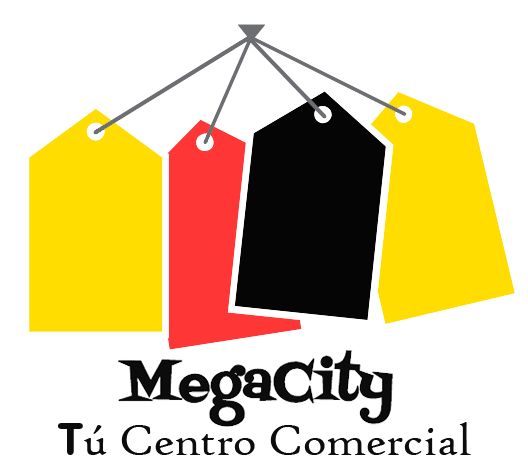
**Documento de Arquitectura de Software**

MegaCity

Ruth Cruz



Armenia, 22-Oct-2017

HISTORIAL DE CAMBIOS

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha** | **Descripción** | **Elaborado por:** |
| 0.0.1 | 18/10/17 | Creación del documento | Yenifer Hernandez – Mauricio Uribe |
| 0.0.2 | 20/10/17 | Especificación de metas y restricciones | Yenifer Hernandez – Mauricio Uribe |
| 0.0.3 | 22/10/17 | Especificación de escenarios y tácticas | Yenifer Hernandez – Mauricio Uribe |

TABLA DE CONTENIDO

[2 LISTA DE FIGURAS 5](#_Toc496462739)

[3 LISTA DE TABLAS 5](#_Toc496462740)

[1. INTRODUCCIÓN 6](#_Toc496462741)

[1.1. Propósito 6](#_Toc496462742)

[1.2. Alcance 6](#_Toc496462743)

[1.3. Definición, acrónimos y abreviaturas. 6](#_Toc496462744)

[2. ARQUITECTURA DE SOFTWARE 8](#_Toc496462745)

[3. METAS Y RESTRICCIONES ARQUITECTÓNICAS 9](#_Toc496462746)

[3.1.1 Priorización de requerimientos 9](#_Toc496462747)

[3.1.2 Resumen de los requerimientos 10](#_Toc496462748)

[3.1.3 Metas 15](#_Toc496462749)

[3.1.4 Restricciones 16](#_Toc496462750)

[4. VISTAS 16](#_Toc496462751)

[4.1. VISTA DE CASOS DE USO 17](#_Toc496462752)

[4.2. VISTA LÓGICA 28](#_Toc496462753)

[4.3. VISTA DE PROCESOS 29](#_Toc496462754)

[4.4. VISTA DE IMPLEMENTACIÓN 29](#_Toc496462755)

[4.5. VISTA FISICA 30](#_Toc496462756)

[4.6. VISTA DE DATOS 31](#_Toc496462757)

[5. ESCENARIOS Y TACTICAS 31](#_Toc496462758)

[5.1.1 Rendimiento o performance 31](#_Toc496462761)

[5.1.2 Gestión de recursos 31](#_Toc496462762)

[5.1.3 Seguridad 32](#_Toc496462763)

[5.1.4 Resistir ataques 32](#_Toc496462764)

[5.1.5 Modificabilidad 32](#_Toc496462765)

[5.1.6 Localización de cambios 32](#_Toc496462766)

[5.1.7 Usabilidad 32](#_Toc496462767)

[5.1.8 Separar la interfaz del resto del sistema 32](#_Toc496462768)

[5.1.9 El usuario toma la iniciativa 32](#_Toc496462769)

[5.1.10 Diálogos simples y naturales 32](#_Toc496462770)

[5.1.11 Hablar el lenguaje del usuario 33](#_Toc496462771)

[5.1.12 Proveer salidas marcadas claramente 33](#_Toc496462772)

[5.1.13 Dar buenos mensajes de error 33](#_Toc496462773)

[5.1.14 Prevenir y manejar errores 33](#_Toc496462774)

# LISTA DE FIGURAS

[Figura 1 arquitectura MegaCity 7](#_Toc496462730)

[Figura 2 Casos de Uso 16](#_Toc496462731)

[Figura 3 Diagrama de Paquetes 27](#_Toc496462732)

[Figura 4 Diagrama de Secuencia 27](#_Toc496462733)

[Figura 5 Diagrama de Clases 28](#_Toc496462734)

[Figura 6 Diagrama de Estados 28](#_Toc496462735)

[Figura 7 Diagrama de Despliegue 29](#_Toc496462736)

[Figura 8 Modelo de Datos 30](#_Toc496462737)

# LISTA DE TABLAS

[Tabla 1 Definiciones 6](#_Toc496462719)

[Tabla 2 Requerimientos Entregable 1 9](#_Toc496462720)

[Tabla 3 Requerimientos Entregable 2 10](#_Toc496462721)

[Tabla 4 Requerimiento Entregable 3 13](#_Toc496462722)

[Tabla 5 Descripción de casos de uso del administrador 16](#_Toc496462723)

[Tabla 6 Descripción de los casos de uso del cliente 25](#_Toc496462724)

# INTRODUCCIÓN

En el presente documento se muestra una descripción detallada del diseño de alto nivel de la aplicación para el centro comercial MegaCity. La cual es una aplicación que busca incentivar, promocionar e incrementar las visitas y/o ventas dentro del centro comercial, permitiéndoles a los clientes que registren sus facturas, indicando el almacén donde hacen las compras y de esta manera acumular puntos para que rediman por grandes premios.

Este diseño será representado por medio del modelo de “4+1 Vistas” [1], cuyo objetivo es mostrar, en cada una de las vistas, una perspectiva o visión de un conjunto de elementos del proyecto y sus relaciones, esto desde el punto de vista de la arquitectura. De esta manera presentar la documentación respectiva de este proyecto de forma ordenada, clara y coherente.

## Propósito

El propósito de este documento es especificar claramente la arquitectura de software a ser usada para el desarrollo de la aplicación para el centro comercial MegaCity .Con este documento se pretenden plasmar en términos arquitectónicos y de diseño todos los requerimientos definidos en el documento de iniciación entregado al cliente en días anteriores.

Este documento va dirigido a los stakeholders implicados en el proceso de desarrollo del software. Estos son para este proyecto los siguientes en mención: Product Owner, Scrum master y equipo (team).

Este documento es la principal fuente de información para empezar con el desarrollo del sistema y la integración de los componentes adquiridos.

## Alcance

Con este documento se pretende definir la arquitectura de software que será utilizada para el desarrollo de la aplicación para el centro comercial MegaCity. Como se verá posteriormente en este documento (ver sección Representación Arquitectónica) esta representación arquitectónica es especificada por las diferentes vistas del sistema (Casos de Uso, Lógica, Procesos, Despliegue e Implementación).También son detallados y direccionados otros aspectos importantes con respecto a los requerimientos no funcionales como lo son el tamaño, desempeño y la calidad de la arquitectura del sistema. Es preciso aclarar que estos diseños se realizan teniendo en cuenta los componentes de la plataforma Firebase para la gestión lógica del desarrollo.

## Definición, acrónimos y abreviaturas.

A continuación se lista algunas de las palabras más mencionadas durante el desarrollo y que pueden llegar a ser desconocidas o ambiguas por su definición técnica; aunque también en el pie de cada página estará la definición de las palabras con esta característica.

Tabla 1 Definiciones

|  |  |
| --- | --- |
| Palabra o Concepto | Definición |
| Ionic | Ionic es un framework gratuito y open source para desarrollar aplicaciones híbridas multiplataforma que utiliza HTML5, CSS (generado por SASS) y Cordova como base. Es uno de los framework del momento por utilizar AngularJS para gestionar las aplicaciones, lo que asegura aplicaciones rápidas y escalables. |
| Firebase | Servicio capaz de proveernos de un Backend en la nube con una fuente de datos en tiempo real y librerías para acceder a ella desde aplicaciones web, iOS y Android. |
| node.js | es un entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome. Node.js usa un modelo de operaciones E/S sin bloqueo y orientado a eventos, que lo hace liviano y eficiente. El ecosistema de paquetes de Node.js, npm, es el ecosistema más grande de librerías de código abierto en el mundo. |
| cordova | es un framework para el desarrollo de aplicaciones móviles propiedad de Adobe Systems que permite a los programadores desarrollar aplicaciones para dispositivos móviles utilizando herramientas web genéricas como JavaScript, HTML5 y CSS3, resultando aplicaciones híbridas. |
| angularJs | Es un proyecto de código abierto, realizado en Javascript que contiene un conjunto de librerías útiles para el desarrollo de aplicaciones web y propone una serie de patrones de diseño para llevarlas a cabo. En pocas palabras, es lo que se conoce como un framework para el desarrollo, en esta caso sobre el lenguaje Javascript con programación del lado del cliente. |
| typescript | es un lenguaje de programación de código abierto desarrollado y presentado por Microsoft hace unos tres años. Es un superconjunto de JavaScript que esencialmente añade capacidades de POO como es el tipado estático y objetos basados en clases. |
| servicio auth-jwt | es un método de autenticación mediante tokens |
| cloudMessagin | es un servicio gratuito que nos permite enviar información desde nuestro servidor a un dispositivo Android. El servicio GCM controla todo lo relacionado con el almacenamiento en cola de los mensajes y su entrega a las aplicaciones. |
| No-SQL | es un término que describe las bases de datos no relacionales de alto desempeño. Las bases de datos NO SQL utilizan varios modelos de datos, incluidos los de documentos, gráficos, claves-valores y columnas. Las bases de datos NO SQL son famosas por la facilidad de desarrollo, el desempeño escalable, la alta disponibilidad y la resiliencia. A continuación, presentamos varios recursos que le servirán para comenzar a usar las bases de datos NO SQL. |

# ARQUITECTURA DE SOFTWARE

Con el fin de describir la arquitectura de la aplicación para el centro comercial MegaCity de acuerdo a los requerimientos por pate del cliente, el tiempo y equipo de trabajo, se analizan diferente estructuras y tecnologías; después de indagar e investigar se toma la decisión de trabajar con diferentes arquitecturas, como cliente servidor para la lógica del negocio y mvvm para el desarrollo de la aplicación. Lo anteriormente descrito se encuentra graficado en la siguiente imagen, donde claramente se evidencian las tecnologías que se están implementando y la manera como estas interactúa.

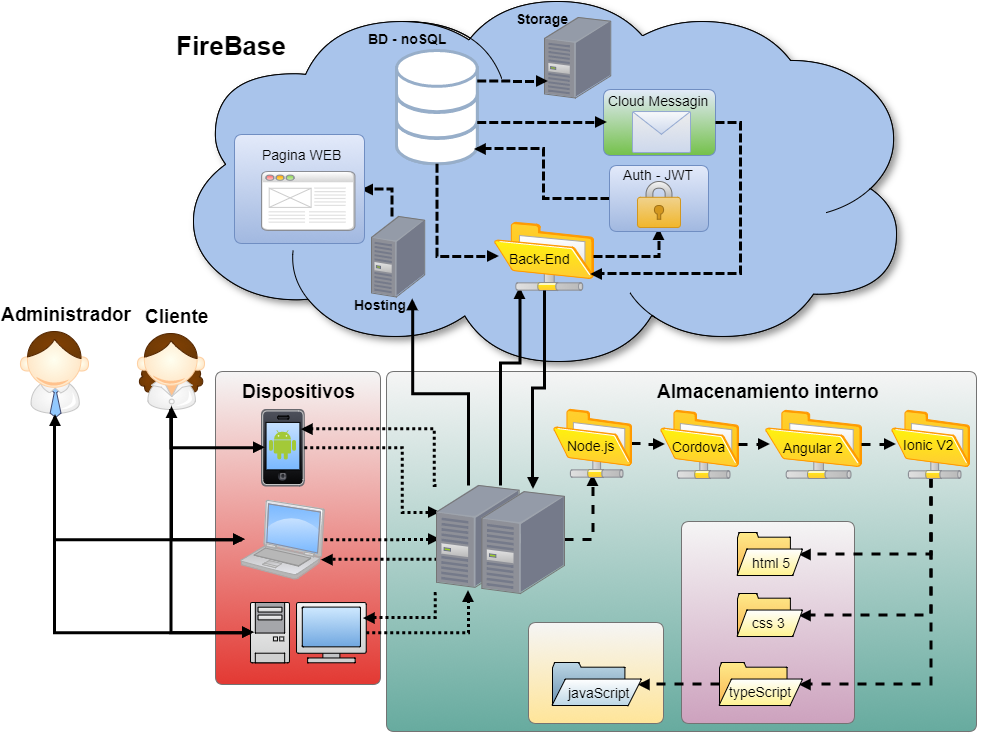


Figura 1 arquitectura MegaCity

De acuerdo con la figura anterior se nota el uso del framework Ionic[[1]](#footnote-1) en su versión 2 para el front-end[[2]](#footnote-2) o construcción de las pantallas o entorno visual de la aplicación; para la gestión de la lógica del negocio, back-end[[3]](#footnote-3) se utiliza la plataforma Firebase[[4]](#footnote-4).

Entonces a continuación se describe las dos divisiones de la arquitectura de la aplicación:

* Back-End: este es implementado sobre la plataforma Firebase de google, en la nube, gratuito, la cual provee una gestión del código del proyecto, como un almacenamiento en tiempo real sobre una base de datos No SQL[[5]](#footnote-5), además unos servicios adicionales y de vital importancia para la ejecución de este proyecto, como la autenticación JWT[[6]](#footnote-6), el cloud messagin[[7]](#footnote-7) , un hosting para el despliegue de la página web del sitio y un almacenamiento directo en el servidor (storage).
* Front-End: se realiza haciendo uso del framework Ionic, el cual se compone de un gestor de servicios node.js v6, framework de aplicaciones Cordova[[8]](#footnote-8), librerías de AngularJs[[9]](#footnote-9), permitiendo realizar entornos visuales de manera ágil haciendo uso de herramientas emergentes como html5, css3, y typescript[[10]](#footnote-10) (un conjunto empaquetado de JavaScript).

Adicional a esta descripción se emplea el modelo de vistas múltiples con el fin de describir aún más la aplicación del centro comercial MegaCity; este modelo organiza una descripción de la arquitectura del software utilizando cinco vistas concurrentes, permitiendo aproximar de manera aislada los intereses de los stakeholders de la arquitectura: los usuarios finales, el equipo de desarrollo, entre otros; y manejar de manera separada los requerimientos funcionales[[11]](#footnote-11).

Entonces empleamos los siguientes diagramas: De casos de uso, de paquetes, de despliegue, de estados, de secuencia y de clases, adicionalmente se definió un modelo de datos, todos estos diagramas serán descritos más adelante dentro de este documento.

# METAS Y RESTRICCIONES ARQUITECTÓNICAS

### Priorización de requerimientos

De acuerdo a lo acordado con el cliente, quien definió en orden que necesitaba con mayor urgencia, se realizó la distribución de todas las funcionalidades o experiencias de usuario en tres sprint’s o entregables, en donde definimos fechas de entrega lo cual quedo diligenciado y acordado con el cliente en el documento de iniciación del proyecto.

### Resumen de los requerimientos

En la siguiente tabla se enumeran, describen y valoran cada uno de los requerimientos, que para facilitar el ciclo de vida del proyecto se proyectaron como historias de usuario.

* Sprint 1

Tabla 2 Requerimientos Entregable 1

|  |  |  |
| --- | --- | --- |
| Historia: H1 | Peso: 5 | |
| Yo como un Admin CC  Quiero Registrarme  Para poder tener privilegios del centro comercial | | |
| Criterios:  Que no se repita el número de identificación  Que no se repita el username | | |
| Historia: H2 | | Peso: 3 |
| yo como Admin CC  Quiero Autenticar para  Poder acceder a la plataforma | | |
| Criterios:  Que identifique que soy un administrador y me asigne los permisos o privilegios necesarios.  Que el sistema solo me permita ingresar desde el navegador | | |
| Historia: H6 | | Peso: 2 |
| yo como Comprador  Quiero Gestionar puntos  Para poder canjearlos por premios | | |
| Criterios:  Que los puntos sean asignados correctamente al usuario | | |
| Historia: H8 | | Peso: 8 |
| yo como Comprador  Quiero Visualizar mis facturas  Para ver que facturas he subido | | |
| Criterios:  Que las facturas estén ordenadas de la mas reciente a la mas antigua | | |
| Historia: H11 | | Peso: 13 |
| yo como Comprador  Quiero Subir foto de la factura  Para hacer efectiva mi compra en el centro comercial y poder acumular puntos | | |
| Criterios:  Que la foto cumpla con las especificaciones de tamaño máximo. | | |
| Historia: H13 | | Peso: 3 |
| yo como Admin CC  Quiero Validar y Aceptar factura  Para verificar que la venta si sea del centro comercial y acumularle puntos a nuestros clientes | | |
| Criterios:  Que las fotos de las facturas cumplan todos los requisitos del CC  Que el estado de la factura validada pase correctamente a el estado validada  Que se asignen los puntos al usuario correspondiente | | |
| Historia: H14 | | Peso: 2 |
| yo como Admin CC  Quiero Ver facturas por validar  Para validar las facturas | | |
| Criterios:  Se deben mostrar las fotos en orden de fecha de registro, las más viejas primero. | | |
| Historia: H15 | | Peso: 2 |
| yo como Admin CC  Quiero Ver facturas validadas  Para analizar cuantas compras se realizan en el centro comercial | | |
| Criterios:  Que las facturas estén ordenadas de la mas reciente a la mas antigua | | |
| Historia: H17 | | Peso: 2 |
| yo como Admin CC  Quiero Denegar facturas  Para no acumular puntos a las personas que no son nuestros clientes, o para descartar las facturas duplicadas o incorrectas | | |
| Criterios:  Que se debe informar al usuario la denegación de la factura y el porque | | |
| Historia: H26 | | Peso: 13 |
| yo como Admin CC  Quiero Notificar a usuarios estado de las facturas  Para indicarle al usuario que su factura ha sido aprobada y sus puntos acumulados o denegada y no obtiene puntaje | | |
| Criterios:  Debe ser una notificación inmediata al usuario. | | |
| Historia: H30 | | Peso: 2 |
| yo como Comprador  Quiero Autenticar  Para poder acceder a la plataforma | | |
| Criterios:  Que la plataforma identifique que soy un cliente y muestre las actividades que puedo realizar  Que pueda ingresar desde mi móvil como desde un navegador | | |

* Sprint 2

Tabla 3 Requerimientos Entregable 2

|  |  |
| --- | --- |
| Historia: H3 | Peso: 2 |
| yo como Admin CC  Quiero Cambiar clave  Para poder cumplir con las políticas de contraseña | |
| Criterios:  Que las contraseñas cumplan con las políticas.  Que se envié un nuevo formulario para que el usuario cambie su contraseña | |

|  |  |
| --- | --- |
| Historia: H4 | Peso: 2 |
| yo como Admin CC  Quiero Cambiar nombre de usuario  Para personalizar mi perfil | |
| Criterios:  Que el nuevo nombre de usuario no exista | |
| Historia: H5 | Peso: 3 |
| yo como Admin CC  Quiero Cambiar foto de usuario  Para personalizar mi avatar | |
| Criterios:  Que la foto cumpla con las especificaciones de tamaño máximo. | |
| Historia: H7 | Peso: 5 |
| yo como Admin CC  Quiero Crear premios  Para atraer a los compradores al centro comercial | |
| Criterios:  Que incluya obligatoriamente la foto del premio  Que el premio no exista en la base de datos | |
| Historia: H18 | Peso: 5 |
| yo como Admin CC  Quiero Crear eventos  Para atraer más clientes al centro comercial e impulsar las ventas | |
| Criterios:  El evento creado no debe existir en DB. | |
| Historia: H19 | Peso: 5 |
| yo como Admin CC  Quiero Crear almacenes  Para conocer que productos ofrece el centro comercial en general y asi poder crear eventos y premios | |
| Criterios:  Almacenes no deben existir previamente. | |
| Historia: H20 | Peso: 5 |
| yo como Admin CC  Quiero Crear categorías  Para poder clásificar de manera correcta cada uno de los almacenes | |
| Criterios:  Categorías no deben existir previemanete. | |
| Historia: H22 | Peso: 2 |
| yo como Comprador  Quiero ver premios canjeados  Para ver que artículos o cosas he recibido como premio | |
| Criterios:  Los premios que se deben mostrar primero son los que puede canjear. | |
| Historia: H23 | Peso: 8 |
| yo como Admin CC  Quiero ver premios  Para conocer que premios se tienen ofertados | |
| Criterios:  Los premios deben salir en oden descendente por fecha de creación. | |
| Historia: H24 | Peso: 5 |
| yo como Comprador  Quiero ver puntos  Para saber con cuantos puntos cuento para luego canjear y que compras me los ha dado | |
| Criterios:  Los puntos se deben mantener actualizados con cada transacción. | |
| Historia: H25 | Peso: 5 |
| yo como Admin CC  Quiero ver almacenes  Para identificar que ofrece cada uno de los almacenes y elaborar estrategias de venta a partir de ellos | |
| Criterios:  Ordenados según políticas del CC | |
| Historia: H28 | Peso: 13 |
| yo como Comprador  Quiero Confirmar canje  Para comprobar que realmente quiero ese premio y que me descuenten esa cantidad de puntos | |
| Criterios:  Los puntos deben ser actualizados inmediatamente. | |
| Historia: H29 | Peso: 5 |
| yo como Comprador  Quiero Registrarme  Para poder tener privilegios del centro comercial | |
| Criterios:  Que permita subir la factura de manera agil ya sea tomar una foto o seleccionarla desde la galería | |
| Historia: H34 | Peso: 3 |
| yo como Comprador  Quiero ver premios  Para conocer que premios se tienen ofertados | |
| Criterios:  Que permita visualizar los puntos que requiere cada premio para ser redimido  Que estén ordenados alfabéticamente | |
| Historia: H35 | Peso: 5 |
| yo como Comprador  Quiero ver almacenes  Para identificar que ofrece cada uno de los almacenes y elaborar estrategias de venta a partir de ellos | |
| Criterios:  Que estén ordenados alfabéticamente | |

* Sprint 3

Tabla 4 Requerimiento Entregable 3

|  |  |
| --- | --- |
| Historia: H9 | Peso: 8 |
| yo como Comprador  Quiero Entrar a la página WEB  Para ver la descripción y detalle del centro comercial y sus almacenes | |
| Criterios:  Que sea corto pero que indique lo necesario  Que los almacenes se puedan filtrar por categoría | |
| Historia: H10 | Peso: 8 |
| yo como Admin  Quiero Entrar a la página WEB  Para ingresar a la aplicación del centro comercial | |
| Criterios:  Que muestre y solicite los datos de acceso | |
| Historia: H21 | Peso: 13 |
| yo como Admin CC  Quiero Generar informes (reportes)  Para tener soporte de las ventas, los premios entregados, los clientes, entre otras y  para tener soporte físico de las acciones dentro del centro comercial | |
| Criterios:  Que estén discriminados por funcionalidad  Que permitan ser exportados e imprimibles  Que estén en formato pdf | |
| Historia: H27 | Peso: 8 |
| yo como Admin CC  Quiero ver informes  Para consultar que informes se han generado y mirar si se genera otro diferente o se  consulta uno ya elaborado | |
| Criterios:  Que estén discriminados por funcionalidad  Que permitan ser exportados e imprimibles  Que estén en formato pdf | |
| Historia: H30 | Peso: 2 |
| yo como Comprador  Quiero Autenticar  Para poder acceder a la plataforma | |
| Criterios:  Que identifique que es administrador y asi asigne los privilegios correspondientes | |
| Historia: H31 | Peso: 2 |
| yo como Comprador quiero Cambiar clave para poder cumplir con las políticas de contraseña | |
| Criterios:  Se debe llevar al usuario a un formulario para que cambie su contraseña | |
| Historia: H32 | Peso: 2 |
| yo como Comprador  Quiero Cambiar nombre de usuario  Para personalizar mi perfil | |
| Criterios:  Que no se repita el número de identificación  Que no se repita el username | |
| Historia: H33 | Peso: 3 |
| yo como Comprador  Quiero Cambiar foto de usuario  Para personalizar mi avatar | |
| Criterios:  La foto debe cumplir los requerimientos máximos. | |
| Historia: H36 | Peso: 1 |
| yo como desarrollador  Quiero generar APK  Para correrla desde los dispositivos móviles y entregarle al cliente | |
| Criterios:  Que contenga todos los requerimientos  Que contenga un splash de acuerdo al logo del centro comercial  Que contenga un icono llamativo y acorde al logo del centro comercial  Que posea el nombre del centro comercial | |

### Metas

Una vez establecido el objetivo central de la aplicación o lo que el cliente desea, que es promover la visita y compra de articulo dentro del centro comercial MegaCity mediante la oferta de premios y eventos se disponen las siguientes metas:

* La aplicación debe permitir tomar la foto de la factura de compra de los clientes.
* La aplicación debe permitir aprobar las facturas, corroborando, lugar de compra o almacén, fecha y valor.
* La aplicación debe permitirle al cliente cargar la foto de la factura desde la galería de su dispositivo móvil o web .
* La aplicación debe permitir acumular puntos a los clientes de acuerdo a sus compras realizadas y registradas.
* La aplicación le debe notificar al cliente por los cambios ejecutados sobre sus facturas registradas, ya sea que se hayan aprobado o rechazado.
* La aplicación debe permitirle al cliente ver el historial de facturas que el a registrado.
* La aplicación le debe permitir al cliente ver, seleccionar o canjear un premio de acuerdo a sus puntos acumulados.
* La aplicación le debe permitir al cliente ver los eventos, los almacenes y los premios canjeados.
* La aplicación le debe permitir al administrador crear, modificar o eliminar premios, eventos y almacenes.
* La aplicación le debe permitir al administrador validar o denegar facturas.
* La aplicación debe contar con una página web informativa que le indique todo lo necesario a los clientes.

### Restricciones

Un vez establecido que se va a realizar dentro de la aplicación se listan las siguientes restricciones:

* El acceso para los clientes debe ser tanto móvil como web.
* La aplicación web debe ser responsiva.
* La aplicación móvil debe ser para sistemas operativos Android versión 4,2 en adelante.
* La aplicación web se debe poder acceder desde cualquier navegador.
* El acceso para los administradores debe ser solo por la aplicación web.
* La página web debe tener la posibilidad de dar acceso a la aplicación web.
* La página web debe ser responsiva.
* La aplicación debe poder descargarse desde la página web.
* La página web debe estar desplegada en la web.

# VISTAS

* Vista lógica: Ofrece soporte a los requerimientos funcionales, lo que el sistema debe proveer en términos de servicios a sus usuarios. En el presente documento, en la vista lógica se mostrará la división del sistema en paquetes.
* Vista de procesos: La vista de procesos permite describir los procesos del sistema y como estos se comunican.
* Vista física: En esta vista se muestra desde la perspectiva de un ingeniero de sistemas todos los componentes físicos del sistema así como las conexiones físicas entre esos componentes que conforman la solución (incluyendo los servicios).
* Vista de desarrollo o de implementación: En esta vista se muestra el sistema desde la perspectiva de un programador y se ocupa de la gestión del software; o en otras palabras, se va a mostrar cómo está dividido el sistema software en componentes y las dependencias que hay entre esos componentes.
* Vista de casos de uso o escenarios: Esta vista sirve para unir a las otras cuatro por medio de secuencias de interacciones que se llegan a establecer entre sus elementos. Tiene dos propósitos, sirve como una guía arquitectónica para descubrir elementos arquitectónicos en el momento del diseño, y para validar e ilustrar el diseño de la arquitectura.

.

## VISTA DE CASOS DE USO

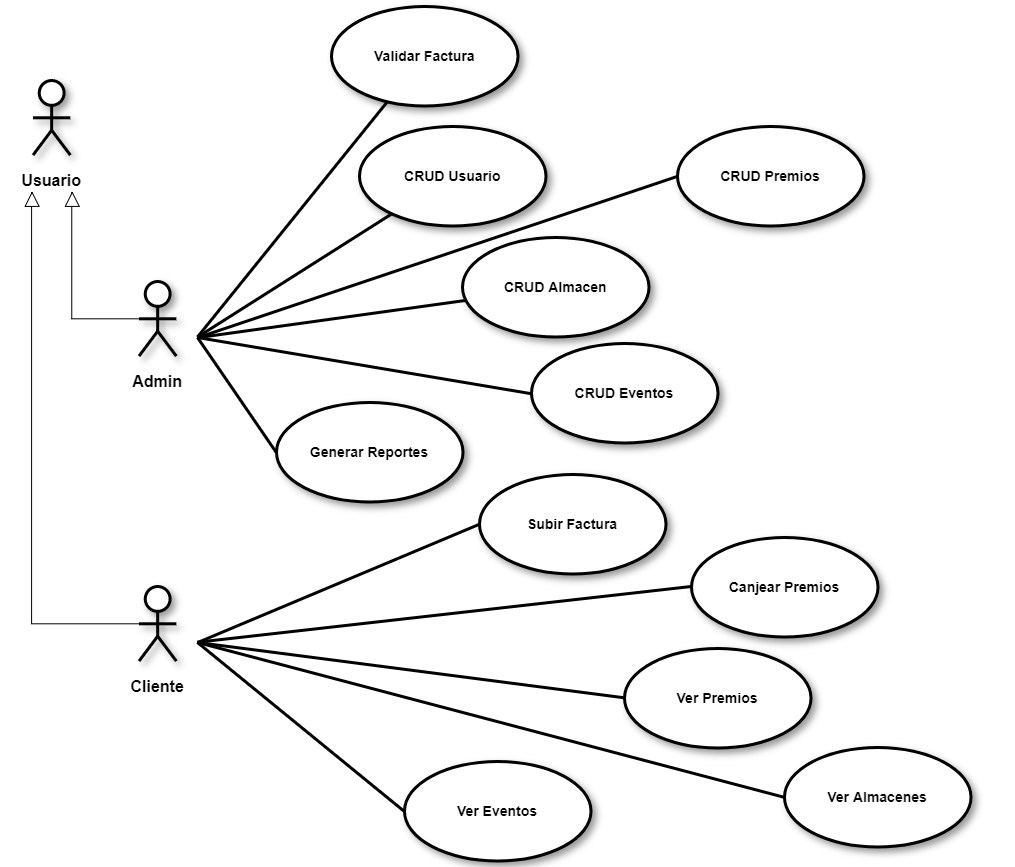


Figura 2 Casos de Uso

A continuación se describe uno a uno los casos de prueba para mayor comprensión de sistema

Tabla Descripción de casos de uso del administrador

|  |  |  |
| --- | --- | --- |
| **Nombre:** | Validar Factura | |
| **Actor:** | Admin | |
| **Descripción:** | El administrador decide si validar o revocar la factura | |
| **Precondición** | Tanto el administrador como el usuario que sube la factura debe estar autenticado en el sistema | |
| **Secuencia**  **Normal** | **Paso** | **Acción** |
| 1 | El administrador selecciona la opción validar facturas |
| 2 | El sistema le muestra una lista de facturas por validar |
| 3 | El administrador selecciona la factura que desea verificar |
| 4 | El sistema muestra los detalles de la factura seleccionada. |
| 5 | El administrador pone el valor de la factura y la acepta. |
| 6 | El sistema designa la factura como validada y asigna al usuario los puntos correspondientes. |
| 7 | El sistema devuelve al administrado a la pantalla principal |
| **Postcondición** | Los puntos de la factura se deben adicionar a los puntos del usuario y se debe notificar de los cambios en su factura | |
| **Excepciones** | **Paso** | **Acción** |
| 1 | El administrador deniega la factura. |
| **Nombre:** | CRUD USUARIO | |
| **Actor:** | Admin | |
| **Descripción:** | El administrador actualiza el estado de los clientes y/o consulta por sus datos | |
| **Precondición** | Tanto el administrador como el usuario que sube la factura debe estar autenticado en el sistema | |
| **Secuencia**  **Normal** | **Paso** | **Acción** |
| 1 | El administrador selecciona la opción administrar clientes |
| 2 | El sistema le muestra un listado de los clientes activos |
| 3 | El administrador selecciona el cliente que desea consultar o modificar su estado |
| 4 | El sistema muestra el detalle del cliente seleccionado. |
| 5 | El administrador visualiza por los campos del cliente o cambia el estado del cliente. |
| 6 | El sistema cambia el estado del cliente en caso de ser esta opción |
| 7 | El sistema devuelve al administrador a la pantalla del listado de los clientes |
| **Postcondición** |  | |
| **Excepciones** | **Paso** | **Acción** |
| 1 | El administrador solo puede modificar el estado del cliente. |
| **Nombre:** | CRUD PREMIOS | |
| **Actor:** | Admin | |
| **Descripción:** | El administrador crea, actualiza, consulta y elimina premios | |
| **Precondición** | El administrador debe estar autenticado en el sistema. | |
| **Secuencia**  **Normal** | **Paso** | **Acción** |
| 1 | El administrador selecciona la opción administrar premios |
| 2 | El sistema le muestra un listado de los premios. |
| 3 | El administrador selecciona el botón crear nuevo premio |
| 4 | El sistema muestra un formulario que le permite ingresar el detalle de premio a crear. |
| 5 | El administrador presiona el botón guardar |
| 6 | El sistema le solicita al administrador una confirmación de la acción. |
| 7 | El sistema valida que los campos obligatorios estén diligenciados |
| 8 | El sistema inserta el registro en la base de datos |
| 9 | El sistema le indica al administrador que el premio fue creado y lo lleva a la pantalla de listado de premios |
| **Secuencia alterna** | **paso** | **Acción** |
| 1 | El administrador selecciona la opción administrar premios |
| 2 | El sistema le muestra un listado de los premios. |
| 3 | El administrador selecciona un premio |
| 4 | El sistema le muestra un formulario con los datos del premio seleccionado |
| 5 | El administrador cambia uno o más datos del premio. |
| 6 | El administrador presiona el botón modificar |
| 7 | El sistema le solicita al administrador una confirmación de la acción. |
| 8 | El sistema verifica que los campos obligatorios estén diligenciados. |
| 9 | El sistema actualiza los campos modificados por el administrador. |
| 10 | El sistema le indica al administrador que los cambios fueron realizados y lo lleva a la pantalla de listado de premios |
| **Secuencia Alterna 2** | **Pasos** | **Acción** |
| 1 | El administrador selecciona la opción administrar premios |
| 2 | El sistema le muestra un listado de los premios. |
| 3 | El administrador selecciona un premio |
| 4 | El sistema le muestra un formulario con los datos del premio seleccionado |
| 5 | El administrador cambia el estado del premio a inactivo (borrado lógico) |
| 6 | El administrador presiona el botón modificar |
| 7 | El sistema le solicita al administrador una confirmación de la acción. |
| 8 | El sistema modifica el estado del premio |
| 9 | El sistema le indica al administrador que los cambios fueron realizados y lo lleva a la pantalla de listado de premios |
| **Postcondición** |  | El listado de premios activos del cliente debe mostrar los cambios ejecutados |
| **Excepciones** | **Paso** | **Acción** |
|  |  |
| **Nombre:** | CRUD ALMACEN | |
| **Actor:** | Admin | |
| **Descripción:** | El administrador crea, actualiza, consulta y elimina almacenes. | |
| **Precondición** | El administrador debe estar autenticado en el sistema. | |
| **Secuencia**  **Normal** | **Paso** | **Acción** |
| 1 | El administrador selecciona la opción administrar almacenes. |
| 2 | El sistema le muestra un listado de los almacenes. |
| 3 | El administrador selecciona el botón crear nuevo almacén. |
| 4 | El sistema muestra un formulario que le permite ingresar el detalle de almacén a crear. |
| 5 | El administrador presiona el botón guardar. |
| 6 | El sistema le solicita al administrador una confirmación de la acción. |
| 7 | El sistema valida que los campos obligatorios estén diligenciados. |
| 8 | El sistema inserta el registro en la base de datos. |
| 9 | El sistema le indica al administrador que el almacén fue creado y lo lleva a la pantalla de listado de almacenes. |
| **Secuencia alterna** | **paso** | **Acción** |
| 1 | El administrador selecciona la opción administrar almacenes |
| 2 | El sistema le muestra un listado de los almacenes. |
| 3 | El administrador selecciona un almacén |
| 4 | El sistema le muestra un formulario con los datos del almacén seleccionado |
| 5 | El administrador cambia uno o más datos del almacén. |
| 6 | El administrador presiona el botón modificar |
| 7 | El sistema le solicita al administrador una confirmación de la acción. |
| 8 | El sistema verifica que los campos obligatorios estén diligenciados. |
| 9 | El sistema actualiza los campos modificados por el administrador. |
| 10 | El sistema le indica al administrador que los cambios fueron realizados y lo lleva a la pantalla de listado de almacenes |
| **Secuencia Alterna 2** | **Pasos** | **Acción** |
| 1 | El administrador selecciona la opción administrar almacenes. |
| 2 | El sistema le muestra un listado de los almacenes. |
| 3 | El administrador selecciona un almacén |
| 4 | El sistema le muestra un formulario con los datos del almacén seleccionado. |
| 5 | El administrador cambia el estado del almacén a inactivo (borrado lógico). |
| 6 | El administrador presiona el botón modificar. |
| 7 | El sistema le solicita al administrador una confirmación de la acción. |
| 8 | El sistema modifica el estado del almacén. |
| 9 | El sistema le indica al administrador que los cambios fueron realizados y lo lleva a la pantalla de listado de almacenes. |
| **Postcondición** |  | El listado de almacenes activos del cliente debe mostrar los cambios ejecutados |
| **Excepciones** | **Paso** | **Acción** |
|  |  |
| **Nombre:** | CRUD EVENTOS | |
| **Actor:** | Admin | |
| **Descripción:** | El administrador crea, actualiza, consulta y elimina eventos. | |
| **Precondición** | El administrador debe estar autenticado en el sistema. | |
| **Secuencia**  **Normal** | **Paso** | **Acción** |
| 1 | El administrador selecciona la opción administrar eventos. |
| 2 | El sistema le muestra un listado de los eventos. |
| 3 | El administrador selecciona el botón crear nuevo evento. |
| 4 | El sistema muestra un formulario que le permite ingresar el detalle de evento a crear. |
| 5 | El administrador presiona el botón guardar. |
| 6 | El sistema le solicita al administrador una confirmación de la acción. |
| 7 | El sistema valida que los campos obligatorios estén diligenciados. |
| 8 | El sistema inserta el registro en la base de datos. |
| 9 | El sistema le indica al administrador que el evento fue creado y lo lleva a la pantalla de listado de almacenes. |
| **Secuencia alterna** | **paso** | **Acción** |
| 1 | El administrador selecciona la opción administrar eventos. |
| 2 | El sistema le muestra un listado de los eventos. |
| 3 | El administrador selecciona un evento |
| 4 | El sistema le muestra un formulario con los datos del evento seleccionado. |
| 5 | El administrador cambia uno o más datos del evento. |
| 6 | El administrador presiona el botón modificar |
| 7 | El sistema le solicita al administrador una confirmación de la acción. |
| 8 | El sistema verifica que los campos obligatorios estén diligenciados. |
| 9 | El sistema actualiza los campos modificados por el administrador. |
| 10 | El sistema le indica al administrador que los cambios fueron realizados y lo lleva a la pantalla de listado de eventos. |
| **Secuencia Alterna 2** | **Pasos** | **Acción** |
| 1 | El administrador selecciona la opción administrar eventos. |
| 2 | El sistema le muestra un listado de los eventos. |
| 3 | El administrador selecciona un evento |
| 4 | El sistema le muestra un formulario con los datos del evento seleccionado. |
| 5 | El administrador cambia el estado del evento a inactivo (borrado lógico). |
| 6 | El administrador presiona el botón modificar. |
| 7 | El sistema le solicita al administrador una confirmación de la acción. |
| 8 | El sistema modifica el estado del evento. |
| 9 | El sistema le indica al administrador que los cambios fueron realizados y lo lleva a la pantalla de listado de eventos. |
| **Postcondición** |  | El listado de eventos activos del cliente debe mostrar los cambios ejecutados |
| **Excepciones** | **Paso** | **Acción** |
|  |  |
| **Nombre:** | GENERAR REPORTES | |
| **Actor:** | Admin | |
| Descripción: | El administrador decide que reporte desea o requiere generar | |
| **Precondición** | El administrador debe estar autenticado en el sistema | |
| **Secuencia**  **Normal** | **Paso** | **Acción** |
| 1 | El administrador selecciona la opción generar reportes |
| 2 | El sistema le muestra una lista de reportes establecidos. |
| 3 | El administrador selecciona el reporte que requiere generar. |
| 4 | El sistema muestra los detalles del reporte. |
| 5 | El sistema le muestra un botón de exportar. |
| 6 | El administrador presiona el botón exportar. |
| 7 | El sistema genera un archivo pdf, listo para su descarga. |
| **Postcondición** |  | |
| **Excepciones** | **Paso** | **Acción** |
| 1 | El administrador deniega la factura. |

|  |  |  |
| --- | --- | --- |
| **Nombre:** | Subir Factura | |
| **Actor:** | Cliente | |
| **Descripción:** | El cliente de registra la factura de su compra por medio de una foto. | |
| **Precondición** | El cliente debe estar autenticado en el sistema | |
| **Secuencia**  **Normal** | **Paso** | **Acción** |
| 1 | El cliente selecciona la opción subir factura, tomándole una foto |
| 2 | El sistema abre la cámara del dispositivo. |
| 3 | El cliente le toma una foto a su factura de compra. |
| 4 | El sistema carga la foto tomada y la visualiza. |
| 5 | El cliente indica el almacén al que pertenece la factura. |
| 6 | El sistema designa la factura como validada y asigna al usuario los puntos correspondientes. |
| 7 | El sistema le solicita al cliente confirmación para registra la factura. |
| 8 | El sistema ingresa el registro a la base de datos. |
| **Secuencia**  **Alterna** | **Pasos** | **Acción** |
|  | **1** | El cliente selecciona la opción subir factura, seleccionado una foto de su galería. |
|  | 2 | El sistema abre la galería del dispositivo. |
|  | 3 | El cliente le selecciona la foto de su factura de compra. |
|  | 4 | El sistema carga la foto tomada y la visualiza. |
|  | 5 | El cliente indica el almacén al que pertenece la factura. |
|  | 6 | El sistema designa la factura como validada y asigna al usuario los puntos correspondientes. |
|  | 7 | El sistema le solicita al cliente confirmación para registra la factura. |
|  | 8 | El sistema ingresa el registro a la base de datos. |
| **Postcondición** | Todas las facturas registradas por los clientes serán visibles para el administrador para su respectiva aprobación | |
| **Excepciones** | **Paso** | **Acción** |
|  |  |
| **Nombre:** | Canjear Premios | |
| **Actor:** | Cliente | |
| **Descripción:** | El cliente canjea premios de acuerdo a sus puntos acumulados. | |
| **Precondición** | El cliente debe estar autenticado en el sistema. | |
| **Secuencia**  **Normal** | **Paso** | **Acción** |
| 1 | El cliente selecciona la opción ver premios. |
| 2 | El sistema le muestra un listado de los premios disponibles en orden de los que puede canjear de acuerdo a sus puntos. |
| 3 | El cliente identifica el premio que quiere y puede canjear. |
| 4 | El sistema le solicita confirmación para canjear el premio. |
| 5 | El sistema valida que los puntos acumulados sean suficientes para canjear el premio. |
| 6 | El sistema descuenta los puntos del premio al cliente. |
| 7 | El sistema crea el registro en los premios del cliente. |
| **Postcondición** | El cliente puede listar los premios canjeados “mis premios” y el cliente tendrá menos puntos acumulados | |
| **Excepciones** | **Paso** | **Acción** |
| 1 | Los puntos del premio son superior a los puntos acumulados por el cliente. |

Tabla Descripción de los casos de uso del cliente

|  |  |  |
| --- | --- | --- |
| **Nombre:** | Ver Premios | |
| **Actor:** | cliente | |
| **Descripción:** | El cliente ve el detalle de cada uno de los premios | |
| **Precondición** | El cliente debe estar autenticado en el sistema. | |
| **Secuencia**  **Normal** | **Paso** | **Acción** |
| 1 | El cliente selección la opción ver premios |
| 2 | El sistema le muestra un listado de los premios con los datos de cada uno |
| **Postcondición** |  |  |
| **Excepciones** | **Paso** | **Acción** |
|  |  |
| **Nombre:** | Ver Almacenes | |
| **Actor:** | cliente | |
| **Descripción:** | El cliente ve el detalle de cada uno de los almacenes | |
| **Precondición** | El cliente debe estar autenticado en el sistema. | |
| **Secuencia**  **Normal** | **Paso** | **Acción** |
| 1 | El cliente selección la opción ver almacenes |
| 2 | El sistema le muestra un listado de los almacenes con los datos de cada uno |
| **Postcondición** |  |  |
| **Excepciones** | **Paso** | **Acción** |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Nombre:** | Ver Eventos | |
| **Actor:** | cliente | |
| **Descripción:** | El cliente ve el detalle de cada uno de los eventos | |
| **Precondición** | El cliente debe estar autenticado en el sistema. | |
| **Secuencia**  **Normal** | **Paso** | **Acción** |
| 1 | El cliente selección la opción ver eventos |
| 2 | El sistema le muestra un listado de los eventos con los datos de cada uno |
| **Postcondición** |  |  |
| **Excepciones** | **Paso** | **Acción** |
|  |  |

## VISTA LÓGICA

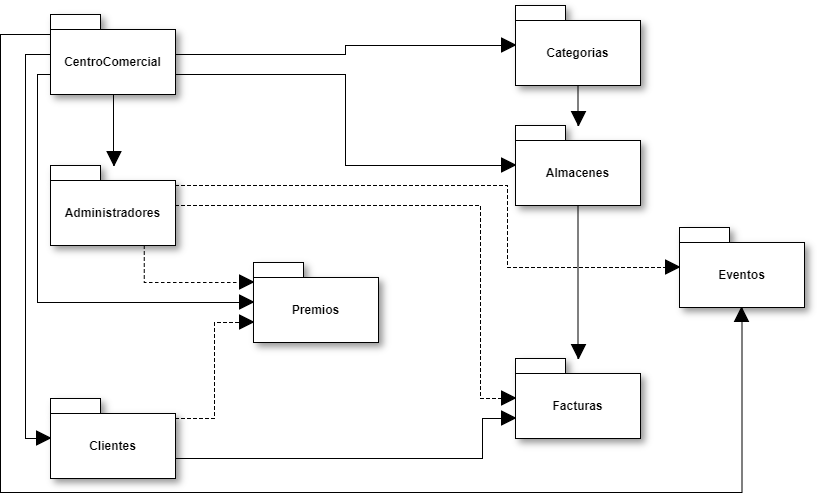


Figura 3 Diagrama de Paquetes

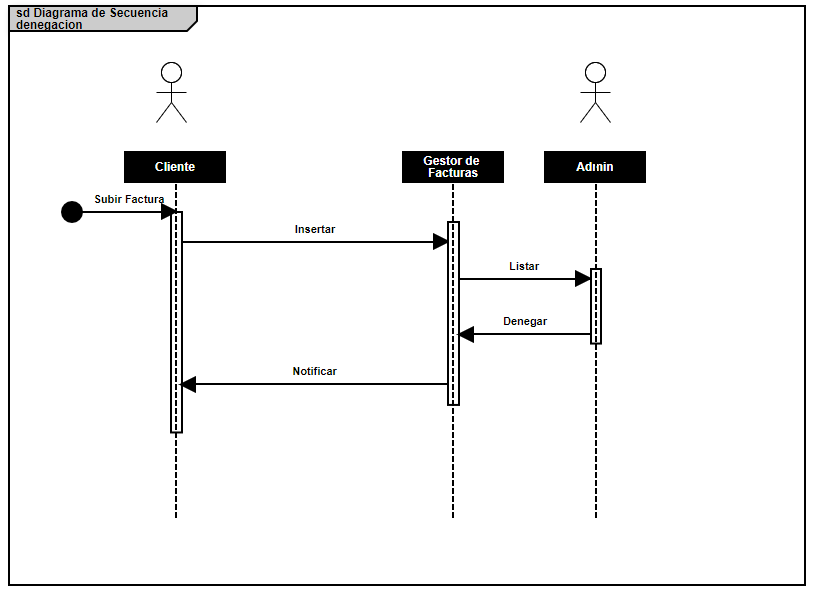


Figura 4 Diagrama de Secuencia

## VISTA DE PROCESOS

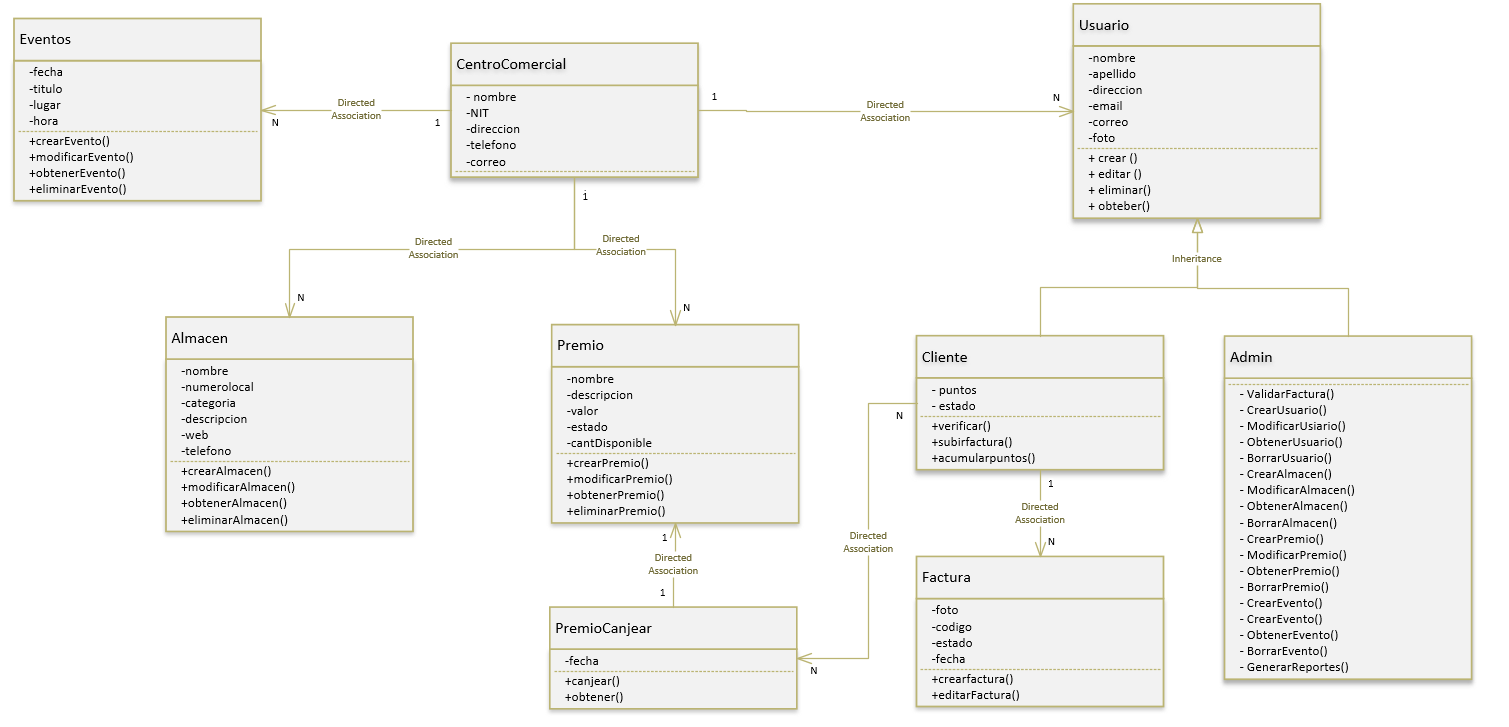


Figura 5 Diagrama de Clases

## VISTA DE IMPLEMENTACIÓN

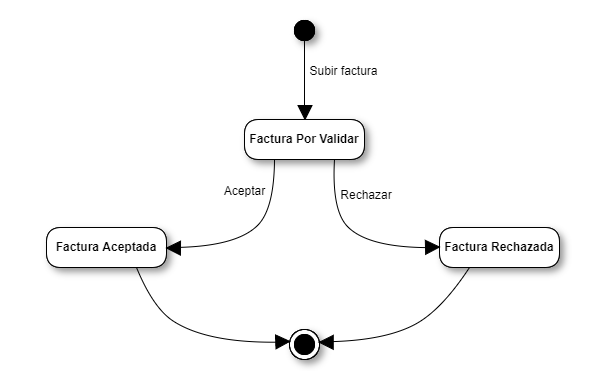


Figura 6 Diagrama de Estados

## VISTA FISICA

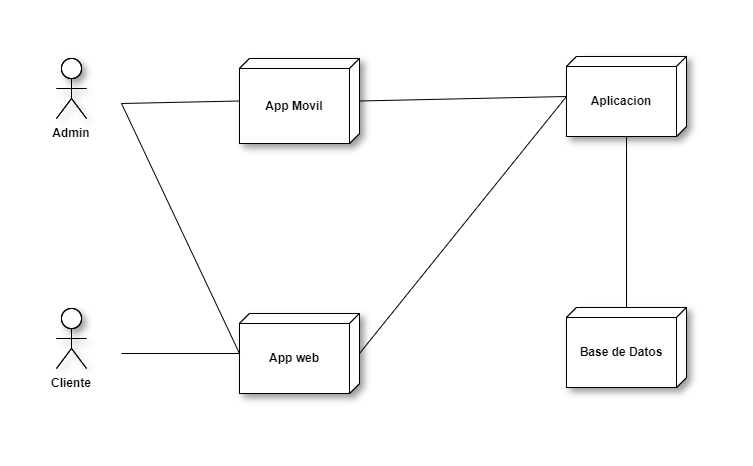


Figura 7 Diagrama de Despliegue

## VISTA DE DATOS

En esta vista se explica la manera como fueron almacenados los datos en la base de datos de la aplicación para el centro comercial MegaCity. Esta vista es vital porque el sistema se encarga de gestionar usuario, actividades y reportes.

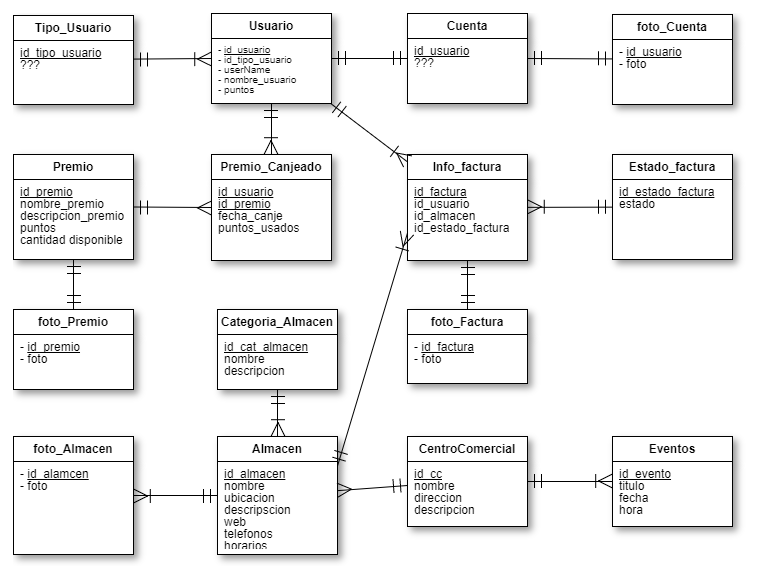


Figura 8 Modelo de Datos

# ESCENARIOS Y TACTICAS

En esta sección se describen las características de software que afectan de manera directa la arquitectura de la aplicación para el centro comercial MegaCity, como los son las tácticas que aportan atributos de calidad y de esta manera medir y evaluar la satisfacción de los usuario.



### Rendimiento o performance

Habilidad de un sistema para responder a un evento que ocurre (interrupciones, mensajes, pedidos de usuarios o paso del tiempo) dentro de un perfil definido.

### Gestión de recursos

De acuerdo con esto y aprovechando la ventaja de gestión de código que ofrece la plataforma Firebase de Google, en la cual se permite configurar hasta 100 usuarios concurrentes, para lo cual se entiende que dentro de la aplicación para el centro comercial MegaCity pueden 100 usuarios consultar al mismo tiempo los datos.

### Seguridad

Habilidad de un sistema para controlar, monitorear y auditar en forma confiable quién puede realizar qué acciones sobre el sistema y sus recursos, y la habilidad para detectar y recuperarse de fallas en los sistemas de seguridad.

### Resistir ataques

La aplicación para el centro comercial MegaCity cuenta con un servicio de autenticación por tokens, lo cual una vez que el usuario se autentica, el sistema le asigna un identificador único para la sesión activa del usuario, una vez cierra su sesión este acceso o este código único caduca, por ende nadie puede clonar la sesión y acceder de manera incorrecta o indebida.

### Modificabilidad

La modificabilidad de un sistema se refiere al atributo de calidad de una arquitectura software relativo al coste del cambio y a la facilidad con que un sistema informático puede adaptarse a los cambios.

### Localización de cambios

La aplicación para el centro comercial MegaCity cuenta una división del código y la interfaz gráfica, además una división de responsabilidades por función, facilitando la comprensión del mismo y la ubicación de cada una dentro de la estructura del proyecto.

### Usabilidad

La facilidad con la cual las personas interactúan con la aplicación en forma efectiva, y sobre como el sistema provee soporte al usuario en este sentido.

### Separar la interfaz del resto del sistema

La aplicación para el centro comercial MegaCity cuenta una división del código y la interfaz gráfica, tanto a nivel estructural como de responsable dentro del equipo de trabajo.

### El usuario toma la iniciativa

La aplicación para el centro comercial MegaCity cuenta con botones visibles como de cancelar, como confirmaciones antes de ejecutar acciones transaccionales, otorgándole así la responsabilidad al usuario que se encuentra manipulando los datos.

### Diálogos simples y naturales

La aplicación para el centro comercial MegaCity cuenta con diálogos simples y claros, indicándole al usuario lo que necesita o puede hacer dentro de la aplicación, además de colores básicos, fáciles de recordar y que anuncian acciones correspondientes al estudio del color.

### Hablar el lenguaje del usuario

La aplicación para el centro comercial MegaCity cuenta diálogos en lenguaje donde se tutea con el usuario, generando un ambiente de confianza con él, para que se sienta cómodo y facilite la comprensión de las acciones a realizar dentro de la aplicación.

### Proveer salidas marcadas claramente

La aplicación para el centro comercial MegaCity cuenta con botones que indican acciones de salir de la aplicación, cancelar una acción, ingresar a ella, guardar o modificar datos.

### Dar buenos mensajes de error

La aplicación para el centro comercial MegaCity cuenta con notificaciones que le permiten identificar al usuario de un error o una falta de acciones por parte de él, para llevar a cabo la acción deseada, le sistema le indica puntualmente en que falló.

### Prevenir y manejar errores

La aplicación para el centro comercial MegaCity cuenta con un manejo adecuado de cada una de las validaciones propias de la gestión de datos como las generadas por la aplicación.

1. Ionic: Ionic es un framework gratuito y open source para desarrollar aplicaciones híbridas multiplataforma que utiliza HTML5, CSS (generado por SASS) y Cordova como base. Es uno de los framework del momento por utilizar AngularJS para gestionar las aplicaciones, lo que asegura aplicaciones rápidas y escalables.  [↑](#footnote-ref-1)
2. Front-End: Son todas aquellas tecnologías que corren del lado del cliente, es decir, todas aquellas tecnologías que corren del lado del navegador web, generalizándose más que nada en tres lenguajes, HTML , CSS Y JavaScript, la persona encargada del front-end, se dedica solo a estas tres tecnologías. [↑](#footnote-ref-2)
3. Back-End: es aquel que se encuentra del lado del servidor, es decir, esta persona se encarga de lenguajes como PHP, Python, .Net, Java, etc, es aquel que se encarga de interactuar con bases de datos, verificar manejos de sesiones de usuarios, montar la página en un servidor, y desde este “servir” todas las vistas que el Front-End crea, es decir, uno como back-end se encarga más que nada de la manipulación de los datos. [↑](#footnote-ref-3)
4. Firebase: Servicio capaz de proveernos de un Backend en la nube con una fuente de datos en tiempo real y librerías para acceder a ella desde aplicaciones web, iOS y Android. [↑](#footnote-ref-4)
5. No SQL: es un término que describe las bases de datos no relacionales de alto desempeño. Las bases de datos No-SQL utilizan varios modelos de datos, incluidos los de documentos, gráficos, claves-valores y columnas. [↑](#footnote-ref-5)
6. JWT: es un método de autenticación mediante tokens. [↑](#footnote-ref-6)
7. Cloud Messagin: es un servicio gratuito que nos permite enviar información desde nuestro servidor a un dispositivo Android. [↑](#footnote-ref-7)
8. Cordova: Es un framework para el desarrollo de aplicaciones móviles propiedad de Adobe Systems que permite a los programadores desarrollar aplicaciones para dispositivos móviles utilizando herramientas web genéricas como JavaScript, HTML5 y CSS3, resultando aplicaciones híbridas. [↑](#footnote-ref-8)
9. AngularJs: Es un proyecto de código abierto, realizado en Javascript que contiene un conjunto de librerías útiles para el desarrollo de aplicaciones web y propone una serie de patrones de diseño para llevarlas a cabo. [↑](#footnote-ref-9)
10. TypeScript: es un lenguaje de programación de código abierto desarrollado y presentado por Microsoft hace unos tres años. Es un superconjunto de JavaScript que esencialmente añade capacidades de POO como es el tipado estático y objetos basados en clases. [↑](#footnote-ref-10)
11. Requerimientos no funcionales: un requisito que sabe bien y especifica criterios que pueden usarse para juzgar la operación de un sistema en lugar de sus comportamientos específicos, ya que éstos corresponden a los requisitos funcionales. [↑](#footnote-ref-11)