

MEMORIA

Gestor de campamentos



ÍNDICE

1.INTRODUCCIÓN	3
1.1 MIEMBROS EQUIPO	
1.2 DOCUMENTACIÓN	
1.3 FACTOR DE PUNTUACIÓN	
2. ARQUITECTURA	4
2.1 DIAGRAMAS DE CLASES	
2.2 DIAGRAMAS DE SECUENCIA	
3. DESARROLLO DEL PROGRAMA SOFTWARE	7
3.1 DISEÑO	
3.2 PATRONES DEL MODELO	
4. PRUEBAS Y RESULTADO FINAL	8

1. INTRODUCCIÓN

1.1. MIEMBROS:

El proyecto Gestor de Campamentos está formado por 6 integrantes:

- Cucicea, Mariana
- García García, Lucía
- Guadilla Gómez, Paula
- Martín Peraza, Airam
- Pérez Salinas, Steven Alexis
- Postigo Ruiz, Sofía

Habiendo trabajado todos en el proyecto de manera equitativa y colaborativa. Demostrando en igual medida un alto nivel de compromiso y dedicación, lo que ha resultado en un trabajo en equipo sólido y eficiente.

1.2. DOCUMENTACIÓN:

La documentación de este proyecto se puede encontrar en los siguientes repositorios de **GIThub**:

→ Documentación:

https://github.com/Tuuturuturuu/DOC_GestorDeCampamentos

→ Modelo:

https://github.com/Tuuturuturuu/MODEL_GestorDeCampamentos

→ Código:

https://github.com/Tuuturuturuu/COD_GestorDeCampamentos

1.3. FACTOR DE PUNTUACIÓN:

El siguiente apartado forma parte de la calificación del proyecto, en el cual pondremos una puntuación a cada uno de los miembros del equipo. Por las razones explicadas en el punto 1 de este documento, hemos decidido que la nota sea igual para cada miembro del grupo.

NOMBRES:	<div>MÁXIMA</div> <div>NOTA</div>						TOTAL:
Mariana							1
Lucía							1
Paula							1
Airam							1
Steven							1
Sofía							1

2. ARQUITECTURA

El proyecto software está basado en una arquitectura multicapa. Cada una de las capas está estructurada por paquetes, siendo estos Presentación, Negocio e Integración, con sus respectivos, patrones, diagramas de secuencia y diagramas de clase de cada una de las entidades pertenecientes al proyecto (Turno, Personal, Actividades, Material, Factura).

2.1 DIAGRAMAS DE CLASES Y DE SECUENCIA:

Los diagramas de clases y de secuencia, los podemos encontrar en el repositorio del Modelo, en la sección de Modelo de Diseño. La distribución de cada uno de los diagramas es la siguiente:

PAQUETE DE INTEGRACIÓN:

- **Actividad:**
 - IActividadDC → **Diagrama de clase**
 - IActividadMaterialDC → **Diagrama de clase**
 - DAOActivadImp
 - DAOActivadMaterialImp
 - DAOActivad
 - DAOActivadMaterial
 - DAOActivad

Diagrama de secuencia (Alta Actividad)

- **Factoría Integración:**
 - IFactorialIntDC → **Diagrama de clase**
 - FactorialIntegración
 - FactoríaIntegraciónImp

- **Factura:**
 - IFactoriaDC → **Diagrama de clase**
 - IGeneraDC → **Diagrama de clase**
 - DAOFacturalImp
 - DAOGeneralImp
 - DAOFactura
 - DAOGenera

- **Material:**
 - IMateriaDC → **Diagrama de clase**
 - DAOMaterialImp
 - DAOMaterial
 - DAOMaterial

Diagrama de secuencia (TODAS LAS FUNCIONES)

- **Personal:**
 - IPersonalDC b → **Diagrama de clase**
 - DAOPersonalImp
 - DAOPersonal
 - DAOPersonal

Diagrama de secuencia (TODAS LAS FUNCIONES)

- **Turno:**
 - ITurnoDC → **Diagrama de clase**
 - DAOTurnoImp
 - DAOTurno
 - DAOTurno → **Diagrama de secuencia**

PAQUETE DE NEGOCIO:

- **Actividad:**
 - NActividadDC → **Diagrama de clases**
 - SA Actividad Imp
 - TActividad
 - TActividad Material
 - SAActividad
 - SAActividad → **Diagramas de secuencia**

Diagrama de secuencia (alta y baja)

- **Factoría Negocio:**
 - FactoríaSADC → **Diagrama de clases**
 - FactoriaSA
 - Factoría SImpl

- **Factura:**
 - FacturaDC → **Diagrama de clases**
 - SAFacturalImp
 - TFactura
 - TFacturaConActividades
 - TGenera
 - SAFactura
 - SAFactura → **Diagramas de secuencia**

Diagrama de secuencia (devolver y cerrar)

- **Material:**
 - MaterialDC → **Diagrama de clases**
 - SAMaterialImp
 - TMaterial
 - SAMaterial
 - SAMaterial → **Diagramas de secuencia**

Diagrama de secuencia (Vincular y listarMaterialPorActividades)

- **Personal:**
 - Personal DC → **Diagramas de clases**
 - SAPersonalImp
 - TPersonal
 - TPersonalCocinero

- TPersonalMonitor
- DAOPersonal
- SAPersonal
- SAPersonal → **Diagramas de secuencia**

Diagrama de secuencia (alta y baja)

- Turno:

- NTurnoDC → **Diagramas de clases**
- SATurnoImp
- TTurno
- SATurno
- SATurno → **Diagramas de secuencia**

Diagrama de secuencia (TODAS LAS FUNCIONES)

PAQUETE DE PRESENTACIÓN: Está dividido en carpetas para cada función. En cada paquete se encuentra el diagrama de clases de cada función junto con el diagrama de secuencia (salvo controlador y factoría presentación las cuales no tienen funciones como las entidades).

- Actividad:

- VAltaActividad
- VBajaActividad
- VModificarActividad
- VMostrarActividad
- VMostrarActividadPorPersonal
- VMostrarTodasActividades
- VVincularActividadConMaterial

- Controlador:

- ControladorDC -> **diagrama de clase**
- Controlador
- ControladorImp
- ControladorImp -> **diagrama de secuencia**

- Factoría Presentación:

- FactoriaVistaDC -> **diagrama de clase**
- FactoriaVistas
- FactoriaVistasImp

- Factura:

- VAbrirFactura
- VCerrarFactura
- VDevolverUnaVenta
- VMostrarFactura
- VMostrarFacturaPorActividad
- VMostrarTodasFacturas

- **Turno:**
 - VAltaTurno
 - VBajaTurno
 - VModificarTurno
 - VMostrarTurno
 - VMostrarTurnos
 - VTurnoDC -> **diagrama de clase general de turno**
 - ControladorTurno -> **diagramas de secuencia de las funciones**

- **Personal:**
 - VAltaPersonal
 - VBajaPersonal
 - VModificarPersonal
 - VMostrarPersonalPorTurno
 - VMostrarTodos
 - VMostrarUno
 - VAltaPersonalCocinero
 - VAltaPersonalPersonal
 - VpersonalDC -> **diagrama de clases general de personal**
 - VPersonal
 - IGUI
 - ControladorPersonal -> **diagramas de secuencia de todas las funciones**

- **Material:**
 - VAltaMterial
 - VBajaMaterial
 - VmostrarActividadesMterial
 - VModificarMterial
 - VMostrarMaterial
 - VMostrarMateriales
 - VVincularMaterialActividad

3. DESARROLLO DEL PROGRAMA SOFTWARE

El código lo podemos encontrar en el repositorio del código, en la sección de SRC.

3.1 DISEÑO

El código está organizado en paquetes, Presentación, Integración y Código, y dentro de estos, un paquete por cada entidad.

3.2 PATRONES DEL MODELO

Los patrones usados en el código son los siguientes:

- Modelo Vista Controlador
- Facade
- Singleton

- Iterator
- Controlador
- Abstract Factory
- Patrón Transferencia (transfer)
- Patrón DAO
- Patrón Servicio Aplicación (SA)

4. PRUEBAS Y RESULTADO FINAL

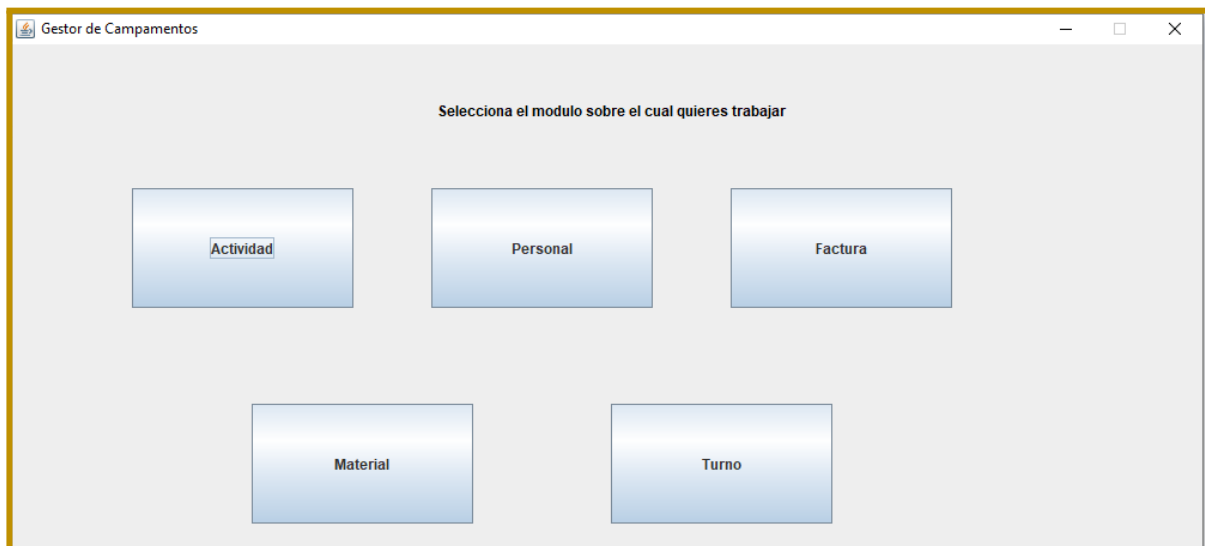
4.1. PRUEBAS:

Las pruebas de la aplicación/JUNIT los podemos encontrar en el siguiente enlace:

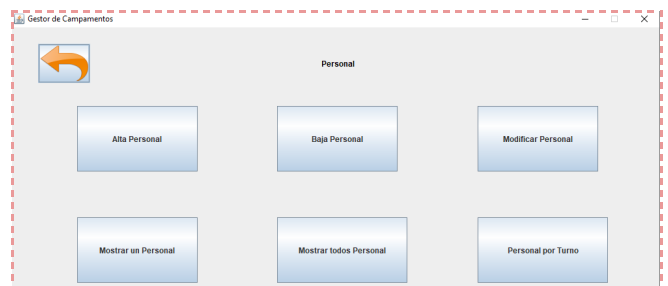
https://github.com/Tuuturuturuu/COD_GestorDeCampamentos/tree/main/CODCampamentos/tests

4.2. RESULTADO FINAL:

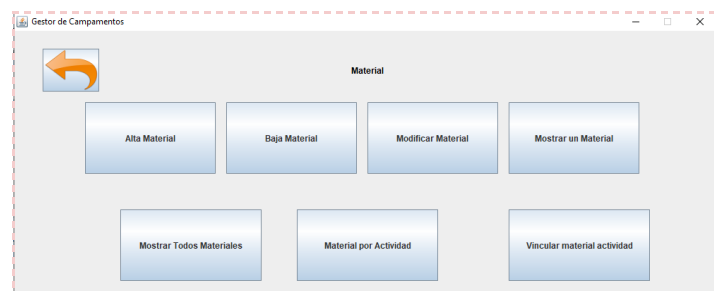
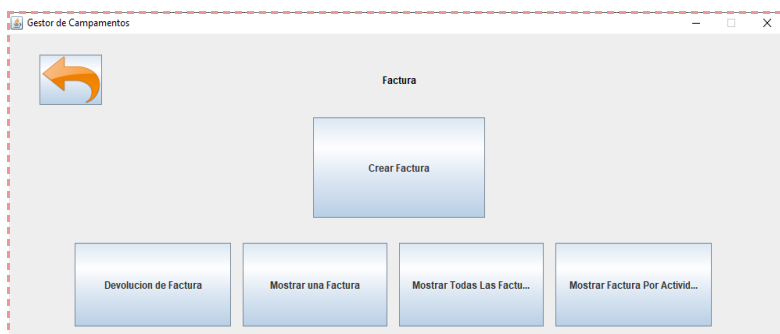
En este apartado se proporcionan una serie de imágenes del proyecto para mostrar el resultado final de la aplicación de forma visual.



VISTA ACTIVIDAD



VISTA PERSONAL



VISTA FACTURA

VISTA TURNO

VISTA MATERIAL

