



**Universidad de Guadalajara**  
**Centro Universitario de Ciencias Exactas e Ingenierías**



# **Diseño lógico-físico**

**Equipo:**

Encinas Mardueño Catherine Michelle (Código: 215440902)

González Olmos Alan (Código: 219748286)

**Carrera:** Ingeniería Informática

**Materia:** Seminario de Solución de Problemas de Ingeniería de Software I

**Sección:** D-01

**Aula:** X-11

**Profesor:** Ávila Cárdenas Karla

## Contenido

Introducción.....	3
Desarrollo .....	4
Conclusiones.....	7

## Introducción

Para esta práctica, seguiremos haciendo uso de diagramas UML, estos diagramas son útiles para visualizar y documentar sistemas de software, pero la terminología puede llegar a ser abrumadora y confusa para alguien fuera del contexto. Por lo que se debe utilizar cierta simbología para que cualquier persona llegue a entender cómo funciona el algoritmo de nuestro software. Algunos de los diagramas incluidos en UML son diagramas de estructuras, diagramas de interacción, diagramas de estado, diagramas de componentes, despliegue, etc. Estos diagramas son considerados diagramas de comportamiento porque describen lo que debe suceder en el sistema y el orden en el que se debe realizar. Es importante realizar un diagrama de comunicación clara y concisa, ya que es necesario que las personas que incluso no formen parte del equipo de desarrollo entiendan el algoritmo y puedan participar en la creación o modificación del algoritmo del sistema.

Algunos de los beneficios de los diagramas UML son:

- Demostrar la lógica de un algoritmo
- Describir los pasos realizados en un caso de uso UML
- Ilustrar un proceso de negocios o flujo de trabajo entre los usuarios del sistema y el sistema mismo.
- Simplificar y mejorar casos de uso complicados.
- Modelar elementos de arquitectura de software.

Algunos de los componentes básicos de los diagramas UML son:

- Acción: Describe un paso en la actividad que los usuarios o el software realiza.
- Nodo de decisión: Una rama condicional en el flujo que se representa con un diamante o rombo. Tiene una entrada y puede incluir dos o más salidas.
- Flujos de control: Otro nombre para los conectores que muestran el flujo entre los pasos de un diagrama.
- Nodo inicial: Indica el inicio de la actividad.
- Nodo terminal: Indica el paso final de una actividad

## Desarrollo

Diagrama de estado:

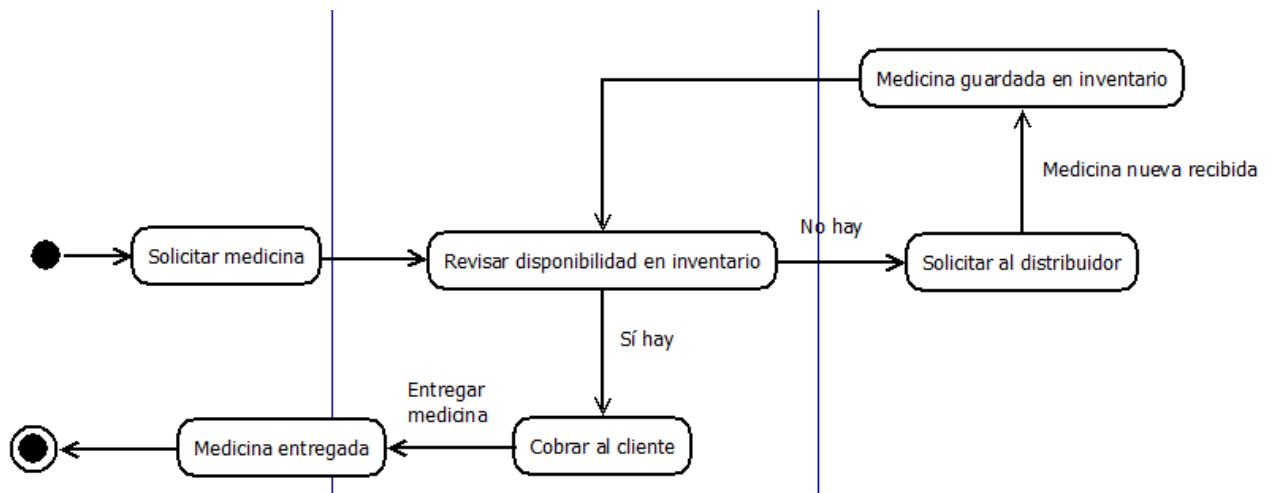
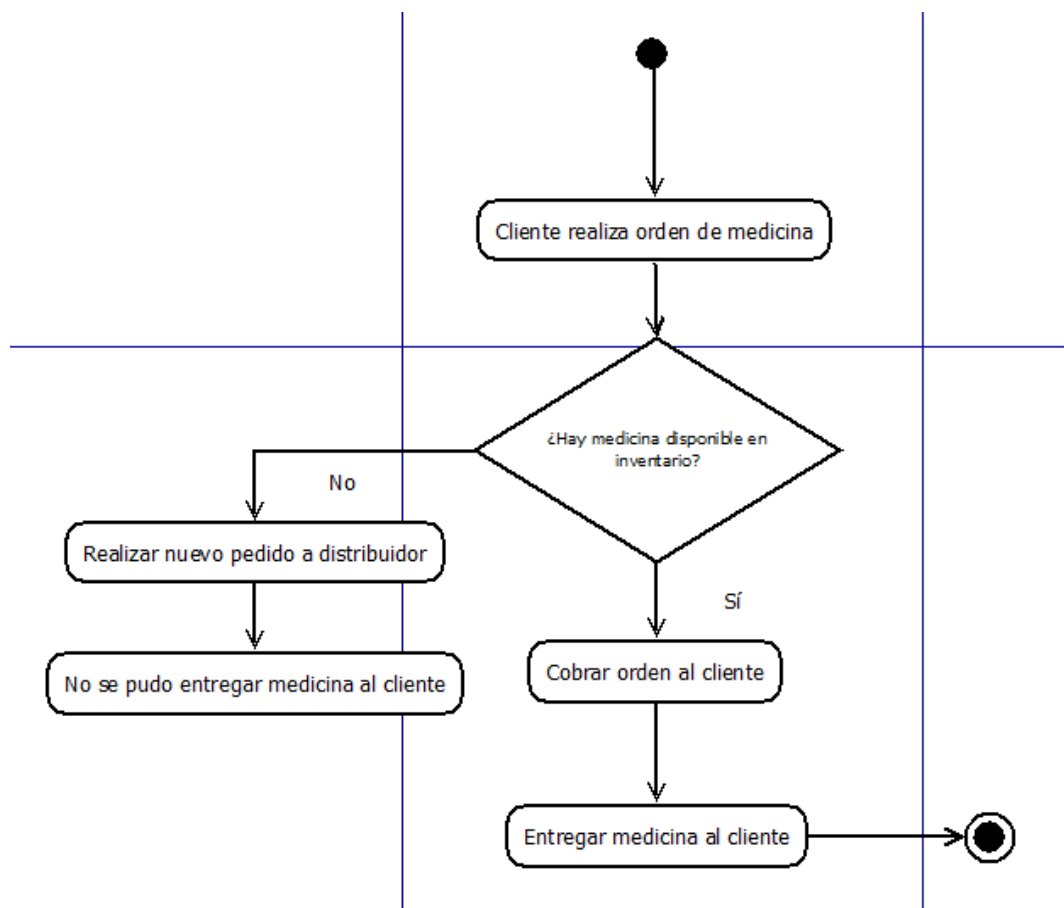


Diagrama de actividades:



### Diagrama de componentes:

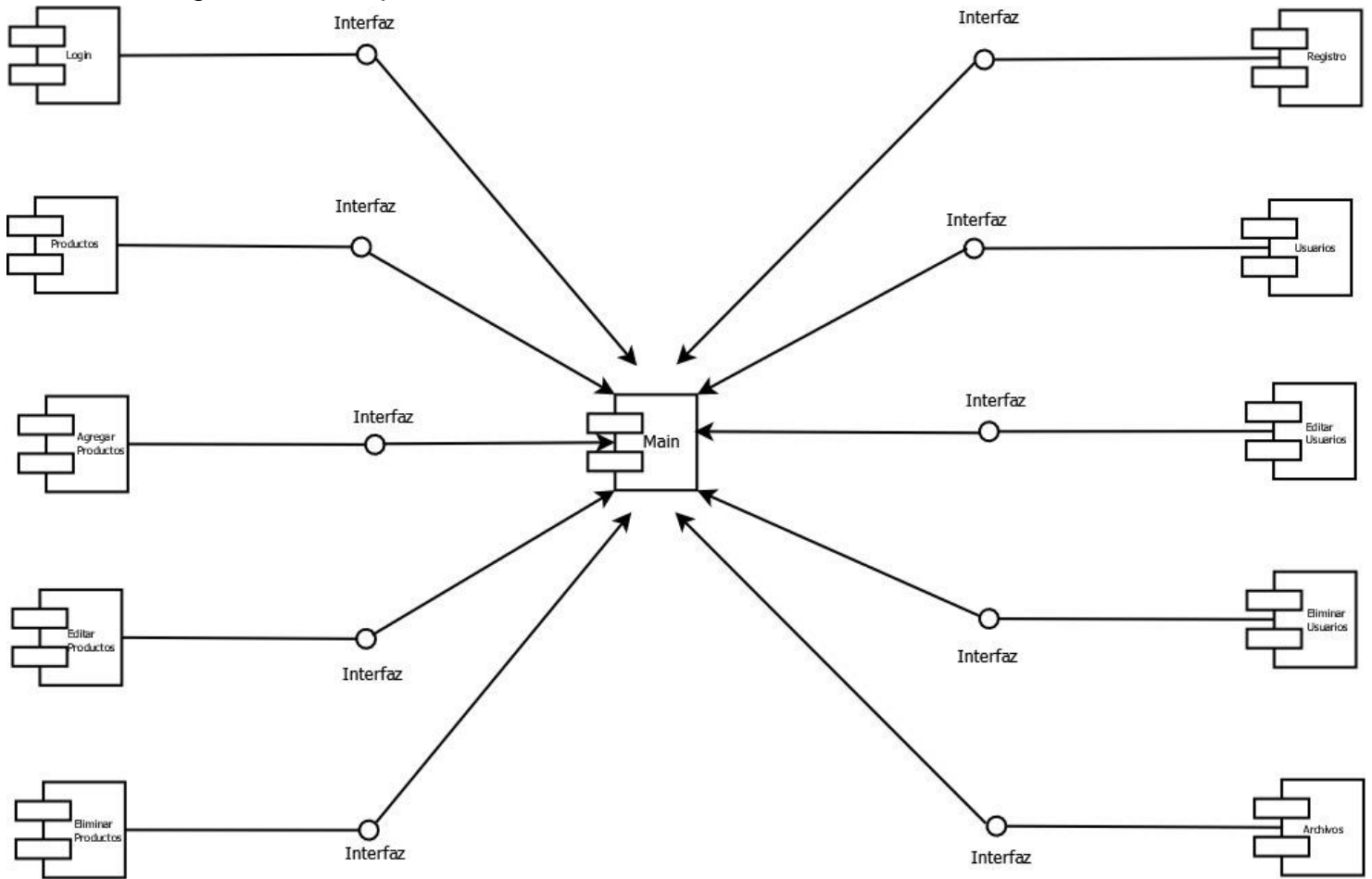
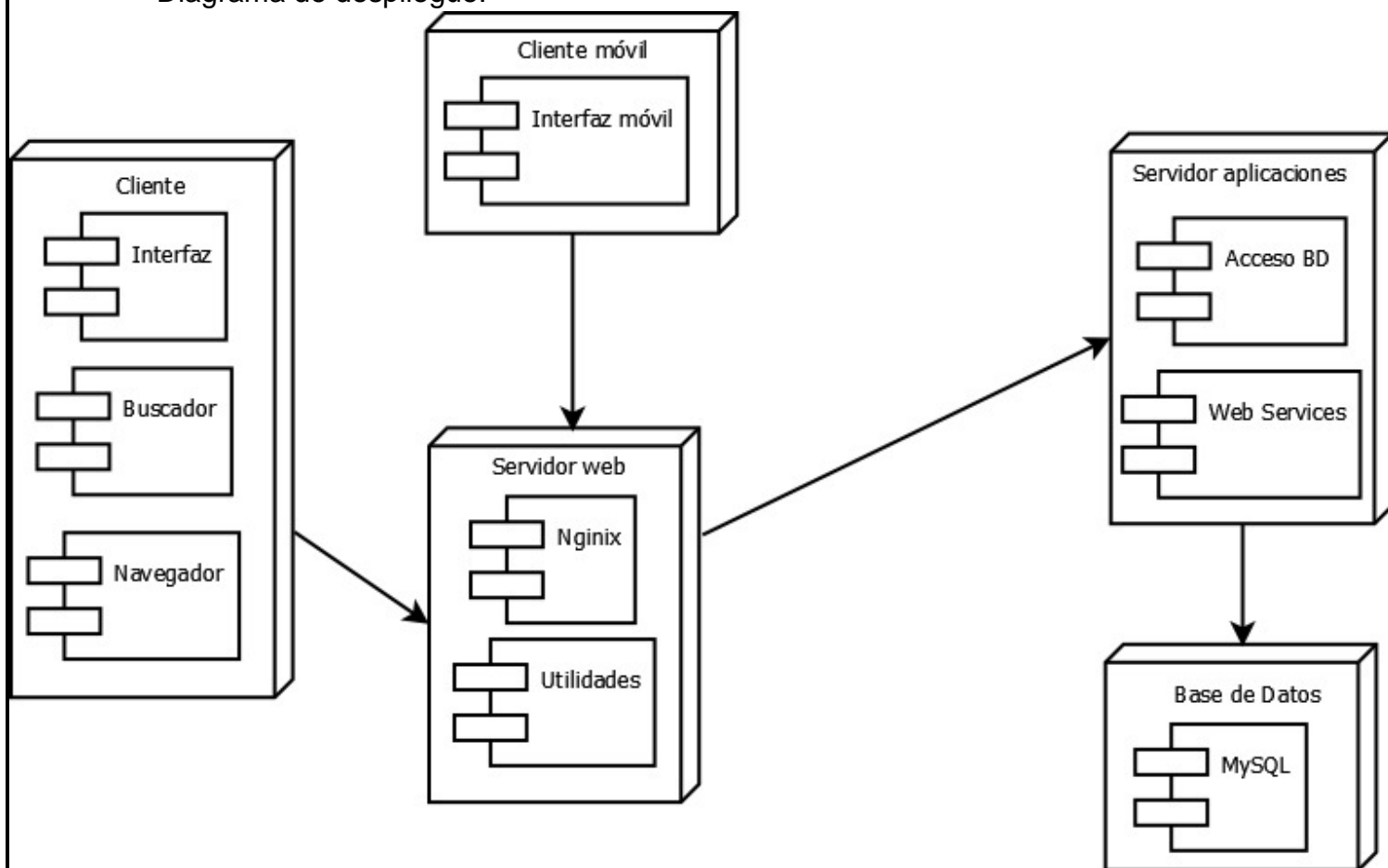


Diagrama de despliegue:



## Conclusiones

*Catherine Encinas:* Mi conclusión acerca de esta actividad es que los diagramas que realizamos nos ayudan a poder comprender mas el como esta conformado nuestro proyecto realizado para la empresa MMedi Health JD. Así podemos tener mas organizado como vamos elaborando dicha plataforma de e-commerce.

Los diagramas UML nos ayudan a poder representar de manera gráfica y clara, algún proyecto de software que este realizando el equipo desarrollador, en el caso de mi compañero y a mí, a comprender mejor el desarrollo de nuestra plataforma e-commerce, para así poder optimizar mas los tiempos, y que este no nos juegue en contra.

Así, sin en algún momento otro equipo desarrollador toma nuestro papel en el software trabajado, pueden entender mejor el como funciona y no tener que preguntar y que en eso el tiempo se les acabe.

En esta actividad como en las anteriores de los diagramas, siento que no se me hicieron tan complicadas, pues mi compañero y yo ya habíamos visto los diagramas UML en la unidad de aprendizaje de Ingeniería de Software I, por lo que se nos fue sencillo realizar dicha práctica.

*Alan González:* Los diagramas de UML utilizados nos ayudaron a mi compañera y a mi visualizar con mayor claridad el algoritmo de compras que se utilizará en la página web y en la farmacéutica. Se nos hicieron útiles para optimizar los procesos de desarrollo y las condiciones para poder producir una mejor transición entre las acciones del sistema.

Así cualquier persona incluso miembros de otros equipos de la clase pueden revisar nuestros diagramas UML y entender con claridad su función, los actores participantes y el orden en el que se deben realizar las acciones.

Esta actividad me pareció muy interesante, además de que a mi compañera y a mi no se nos dificulto tanto, ya que estos diagramas ya los habíamos visto en la clase de Ingeniería de Software I.