

Product Backlog

QueMenges

Autors:

Yang Chen

Daniel Alberto Vaca Davalos

Nour Kobbi

Liyang Sun

1 Visión del usuario.

1.1 Gestión de cuentas y creación de perfiles de usuario.

- **Historia 1** Como usuario quiero poder mirar los menús diferentes a cada momento para así saber las diferentes opciones de comida a elegir.
- **Historia 2** Como usuario quiero poder identificarme como usuario o por plataformas altamente usadas como Google o Outlook, para poder facilitar el uso y registramiento dentro de la aplicación.
- **Historia 3** Como usuario registrado quiero poder actualizar en cualquier momento mi perfil, para mantener mis datos actualizados.
- **Historia 4** Como usuario registrado quiero dar de baja mi perfil en cualquier momento.
- **Historia 5** Como usuario quiero elegir el tipo de cuenta que crearé (cliente, repartidor o cocinero), para determinar mi rol en la aplicación.
- **Historia 6** Como usuario quiero poder ser los 3 tipos de roles de la aplicación, para adaptarme a mis necesidades.

1.2 Realización de comandas.

- **Historia 7** Como usuario quiero poder tener una preferencia de platos, a través de datos proporcionados en mi perfil, como pueden ser alérgenos, de esta manera los primeros platos que me salgan serán platos de mi agrado.
- **Historia 8** Como usuario quiero poder ver el tiempo de espera de la entrega de los platos para así poder tener un rasgo más a elegir entre las diferentes opciones.
- **Historia 9** Como usuario quiero poder tener una lista de platos o menús individuales como también de cocineros que ofrecen sus servicios culinarios, en este último caso además, querría poder ver los platos que ofrece dicho cocinero, todo esto para poder tener diferentes vistas a elegir, no solo platos como tal, sino también su cocinero.
- **Historia 10** Como usuario quiero que la lista que se me muestre, sea de cocineros o platos, este ordenado según la distancia física entre usuario y cocinero, así habrá un rasgo más que me hará decantar por un plato u otro.
- **Historia 11** Como usuario quiero poder tener un buscador tanto de platos como de cocineros, esto ayudará a poder tener una búsqueda más específica en el caso de que sepa de antemano que o de quien querré pedir.
- **Historia 12** Como cocinero quiero poder aparecer en una lista de no disponibles, que indique mi horario de disponibilidad, de esta manera los usuarios sabrán cuando no y cuando si enviarme comandas.
- **Historia 13** Como usuario quiero poder reservar un plato a un cocinero para una hora a la que esté disponible, así no hay que estar atento a que llegue la hora para poder pedir, ya se deja pedido de antemano.
- **Historia 14** Como usuario quiero poder revisar los ingredientes que tiene un plato, como también modificar alguno que sea indicado por el cocinero como opcional. Así puedo quitar aquellos ingredientes que no sean de mi agrado.

- **Historia 15** Como usuario quiero poder consultar los alergenicos, para así poder saber si ese plato puede ser consumido o no.
- **Historia 16** Como cocinero y usuario quiero que una vez el plato haya sido seleccionado se cierre la comanda y se proceda el envío, así evitamos que haya errores.
- **Historia 17** Como usuario, no podré tener más de tres comandas activas, así se garantiza el correcto funcionamiento de todo.
- **Historia 18** Como usuario quiero poder mirar todas las comandas activas como también las hechas en el pasado, con la opción de poder repetir una misma comanda si es necesario, de esta manera puedo controlar mi comanda a la vez que agilizo el proceso de pedir reiterativamente una misma comanda.

1.3 Envío de comandas.

- **Historia 19** Como cliente quiero elegir entre los 3 métodos de entrega, para elegir la opción que más me convenga.
- **Historia 20** Como cliente quiero que el precio de entrega se calcule en la aplicación directamente, para así saber qué opción de entrega me conviene más.
- **Historia 21** Como cliente que ha pagado un pedido quiero seguir el progreso de mi envío en todo momento, para saber qué mi comida ya está en camino.
- **Historia 22** Como cliente que ha pagado un pedido quiero seguir en todo momento por donde va el repartidor con mi pedido para saber cuánto le falta para llegar.

1.4 Pago de comandas.

- **Historia 23** Como cliente que ha pagado un pedido quiero seguir el pedido en Google Maps para saber visualmente por donde va.
- **Historia 24** Como cliente, quiero poder elegir entre pagar con tarjeta bancaria o Paypal, para poder pagar con el método que me convenga.
- **Historia 25** Como cliente, quiero que se me redirija a la plataforma correspondiente a autenticar mis datos, para poder completar la transacción de manera segura.
- **Historia 26** Como repartidor o cocinero, quiero que la aplicación ingrese directamente el pago del pedido a mi cuenta bancaria para recibir mi pago correctamente.
- **Historia 27** Como repartidor o cocinero, quiero vincular mi cuenta bancaria a mi perfil en cualquier momento para recibir todos los pagos a la cuenta especificada.

1.5 Valoración de las comandas.

- **Historia 28** Como cliente quiero poder valorar la comanda recibida, pudiendo indicar diferentes aspectos como puntualidad o estado de la comida, de esta manera puedo contribuir a otros usuarios a la hora de decidir que o de quien comer.
- **Historia 29** Como cocinero quiero que esas valoraciones se puedan asociar a mí y que me dé una pequeña ventaja a la hora de salir en las listas de cocineros o platos disponibles, de esta manera se recompensará mi esfuerzo.

1.6 Funcionalidades adicionales.

- **Historia 30** Como cliente quiero poder proponer platos de mi agrado para consumir, de esta manera podré consumir aquello que quiera aunque no esté disponible.
- **Historia 31** Como cliente quiero que esos platos elegidos por mí aparezcan en un menú específico por el cual el cocinero se comprometerá a cocinarlo, como también poder tener un chat privado con él, de esta manera me aseguro de que el plato sea cocinado a mi gusto.
- **Historia 32** Como cliente quiero poder tener la opción a pagar un plan mensual que ofrezca una tarifa plana de envíos y un programa de puntos, que se consiga pagando por comidas en la aplicación y que pueden canjearse por un descuento en la comida en un futuro, de esta manera puedo tener un beneficio extra de otra manera.

1.7 Otras consideraciones.

- **Historia 33** Como usuario quiero que la aplicación esté disponible en catalán, castellano e inglés, para poder usar la interfaz en el idioma con el que me sienta más cómoda.
- **Historia 34** Como usuario quiero poder acceder a la aplicación tanto desde la versión web como desde la versión móvil, para adaptarme al dispositivo que esté utilizando en cada momento.
- **Historia 35** Como usuario quiero que la aplicación móvil funcione tanto en Android como en IOS para asegurarme de que mi sistema operativo está soportado.
- **Historia 36** Como usuario quiero que la aplicación utilice el tipo de letra Calibri en todos los apartados, para mantener la coherencia visual.
- **Historia 37** Como usuario quiero que el color principal de la aplicación sea naranja y que el logo esté visible en todas las pantallas, para identificar fácilmente la marca de QueMenges.
- **Historia 38** Como usuario quiero que la aplicación tenga un tiempo de respuesta inferior de 5 ms cuando realizo una petición, para que la experiencia sea rápida y eficiente.
- **Historia 39** Como administrador quiero que la aplicación pueda soportar hasta 1000 usuarios conectados simultáneamente, para garantizar la estabilidad en horas punta.
- **Historia 40** Como usuario quiero que la aplicación utilice el protocolo OpenVPN para la encriptación de mis datos personales y de pago, para garantizar la seguridad y privacidad de mi información.

2 Lista de requerimientos.

2.1 Grupo 1: Gestión de cuentas y creación de perfiles de usuario.

Requisitos Funcionales

RF-1-01:Un usuario podrá consultar los menús, esté o no registrado.

RF-1-02:Solo los usuarios registrados podrán hacer pedidos.

RF-1-03: Los usuarios podrán registrarse con una cuenta de correo.

RF-1-04: Los usuarios podrán registrarse a través de shortcuts, en este caso de Gmail y Outlook.

RF-1-05: Los clientes que vayan a realizar un pedido deberán completar un perfil donde completarán sus datos, entre ellos un nombre de usuario.

RF-1-06: Un usuario puede actualizar en todo momento su perfil, incluso cambiar sus roles dentro de la aplicación.

RF-1-07: Un usuario puede eliminar si lo desea su perfil.

RF-1-08: A la hora de registrarse, el usuario tiene que indicar que tipo de cuenta se puede crear (cliente, cocinero o repartidor).

RF-1-09: Un usuario puede tener todos los tipos de perfiles disponibles si lo desea.

Requisitos no Funcionales

RNF-1-01: ¿El nombre de usuario tendrá 5 caracteres como mínimo y deberá de ser único.

RNF-1-02: ¿La contraseña de la cuenta tendrá como mínimo 8 caracteres, de los cuales mínimo 2 serán mayúsculas y 2 caracteres especiales.

2.2 Grupo 2: Realización de comandas.

Requisitos Funcionales

RF-2-01 : Los platos y menús se muestran priorizando aquellos que cumplen con las restricciones alimentarias del perfil del cliente.

RF-2-02 : Se muestra el tiempo estimado de entrega de cada plato o menú.

RF-2-03 : Los clientes pueden elegir entre dos vistas: la lista de platos y menús o la lista de cocineros disponibles.

RF-2-04 : Si se consulta un cocinero, se muestra la lista de todos los platos que ofrece dicho cocinero.

RF-2-05 : Los clientes pueden utilizar un buscador para buscar cocineros o platos específicos.

RF-2-06 : Los cocineros pueden aparecer como "no disponibles", indicando la hora a partir de la cual aceptarán pedidos, a la cual los clientes pueden reservar pedidos para entonces.

RF-2-07 : Los clientes pueden consultar la lista de ingredientes de cada plato, así como sus alérgenos, y si es necesario, eliminar algún ingrediente opcional.

RF-2-08 : El cliente puede confirmar el plato y cerrar el pedido para iniciar el envío.

RF-2-09 : Los clientes pueden consultar el historial de pedidos realizados y, si es necesario y posible, repetir pedidos anteriores.

Requisitos no Funcionales

RNF-2-01 : El sistema debe mostrar los resultados priorizados según la distancia y las restricciones alimentarias de manera eficiente.

RNF-2-02: El sistema debe limitar a tres los pedidos activos por cliente.

2.3 Grupo 3: Envío de comandas

Requisitos Funcionales

RF-3-01: El sistema ofrece tres opciones de envío: Repartidor contratado por la aplicación.

El cocinero entrega el pedido. El cliente recoge el pedido en el domicilio del cocinero.
RF-3-02: El costo de envío se calcula según la distancia entre el cocinero y el cliente.
RF-3-03: El envío es gratuito si el cliente recoge el pedido personalmente.
RF-3-04: Una vez pagado el pedido e iniciado el envío, el cliente puede seguir su progreso en tiempo real.
RF-3-05: Se muestra la posición actual del repartidor en un mapa en la web.

Requisitos no Funcionales

RNF-3-01: Integración con Google Maps para la visualización de la posición del repartidor en tiempo real.

2.4 Grupo 4: Pago de la comanda

Requisitos Funcionales

RF-4-01: El sistema debe permitir el pago de la comanda desde la aplicación.
RF-4-02: Debe ofrecer métodos de pago mediante tarjeta bancaria y PayPal.
RF-4-03: El sistema debe realizar el ingreso del dinero al cocinero y al repartidor según los datos bancarios asociados a su perfil.

Requisitos no Funcionales

RNF-4-01: El sistema debe conectarse con la plataforma de pago correspondiente para autenticar la transacción.

2.5 Grupo 5: Valoración de la comanda

Requisitos Funcionales

RF-5-01: El cliente debe poder valorar la comanda después de recibirla.
RF-5-02: La valoración incluirá criterios como puntualidad y estado del plato.
RF-5-03: El sistema asociará la valoración con el cocinero que ha preparado el plato.
RF-5-04: Las valoraciones influirán en el posicionamiento de los cocineros en el listado de platos disponibles.
RF-5-05: Los clientes deben poder ver las valoraciones de otros usuarios antes de hacer un pedido.

2.6 Grupo 6: Funcionalidades Adicionales

Requisitos Funcionales

RF-6-01: El usuario debe poder proponer platos a consumir aparte de los mostrados en la lista de platos.
RF-6-02: Estos platos propuestos por el usuario deben aparecer en un menú distinto, que permitirá la comunicación usuario-cocinero a través de un chat.
RF-6-03: La aplicación debe poder tener la opción de pagar una mensualidad que ofrezca una tarifa plana de envíos, como también un sistema de puntos, los cuales se consiguen pagando comandas dentro de la aplicación, y gracias a estos poder obtener descuentos en futuras comandas.

2.7 Grupo 7: Otras consideraciones

Requisitos Funcionales

RF-7-01:El software debe estar disponible tanto en catalán, castellano e inglés.

RF-7-02:El software debe poder ser accedida tanto por web como desde una aplicación móvil.

Requisitos no Funcionales

RNF-7-01:El software debe poder funcionar tanto en Android como en IOS.

RNF-7-02:El software debe usar la tipografía estilo Calibri.

RNF-7-03:El color principal de la aplicación debe ser el naranja.

RNF-7-04:El logo debe poder ser visible en todas las pantallas.

RNF-7-05:Las peticiones hechas por el usuario deben ser respondidas en un tiempo inferior a 5ms.

RNF-7-06:El software debe poder soportar hasta 1000 usuarios simultáneos.

RNF-7-07:El software debe usar el protocolo OpenVPN para encriptación de datos y pagos.

3 Product Backlog

- Planificación del proyecto
- Creación del tablero en Trello
- Configuración del repositorio en Github
- Reunión inicial
- Extraer requisitos de Pagos y Valoraciones
- Describir el proyecto en Github
- Crear documento para las historias de usuario
- Extraer requisitos de Gestión de Cuentas y Perfiles de Usuario
- Crear el documento para el product backlog
- Creación la 1ra versión del documento de especificaciones
- Extraer requisitos de Funcionalidades Adicionales y RN
- Elaboración de historias de usuario
- Creación del documento de requisitos
- Analisis de los requisitos del proyecto
- Documentar en LaTeX las historias de usuario
- Revisar las historias de usuario
- Extraer requisitos de Pedidos y Envíos
- Extracción de las historias de usuario del bloque 1
- Extracción de las historias de usuario del bloque 2, 5 y 6
- Extracción de las historias de usuario del bloque 3 y 4

- Extracción de las historias de usuario del bloque de funciones adicionales
- Mejorar el documento de especificaciones
- Añadir toda la información en el documento de especificaciones
- Revisión de requisitos
- Actualizar el product backlog
- Subir la versión definitiva de las especificaciones
- Verificar que todo esta en GitHub
- Crear la presentación del proyecto del 1er sprint
- Diseñar y realizar la presentación del 1er sprint
- Actualizar el documento de especificaciones
- Crear documento para el diagrama de especificación de casos de uso
- Crear diagrama de clases
- Crear diagrama de secuencias
- Identificar las especificaciones de nuestro sistema
- Actualización de las carpetas de requisitos en GitHub.
- Revisión de requisitos 2.0 (revisar subtipos de los requisitos no funcionales).
- Actualización de las carpeta de requisitos de Github.
- Revisión de requisitos 2.0
- Evaluar RNF 1-2 y proponer ajustes.
- Evaluar RNF 5,6-7 y proponer ajustes.
- Creación Presentación Sprint 2.
- Actualizar especificaciones en le documento del usuario 4 (no registrado).
- Actualización Product BackLog 2.0.
- Validar repositorio para el Sprint 2.
- Actualizar BurnDown Chart Sprint 2.
- Relacionar los requisitos 6-7.
- Relacionar los requisitos 1-3.
- Actualizar versión final presentación Sprint 2.
- Actualizar Diagramas de caso de uso en el documento compartido.
- Subir y compartir el diagrama final al Github.
- Unificar diagramas individuales en un solo diagrama final.
- Revisión final del diagrama de casos de uso.
- Revisar y validar los diagramas individuales.

- Dibujar diagrama de casos bloque 1.
- Dibujar diagrama de casos bloque 2.
- Dibujar diagrama de casos bloque 3.
- Dibujar diagrama de casos bloque 4.
- Dibujar diagrama de casos bloque 5.
- Dibujar diagrama de casos bloque 6.
- Dibujar diagrama de casos bloque 7.
- Revisar la nueva versión de relacionar los requisitos.
- Evaluar RNF 3-4 y proponer ajustes.
- Validación de observaciones sobre RNF.
- Relacionar los requisitos 4-5.
- Plantilla Diagrama Casos de uso.
- Validar el repositorio para el Sprint 3.
- Revisar el diagrama Cas d'Us final.
- Creación Burn-Down chart.
- Corregir la plantilla del bloque 1.
- Corregir la plantilla del bloque 2 y 3.
- Corregir la plantilla del bloque 4 y 5.
- Corregir la plantilla del bloque 6.
- Revisar Especificación bloque 1 y 2.
- Revisar Especificación bloque 3.
- Revisar Especificación bloque 4 y 5.
- Revisar Especificación bloque 6.
- Diagrama de Clase del bloque 1 y 2.
- Subir código UML bloque 1 y 2 al github.
- Diagrama de Clase del bloque 4.
- Subir código UML bloque 4 al github.
- Subir código UML bloque 5 y 6 al github.
- Diagrama de Clase del bloque 5 y 6.
- Subir código UML bloque 3 al github.
- Diagrama de Clase del bloque 3.
- Unificar los diagramas en un diagrama final y subirlo al github.
- Validar los diagramas de cada bloque.

- Revisión de los codigos UML subidos a github.
- Actualizar BurnDown Chart Sprint 3.
- Completar Diagramas de Clase 1 y 2.
- Completar Diagrama de Clase 3.
- Completar Diagrama de Clase 4.
- Completar Diagramas de Clase 5 y 6.
- Subir Codigos UML Diagramas de Clase 1 y 2.
- Subir Codigos UML Diagramas de Clase 3.
- Subir Codigos UML Diagramas de Clase 4.
- Subir Codigos UML Diagramas de Clase 5 y 6.
- Actualizar Product Backlog.
- Crear y actualizar BurnDown Chart.
- Reporte diario de los Daily Meetings al Github