

UNIVERSIDAD DEL VALLE DE GUATEMALA
CAMPUS CENTRAL
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
PROGRAMACIÓN ORIENTADA A OBJETOS



ENTREGA 3:

Priorización de requisitos, persistencia de datos, planificación y desarrollo.

POR:

Julio Orellana: 24207

Ginebra Estrada: 24079

Sofia Batz: 241281

Ricardo Rodríguez: 18907

28 de septiembre de 2024, Guatemala, Guatemala.



Requisitos Funcionales Priorizados.

Como equipo decidimos dar prioridad a algunos de los requisitos nuevos que surgieron para esta entrega y mantener como prioridad 1 a los requisitos planteados anteriormente, sin embargo, para esta entrega como equipo llegamos a la conclusión de priorizar los siguientes requisitos:

- Diseñar, crear y conectar la base de datos MySQL del sistema.
- Crear las tablas necesarias en la base de datos y agregar datos genéricos para la prueba de las mismas.
- Implementar una clase que sirva como gestor de base de datos para el proyecto en el cual se puedan realizar consultas directamente para contar con persistencia de datos.
- Crear una clase encargada de gestionar Seguridad en los datos utilizada para encriptar y desencriptar sincrónicamente información.
- Estructurar el proyecto con una arquitectura de software adecuada y escoger los patrones de diseño para el código.
- Crear clase de autenticación.
- Crear clase de Productos ya que será el principal objeto a utilizar en el Sistema.
- Crear clases de gestión de productos para seguir el patrón de diseño seleccionado e implementar métodos generales para otros objetos.



Tecnología para persistencia de datos.

Como equipo estuvimos reuniéndonos en varias ocasiones para decidir que tecnología y/o método utilizar para la persistencia de los datos en nuestro sistema, después de todas estas reuniones tal y como mencionamos en la fase 2, nos decidimos utilizar una base de datos relacional (SQL) ya que nos ofrecen algunas ventajas notables ante las demás tecnologías que existen para el mismo motivo, algunas de estas ventajas son:

- Seguridad de los datos: Las bases de datos SQL nos ofrecen el modelo ACID (Atomicidad, Consistencia, Aislamiento, Durabilidad), el cual nos garantiza una integridad de los datos sensibles como puede ser el stock de la empresa y las credenciales de nuestros usuarios, tanto como administradores, clientes y empleados.
- Relación entre tablas: Las bases de datos SQL nos permiten tener relación directa entre una tabla y otra mediante claves foráneas lo cual es de gran utilidad para nuestro sistema para distintos tipos de datos como productos, proveedores, clientes, ventas, entre otros.
- Consultas complejas: SQL nos permite realizar consultas complejas a la base de datos como puede ser unir varias tablas para generar informes detallados. Esto es muy ventajoso para nuestro sistema de inventarios y la gestión correcta de los datos.
- Soporte: Al tratarse de bases de datos encontramos más soporte para Java de bases de datos SQL por lo cual se nos facilita la implementación de las mismas al sistema.

Después de que como equipo decidimos implementar una base de datos SQL nos quedaba decidir una de las partes más importantes, que sistema de gestión de base de datos utilizar, tratándose de bases de datos SQL teníamos 2 potenciales opciones: PostgreSQL y MySQL. Esto nos llevó a describir qué ventajas y desventajas nos ofrece cada uno de estos gestores de bases de datos para nuestro proyecto, las cuales se describen resumidamente en la siguiente tabla.

Característica	Ventaja - MySQL	Desventaja - MySQL	Ventaja - PostgreSQL	Desventaja - PostgreSQL
Facilidad en la implementación	Fácil de usar y configurar especialmente para proyectos simples como el nuestro.	No es tan adecuada para aplicaciones más complejas	Mayor personalización y flexibilidad al configurarla.	Más difícil de entender y utilizar.
SQL	Menos estricto con el estándar SQL lo cual	Puede llevar a inconsistencia en los datos y	Cumple estrictamente el estándar SQL lo	Menos tolerante con malas prácticas de

UNIVERSIDAD DEL VALLE DE GUATEMALA
 CAMPUS CENTRAL
 FACULTAD DE INGENIERÍA
 DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
 PROGRAMACIÓN ORIENTADA A OBJETOS



	lleva a mayor flexibilidad y facilidad de uso.	generar problemas.	cual nos otorga una mayor integridad en los datos.	diseño de bases de datos.
ACID	Es más simple en términos funcionales, lo cual es eficiente para un sistema sencillo.	Algunos de sus motores no soportan transacciones.	Soporte completo para cualquier tipo de operación	Para aplicaciones pequeñas puede generar un tráfico (carga) muy alto.
Orientación a Objetos	Facilidad para guardar los atributos de los objetos si la tabla tiene una buena estructura.	No tiene Orientación a objetos directamente ya que sigue un modelo más estrictamente relacional.	Tiene un modelo orientado a objetos lo cual facilita la manipulación y almacenamiento de los datos de cada uno de los objetos en el sistema.	Para tener todas las ventajas de su orientación a objetos se debe tener conocimiento un poco avanzado para realizar las consultas adecuadas.

Esta tabla nos ayudó a identificar cual de las 2 se adecuaba mejor al proyecto, aunque la PostgreSQL sonaba mejor por su orientación a objetos nos decidimos en utilizar una **MySQL** por su facilidad en la implementación del sistema ya que aunque se está trabajando con objetos no necesariamente necesitamos realizar consultas tan avanzadas para llegar al objetivo.

Como servicio de alojamiento para nuestra base de datos escogimos a **Railway** ya que nos ofrece una versión gratuita la cual no permite elaborar el sistema sin ningún inconveniente y poder acceder a nuestros datos desde cualquier lugar solamente con las credenciales proporcionadas por el servicio. Su facilidad en la implementación fue otro factor muy importante en la toma de esta decisión

UNIVERSIDAD DEL VALLE DE GUATEMALA
CAMPUS CENTRAL
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
PROGRAMACIÓN ORIENTADA A OBJETOS



Planificación y gestión de las tareas a realizar.

https://docs.google.com/spreadsheets/d/1rh1xs0_8bRVKSGprF--jVA77RyvPHfshfYC7-eyt3U/edit?usp=sharing

Repositorio de github del proyecto.

https://github.com/Julio-orellana/UVG_JAVA_PROJECT



Historial de Commits:

```
commit 44db3de778ae50a16a4e9be1ef4e35c98e72836c
Author: Julio <julioes134@outlook.es>
Date:   Wed Sep 25 15:08:29 2024 -0600

    Creacion de paquete utils que incluye la interfaz ILoadEnv para cargar va
bles de entorno

commit 4fb6935c9cd33d70cef7c4532d72f68b6adc25cd
Author: Julio <julioes134@outlook.es>
Date:   Wed Sep 25 15:06:31 2024 -0600

    Paquete Security y clase Encryption creados para gestionar seguridad de
datos

commit fc520e4bba935eb1aa1ca900f39c58bec48afeea
Author: Julio <julioes134@outlook.es>
Date:   Wed Sep 25 15:05:37 2024 -0600

    Paquete Exception creado para gestionar excepciones personalizadas

commit aa2c17baec0633464a26753cb1f13e1e6473798c
Author: Julio <julioes134@outlook.es>
Date:   Wed Sep 25 15:04:30 2024 -0600

    NUEVAS IMPLEMENTACIONES AL GESTOR DE BASE DE DATOS

commit 26d38917441f5053702204f922aafbe09764fffd
Author: ricrod01 <rod18907@uvvg.edu.gt>
Date:   Wed Sep 25 12:27:20 2024 -0600

    Parte Ricardo Rodríguez commit2, arreglé clase Gestión y Producto

commit e446c743e9222efaa24dca2e944ace3a5c22795e
Author: Julio <julioes134@outlook.es>
Date:   Tue Sep 24 21:55:12 2024 -0600

    Refactorizacion del codigo

commit ee381d547edda70cd7fd859a072c28d36d7d2599
Merge: 5aa4cdc ca40f4b
Author: Julio-orellana <84369228+Julio-orellana@users.noreply.github.com>
Date:   Tue Sep 24 19:11:37 2024 -0600

    Merge pull request #4 from Julio-orellana/development

    CONEXION A LA BASE DE DATOS

commit ca40f4b0536e1d22023e02f232b843784d7f280e
Author: Julio <julioes134@outlook.es>
Date:   Tue Sep 24 19:08:49 2024 -0600

    CONEXION A LA BASE DE DATOS

commit 5aa4cdcdf57ab13e359c8ad222909b659e0b51b5
Author: ricrod01 <rod18907@uvvg.edu.gt>
Date:   Sun Sep 22 19:29:11 2024 -0600

    Parte Ricardo Rodríguez, commit 1
```

UNIVERSIDAD DEL VALLE DE GUATEMALA
CAMPUS CENTRAL
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
PROGRAMACIÓN ORIENTADA A OBJETOS



```
commit 884030e80f33153164da2411fb7d75985c6ba139 (HEAD -> main, origin/main, origin/HEAD)
Author: ricrod01 <rod18907@uvg.edu.gt>
Date: Sat Sep 28 21:35:44 2024 -0600

    Parte Ricardo Rodríguez commit4, agregué algunos métodos en la clase de productos

commit 460781450496c7dd8528850eebc1879fa4f0901b
Author: ricrod01 <rod18907@uvg.edu.gt>
Date: Sat Sep 28 09:31:31 2024 -0600

    Parte Ricardo Rodríguez commit3, Cambios en las clases de Gestion y Productos

commit 7bea61d36600d660a4a707a708ef91c334cdd7ee
Merge: 26d3891 296f9da
Author: ricrod01 <rod18907@uvg.edu.gt>
Date: Sat Sep 28 09:19:03 2024 -0600

    Merge branch 'main' of https://github.com/Julio-orellana/UVG_JAVA_PROJECT

commit 296f9da86ea7a294d4546603523d434bd28ec8dc
...skipping...
commit 884030e80f33153164da2411fb7d75985c6ba139 (HEAD -> main, origin/main, origin/HEAD)
Author: ricrod01 <rod18907@uvg.edu.gt>
Date: Sat Sep 28 21:35:44 2024 -0600

    Parte Ricardo Rodríguez commit4, agregué algunos métodos en la clase de productos

commit 460781450496c7dd8528850eebc1879fa4f0901b
Author: ricrod01 <rod18907@uvg.edu.gt>
Date: Sat Sep 28 09:31:31 2024 -0600

    Parte Ricardo Rodríguez commit3, Cambios en las clases de Gestion y Productos

commit 7bea61d36600d660a4a707a708ef91c334cdd7ee
Merge: 26d3891 296f9da
Author: ricrod01 <rod18907@uvg.edu.gt>
Date: Sat Sep 28 09:19:03 2024 -0600

    Merge branch 'main' of https://github.com/Julio-orellana/UVG_JAVA_PROJECT

commit 296f9da86ea7a294d4546603523d434bd28ec8dc
Author: Julio <julioes134@outlook.es>
Date: Wed Sep 25 15:10:45 2024 -0600

    .env file added to gitignore

commit 2b2e10c142fae9c330722af2355ee624671ec6a7
Author: Julio <julioes134@outlook.es>
Date: Wed Sep 25 15:09:53 2024 -0600

    Metodo para realizar actualizaciones agregado, refactorizacion del codigo y prueba de los objetos
```




FORMULARIOS DE GESTIÓN DE TIEMPO

Nombre: José Rodríguez
 Carné: 18907

Fecha	Inicio	Fin	Tiempo Interrupción (min)	Tiempo trabajado (min)	Tarea	Comentarios
21/09/2024	20:00	21:30	10	80	Requisitos funcionales	Vi un video
22/09/2024	07:00	07:45	0	45	Requisitos funcionales	N/A
25/09/2024	10:00	12:00	0	120	Requisitos funcionales	N/A
28/09/2024	9:00	10:00	0	60	Requisitos funcionales	N/A

Nombre: Julio Orellana
 Carné: 24207

Fecha	Inicio	Fin	Tiempo Interrupción (min)	Tiempo trabajado (min)	Tarea	Comentarios
24/09/2024	3:30 pm	11:00 pm	240 min	210 min	Creación, conexión e implementación de la base de datos	Lectura de documentación para aprender a utilizar el JDBC de MySQL en java.
25/09/2024	10:00 am	3:30 pm	90 - 120 min	180 - 210 min	Creación del Paquete exceptions para excepciones personalizadas	Ejercicio 4 POO

UNIVERSIDAD DEL VALLE DE GUATEMALA
CAMPUS CENTRAL
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
PROGRAMACIÓN ORIENTADA A OBJETOS



25/09/2024	5:15 pm	6:00 pm	0 min	45 min	Creación de la clase Encryption y el paquete Security	N/A
25/09/2024	6:00 pm	6:05 pm	0 min	5 min	Creación de la interfaz encargada de cargar las variables de entorno y el paquete utils	N/A
25/09/2024	9:00 pm	9:30 pm	0 min	30 min	Prueba de las clases creando consultas y nuevas implementaciones	N/A

Nombre: Ginebra Estrada
Carné: 24079

Fecha	Inicio	Fin	Tiempo Interrupción (min)	Tiempo trabajado (min)	Tarea	Comentarios
28/09/2024	11:11 a.m.	2:00 p.m.	20 min	149	Gestionar categorías de productos	tiktok
28/09/2024	3:41 p.m.	5:30 p.m.	25 min	84	Generar reporte de ventas	tiktok y aperitivo
28/09/2024	6:15 p.m.				Generar reporte de ventas	

UNIVERSIDAD DEL VALLE DE GUATEMALA
CAMPUS CENTRAL
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
PROGRAMACIÓN ORIENTADA A OBJETOS



Nombre: Sofía Batz

Carné: 241281

Fecha	Inicio	Fin	Tiempo Interrupción (min)	Tiempo trabajado (min)	Tarea	Comentarios
27/09/2024	5:30 p.m	9:00 p.m	70	140	Analisis Clase Clientes	Cena
28/09/2024	7:30 p.m	10:15 p.m	45	180	Clase Clientes	Salida familiar