



Arquitectura de Software - grupo 2

Proyecto de software: UNdomicilio

Caso de estudio: Domicilios.com



Documento de Descripción de Arquitectura (ADD) v5.0

Integrantes

Christian Alfonso

David Rico

Sebastián Guerrero

Brayan Prieto

Julián Díaz

Gabriel Bejarano

Bogotá, DC. Enero de 2020

TABLA DE CONTENIDO

INTRODUCCIÓN	3
Nombre del sistema de software	3
Descripción del sistema de software	3
REQUISITOS FUNCIONALES (RF)	4
Historias de usuario	4
REQUISITOS NO FUNCIONALES (RNF)	8
Lenguajes y frameworks	8
Microservicio de pedidos	8
Microservicio de clientes y establecimientos	8
Microservicio Factura	9
Microservicio de mensajería	9
Microservicio de productos / publicaciones	9
Estilo y patrones de arquitectura	10
Plataforma (contenedores)	10
DISEÑO ARQUITECTÓNICO CON DESCRIPCIÓN POR VISTA	10
Vista de descomposición	10
Vista de capas	11
Vista de componentes y conectores	11
Vista de despliegue	12
Vista de modelos de datos (Diagramas E/R)	13

1. INTRODUCCIÓN

1.1. Nombre del sistema de software

Este documento denominado **Documento de Descripción de Arquitectura (ADD)**, tiene la finalidad de describir desde diversas perspectivas, la aplicación de software que se va a desarrollar durante el segundo semestre académico 2019-II, de manera que sea entendible por los distintos stakeholders o interesados.

El nombre de la aplicación es: **UNdomicilio**, la cual se deriva de la ya existente **domicilios.com**.

1.2. Descripción del sistema de software

Esta aplicación, tomará como base la plataforma existente Domicilios.com, buscando ser un prototipo de la ya mencionada con los requerimientos más básicos tanto funcionales como no funcionales.

Teniendo en cuenta lo anterior, la aplicación se trata de una plataforma en la web, que permita la interacción de básicamente dos tipos de usuarios (**clientes y establecimientos**). En donde el establecimiento pueda ofrecer su marca, sus productos y ofertas. Esto quiere decir, que no son solamente restaurantes o establecimientos de comida, sino también droguerías, almacenes, misceláneas, etc. Por otro lado, debe permitir a los clientes ingresar a la plataforma para descubrir productos de su interés, ya sea que busque uno en particular, o quiera ver el catálogo de productos que ofrecen los distintos establecimientos.

Adicionalmente, la plataforma debe proveer una interfaz de comunicación entre el cliente y el establecimiento, donde se pueda hacer un seguimiento de una compra (pedido) hecha en la plataforma, método de pago, punto de partida y punto de destino del pedido, y comunicación vía chat de texto entre el cliente y el establecimiento.

2. REQUISITOS FUNCIONALES (RF)

2.1. Historias de usuario

En esta sección, se presentan los requerimientos funcionales del sistema, por medio de **historias de usuario (HU)**.

Las historias de usuario son descripciones informales, desde la perspectiva del usuario final, que explican la funcionalidad deseada del sistema.

ítem	Descripción
Identificación	HU-01
Título	Publicación de productos y promociones
Descripción	<ul style="list-style-type: none">El sistema permitirá al usuario de tipo establecimiento hacer publicaciones sobre productos que ofrece con una breve descripción, su precio e imagen.El sistema permitirá realizar publicaciones sobre alguna promoción que tenga cada usuario de tipo establecimiento junto con una descripción, condiciones e imagen(es).
Criterios de aceptación	<ul style="list-style-type: none">Publicación de un producto en la sección del establecimientoPublicación de una promoción donde involucre uno o más productos existentes

ítem	Descripción
Identificación	HU-02
Título	Información de establecimientos
Descripción	<ul style="list-style-type: none">El sistema le permitirá al usuario tipo cliente consultar información de interés sobre los establecimientos registrados como una descripción, la ubicación, el horario de atención, productos que manejan y calificaciones otorgadas.También debe permitir al usuario consultar las promociones vigentes publicadas por los

	establecimientos y la posibilidad de filtrar las publicaciones de acuerdo al tipo de establecimiento, productos que ofrecen, fecha, hora entre otros.
Criterios de aceptación	<ul style="list-style-type: none"> • Ver la información completa y correcta de un establecimiento. • Aplicación exitosa de distintos tipos de filtros por parte del cliente.

ítem	Descripción
Identificación	HU-03
Título	Información de clientes
Descripción	<ul style="list-style-type: none"> • El sistema le permitirá al usuario tipo cliente consultar información de interés sobre los establecimientos registrados como una descripción, la ubicación, el horario de atención, productos que manejan y calificaciones otorgadas. • También debe permitir al usuario consultar las promociones vigentes publicadas por los establecimientos y la posibilidad de filtrar las publicaciones de acuerdo al tipo de establecimiento, productos que ofrecen, fecha, hora entre otros.
Criterios de aceptación	<ul style="list-style-type: none"> • Ver la información completa y correcta de un establecimiento. • Aplicación exitosa de distintos tipos de filtros por parte del cliente.

ítem	Descripción
Identificación	HU-04
Título	Facturación y cobros
Descripción	<ul style="list-style-type: none"> • El sistema le permitirá a los usuarios tipo cliente hacer sus pedidos a los usuarios de tipo establecimiento. • El sistema le permitirá a los usuarios tipo establecimiento dar a conocer los métodos de pago que manejan.

	<ul style="list-style-type: none"> • El sistema permitirá los usuarios tipo cliente elegir el método de pago para los pedidos, adiciones a su pedido y observaciones extras. • El sistema ofrecerá a los usuarios tipo cliente una factura al recibir cada pedido y así tener el resumen de la transacción
Criterios de aceptación	<ul style="list-style-type: none"> • Generación de un pedido de forma exitosa • Gestión correcta de los campos adicionales que puede tener un pedido, y que estos datos sean consistentes con los datos que le llegan al establecimiento.

ítem	Descripción
Identificación	HU-05
Título	Mensajería
Descripción	<ul style="list-style-type: none"> • El sistema permitirá el envío de mensajes entre los usuarios tipo cliente y los usuarios tipo establecimiento • El sistema deberá enviar notificaciones a los usuarios tipo cliente sobre nuevas promociones de los usuarios tipo establecimiento. • El sistema mensajes notificando a los establecimientos sobre cualquier novedad que el cliente realice sobre un pedido.
Criterios de aceptación	<ul style="list-style-type: none"> • Generación correcta de notificaciones y mensajes. • Las notificaciones y mensajes deben llegar a los usuarios apropiados.

ítem	Descripción
Identificación	HU-06
Título	Creación de usuarios
Descripción	<ul style="list-style-type: none"> • El sistema permitirá que los usuarios creen su cuenta llenando un formulario simple con sus datos básicos personales (nombres, apellidos, fecha de nacimiento, nombre de usuario, documento de identidad, correo

	electrónico, contraseña, etc.) al igual que el tipo de usuario que será.
Criterios de aceptación	<ul style="list-style-type: none"> • Permitir la creación de un usuario de tipo cliente • Permitir la creación de un usuario de tipo establecimiento

ítem	Descripción
Identificación	HU-07
Título	Inicio de sesión
Descripción	<ul style="list-style-type: none"> • El sistema permitirá que los usuarios inicien sesión con su nombre de usuario/correo y contraseña, cargando la información correctamente de su perfil. Además, el sistema le permitirá al usuario cerrar su sesión de forma segura.
Criterios de aceptación	<ul style="list-style-type: none"> • Que el sistema cargue correctamente la información para un perfil específico

ítem	Descripción
Identificación	HU-08
Título	Generación de pedidos
Descripción	<ul style="list-style-type: none"> • El sistema permitirá a los usuarios tipo cliente realizar uno o más pedidos a uno o más usuarios tipo establecimiento • El sistema permitirá modificar los productos de los pedidos antes de confirmarlos • El sistema le ofrecerá a los clientes tipo usuario un detalle de los precios de los productos del pedido y del costo total • El sistema le dará a los usuarios de tipo cliente y tipo usuario un identificador único de pedido para consultarlos posteriormente • El sistema le permitirá al usuario tipo cliente confirmar el pedido para que este le sea despachado
Criterios de aceptación	<ul style="list-style-type: none"> • Que el sistema cargue correctamente la información

	para un perfil específico
--	---------------------------

3. REQUISITOS NO FUNCIONALES (RNF)

3.1. Lenguajes y frameworks (Back-end)

Para el desarrollo de la aplicación, más precisamente el back-end de esta, se cuenta con 5 servicios principales (en la siguiente sección se detalla la arquitectura). A continuación se describen estos 5 servicios y las tecnologías con los cuales se van a desarrollar.

3.1.1. Microservicio de pedidos

Es el microservicio encargado de procesar las peticiones del cliente para generar un pedido con los productos/promociones que haya seleccionado. Será el encargado de gestionar para cada pedido: el usuario cliente que lo solicitó, que productos incluye, la fecha en que se solicitó, la fecha en que se finalizó (si aplica), el estado en que se encuentra el pedido y su descripción o anotaciones.

Este microservicio será construido mediante el entorno de ejecución de JavaScript **Nodejs**. En cuanto a la base de datos donde se almacenará la información, será desarrollada en el motor de base de datos relacionales **MySQL**.

3.1.2. Microservicio de clientes y establecimientos

Es el microservicio encargado de procesar el perfil de los dos tipos de usuarios de la aplicación: **cliente y establecimiento**. Involucra los procedimientos necesarios para actualizar estos perfiles, almacenar esta información consistentemente, y enviar dicha información a la capa de presentación (más adelante) de forma adecuada.

Este microservicio será construido mediante el framework **Express.js** de Nodejs. En cuanto a la base de datos donde se almacenará la información, será desarrollada en el motor de base de datos de tipo noSQL conocido como **MongoDB**.

3.1.3. Microservicio Factura

Se encarga del manejo de las facturas, almacenando la suma total de la factura y sumándole el costo del impuesto IVA

Este microservicio será construido mediante el framework de Python **Django**. En cuanto a la base de datos donde se almacenará la información, será desarrollada en el motor de base de datos relacional **SQLite3**.

3.1.4. Microservicio de mensajería

Es el microservicio encargado de administrar la comunicación cliente y establecimiento. Permitiendo una comunicación eficiente entre ambos durante la ejecución de un pedido, y un tiempo después en caso de algún comentario extraordinario (queja, reclamo, inquietud o sugerencia) del cliente o establecimiento.

Este microservicio será construido mediante el framework **Ruby on Rails**. En cuanto a la base de datos donde se almacenará la información, será desarrollada en el motor de base de datos relacional conocido como **PostgreSQL**.

3.1.5. Microservicio de productos / publicaciones

Es el microservicio encargado de administrar los productos y promociones que un establecimiento ofrece. Permite gestionar y almacenar información que es de interés para el usuario, ofrece herramientas de filtros, búsqueda por palabras clave y funciones acordes con el tipo de productos a partir del tipo de establecimiento.

Este microservicio será construido mediante el entorno de ejecución de JavaScript **Nodejs**. En cuanto a la base de datos donde se almacenará la

información, será desarrollada en el motor de base de datos relacionales **MySQL**.

3.2. Lenguajes y frameworks (Front-end)

Para el desarrollo de la aplicación, en el lado del front-end, se realizó un consenso entre todo el equipo que involucra a todos sus microservicios. Si bien, hay algunos microservicios que cuentan con más de una vista, la verdad es que todas las vistas están enlazadas entre sí. Por este y otros motivos, se decidió trabajar con el mismo framework para front: Este es **React.js**.

Por su amplio uso y recepción positiva entre la comunidad de desarrolladores y también por la versatilidad y extensa documentación que posee el lenguaje de programación **JavaScript**.

3.3. Estilo y patrones de arquitectura

El patrón de arquitectura para el desarrollo del proyecto es el patrón por capas, en este caso microservicios con REST, de igual manera el estilo escogido es el de capas con unión de un estilo orientado a servicios.

Para esta entrega se tendrán dos escenarios, uno que es el convencional, manejado durante todos los sprints anteriores. Y un nuevo escenario, con los componentes software escalados a **2 instancias**.

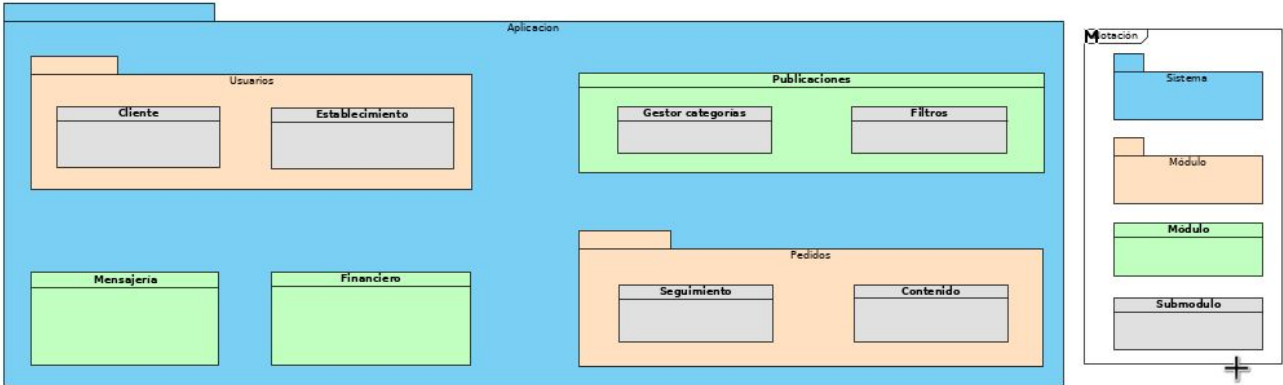
3.4. Plataforma (contenedores)

La aplicación será desplegada y ejecutada sobre plataformas web usando los servicios de Google de 'Computación en la nube'. Este servicio es comúnmente conocido como **Google Cloud**.

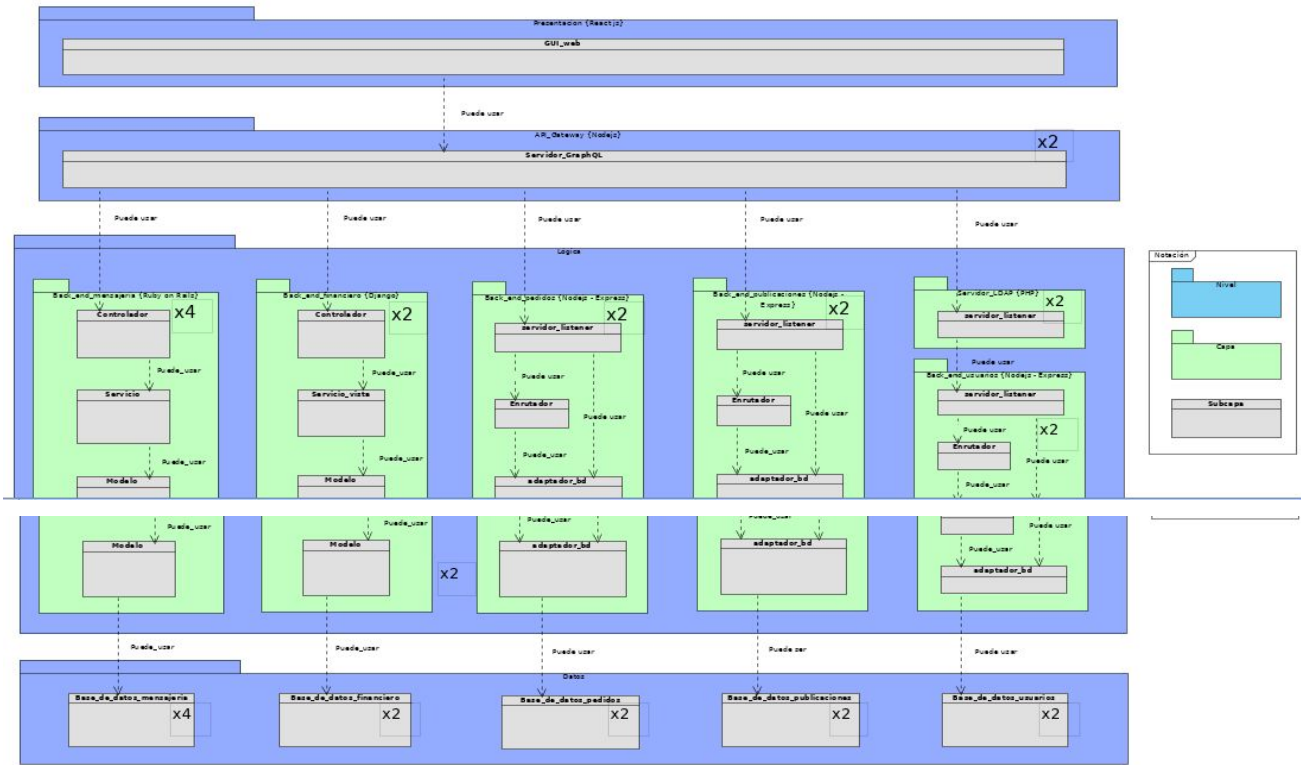
4. DISEÑO ARQUITECTÓNICO CON DESCRIPCIÓN POR VISTA

En la presente sección se describen las principales vistas que describen la disposición y composición de los microservicios ya mencionados.

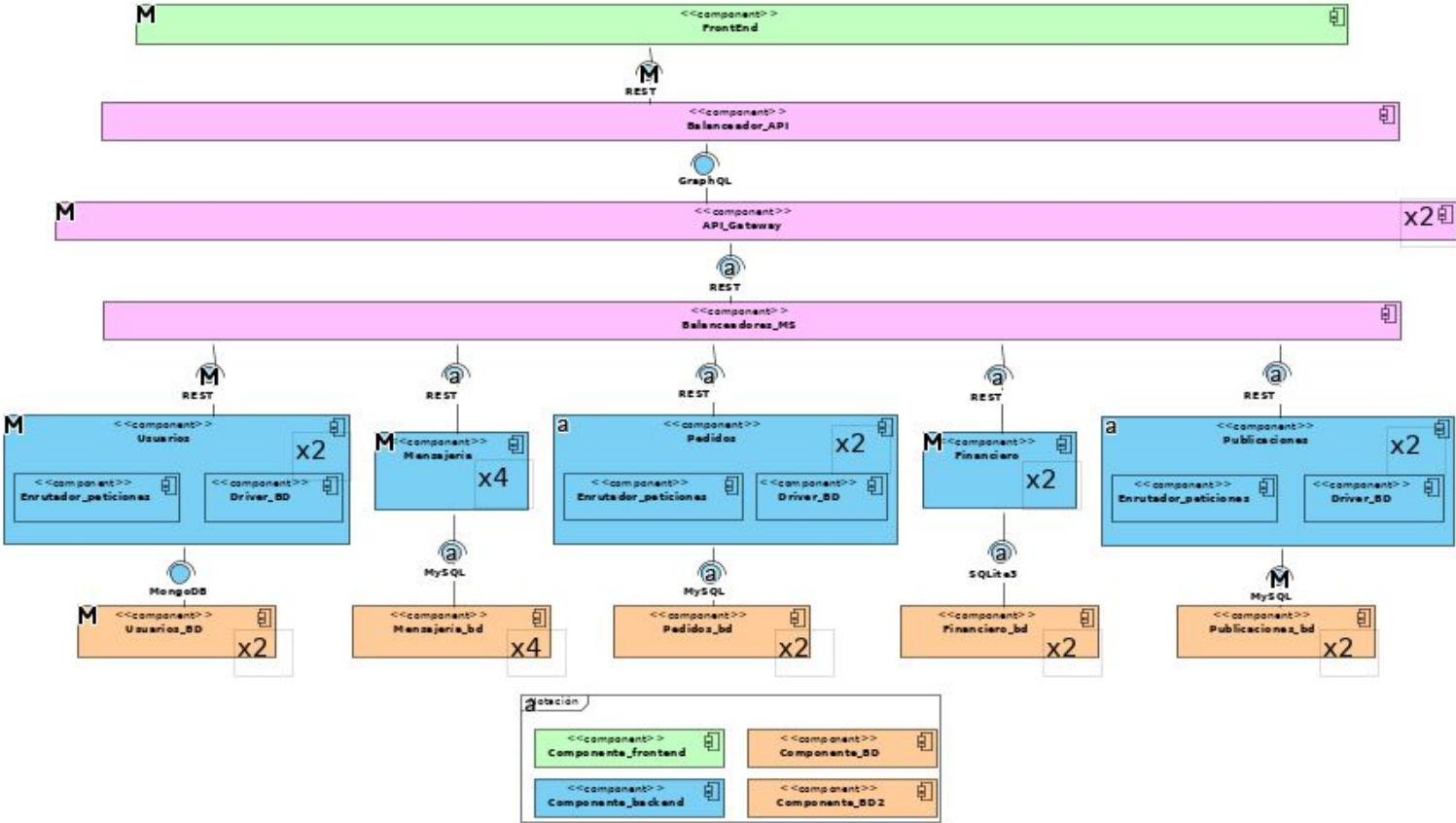
4.1. Vista de descomposición



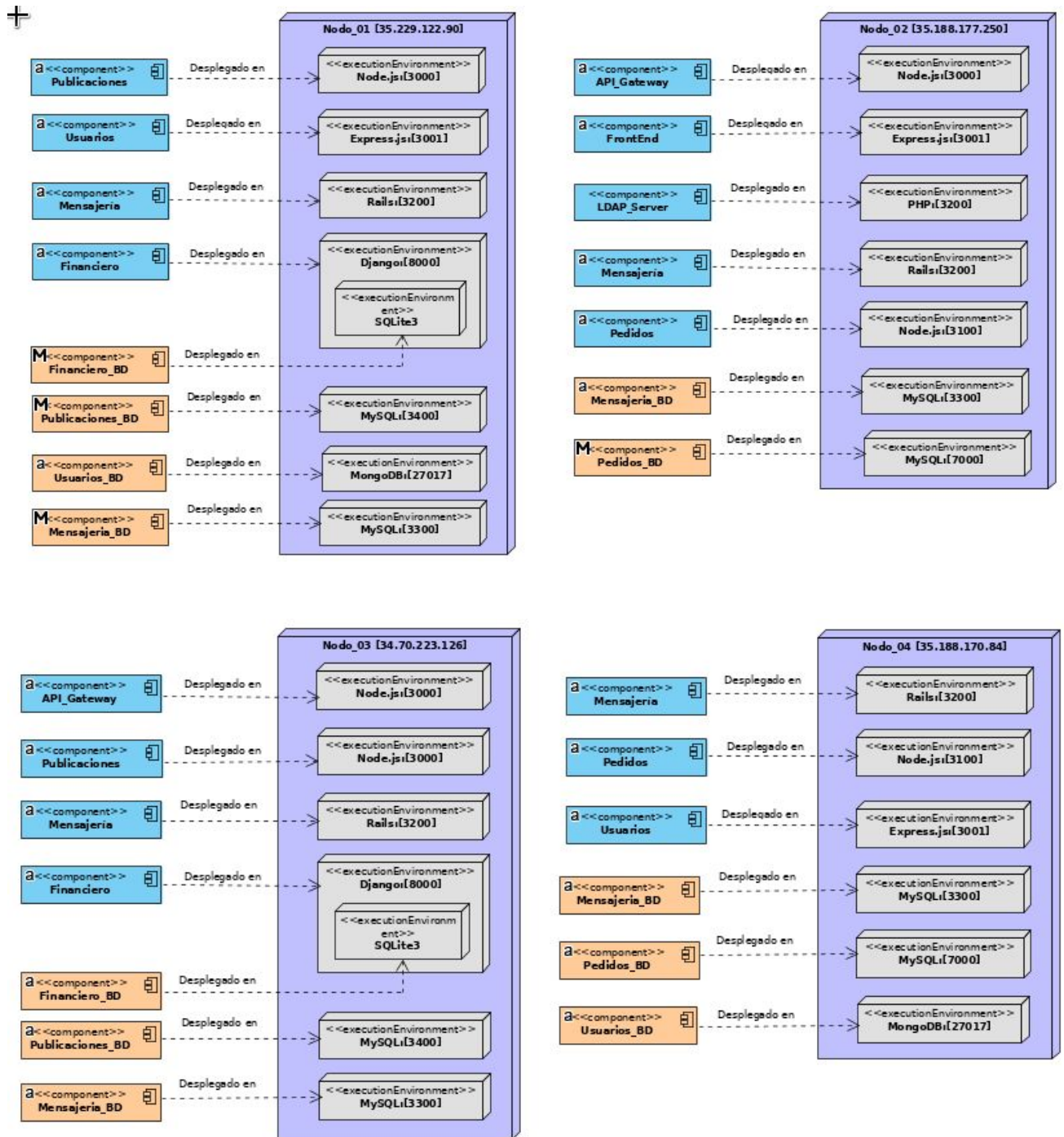
4.2. Vista de capas



4.3. Vista de componentes y conectores



4.4. Vista de despliegue



4.5. Vista de modelos de datos (Diagramas E/R)

