. Pedidos 4.Productos
5. ¿Hay alguna calificación más que quiera añadir a la uva? Por ejemplo, tipo de uva.
 - (a) El tipo de uva debería estar incluido, a parte de eso ningún dato más.

Chapter 3

Análisis de riesgos

Table 3.1: Evaluación de Riesgos del Proyecto

RIESGO	DESCRIPCION	ESTADO	PLAN	I	P	I*P
Tiempo reducido	No tenemos tiempo suficiente para la entrega del sistema pedido por el cliente	Mitigated	Al hablar con el cliente, dejar muy claro el tiempo que necesitamos para hacer lo que nos pide, o lo que podemos llegar a hacer en el tiempo que él quiere	0.9	0.8	0.72
No es factible la implementación del cliente	Lo que nos pide el cliente es algo muy complejo, no podemos llegar a conseguir esta implementación	Mitigated	Explicar al cliente que dicha funcionalidad no se puede añadir y las razones por las que no es posible	1	0.5	0.5
Pérdida o ruptura del código	El código de la aplicación se pierde o rompemos el código, y no disponemos de ningún backup	Resolved	Desde el comienzo del proyecto usaremos github, de esta forma tendremos un control de versiones	1	0.5	0.5

Continued on next page

Table 3.1: Evaluación de Riesgos del Proyecto (Continued)

RIESGO	DESCRIPCION	ESTADO	PLAN	I	P	I*P
Cambio de requisitos por parte del cliente	El cliente decide cambiar la mayor parte de la aplicación	Mitigated	Explicar al cliente todas las funcionalidades que tendrá la app y darle un plazo de tiempo para realizar los cambios que él desee, después de este plazo no podrá realizar más cambios	0.9	0.5	0.45
El cliente no comprende el software	Al realizar la entrega del software, el cliente no comprende como utilizar sus funciones, es decir, no comprende como usar nuestro software	Owned	Antonio Cabrera se ofrece a hacer una guía de uso de la aplicación, donde se explicarán todas las funcionalidades de esta	0.7	0.5	0.35
No se divide correctamente el trabajo	El trabajo se reparte erróneamente, entonces el trabajo no avanza adecuadamente	Owned	Antonio Pérez (CEO) decide como dividir correctamente el trabajo entre todos. Deberá ver que partes del proyecto dependen de otras, entre otras cosas	0.6	0.4	0.24
Fallo en la terminal	Las terminales del cliente no funcionan correctamente (no están actualizadas, etc...), no se puede implementar el software en las terminales	Accepted	Antes de comenzar el proyecto, comprobar si el software se puede implementar en las terminales del cliente, en caso de que esto no se pueda hacer, informar al cliente	0.7	0.3	0.21

Continued on next page

Table 3.1: Evaluación de Riesgos del Proyecto (Continued)

RIESGO	DESCRIPCION	ESTADO	PLAN	I	P	I*P
El cliente no comprende el alcance del software	Al entregar el código el cliente piensa que la aplicación tiene más opciones de las que tiene realmente	Mitigated	Desde un principio tenemos que explicarle al cliente con mucho detalle que va a tener la aplicación y lo que hará cada funcionalidad	0.6	0.3	0.18
No hay suficiente almacenaje	El almacenaje en la nube contratado no es suficiente para todos los datos, el cliente tiene más datos de los que podemos almacenar	Owned	Alejandro Jiménez contactará con la empresa suministradora del almacenaje. Para contratar más capacidad en el almacenaje del proyecto	0.8	0.2	0.16
El almacenaje en la nube se cae	La tecnología que usamos para el almacenaje de los datos se cae y deja inactivo todo el sistema	Accepted	Realizar una queja formal a la empresa suministradora de la nube y esperar a que se solucione	0.9	0.1	0.09

Chapter 4

Épicas y User Stories

4.1 Épicas

Hemos agrupado nuestras user stories en 7 épicas:

1. Gestión de usuarios
2. Gestión de productos
3. Gestión de pedidos
4. Gestión de facturas
5. Gestión de clientes
6. Gestión de login
7. Gestión de base de datos

4.2 User Stories

1. Gestión de base de datos:
 - (a) Como dueña quiero que no se puedan eliminar los datos.
Requisito: Los datos se pueden poner a 0, pero no eliminar.
 - (b) Como dueña quiero que exista una confirmación y validación de los datos antes de que sean introducidos.
Requisito: Botón para guardar los datos, funciones que validan dichos datos.
2. Gestión de clientes:
 - (a) Como cliente quiero que mis datos de compra se guarden, para pedir la misma cantidad de productos en mi próxima compra.
Requisito: Añadir función de volver a pedir (guardar antiguos pedidos para volver a pedir).

- (b) Como dueña quiero que existan dos tipos de clientes, empresas e individuales.
Requisito: Crear dos tablas en la base de datos, para individuales y otro para las empresas.
 - (c) Como dueña quiero que se guarden los cumpleaños de mis clientes, para poder felicitarlos y enviarles una caja regalo.
Requisito: Sistema de alerta de cumpleaños de los clientes.
3. Gestión de facturas:
- (a) Como cliente quiero recibir una factura con todos los datos del pedido
Requisito: Creación de factura con todos los datos necesarios.
 - (b) Como trabajador quiero que la factura se haga sola y así ahorrar tiempo.
Requisito: Devolver la factura completa sobre un pedido seleccionado.
4. Gestión de login:
- (a) Como dueña quiero que mis trabajadores no bloqueen la aplicación por poner la contraseña mal varias veces, para que no se quede un empleado sin trabajar.
Requisito: Que no exista un límite de intentos al introducir la contraseña.
 - (b) Como dueña quiero que la forma de iniciar sesión sea introduciendo el email y la contraseña de cada uno de mis trabajadores.
Requisito: Cuentas con email y contraseña.
5. Gestión de productos:
- (a) Como trabajador quiero poder añadir productos de forma eficaz, tener todos los productos existentes y modificar la cantidad almacenada.
Requisito: UX fácil de entender y simple, poder modificar las cantidades.
6. Gestión de pedidos:
- (a) Como trabajador quiero poder ver todos los datos de cada pedido, el estado en el que se encuentra, y el importe total con IVA y sin IVA.
Requisito: Base de datos de pedidos con variable de estado, **importe+IVA**, **importe-IVA**.
7. Gestión de usuarios:
- (a) Como dueña quiero que solo exista un administrador que sea mi gerente.
Requisito: Role de administrador, únicamente para el gerente de la bodega.
 - (b) Como trabajador voy a poder acceder a todos los campos de la aplicación, excepto la gestión de empleados.
Requisito: Dejar acceso a los trabajadores de todo.

Chapter 5

Backlog

US	PRIORIDAD	ESTIMACIÓN	DESCRIPCIÓN
GESTIÓN DE BASE DE DATOS Como dueña quiero que exista una confirmación y validación de los datos antes de que sean introducidos.	● URGENTE	S	Botón para guardar los datos, funciones que validan dichos datos.
Como dueña quiero que no se puedan eliminar los datos.	● IMPORTANTE	XS	Los datos se pueden poner a 0, pero no eliminar.
GESTIÓN DE FACTURAS Como trabajador quiero que la factura se haga sola y así ahorrar tiempo.	● URGENTE	M	Método para añadir nuevos clientes a la base de datos
Como cliente quiero recibir una factura con todos los datos del pedido.	● IMPORTANTE	M	Añadir función de volver a pedir (guardar antiguos pedidos para volver a pedir).

Figure 5.1: Primera parte del Backlog

ÉPICA/US	PRIORIDAD	ESTIMACIÓN	DESCRIPCIÓN
GESTIÓN DE CLIENTES			
Como dueña quiero que existan dos tipos de clientes, empresas e individuales.	URGENTE	S	Crear dos tablas en la base de datos, para individuales y otro para las empresas.
Como cliente quiero que mis datos de compra se guarden, para pedir el mismo pedido de productos la próxima vez.	URGENTE	S	Creación de factura con todos los datos necesarios (cliente, producto, precio, etc).
Como trabajador quiero poder buscar a los clientes.	IMPORTANTE		Buscador con filtro, de clientes.
Como dueña quiero que se guarden los cumpleaños de mis clientes, para poder felicitarlos y enviarles una caja regalo.	NORMAL	XS	Sistema de alerta sobre los cumpleaños de los clientes.
Como dueña quiero una forma fácil de añadir nuevos clientes.	NORMAL	S	Devolver la factura completa sobre un pedido seleccionado.

Figure 5.2: Segunda parte del Backlog

ÉPICA/US	PRIORIDAD	ESTIMACIÓN	DESCRIPCIÓN
GESTIÓN DE LOGIN			
Como dueña quiero que la forma de iniciar sesión sea introduciendo el email y la contraseña de cada uno de mis trabajadores.	URGENTE	S	Creación de cuentas de usuario con email y contraseña, además de otros campos.
Como dueña quiero que mis trabajadores no bloqueen la aplicación por poner la contraseña mal varias veces, para que no se quede un empleado sin trabajar.	IMPORTANTE	XS	Que no exista un límite de intentos al introducir la contraseña.
GESTIÓN DE PRODUCTOS			
Como trabajador quiero un apartado donde ver todos los productos fácilmente.	URGENTE	S	UX fácil de entender y simple, poder modificar las cantidades.

Figure 5.3: Tercera parte del backlog

ÉPICA/US	PRIORIDAD	ESTIMACIÓN	DESCRIPCIÓN
GESTIÓN DE PRODUCTOS			
Como trabajador quiero guardar los datos de los productos.	IMPORTANTE	XS	Poder añadir productos a la base de datos
Como trabajador quiero poder buscar un producto específico.		S	
GESTIÓN DE PEDIDOS			
Como trabajador quiero poder añadir productos a la aplicación.	IMPORTANTE	XS	Función para agregar nuevos valores de pedidos.
Como trabajador quiero poder ver todos los datos de cada pedido.		S	Página de pedidos donde se pueden ver todos, con las cantidades y el cliente.
Como trabajador quiero buscar un pedido en específico rápidamente.		S	Buscador de pedidos, con un filtro para las diferentes variables.

Figure 5.4: Cuarta parte del Backlog

ÉPICA/US	PRIORIDAD	ESTIMACIÓN	DESCRIPCIÓN
GESTIÓN DE USUARIOS			
Como dueña quiero que solo exista un administrador, que sea mi gerente.	URGENTE	XS	Crear roles dentro de la aplicación, pero que solo haya una cuenta con el role de admin.
Como trabajador quiero poder acceder a todos los campos de la aplicación, excepto la gestión de empleados.		S	Permitir el acceso a la web a las personas que tengan una cuenta, pero no dejarles entrar en la sección de empleados.
GESTIÓN DE EMPLEADOS			
Como gerente quiero añadir nuevos empleados y poder modificar los ya existentes.	IMPORTANTE	S	Permitir que el gerente (admin) pueda añadir nuevos empleados a la base de datos.
Como gerente quiero buscar un empleado específico fácilmente.		S	Buscador de empleados, con un filtro de diferentes variables.

Figure 5.5: Quinta parte del Backlog

Chapter 6

User Story Mapping

El mapa de historias de usuarios es una forma muy visual de organizar las historias de usuario y sus tareas relacionadas.

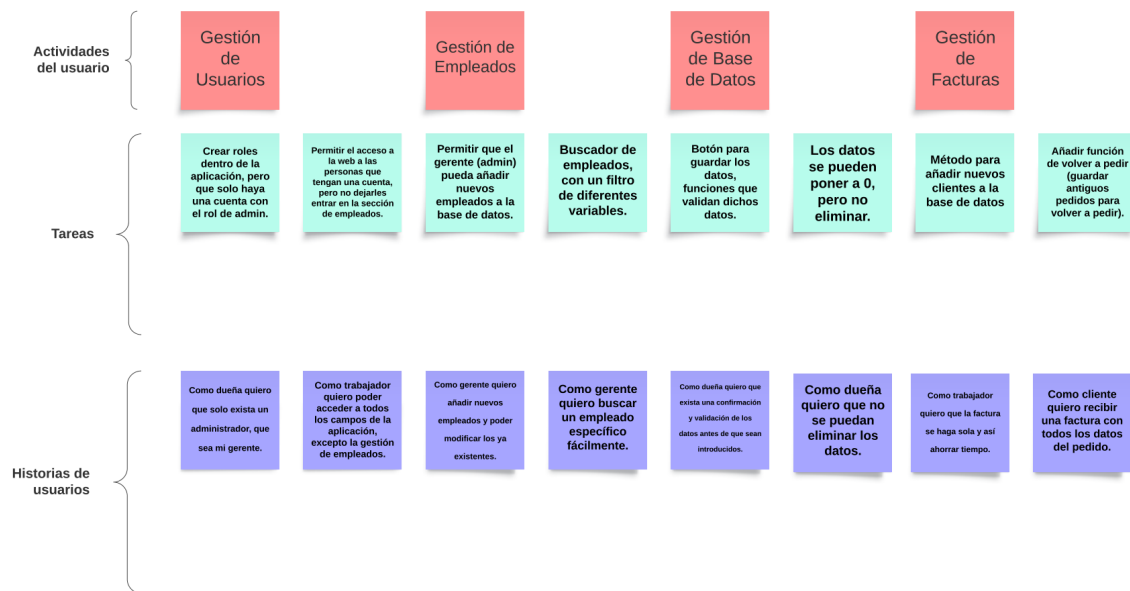


Figure 6.1: Pimera parte del mapa de las historias de usuario

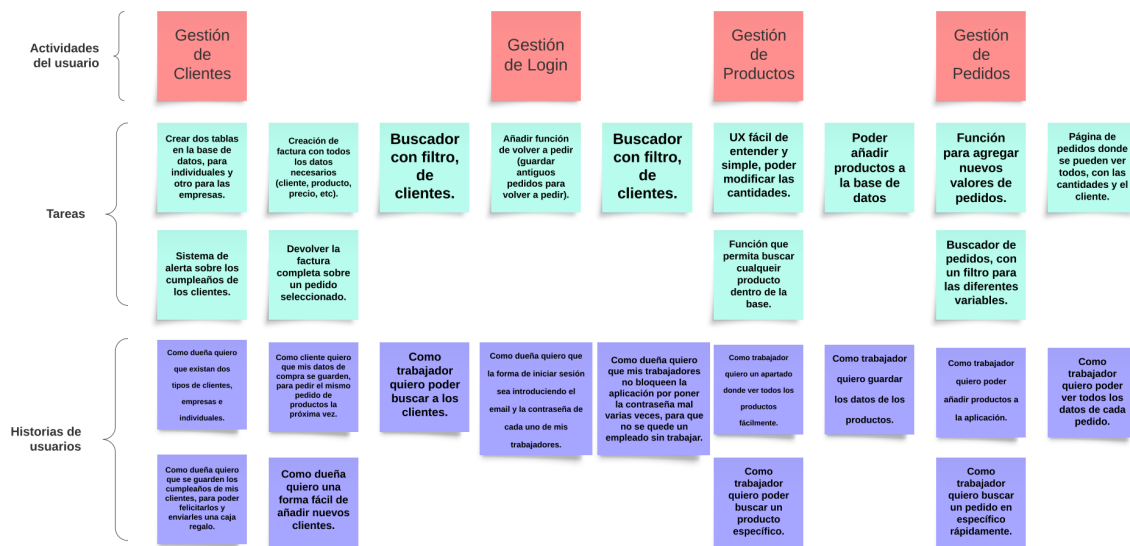


Figure 6.2: Segunda parte del mapa de las historias de usuario

Chapter 7

Casos de uso

7.1 CoreBodegas

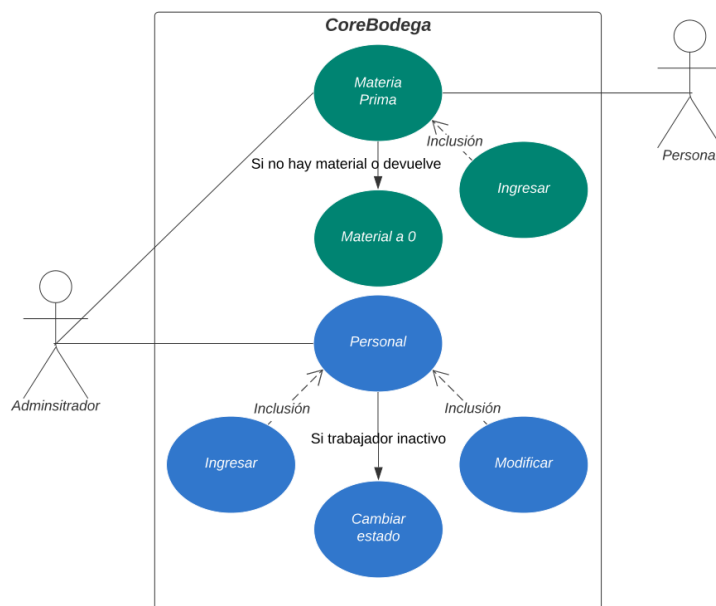


Figure 7.1: Diagrama de casos de uso de CoreBodega

7.2 CoreReservas



Figure 7.2: Diagrama de casos de uso de CoreReservas

Chapter 9

Diagramas de secuencia

9.1 Buscar, añadir y modificar elementos

En este diagrama de secuencia se indica como funcionarían las funciones de buscar, añadir y modificar elementos de la bodega.

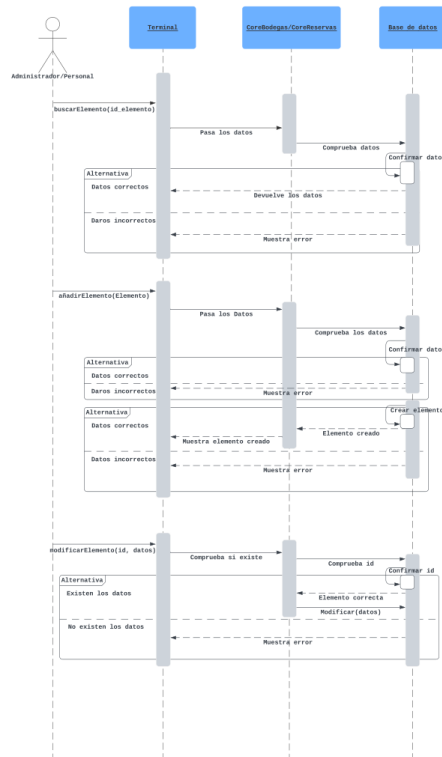


Figure 9.1: Diagrama de secuencia de elementos

9.2 Inicio de sesión

En el segundo diagrama de secuencia veremos como funcionaría el programa de inicio de sesión.

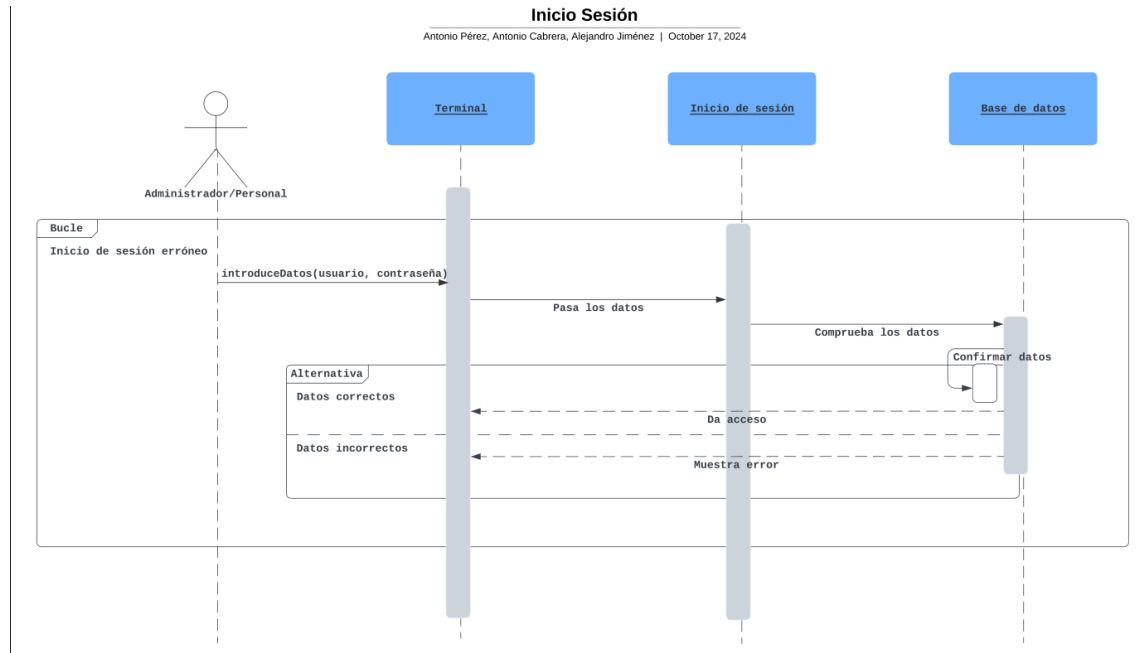


Figure 9.2: Diagrama de secuencia de inicio de sesión

9.3 Personal secuencia

Por último, en el siguiente diagrama se muestra el funcionamiento de las funciones que tienen que ver con la manipulación del personal.

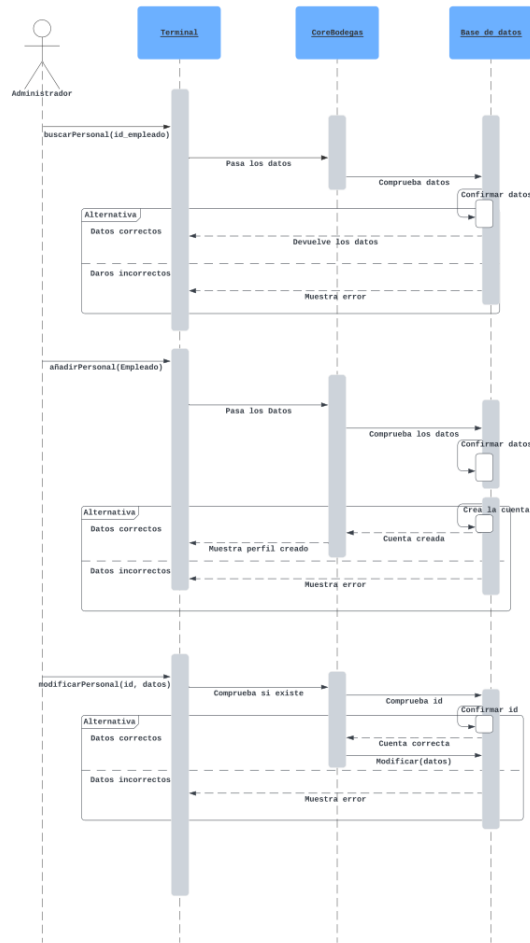


Figure 9.3: Diagrama de secuencia del personal

Chapter 10

Diagrama de Gantt

El diagrama de Gantt es una herramienta que utilizan los equipos para ver de forma visual las tareas que hay que realizar, con el tiempo estimado para cada una.

En nuestro caso, hemos dividido el diagrama por semanas a lo largo de los meses de octubre, noviembre y diciembre. Cada tarea es una user storie que deberá estar implementada en el tiempo indicado.

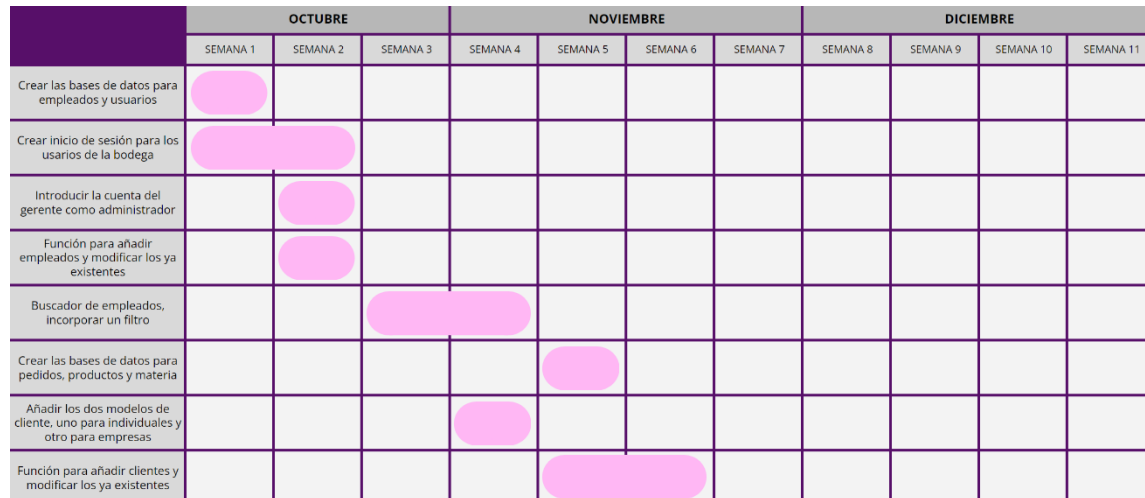


Figure 10.1: Primera parte del diagrama de Gantt

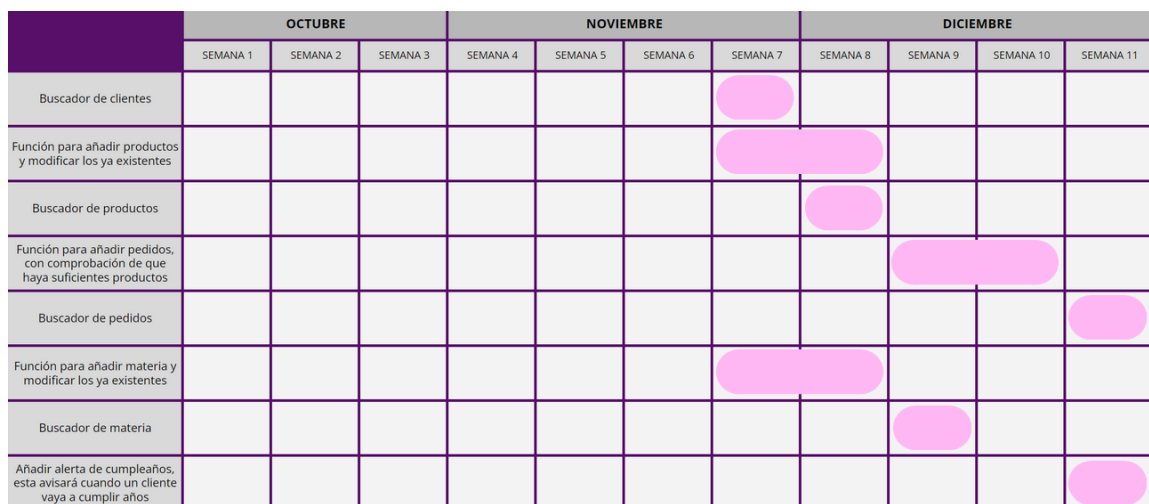


Figure 10.2: Segunda parte del diagrama de Gantt

Chapter 11

Acta de entrevista post mortem

ACTA DE LA ENTREVISTA POST MORTEM REALIZADA POR LOS PARTICIPANTES DE LA EMPRESA K-VATA, PARA REVISAR EL DESARROLLO DEL PROYECTO.

En U-tad, a dieciséis de diciembre de dos mil veinticuatro, siendo las 19:00 horas, se reúnen los siguientes integrantes de la empresa K-VATA para debatir sobre los aspectos realizados durante el desarrollo del proyecto.

- Alejandro Jiménez García
- Antonio Cabrera Landín
- Antonio Pérez Marquéz

En la reunión se realizaron las siguientes aportaciones:

1. ¿Se ha trabajado correctamente durante todo el proyecto?
 - (a) Antonio Cabrera: Si, en general se ha aplicado una buena metodología de trabajo durante todo el proyecto.
 - (b) Antonio Pérez: Si, desde un primer momento hemos trabajado en conjunto, discutiendo en grupo cómo hacer las partes del proyecto y que cosas no teníamos claras.
 - (c) Alejandro Jiménez: Si, desde un principio se decidió que todo se discutía con los demás integrantes para poder aclarar y resolver todos los problemas.
2. ¿La organización ha estado impuesta desde un principio?
 - (a) Antonio Cabrera: No, al principio estábamos menos organizados, pero a lo largo de las semanas hemos ido mejorando nuestra organización.
 - (b) Antonio Pérez: No, al comienzo no seguíamos ningún método de organización, no ha sido hasta la última entrega cuando hemos aplicado un buen método de como organizarnos el trabajo.
 - (c) Alejandro Jiménez: No, nos costó al principio organizarnos fue bastante complicado porque no teníamos claras todas las necesidades del cliente. Pero tras saber todo, nos organizamos y realizamos un buen trabajo.

3. ¿Todos los integrantes del proyecto han trabajado adecuadamente?
- (a) Antonio Cabrera: Si, todos han trabajado adecuadamente.
 - (b) Antonio Pérez: Si, nos hemos dividido los trabajos de forma equitativa y los tres hemos trabajado una misma cantidad de tiempo, cumpliendo con las fechas y lo que teníamos que entregar.
 - (c) Alejandro Jiménez: Si, en los trabajos todos los integrantes hemos participado aunque hayan estado “divididos”.
4. ¿Se ha entendido desde un principio lo que el cliente nos ha pedido?
- (a) Antonio Cabrera: No del todo, no fue hasta que tuvimos las reuniones con el cliente que entendimos realmente lo que quería.
 - (b) Antonio Pérez: Si, desde el principio teníamos la idea correcta de lo que quería el cliente, aunque hubieran ciertas cosas que no estaban explicadas claramente, pero estas dudas las hemos resuelto en las reuniones con el cliente.
 - (c) Alejandro Jiménez: A medias, desde un principio tuvimos una idea de lo que necesitaba el cliente, pero no fue hasta la primera reunión que aclaramos todas esas dudas y pudimos trabajar al cien por cien.
5. ¿Ha existido una buena comunicación entre los integrantes?
- (a) Antonio Cabrera: Si, la comunicación entre los miembros del equipo ha sido fluida.
 - (b) Antonio Pérez: Si, hemos comunicado todo lo que había que hacer en cada momento, y se ha transmitido claramente quien se encarga de que parte del trabajo.
 - (c) Alejandro Jiménez: Si, todos los avances que se han producido durante el desarrollo del trabajo se han ido comunicando, por lo que ha habido una buena comunicación
6. ¿Puedes mencionar tres cosas que salieron bien durante este proyecto?
- (a) Antonio Cabrera: Las entrevistas con el cliente, el ambiente en el equipo y la recogida de requisitos
 - (b) Antonio Pérez: La comunicación entre los miembros del grupo, el reparto de actividades entre cada miembro del equipo y la preparación a las reuniones con el cliente.
 - (c) Alejandro Jiménez: La buena relación que hemos tenido, que ha ayudado a que haya una buena comunicación; el desarrollo de los UML, que nos permitieron darle una mayor perspectiva al trabajo; y las presentaciones al cliente, que nos permitieron pulir nuestros errores o cambiarlos totalmente.
7. ¿Puedes mencionar tres cosas que no salieron bien durante este proyecto?
- (a) Antonio Cabrera: Solo puedo encontrar dos cosas que no salieron del todo bien: la primera vez que hicimos las épicas no salieron bien del todo; y la segunda cosa que se puede mejorar es la constancia de trabajo ya que algunas semanas trabajamos mucho y otras menos.

- (b) Antonio Pérez: La constancia en el trabajo, algunas semanas trabajamos demasiado en los proyectos de la asignatura, lo que llevó a que las siguientes semanas no tuviéramos tantas ganas de realizar el trabajo; la organización, no nos hemos organizado desde un primer momento; y no comenzar con tiempo el prototipo de la aplicación.
 - (c) Alejandro Jiménez: Desde un principio no hemos sido constantes, lo que ha generado épocas de gran cantidad de trabajo; la realización de las historias de usuarios, al principio no teníamos claro cómo crearlas pero conseguimos solucionarlo; y el prototipo, nos atrasamos con las demás partes lo que llevó a no terminarlo.
8. ¿Tienes alguna sugerencia sobre cómo podemos mejorar para el próximo proyecto?
- (a) Antonio Cabrera: Podríamos agendar reuniones cada dos días para mantener una constancia adecuada.
 - (b) Antonio Pérez: Aplicar una metodología ágil desde un primer momento (como la que proponemos nosotros), para organizarnos mejor; y tomar pausas al realizar los trabajos, para no estar quemados en las siguientes tareas.
 - (c) Alejandro Jiménez: Se podrían aplicar metodologías como SCRUM para poder llevar mejor el proyecto sin que haya grandes acumulaciones de trabajo.
9. ¿Te gustaría volver a trabajar en un proyecto como este? ¿Por qué o por qué no?
- (a) Antonio Cabrera: Me gustaría más trabajar en un trabajo práctico ya que este era muy teórico.
 - (b) Antonio Pérez: Si, me gusto mucho el ambiente de trabajo, además, me ha gustado realizar un proyecto desde cero. Y tengo ganas de aplicar desde un primer momento estrategias de trabajo que hemos ido descubriendo.
 - (c) Alejandro Jiménez: La realización de este proyecto nos ha dado una perspectiva nueva de cómo se realizan los trabajos y no como habíamos hecho con anterioridad, pero me gustaría trabajar más en una parte práctica.
10. Después de esta experiencia, si pudieras volver a hacer este proyecto, ¿qué harías de forma diferente?
- (a) Antonio Cabrera: Intentaría hacer una prueba de concepto del proyecto funcionando de verdad.
 - (b) Antonio Pérez: Empezaría con el desarrollo del prototipo mucho antes, en vez de esperar a que el resto de apartados estén bien pulidos, de esta forma podríamos tener un feedback del cliente a medida que avanzamos con la aplicación.
 - (c) Alejandro Jiménez: Desde un principio realizaría una agenda para no acumular trabajo.