.

**Proyecto**

**Comedor-UACM**

**SDD:**

**PARTICIPANTES:**

Daniel Alejandro Valdivia Romero 16-011-0560

Balamm Salvador Lucio Lopez 13-011-0277

Johan Arturo Pérez Hernández 15-011-0412

**Índice**

**1. Introducción**

**2. Objetivos**

**2.1 Patrones Elegidos**

**3. Diagramas**

* **Diagrama Funcional**
* **Diagrama de Casos de Uso**
* **Diagrama de clases.**
* **Diagrama de dependencias.**
* **Figura. Interfaz.**

**Introducción**

El presente proyecto propone desarrollar una aplicación para el sistema operativo ANDROID al servicio del comedor de la Universidad Autónoma de la Ciudad de México, con la finalidad de informar los platillos ofertados del día a la comunidad universitaria, también se mostrara los horario y los menús actualizados de forma diaria, así mismo también se mostrara información de los encargados en turno para cualquier incidente que se pudiera presentar, se optó por no enfocarnos solo en un plantel sino en todos los planteles para que sea una app en la que todos los estudiantes tengan al alcance esta herramienta.

**Objetivos:**

* Convertir a **UACM-Comedor** en un sistema de calidad que sea altamente competitivo, que sea útil a nivel institucional.
* Agilizar tiempos de los comedore.
* Desarrollar el conocimiento de la plataforma.
* Incrementar el número de reseñas de usuarios para conocer su opinión y brindar seguimientos personalizados.
* Desarrollar una plataforma intuitiva que sea funcional para cualquier usuario que tenga la necesidad de utilizarlo sin importar qué tan familiarizado se encuentre con las aplicaciones actuales.
* Potenciar la imagen de la plataforma, creando vínculos de cercanía entre usuarios y comunidad universitaria

**Patrones Elegidos**

1. **MVVM (Modelo Vista Vista-Modelo)**

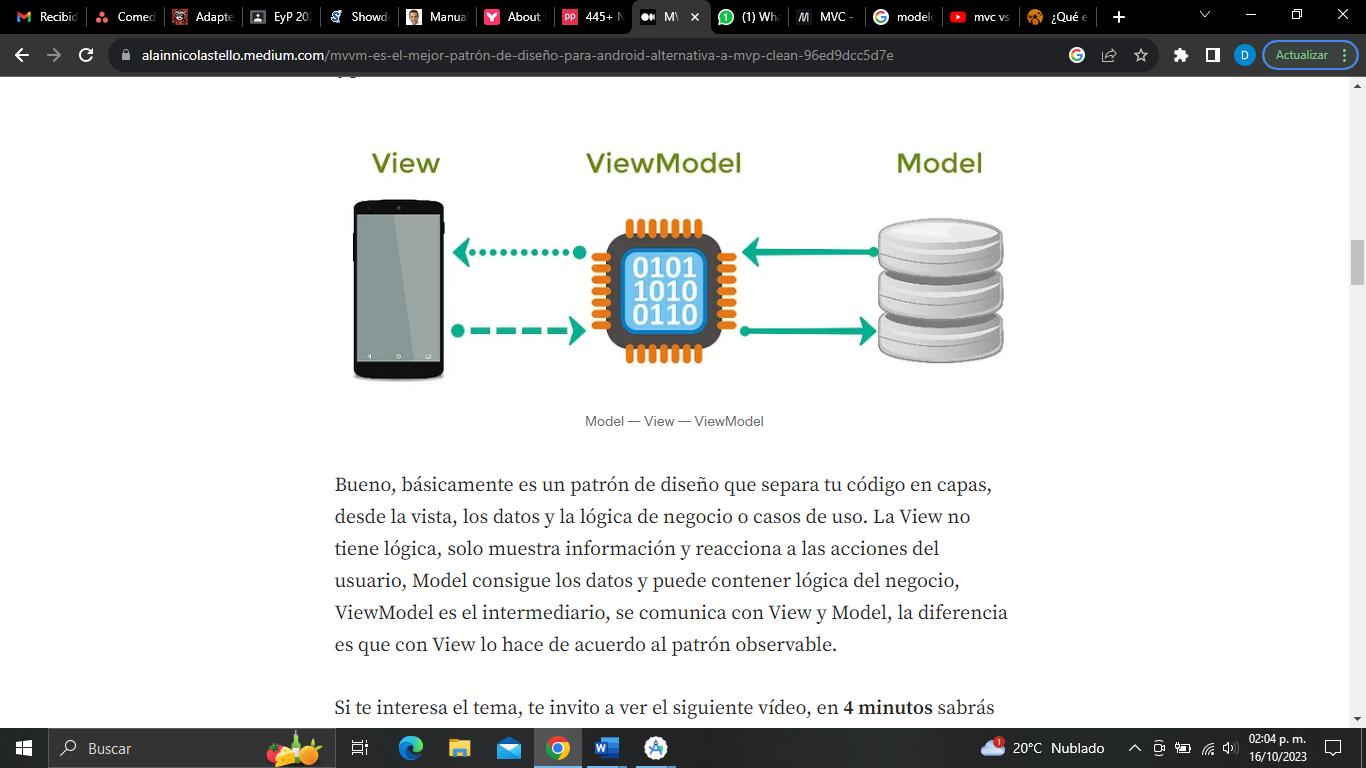
Es un patrón de diseño de software comúnmente utilizado para implementar interfaces de usuario, datos y lógica de control.

Separa el código en capas:

• View: solo muestra la infromacion y reacciona a las acciones del usuario

• Model: consigue los datos y puede contener lógica de negocio

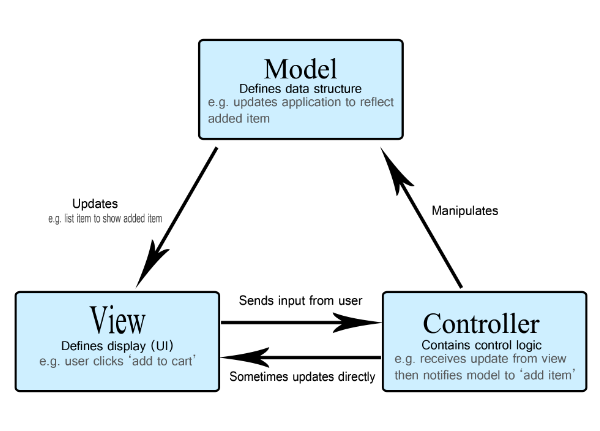
• View-Model: es el intermediario, se comunica con View y Model.



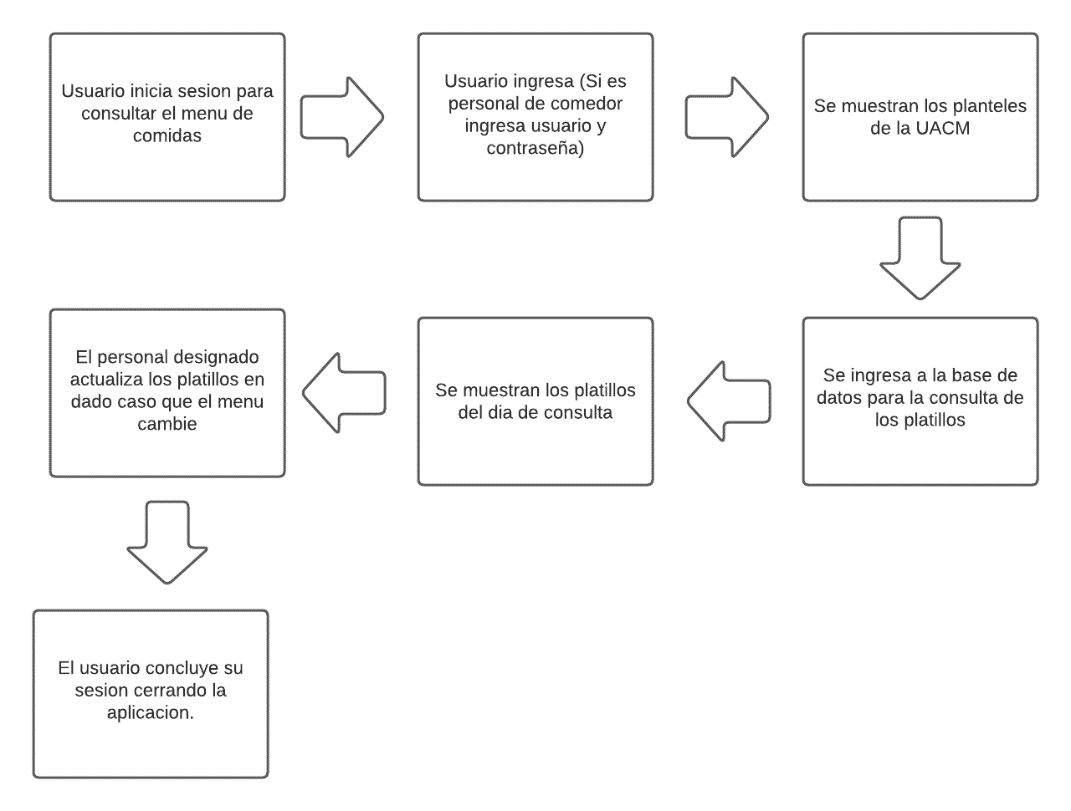
1. **MVC (Modelo Vista Controlador)**

Es un patrón en el diseño de software comúnmente utilizado para implementar interfaces de usuario, datos y lógica de control. Enfatiza una separación entre la lógica de negocios y su visualización.

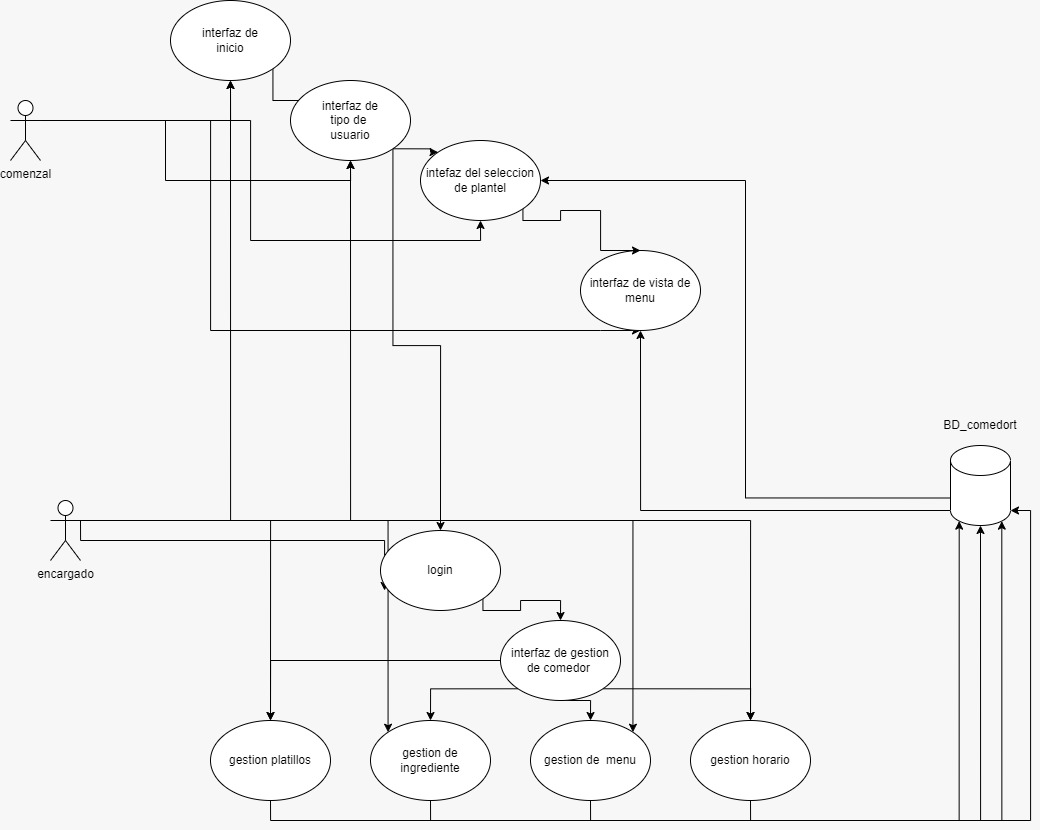
Es por ello que decidimos tomarlo como base para la elaboración del proyecto.

****

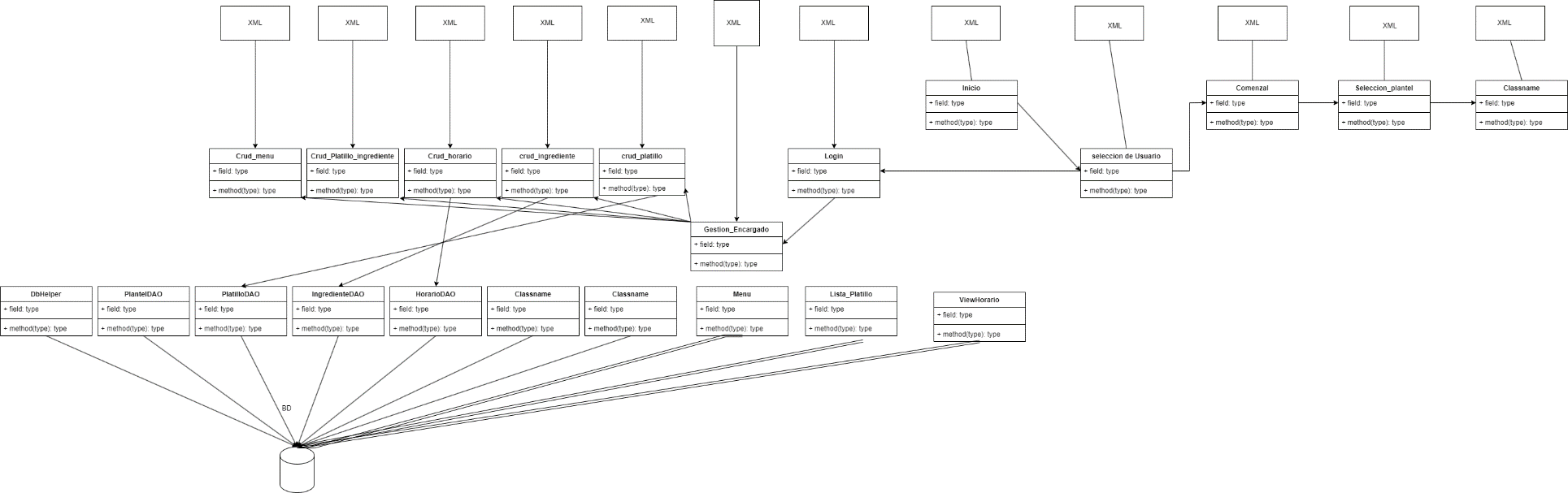
**Diagrama Funcional**

****

**Diagrama Casos de Uso**

****

**Diagrama de Clases**

****