

## **Sistema de gestión Gimnasio - Entregable #1**

Christopher Daniel Calvo Carvajal

Wong Gutiérrez Ángel Gabriel

Universidad Politécnica Internacional

90501-Técnicas de Programación

Profesor: Luis Felipe Mora Umaña

6 de noviembre del 2024.

## Índice

Decisión de diseño.....	4
División de proyecto .....	8
Epic 1: Gestión de Usuarios .....	8
Feature 1.1: Roles de los Usuarios y su autenticación.....	8
Product Backlog Item 1.1.1.....	8
Product Backlog Item 1.1.2.....	8
Product Backlog Item 1.1.3.....	9
Epic 2: Gestión de Membresías.....	9
Feature 2.1: Renovación y Notificaciones de Membresías.....	9
Product Backlog Item 2.1.1.....	10
Product Backlog Item 2.1.2.....	10
Epic 3: Gestión de Clases y Reservas .....	11
Feature 3.1: Programación de Clases y Reservas de Espacios .....	11
Product Backlog Item 3.1.1.....	11
Product Backlog Item 3.1.2.....	11
Epic 4: Gestión de Inventario .....	12
Feature 4.1: Control de Máquinas y Equipos .....	12
Product Backlog Item 4.1.1.....	12
Product Backlog Item 4.1.2.....	12

Epic 5: Reportes .....	13
Feature 5.1: Generación de Reportes para Entrenadores .....	13
Product Backlog Item 5.1.1.....	13
Product Backlog Item 5.1.2.....	13
Epic 6: Facturación .....	14
Feature 6.1: Gestión de Facturación .....	14
Product Backlog Item 6.1.1.....	14
Análisis Aprendizajes.....	15
Análisis Conclusiones.....	15

## Decisión de diseño

Para la decisión de diseño se realizó una reunión en la cual se concretó en pseudo código una idea aproximada de cómo se realizarían las distintas épicas.

### 1- Usuarios:

Clase UserController

Método iniciarSesion(usuarioID)

Llamar al método buscarUsuarioPorID(usuarioID)

Retornar Usuario si existe

FinMétodo

FinClase

### 2- Membresías:

Clase membershipController

Método verificarVencimiento(ClienteID)

Buscar membresía de ClienteID

If (fecha actual >= fechaVencimiento - 5 días)

Retornar mensaje de aviso

Else

FinMétodo

FinClase

### 3- Clases y Reservas:

Clase ClassController

Método reservarClase(ClienteID, ClaseID)

Verificar disponibilidad en Clase

If (disponible)

Reservar cupo para clienteID en Clase

Agregar reserva en el perfil del cliente y entrenador

Else

FinMétodo

FinClase

### 4- Inventario:

Clase inventoryController

Método verificarEstadoMaquinas()

For cada Maquina en Inventario

If (Vida útil restante  $\leq$  3 meses)

Agregar maquina a la lista de notificación

Else

Fin del For

FinMétodo

FinClase

5- Reportes:

Clase ReportController

Método generarReporteMatricula()

Generar reporte de crecimiento o decremento de matrícula

Retornar reporte

FinMétodo

Método generarReporteFinanciero()

Calcular ingresos y egresos por mes

Retornar reporte financiero

FinMétodo

Método generarReporteClases()

Analizar clases populares

Retornar reporte de clases y horarios más atractivos

FinMétodo

FinClase

6- Facturacion:

Clase invoiceController

Método generarFactura(clienteID)

Crear una nueva factura para ClienteID

Almacenar factura

FinMétodo

Método consultarFactura(clienteID, fecha)

Buscar factura según cliente y fecha

Retornar factura

FinMétodo

FinClase

## División de proyecto

### Epic 1: Gestión de Usuarios

#### Feature 1.1: Roles de los Usuarios y su autenticación

Permitir el acceso al sistema a los usuarios (clientes y entrenadores) con roles específicos definidos.

##### ***Product Backlog Item 1.1.1***

- **Descripción:** Configurar el sistema para que dos usuarios con rol de administrador puedan iniciar sesión.
- **Criterio de Aceptación:**
  - Los administradores deben poder iniciar sesión con permisos de administrador.
  - La configuración debe permitir que ambos administradores se registren con contraseñas independientes.
- **Tareas:**
  - Crear usuarios para los administradores con permisos completos.
  - Asignar una contraseña correspondiente a cada administrador.

##### ***Product Backlog Item 1.1.2***

- **Descripción:** Inicializar usuarios y asignar permisos y horarios.
- **Criterio de Aceptación:**



- El sistema contara con 2 administradores, 98 usuarios registrados y 6 entrenadores con sus respectivos permisos, horarios y puntos fuertes.
- **Tareas:**
  - Inicializar 2 administradores.
  - Inicializar 6 entrenadores y asignarles permisos de administración.
  - Asignar horarios y puntos fuertes a cada entrenador.
  - Dar permisos a los administradores y configurar contraseñas para cada uno, diferenciando los accesos de administrador, usuario y entrenador.

### ***Product Backlog Item 1.1.3***

- **Descripción:** Inicializar las clases y horarios.
- **Criterio de Aceptación:**
  - Las clases y horarios deben estar inicializados y visibles para todos los usuarios.

## **Epic 2: Gestión de Membresías**

### **Feature 2.1: Renovación y Notificaciones de Membresías**

Gestionar las membresías de los clientes y enviar notificaciones de vencimiento al llegar la fecha límite.

***Product Backlog Item 2.1.1***

- **Descripción:** Crear un sistema de calendario en el que se pueda gestionar la renovación de membresías.
- **Criterio de Aceptación:**
  - El calendario debe permitir establecer fechas de pago y enviar notificaciones al usuario 5 días antes de la fecha de vencimiento.
- **Tareas:**
  - Crear calendario y agregar la fecha de entrada de 3 clientes.
  - Establecer fecha de pago de membresía.
  - Mostrar notificación emergente al cliente cuando falten 5 días para el pago.

***Product Backlog Item 2.1.2***

- **Descripción:** Configurar pantalla de pago de membresía.
- **Criterio de Aceptación:**
  - La pantalla de pago debe permitir agregar y actualizar la membresía, así como realizar el pago anticipado.
- **Tareas:**
  - Crear una pantalla de pago con un calendario para fechas de pago fijas (29 de cada mes).
  - Implementar funciones para bloquear acceso al usuario si no se ha pagado a tiempo y dar opción de pagar con tarjeta.

- Permitir que el usuario seleccione clases y entrenadores tras actualizar la membresía.

### **Epic 3: Gestión de Clases y Reservas**

#### **Feature 3.1: Programación de Clases y Reservas de Espacios**

Gestionar la programación de clases y reservas de espacios.

##### ***Product Backlog Item 3.1.1***

- **Descripción:** Crear una tabla de reservas visible para los usuarios.
- **Criterio de Aceptación:**
  - La tabla de reservas debe ser visible al iniciar sesión y mostrar las reservas actuales.
- **Tareas:**
  - Mostrar una tabla que permita a los usuarios reservar y llevar un registro visible al iniciar sesión.

##### ***Product Backlog Item 3.1.2***

- **Descripción:** Crear lista de clases con horarios y categorías.
- **Criterio de Aceptación:**
  - Los usuarios deben poder seleccionar clases de Zumba, Funcional y Cardio Dance en horarios definidos.
- **Tareas:**

- Crear lista de clases y horarios disponibles para las categorías Zumba, Funcional y Cardio Dance en los horarios de mañana, tarde y noche.

## **Epic 4: Gestión de Inventario**

### **Feature 4.1: Control de Máquinas y Equipos**

Gestionar el inventario de máquinas y notificar el próximo mantenimiento.

#### ***Product Backlog Item 4.1.1***

- **Descripción:** Crear registro de inventario de máquinas con tiempo de vida y caducidad.
- **Criterio de Aceptación:**
  - El registro de inventario debe incluir fecha de entrada, tiempo de vida y fecha de caducidad.
- **Tareas:**
  - Crear una tabla de registro de inventario con un calendario para gestionar la caducidad.

#### ***Product Backlog Item 4.1.2***

- **Descripción:** Crear alertas de caducidad de equipos.
- **Criterio de Aceptación:**
  - El sistema debe notificar cuando falten 3 meses para la caducidad de un equipo.

- **Tareas:**
  - Implementar alertas que indiquen 3 meses antes de la caducidad de cada equipo.

## **Epic 5: Reportes**

### **Feature 5.1: Generación de Reportes para Entrenadores**

Crear reportes sobre el crecimiento de matrículas, informes contables y popularidad de clases.

#### ***Product Backlog Item 5.1.1***

- **Descripción:** Crear reportes de inscripción de clases.
- **Criterio de Aceptación:**
  - La lista de clientes inscritos en las clases debe ser visible en una tabla con la opción de mostrar ingresos mensuales.
- **Tareas:**
  - Crear una tabla con la lista de clientes inscritos en diferentes clases.
  - Crear un informe de ingresos mensual basado en el costo de membresía.

#### ***Product Backlog Item 5.1.2***

- **Descripción:** Generar reportes financieros y de balance general.
- **Criterio de Aceptación:**

- El sistema deberá mostrar el total de membresías, compras de máquinas y balance general debajo de la tabla de clientes.
- **Tareas:**
  - Configurar costos de compra de máquinas y calcular el balance general.
  - Imprimir el reporte financiero con total de membresías y costos.

## **Epic 6: Facturación**

### **Feature 6.1: Gestión de Facturación**

Gestionar las facturas de membresías, clases y otros servicios.

#### ***Product Backlog Item 6.1.1***

- **Descripción:** Almacenar y mostrar información de pago en el perfil del cliente.
- **Criterio de Aceptación:**
  - La información de pago debe ser visible en el perfil del cliente y accesible desde una pantalla de factura.
- **Tareas:**
  - Crear una tabla que almacene información de pago mensual de membresías y reservas.
  - Generar una pantalla de factura que muestre la información almacenada y permita verla desde el perfil del cliente.

### **Análisis Aprendizajes**

Con el proyecto que desarrollamos, aprendimos a como organizar correctamente un proyecto en Git y Trello. Además de que la constante comunicación y el uso de herramientas antes mencionadas nos ayudo a agilizar y distribuir la carga de trabajo a la hora de trabajar en equipo.

### **Análisis Conclusiones**

El desarrollo del código para programar un sistema de gestión de gimnasio presento un desafío significativo y un importante proceso de aprendizaje, debido a la necesidad de afrontar una curva de aprendizaje notable. Este acercamiento a situaciones similares a las que se enfrentan en una empresa y al desarrollar un programa para un cliente mejora la preparación para responder de manera efectiva a las problemáticas en la informática.