Universidad del Cauca  
Facultad de Ingeniería Electrónica y Telecomunicaciones  
Ingeniería de Sistemas

Ingeniería de Software II

RlxdFood app – Aplicación de Delivery para los restaurantes de Popayán

Colaboradores:

Juan David Muñoz Pasquel

Stakeholders:

Julio Ariel Hurtado  
 Santiago Hyun Dorado  
Libardo Pantoja

Popayán – Cauca

2021

**Tabla de contenido**

|  |  |
| --- | --- |
| 1. Resumen | **1** |
| 1. Vista de requerimientos:  |  | | --- | | * 1. Historias épicas e Historias de usuario | | * 1. Escenarios de calidad | | * 1. Criterios de aceptación | | **2**  **3**  **4** |
| 1. Representación de la arquitectura | **5** |
| 1. Modelo de contexto (Diagrama C4) | **6** |
| 1. Vista lógica:  |  | | --- | | * 1. Vista de contenedores y componentes (Diagrama C4) | | * 1. Vista de módulos (UML) | | * 1. Vista de componentes y conectores (UML) | | **7**  **8**  **9** |
| 1. Vista de implementación  |  | | --- | | * 1. Vista de implementación (Maven, UML) | | **10** |
| 1. Vista de instalación  |  | | --- | | * 1. Vista de instalación (Allocation, UML) | | **11** |
| 1. Vista de datos  |  | | --- | | * 1. Modelo Entidad – Relación | | **12** |
| 1. Resumen de las decisiones de arquitectura y su justificación | **13** |

1. **Resumen**

Una de las actividades con mayor demanda, aceptación y oportunidad de crecimiento que ha traído consigo la expansión de las tecnologías de la información a nivel global, es el Delivery (Servicio a través del cual un restaurante reparte sus platos a domicilio) a través de las plataformas digitales; algunos ejemplos son las plataformas “IFood”, “Rappi”, “Uber eats” entre otras muchas aplicaciones existentes que brindan la opción a los restaurantes de tomar parte en la denominada transformación digital. RlxdFood surge como una nueva opción de Delivery para los restaurantes de la ciudad de Popayán, esta es la aplicación que se va a construir como solución y aporte a la opción de Delivery de los distintos restaurantes, partiendo de una necesidad común y unos requerimientos específicos; en este documento se plantea la documentación de todas (o al menos la mayoría) de las decisiones tomadas arquitecturalmente hablando para llevar a cabo el desarrollo de este Software, las vistas que se han utilizado y los requerimientos desde los cuales se partió para el desarrollo.

1. **Vista de requerimientos**
   1. **Historias épicas e Historias de usuario**

|  |  |  |  |
| --- | --- | --- | --- |
| **Prioridad** | **Código** | **Historia épica** | **Contexto** |
| 1 | HE01 | Yo **como** administrador del restaurante **necesito** registrar los componentes del almuerzo ejecutivo **para** posteriormente ofrecer a través de la plataforma el almuerzo de un día determinado | Un componente del almuerzo ejecutivo debe tener un id, nombre, tipo (entrada, principio, proteína o bebida). Ejemplos de componentes son: Sopa de verduras, sopa de carantanta, frijoles, lentejas, pollo frito, pollo sudado, limonada, jugo de tomate etc. |
| **2** | HE02 | Yo **como** administrador del restaurante **necesito** registrar el almuerzo ejecutivo del día en el sistema **para** que los clientes sepan que ofrece el restaurante como entrada, principio, proteína y bebida. | El restaurante puede ofrecer uno o más componentes por entrada, principio, proteína y bebida. Por ejemplo, un cliente puede elegir de principio: frijoles, arvejas o lentejas; de entrada: Sopa de verduras, sopa de maíz o frutas. Cuando algún componente se agote, el administrador lo puede quitar de la oferta para que los clientes no lo soliciten más. Se debe además poder seleccionar y agregar una imagen del almuerzo ejecutivo. Además, se puede agregar una ligera descripción del plato ejecutivo que motivará al cliente a realizar el pedido. |
| **3** | HE03 | Yo **como** administrador del restaurante **necesito** registrar los platos a la carta que ofrece el restaurante **para** que los clientes los puedan conocer y pedir a través del sistema. | Cada plato a la carta debe tener un nombre, una descripción y una imagen.  Ejemplos de plato especiales son: pechugas rellenas con tocino y queso, bandeja paisa, italian sausage and peppers, cheesy broccoli stuffed chicken breast, etc. |
| **4** | HE04 | Yo **como** visitante **necesito** entrar a la plataforma y visualizar los restaurantes que tiene la ciudad **para** elegir uno en particular y visualizar los platos ejecutivos y a la carta que ofrece para ese día. | Los usuarios siempre quieren visualizar todos los platos, filtrar los platos de su  preferencia y mirar el contenido que ofrece el restaurante sin hacer ningún tipo de registro o inicio de sesión. |
| **5** | HE05 | Yo **como** cliente **necesito** hacer un determinado pedido del restaurante elegido **para** que el restaurante lo envíe a domicilio. | El pedido puede tener uno o más platos tanto ejecutivos como a la carta. De cada plato se puede solicitar una cantidad entre 1 y 10. Para hacer el pedido el cliente debe estar registrado previamente en la plataforma. El cliente puede elegir cancelar el pedido mediante PSE, tarjeta de crédito, o cancelar en efectivo cuando el mensajero le entregue el pedido en casa. El sistema debe informar el tiempo  estimado de entrega del pedido. |
| **6** | HE06 | Yo **como** administrador o chef del restaurante **necesito** acceder a los pedidos **para** hacer la preparación y el despacho de los mismos hacia los clientes. | Se debe tener un listado en orden cronológico para conocer los pedidos que hay pendientes y poder prepararlos. |
| **7** | HE07 | Yo **como** administrador o chef del restaurante **necesito** despachar los pedidos **para** hacer el envío hacia las ubicaciones de los clientes. | Cuando se despache un pedido se debe elegir el mensajero que entregará el  pedido. Se debe mostrar una lista con los mensajeros que se encuentran actualmente disponibles. Cuando un mensajero es seleccionado, se debe cambiar su estado a ocupado. |
| **8** | HE08 | Yo **como** mensajero **necesito** acceder a la plataforma **para** cambiar el estado de un pedido cuando ha sido entregado satisfactoriamente a un cliente. | Todos los pedidos se deben finalizar cambiando de estado a entregado. En este caso ya no deberá aparecer en la lista principal de pedidos, pero deberá mantenerse en el sistema para saber el historial de pedidos realizados. Una vez  entregados todos los pedidos asignados, el mensajero debe cambiar su estado a disponible. |
| **9** | HE09 | Yo **como** súper-administrador de la plataforma **necesito** registrar un nuevo restaurante **para** que el restaurante pueda  posteriormente ofrecer sus platos, pedidos a través de la plataforma. | Un restaurante tiene un nit, nombre, eslogan, propietario, dirección, teléfonos, ciudad, administrador y fotografía. El listado de restaurantes se debe mostrar en una tabla que permita filtrar de acuerdo a su nombre, nit o el nombre del administrador. |
| **10** | HE10 | Yo **como** cliente **necesito** registrarme en la plataforma **para** poder hacer pedidos. | Los datos del registro del cliente son: nombres, fecha nacimiento, género, email, contraseña y celular. No se pueden registrar menores de edad. |
| **11** | HE11 | Yo **como** cliente, administrador,  mensajero, chef **necesito** iniciar sesión en el sistema **para** acceder a las funcionalidades o servicios que ofrece la plataforma. | Para iniciar sesión se requiere email y contraseña. Debe existir un único email  por usuario y la contraseña debe tener un grado medio de seguridad. Como medida  de seguridad, la contraseña debe ser encriptada al momento de guardar y desencriptada al iniciar la sesión. |
| **12** | HE12 | Yo **como** administrador del restaurante **necesito** acceder a los reportes de ventas **para** conocer cuánto dinero ha ingresado al restaurante. | El reporte debe solicitar un rango de fechas y debe mostrar los datos en una  gráfica de barras. |

A partir de las historias épicas, se presentan las siguientes historias de usuario (se extraen algunas, pero pueden ampliarse a lo largo del tiempo) se pretende que, a medida que se avance en la realización de las diferentes historias de usuario – junto a sus correspondientes tareas, el sistema sea robustecido y que brinde cada vez mejores funcionalidades para cada uno de los actores que utilicen el sistema. Es importante resaltar que algunas de las historias marcadas como épicas, están muy explícitas, claras y cortas, pudiendo considerarse como historias de usuario; es por esta razón que algunas de las historias de usuario se toman tal cual se encuentran como historias épicas, además de dejar en blanco algunos de los campos de contexto, puesto que, a nivel de historias épicas, el requerimiento es claro.

|  |  |  |  |
| --- | --- | --- | --- |
| **Historia épica** | **Historia de usuario** | **Historia de usuario** | **Contexto** |
| HE01 | HU01 | Yo **como** administrador del restaurante **necesito** agregar los componentes del almuerzo ejecutivo **para** posteriormente ofrecer a través de la plataforma el almuerzo que se oferta en un día determinado. | Id, nombre y tipo del componente de almuerzo son obligatorios. El sistema debe validar que no existan componentes repetidos. |
| HE01 | HU02 | Yo **como** administrador del restaurante **necesito** buscar un componente en particular **para** conocer su información registrada. | La búsqueda es realizada por id o por nombre dependiendo de la necesidad que se tenga. |
| HE01 | HU03 | Yo **como** administrador del restaurante **necesito** listar los componentes del almuerzo ejecutivo **para** posteriormente ofrecer a través de la plataforma el almuerzo que se oferta en un día determinado. | El listado debe mostrar los campos id, nombre y tipo. Se debe paginar los resultados en grupos de 5, 10 y 20 registros. Hacia futuro se debe poder hacer filtros y ordenamientos por alguna columna elegida. |
| HE02 | HU04 | Yo **como** administrador del restaurante **necesito** registrar el almuerzo ejecutivo del día en el sistema **para** que los clientes sepan lo que oferta el restaurante como entrada, principio, proteína y bebida. | Se debe poder elegir cuales  componentes se ofrecerán de entrada, principio, proteína y bebida. Se pueden elegir uno o varios, máximo cinco por categoría. Además, se debe poder elegir la imagen del plato ejecutivo y escribir la descripción del plato que motive al cliente a hacer el  pedido. |
| HE03 | HU05 | Yo **como** visitante **necesito** entrar a la plataforma y listar los restaurantes que tiene la ciudad **para** elegir uno en particular y ver los platos a la carta que se ofrecen. | El listado de los diferentes restaurantes debe contener el nombre, la dirección, los teléfonos y la fotografía. Una vez se ingrese en uno de los restaurantes se debe mostrar incluir el slogan. |
| HE03 | HU06 | Yo **como** visitante **necesito** entrar a la plataforma y listar los diferentes platos a la carta **para** poder ver lo que oferta el restaurante. | El listado debe contener el nombre y la descripción del plato; la visualización de cada plato (tal como en el plato ejecutivo) debe ser de manera agradable y llamativa de tal motivo que incita a realizar el pedido, por ejemplo, mostrando en primera instancia la imagen del plato. |
| HE03 | HU07 | Yo **como** visitante **necesito** entrar a la plataforma y listar los componentes del plato ejecutivo **para** poder ver lo que oferta el restaurante en un día determinado. | El listado debe contener el nombre y la descripción del plato; la visualización de cada plato (tal como en el plato a la carta) debe ser de manera agradable y llamativa de tal motivo que incita a realizar el pedido, por ejemplo, mostrando en primera instancia la imagen del plato. |
| HE05 | HU08 | Yo **como** cliente **necesito** inicializar mi pedido **para** que tenga como estado “Solicitado” | El pedido consta de varios estados que van cambiando desde que es recibido por el restaurante hasta que es entregado por el repartidor.  El cliente debe estar registrado en la plataforma y tener una sesión activa para poder realizar el pedido. |
| HE05 | HU09 | Yo **como** cliente **necesito** seleccionar la forma de pago **para** poder cancelar el pedido solicitado. | Opciones de cancelación son PSE, tarjeta de crédito y efectivo al momento de recibir el pedido. Es necesario considerar un aumento en las formas de pago hacia futuro. |
| HE06 | HU10 | Yo **como** administrador o chef del restaurante **necesito** listar los pedidos **para** poder realizar su preparación. | Se debe tener un listado en orden cronológico para conocer los pedidos que hay pendientes y poder prepararlos. |
| HE06 | HU11 | Yo **como** administrador o chef del restaurante **necesito** cambiar el estado de los pedidos de “Recibido” a “En preparación” **para** que el cliente conozca cómo va su pedido. | El pedido puede encontrarse en varios estados: “Solicitado”, “Recibido”, “En preparación”, “Enviado”, “En camino” y “Entregado”. |
| HE07 | HU12 | Yo **como** administrador o chef del restaurante **necesito** cambiar el estado del pedido de “En preparación” a “Enviado” **para** que el cliente esté al tanto. |  |
| HE07 | HU13 | Yo **como** administrador o chef del restaurante **necesito** listar los repartidores disponibles **para** seleccionar uno para entregar el pedido. | Se debe mostrar un listado con los repartidores cercanos disponibles |
| HE08 | HU14 | Yo **como** mensajero **necesito** acceder a la plataforma **para** poder cambiar el estado de un pedido de “En camino” a “Entregado”. |  |
| HE08 | HU15 | Yo **como** mensajero **necesito** mirar en la plataforma la mejor ruta hacia la dirección estipulada **para** llevar el pedido en el menor tiempo posible. | La aplicación utilizará servicios de ubicación de terceros, se hará el consumo del api “Geolocation\_api” |
| HE08 | HU16 | Yo **como** mensajero **necesito** cambiar mi estado “Disponible” – “Ocupado” al momento de ir en camino a entregar un pedido **Para** que no se me asignen otros pedidos. | La aplicación puede asignarle más de un pedido mientras el mensajero tenga su estado en “Disponible”. Una vez el estado cambie a “Ocupado” no se le asignarán nuevos hasta que se encuentre en “Disponible” nuevamente |
| HE08 | HU17 | Yo **como** administrador del restaurante **necesito** acceder al historial principal de los pedidos **para** consultar la información de cualquiera de ellos. | La plataforma guarda registro de cada pedido realizado, la fecha, la hora, cliente y repartidor. |
| HE09 | HU18 | Yo **como** súper – administrador de la plataforma **necesito** registrar un nuevo restaurante **para** que posteriormente pueda ofertar sus platos a través de la plataforma. | No pueden existir restaurantes repetidos, la plataforma debe validar antes de ingresar cualquier restaurante. Cualquier restaurante puede ser eliminado en cualquier momento. |
| HE10 | HU19 | Yo **como** cliente **necesito** registrarme en la plataforma **para** poder realizar pedidos. | Datos del registro: Nombres, fecha de nacimiento, género, email, contraseña y celular. No se pueden registrar menores de edad. |
| HE11 | HU20 | Yo **como** cliente, administrador, mensajero o chef del restaurante **necesito** iniciar sesión en el sistema **para** poder acceder a las funcionalidades que ofrece. | La plataforma debe validar que, en efecto, exista la cuenta con la que se intenta ingresar, de no existir, se mostrará la opción, al usuario, de registrarse. |
| HE12 | HU21 | Yo **como** administrador del restaurante **necesito** acceder a los reportes de ventas **para** conocer cuánto dinero ha ingresado al restaurante. |  |

* 1. **Escenarios de calidad y criterios de aceptación**

Para la selección de escenarios y atributos de calidad y los criterios de aceptación, se realiza un MiniQAW (una versión más ligera del Quality Attribute Workshop – método creado por el SEI para la captura de atributos de calidad, basado en las necesidades de los Stakeholders; este método por lo general requiere mucho tiempo y la colaboración de todos los Stakeholders); el MiniQAW permite ahorrar tiempo con nuevas actividades que facilitan la selección de los atributos de calidad más importantes para los interesados.

El MiniQAW se realizó en 4 etapas, las cuales se mencionan a continuación:

1. **Mapa de empatía de los StakeHolders.**

El Mapa de empatía de los stakeholders es un ejercicio de lluvia de ideas y visualización que da voz a los stakeholders ausentes. Durante el ejercicio, los participantes se identifican con los stakeholders ausentes y especifican preocupaciones sobre la calidad desde su perspectiva. Como resultado, los participantes experimentan de primera mano las prioridades y requisitos en conflicto que pueden existir en los requisitos del sistema.

Esta actividad se divide en 6 puntos característicos:

1. **Identificación de StakeHolders involucrados en el proyecto.**

|  |  |  |
| --- | --- | --- |
| **Stakeholders** | **Quality feature** | **Priority level** |
| Cliente registrado en la plataforma | Adecuación funcional | 1 |
| Visitante de la plataforma | Eficiencia de desempeño | 2 |
| Chef - Cocineros | Compatibilidad | 3 |
| Repartidor | Usabilidad | 4 |
| Administrador de restaurante | Fiabilidad | 5 |
| Súper administrador | Seguridad | 6 |
| Desarrolladores | Mantenibilidad |  |
| Restaurantes Popayán (Escalable) | Portabilidad |  |

1. **Selección de StakeHolders que no se encuentran presentes en el workshop.**

En este caso, se cuenta con un único integrante para la realización del MiniQAW, por ende, se realiza la selección de todos los interesados.

1. **Estimación de prioridades de calidad para las partes interesadas seleccionadas (ausentes) y Almacenamiento de la priorización en el Empathy map.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Empathy map** | | | |
| **Selected stakeholder** | **Priority** | **Quality feature** | **Rationale** |
| Cliente registrado en la plataforma | 1 | Usabilidad | El Software debe tener la capacidad de ser entendido, aprendido, usado y atractivo para el usuario. El usuario debe poder encontrar un menú de opciones en 3 segundos o menos, más tiempo podría afectar la capacidad de ser atractivo para los usuarios. |
| Cliente registrado en la plataforma | 6 | Adecuación funcional | El Software debe brindar al cliente registrado algunas ciertas capacidades, como lo es pago en línea, por ejemplo, solicitud de un pedido, cancelación de un pedido (Cuando este se encuentre en estado "Solicitado" o "Recibido") |
| Cliente registrado en la plataforma | 3 | Seguridad | El Software estará en contacto con datos e información sensibles, como lo es la información bancaria de diferentes clientes (solicitada para los pagos en línea). Se debe garantizar que esta información se preserve confidencial e integra. |
| Cliente registrado en la plataforma | 5 | Compatibilidad | Al menos 3 de cada 5 pedidos se realizarán con tarjetas de crédito, débito y otras formas de pago en línea. La aplicación debe estar en capacidad de coexistir satisfactoriamente con servicios de terceros que brinden este servicio (Como por ejemplo la API que brinde PSE) |
| Visitante de la plataforma | 1 | Usabilidad | La plataforma debe ser agradable para los usuarios principales, contribuyendo a que vuelvan a utilizarla e incluso que la recomienden. El visitante (Usuario sin registro) debe poder encontrar el listado de los diferentes restaurantes, el listado de los platos y diferentes menús de opciones en 3 segundos máximo. |
| Visitante de la plataforma | 4 | Portabilidad | La portabilidad podría influir en al menos un 30% en que el visitante decida registrarse o no en la plataforma (en adelante, dependerá de criterios de usabilidad) pues es más cómodo poder realizar un pedido desde diferentes dispositivos a estar limitado por uno sólo. |
| Chef - Cocineros | 1 | Eficiencia de desempeño | Es importante pensar en situaciones de estrés para el Software (gran procesamiento) y uno de estos contextos, puede ser pensado cuando el restaurante tenga muchos pedidos (Un restaurante en un buen día podría tener entre 30 y 50 pedidos en una hora - Suponiendo un intervalo de tiempo de almuerzo: 11 - 2, se tendrá una gran cantidad de pedidos) y es en estos momentos, cuando el Software debe responder en el menor tiempo, para que los cocineros puedan proceder con la preparación y el pedido no se retrase. |
| Chef - Cocineros | 2 | Adecuación funcional | A nivel del Software, los cocineros tendrán ciertas funcionalidades como, por ejemplo, el cambio de estado de un pedido ("Recibido" - "En preparación"; "En preparación" - "Enviado") o la selección de un repartidor, listar los platos en orden cronológico, entre otras. EL sistema debe mostrar una completitud funcional frente a este conjunto de operaciones para los cocineros. |
| Chef - Cocineros | 3 | Fiabilidad | Si un día en el que se tienen muchos pedidos, el sistema no puede responder durante un largo intervalo de tiempo como lo sería 11 - 2 por ejemplo, los cocineros dejarían de preparar una gran cantidad de pedidos, generando una pérdida para el restaurante. |
| Repartidor | 2 | Eficiencia de desempeño | Es importante que durante el recorrido de un pedido (desde que es recogido por el repartidor hasta que es entregado en la dirección estipulada) el sistema pueda responder a las peticiones del repartidor (puesto que se pueden presentar inconvenientes) ya que el repartidor puede llevar uno o más platos en su recorrido. |
| Repartidor | 1 | Portabilidad | El sistema debe garantizar la facilidad de instalación y la adaptabilidad en diferentes Sistemas hardware y Software. |
| Repartidor | 3 | Adecuación funcional | El Software debe brindar al repartidor registrado algunas ciertas capacidades, como lo es el cambio de estado del pedido ("Enviado" - "En camino"; "En camino" - "Entregado"), cambio de estado propio ("Disponible" - "Ocupado"; "Ocupado" - "Disponible"), seguimiento del mapa trazado con la ruta más óptima para la entrega del pedido. |
| Repartidor | 4 | Fiabilidad | El sistema debe responder y estar disponible cuando el repartidor parta con su pedido o cuando se le solicite que lo recoja, puesto que un fallo sin su correcta recuperación, podría ocasionar la pérdida de muchos de los pedidos solicitados en un día. |
| Repartidor | 5 | Compatibilidad | El repartidor contará con una ruta óptima trazada por parte de un sistema de GPS. Esta ruta óptima puede ahorrar un 10% - 15% del tiempo de entrega. |
| Administrador de restaurante | 1 | Adecuación funcional | Un administrador del restaurante cuenta con funcionalidades extras a por ejemplo un cliente, estas funcionalidades pueden ser la gestión de componentes de un plato ejecutivo, la gestión de platos (tanto a la carta como ejecutivos - en un futuro podrían existir más tipos dependiendo del restaurante), la traza de ventas del restaurante en cuestión, entre otras. El sistema debe garantizar una completitud funcional de al menos 70% de estas, donde se involucre la gestión de componentes y de platos. |
| Administrador de restaurante | 4 | Fiabilidad | El administrador del restaurante necesita que el sistema sea tolerante a fallos, y que se encuentre en disponibilidad en momentos que se requiere añadir los componentes del plato ejecutivo y los platos como tal. |
| Administrador de restaurante | 6 | Usabilidad | El administrador del restaurante necesita poder ofertar los platos del restaurante. La necesidad del administrador del restaurante es poder subir sus componentes y sus platos y que se pueda hacer la gestión necesaria, la usabilidad es puntuada con un 6 para el administrador del restaurante porque si bien es necesaria porque también debe facilitarse al administrador la gestión de los platos, mediante una aplicación auto descriptible e intuitiva, este atributo no es el más importante. |
| Administrador de restaurante | 5 | Seguridad | A nivel del administrador del restaurante, se tienen datos e información sensible que se debe preservar de manera integra y confidencial. Un ejemplo de información sensible puede ser las credenciales del mismo administrador y la traza de ventas del restaurante. |
| Súper administrador | 2 | Mantenibilidad | Por los menos 2 de los requerimientos cambiarán, solamente a corto plazo. A lo largo del tiempo, se sumarán algunos otros y el Software debe estar en la capacidad de responder a esta evolución. Para el súper administrador es uno de los atributos más importantes puesto que es quien se encargará de realizar el mantenimiento correctivo y preventivo en caso de necesitarse a la plataforma. En caso de que se requiera una evolución en el sistema y el este no la soporte, el súper administrador debe tomar las decisiones correspondientes, si un problema se presenta y se presenta retiradamente, puede generar grandes pérdidas para los restaurantes. |
| Súper administrador | 1 | Adecuación funcional | El súper administrador debe tener ciertas funcionalidades exclusivas de este rol, es decir, el sistema debe brindar una pertinencia funcional acertada (en el caso del súper administrador debe ser de no menos de 95% su pertinencia funcional – en otros casos puede llegar a ser menor) para que el súper administrador cuente con las herramientas para realizar las revisiones correctivas y previsorias necesarias. |
| Desarrolladores | 1 | Adecuación funcional |  |
| Desarrolladores | 2 | Mantenibilidad | El administrador del restaurante necesita que el sistema sea tolerante a fallos, y que se encuentre en disponibilidad en momentos que se requiere añadir los componentes del plato ejecutivo y los platos como tal. |
| Restaurantes Popayán (Escalable) | 1 | Eficiencia de desempeño | El administrador del restaurante necesita que el sistema sea tolerante a fallos, y que se encuentre en disponibilidad en momentos que se requiere añadir los componentes del plato ejecutivo y los platos como tal. |
| Restaurantes Popayán (Escalable) | 2 | Fiabilidad | El administrador del restaurante necesita que el sistema sea tolerante a fallos, y que se encuentre en disponibilidad en momentos que se requiere añadir los componentes del plato ejecutivo y los platos como tal. |
| Restaurantes Popayán (Escalable) | 3 | Seguridad | El administrador del restaurante necesita que el sistema sea tolerante a fallos, y que se encuentre en disponibilidad en momentos que se requiere añadir los componentes del plato ejecutivo y los platos como tal. |
| Restaurantes Popayán (Escalable) | 4 | Adecuación funcional | El administrador del restaurante necesita que el sistema sea tolerante a fallos, y que se encuentre en disponibilidad en momentos que se requiere añadir los componentes del plato ejecutivo y los platos como tal. |
| Restaurantes Popayán (Escalable) | 6 | Usabilidad | Un restaurante quiere poder ofertar sus platos del día y sus platos a la carta, la usabilidad tiene una puntuación de 6 para "Restaurantes Popayán (Escalable)" debido a que no es lo más importante para los restaurantes, lo más importante es poder ofertar sus platos y que la mayor cantidad posible de clientes compren (Por esta razón, la usabilidad está mejor puntuada para el Cliente y el visitante) |
| Restaurantes Popayán (Escalable) | 5 | Mantenibilidad | El sistema deberá evolucionar a lo largo del tiempo, a medida que los requerimientos de los restaurantes y los diferentes stakeholders vayan cambiando. El sistema debe estar en capacidad de responder a esta evolución constante y adaptarse de la mejor manera, tanto eficiente como efectivamente, para no afectar a ninguno de los involucrados y, por el contrario, brindarles el mejor servicio. Si los requerimientos de 3 restaurantes, por ejemplo, evolucionan y el sistema no puede responder a esto, se tendría una pérdida de estos 3 restaurantes, representando una pérdida significativa en gran cantidad de clientes. |

1. **Selección de los atributos de calidad con más prioridad para todos los StakeHolders y modificación de la red de atributos**
2. **Lluvia de ideas de escenarios.**

El objetivo de esta actividad es establecer lo que necesitan y esperan los participantes del sistema. Para ello, los participantes deben establecer escenarios de calidad en dónde imaginen condiciones en las que se necesite la adecuación de un determinado atributo de calidad. Por ejemplo: Generalmente, las personas solicitan más comida a determinadas horas del día, en las cuales el sistema debe estar siempre disponible y ser capaz de responder con un promedio aproximado de 200 peticiones por minuto.

1. **Priorización de atributos.**
2. **Refinamiento de escenarios.**
3. **Representación de la arquitectura**
4. **Modelo de contexto**

En este diagrama1 se tiene como punto referencial la aplicación que se va a construir “RlxdFood”, la cual se ve en relación con seis actores principales y tres sistemas adicionales o de soporte a la aplicación. Los actores principales que se han identificado hasta el momento son: Chef – cocineros, repartidor, visitante, cliente, administrador del restaurante y súper administrador. Por otro lado, los sistemas que actúan en conjunto a la aplicación principal “RlxdFood” son: GPS system, E-mail system, online payment system (por el momento no se ha tomado decisión sobre el sistema específico o el api de terceros que se utilizará). Cada actor y sistema juega un rol importante dentro de la aplicación, los cuales se ven a continuación

* **Chef – cocineros:** Encargados de preparar los platos conforme a lo solicitado en el pedido. A nivel de la plataforma, quienes se encarguen de este rol tendrán la tarea de actualizar los estados del pedido “Recibido” – “En preparación” y “En preparación” – “Enviado”.
* **Repartidor:** Persona encargada de recoger el pedido solicitado y transportarlo hacia la dirección estipulada. Dentro de la plataforma, esta persona también tiene la tarea de cambiar los estados del pedido “Enviado” – “En camino” y “En camino” – “Entregado” y, además de cambiar el estado del pedido como tal, también deberá cambiar su estado “Disponible” – “Ocupado” y “Ocupado” – “Disponible”
* **Visitante:** Un visitante es todo aquel que ingresa a la plataforma, puede mirar los restaurantes listados, los platos tanto ejecutivos como a la carta que ofrecen los diferentes restaurantes sin ningún tipo de inicio de sesión. Si un visitante desea realizar un pedido, debe registrarse en la plataforma y pasaría a ser llamado “cliente”.
* **Cliente:** Un cliente es todo aquel que se ha registrado satisfactoriamente en la plataforma, puede realizar pedidos, observar el estado de su pedido y el tiempo estimado que tardará en ser entregado, además de las diferentes opciones de pago como lo son: tarjeta de crédito, PSE y efectivo.
* **Administrador del restaurante:** Un administrador del restaurante es quien se encarga de la gestión de los diferentes platos. La gestión puede incluir registro, actualización, eliminado de un componente de un plato ejecutivo, registro y actualización del plato ejecutivo como tal, registro de platos a la carta, entre otros.
* **Súper administrador:** Es quien se encarga de brindar soporte técnico a la aplicación (mantenimiento y solución de bugs) además, este súper administrador tiene como tarea la gestión de restaurantes, lo que incluye el registro, la actualización, eliminado, entre otros.
* **GPS System:** Sistema que brindará apoyo a la aplicación en el seguimiento del pedido en tiempo real. En adición, también proporcionará una ruta al repartidor desde el restaurante hasta la dirección estipulada para entregar los diferentes pedidos.
* **E – mail System:** El sistema de E – mail brindará apoyo a la aplicación con el envío de mensajes, notificaciones a los diferentes actores y dependiendo de las diferentes situaciones.
* **Online payment System:** El sistema de pago en línea será utilizado como medio para realizar los pagos de los diferentes pedidos que no se den en modalidad de “pago con dinero en efectivo

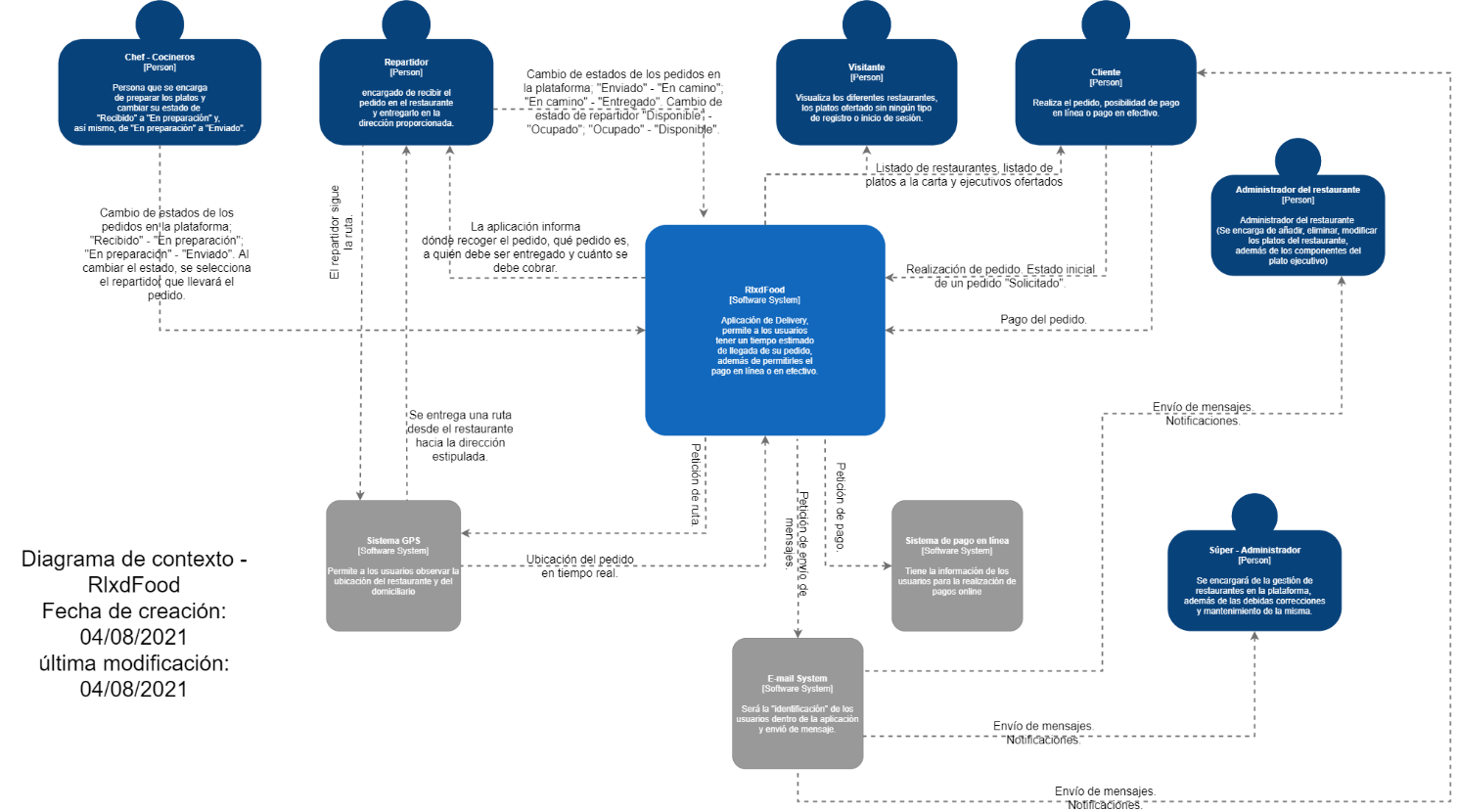
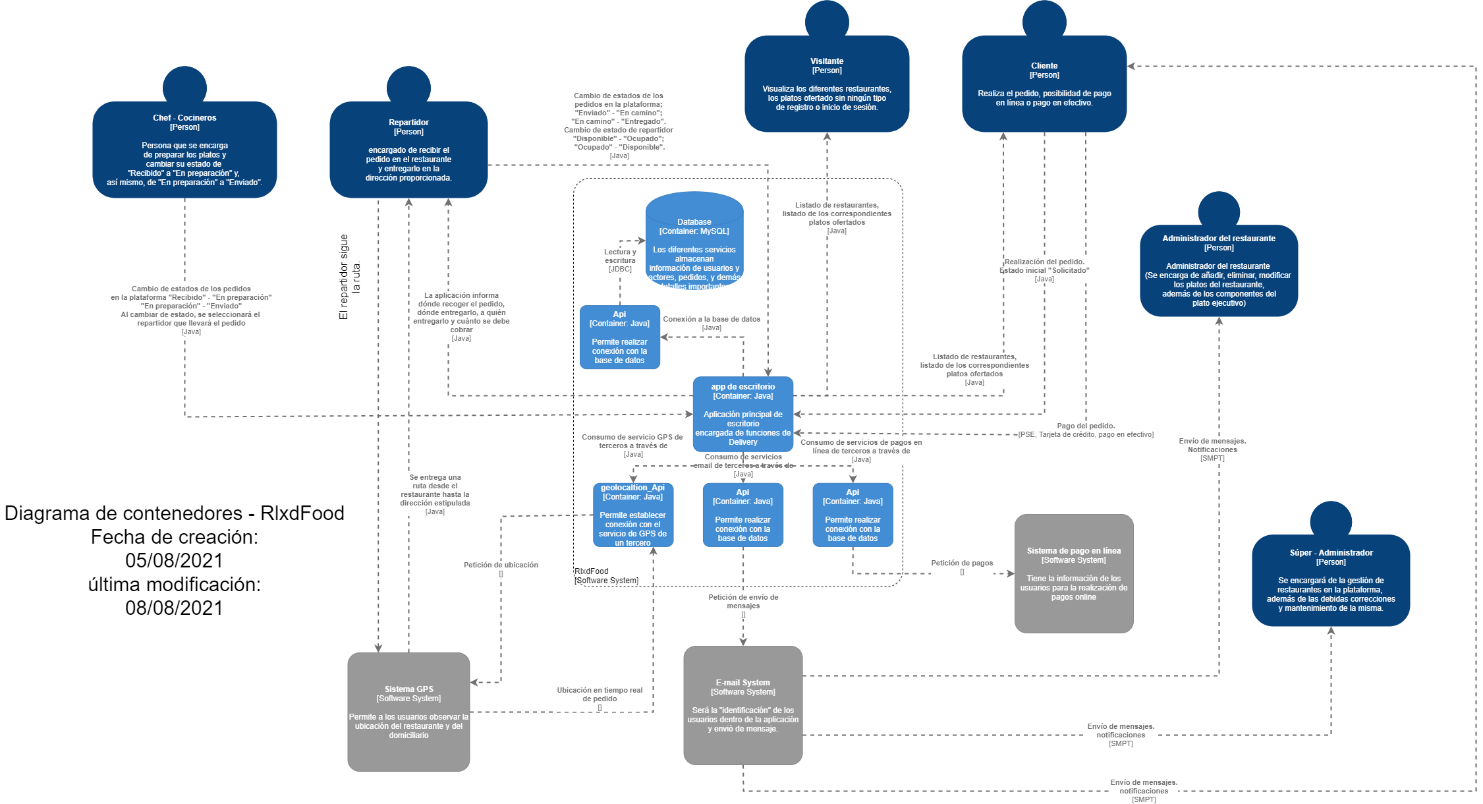
****

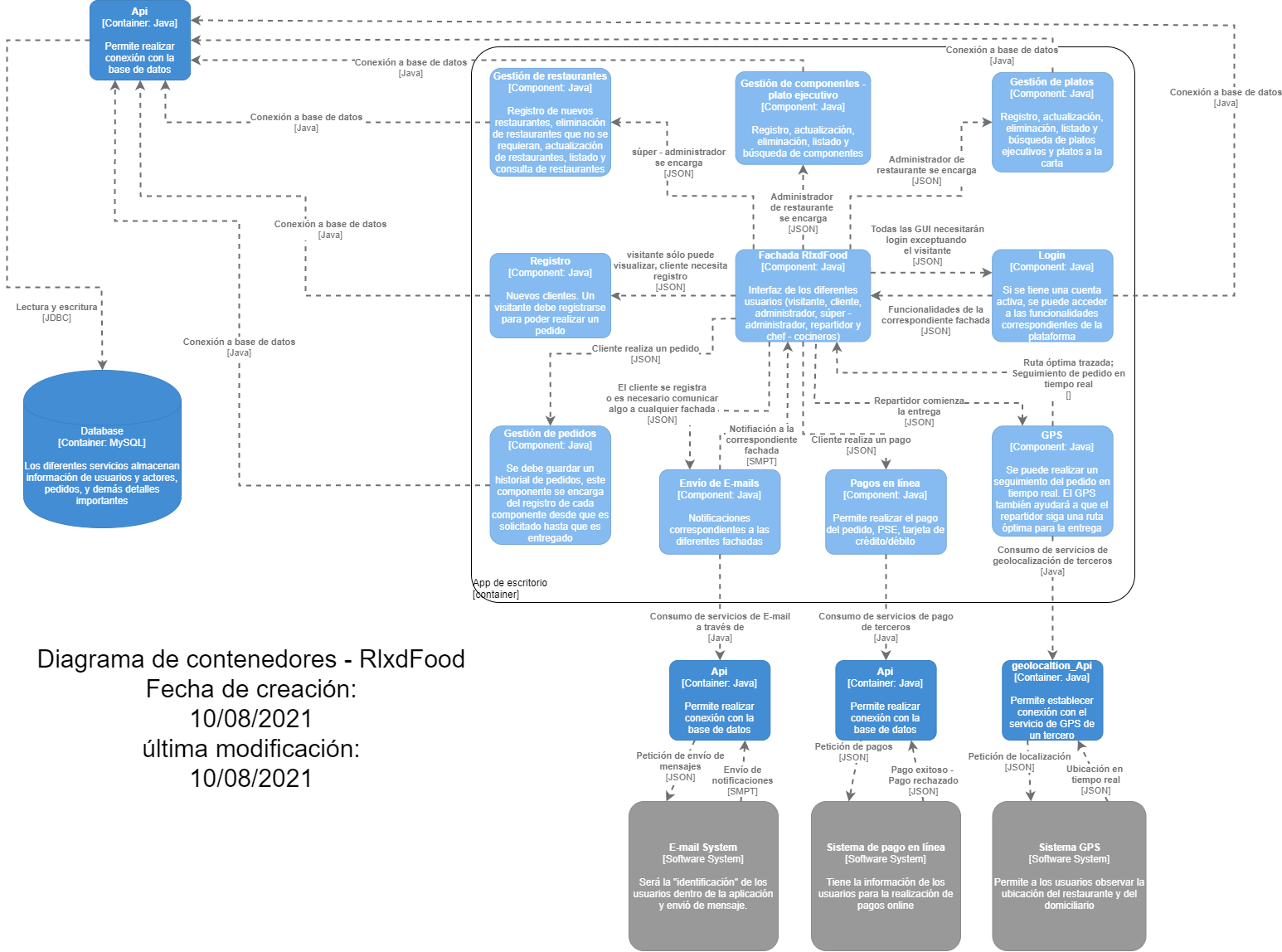
Figura Modelo de contexto de la aplicación RlxdFood

[**https://drive.google.com/file/d/12G9ttxV\_cWGGWpPNu5ZZOt9sB2iiDNbY/view?usp=sharing**](https://drive.google.com/file/d/12G9ttxV_cWGGWpPNu5ZZOt9sB2iiDNbY/view?usp=sharing)

1. **Vista lógica**
   1. **Vista de contenedores y componentes**

****

[**https://drive.google.com/file/d/1rYT\_ZSc9Kqlqmpmlz83gdq\_QZG9FB\_kG/view?usp=sharing**](https://drive.google.com/file/d/1rYT_ZSc9Kqlqmpmlz83gdq_QZG9FB_kG/view?usp=sharing)

****

[**https://drive.google.com/file/d/1ARPTh7ZWDu7MmjACByjsEvGm5M-7N8Oe/view?usp=sharing**](https://drive.google.com/file/d/1ARPTh7ZWDu7MmjACByjsEvGm5M-7N8Oe/view?usp=sharing)

* 1. **Vista de módulos**
  2. **Vista de componentes y conectores**

1. **Vista de implementación**
2. **Vista de instalación**
3. **Modelo Entidad – Relación**
4. **Resumen de las decisiones de arquitectura y su justificación**