

HYRULE EVENTOS

MANUAL TECNICO



1 TABLA DE CONTENIDO

1	TABLA DE CONTENIDO	1
2	INTRODUCCION	2
3	OBJETIVOS	2
3.1	OBJETIVO GENERAL	2
3.2	OBJETIVOS ESPECIFICOS	2
4	ALCANCE.....	2
5	REQUERIMIENTOS TECNICOS	3
5.1	HARDWARE.....	3
5.1.1	REQUISITOS DEL SISTEMA.....	3
5.2	SOFTWARE	3
5.2.1	TECNOLOGIAS IMPLEMENTADAS.....	3
6	ARQUITECTURA GENERAL	6
7	CONFIGURACION DEL ENTORNO	6
8	ESTRUCTURA DEL PROYECTO.....	7
9	BASE DE DATOS.....	8
9.1	MODELO RELACIONAL	8
10	GENERACION DE REPORTES Y CERTIFICADOS.....	8
11	COMPILACION Y EJECUCION.....	8
11.1	DESDE NETBEANS	8
11.2	DESDE LINEA DE COMANDOS	9
12	CONCLUSIONES	9
13	DESPEDIDA	9

2 INTRODUCCION

El sistema **Eventos Hyrule** solicitado por el Reino de Hyrule ha sido analizado, desarrollado, sometido a pruebas de diferente tipo y al mismo tiempo dando soporte en cualquier fallo que se pudo haber presentado durante las mismas. Con el fin de presentar una práctica de la mejor manera y rendimiento posible a las personas encargadas de la calificación o a cualquier persona que pueda tener acceso a dicha práctica.

Tratando de cumplir con los requerimientos establecidos en el documento proporcionado por el laboratorio del curso, así también implementando los conocimientos adquiridos durante las clases magistrales vistos hasta la fecha como lo son: Algoritmos, pseudocódigos, diagramas de clase, diagramas de entidad relación, diagrama de tablas, base de datos, programación estructurada modular y POO. Todo dentro del lenguaje de alto nivel Java, se presenta el siguiente manual técnico.

3 OBJETIVOS

3.1 OBJETIVO GENERAL

Este documento tiene como objetivo y propósito principal presentar el proceso de desarrollo del sistema **Eventos Hyrule - Triforce Software**. Al mismo tiempo dar referencias de como interactuar con el programa para futuras actualizaciones o darle un mantenimiento adecuado en caso de errores o caídas.

3.2 OBJETIVOS ESPECIFICOS

1. **Mostrar el código fuente para una posible mejora en el sistema *Eventos Hyrule - Triforce Software*.**
2. *Requisitos para una satisfactoria ejecución del programa.*
3. *Manifestar evidencias del diseño del sistema antes de implementarlo (Pseudocódigo, estados, etc.).*

4 ALCANCE

Este documento está especialmente diseñado para ser interpretado por estudiantes de programación o programadores, aunque también puede ser interpretado por personas que tengan conocimientos básicos en programación: Algoritmos, pseudocódigo, base de datos, NetBeans, Java, etc.

5 REQUERIMIENTOS TECNICOS

5.1 HARDWARE

5.1.1 REQUISITOS DEL SISTEMA

Mínimo: Procesador de 500 MhZ, 96 MB de RAM, tarjeta de video de 16 MB, Windows XP o versiones posteriores.

Recomendado: Procesador de 800 MhZ, 128 MB de RAM, tarjeta de video de 32 MB, Windows XP o versiones posteriores.

5.2 SOFTWARE

5.2.1 TECNOLOGIAS IMPLEMENTADAS

Java JDK: Java™ Development Kit (JDK) es un software para los desarrolladores de Java. Incluye el intérprete Java, clases Java y herramientas de desarrollo Java (JDT): compilador, depurador, desensamblador, visor de applets, generador de archivos de apéndice y generador de documentación. El JDK le permite escribir aplicaciones que se desarrollan una sola vez y se ejecutan en cualquier lugar de cualquier máquina virtual Java. Las aplicaciones Java desarrolladas con el JDK en un sistema se pueden usar en otro sistema sin tener que cambiar ni recompilar el código. Los archivos de clase Java son portables a cualquier máquina virtual Java estándar.



Recomiendo tener instalado en el ordenador la versión 21.0.4 de Java JDK.

Apache Maven: Apache Maven es una herramienta que estandariza la configuración de un proyecto en todo su ciclo de vida, como por ejemplo en todas las fases de compilación y empaquetado y la instalación de mecanismos de distribución de librerías, para que puedan ser utilizadas por otros desarrolladores y equipos de desarrollo.



También contempla temas relacionados con la integración continua, para poder realizar la ejecución de test unitarios y pruebas automatizadas, test de integración, etc.

Recomiendo tener instalado en el ordenador la versión 3.9.6 de Apache Maven.

Apache NetBeans: NetBeans es un IDE o entorno de desarrollo integrado, basado en el lenguaje Java y ejecutado en Swing.

De esta forma, NetBeans o Apache NetBeans es una aplicación de código abierto, que ha cobrado bastante popularidad en los últimos años.

Este IDE, orientado principalmente a las apps de Java, ofrece diferentes herramientas digitales como editor de texto, código, compilador, interfaz gráfica de usuario; además de un depurador.

Por otro lado, cabe destacar que NetBeans facilita la creación de aplicaciones estructuradas, ya que están basadas en un conjunto de módulos. Así, se favorece el desarrollo de las diversas funciones de una manera independiente y pudiendo también reutilizar los componentes.



Recomiendo tener en el ordenador la versión 21 de Apache NetBeans.

MySQL: MySQL es, como su nombre indica, un sistema de gestión de bases de datos relacionales o SGBD basado en SQL. En la actualidad, este **software de código abierto** forma parte de Oracle, la empresa que también desarrolló el lenguaje de programación Java.



MySQL almacena, gestiona y muestra datos en tablas. Funciona como un **sistema cliente-servidor**. Mientras que la base de datos actúa como un servidor en el que se almacena toda la información relevante, el software puede verse como un cliente. Con la ayuda del software, los usuarios de la base de datos relacional pueden formular diversas consultas, denominadas “**queries**”, en el lenguaje de consulta SQL y enviarlas al sistema de base de datos. Estos son procesados por MySQL, por lo que el acceso a los datos es también una parte importante de MySQL.

Windows 10: Windows 10 es uno de los sistemas operativos más utilizados en todo el mundo y su lanzamiento se produjo allá por julio de 2015. Los sistemas operativos como Windows se encargan de gestionar el hardware de nuestro ordenador para poder ejecutar aplicaciones y programas, ya sean propias o desarrolladas por terceros.



Actualmente, Windows 10 goza de mucha popularidad entre los usuarios de PC, incluso con la reciente salida de Windows 11 al mercado. Su gran rango

de adaptación y optimización lo hacen una gran opción incluso para ordenadores más austeros.

Para este proyecto recomiendo versiones de Windows posteriores a Windows 7, aunque queda a disposición del desarrollador escoger el OS que más le convenga.

Git: es un sistema de control de versiones de código fuente. Es una herramienta que se utiliza para llevar un registro de los cambios en el código fuente de un proyecto y permitir a varios desarrolladores trabajar en el mismo proyecto de manera simultánea.



Con Git, es posible crear un repositorio que contenga el código fuente de un proyecto y llevar un registro de todos los cambios realizados en el repositorio. Los desarrolladores pueden colaborar en el mismo proyecto al hacer «commits» (guardar cambios) en el repositorio y luego «pushear» (enviar) estos cambios a un servidor centralizado para que puedan ser utilizados por otros desarrolladores.

Git es una herramienta muy popular en la comunidad de desarrollo de software debido a su flexibilidad, velocidad y capacidad para manejar grandes proyectos. También es ampliamente utilizado en proyectos de código abierto, ya que permite a cualquier persona contribuir al proyecto y hacer seguimiento de los cambios realizados en el código fuente.

GitHub: es una plataforma de desarrollo colaborativo que aloja proyectos en la nube utilizando el sistema de control de versiones llamado Git. Ayuda a los desarrolladores a almacenar y administrar el código llevando un registro de cambios. Generalmente el código es abierto, lo que permite realizar proyectos compartidos y mantener el seguimiento detallado de su progreso. La plataforma GitHub también funciona como red social conectando a los desarrolladores con los usuarios. Como usuario puedes descargar programas o aplicaciones, y de la misma manera puedes aportar a su desarrollo ofreciendo mejoras y discutir las cuestiones que te interesan en foros temáticos.



6 ARQUITECTURA GENERAL

El sistema sigue una **arquitectura en capas**, lo que facilita la separación de responsabilidades:

- **Capa de Conexión (DBConnection):** Maneja la conexión con MySQL.
 - **Capa DAO (Data Access Objects):** Contiene la lógica para interactuar con la base de datos.
 - **Capa de Servicios (Lógica del negocio):** Aplica reglas de negocio (validaciones de pagos, cupos, duplicados, etc.).
 - **Capa de Presentación (Frontend GUI con Swing/NetBeans GUI Builder):** Formularios que permiten al usuario final interactuar con el sistema.
 - **Capa de Utilidades:** Generación de reportes y certificados en HTML.
-

7 CONFIGURACION DEL ENTORNO

- **Java JDK 21**
- **Apache Maven 3.9.6**
- **NetBeans 21**
- **MySQL Server 9.2.0 y MySQL Workbench 8.0 CE**
- **Dependencias Maven**

En el archivo pom.xml se incluye:

```
<dependency>
  <groupId>com.mysql</groupId>
  <artifactId>mysql-connector-j</artifactId>
  <version>9.3.0</version>
</dependency>
```

Se usa además el **maven-shade-plugin** para generar un **JAR ejecutable con todas las dependencias incluidas**.


- **Configuración de la Base de Datos**
 1. Crear un esquema en MySQL:
 2. `CREATE DATABASE eventos_hyrule;`
-

3. *Importar el script SQL proporcionado en la carpeta /sql.*


4. *Configurar credenciales en DBConnection.java:*

```
private static final String USER_NAME = "root";  
private static final String PASSWORD = "*****";  
private static final String SCHEMA = "eventos_hyrule";
```

8 ESTRUCTURA DEL PROYECTO

 *src/main/java/com/hyrule/eventos/hyrule*


- *EventosHyrule.java: Clase principal con main().*

 *dbconnection*


- *DBConnection.java: Clase para gestionar conexión con MySQL.*

 *dao*

- *Clases DAO para cada entidad: EventoDAO, ParticipanteDAO, InscripcionDAO, PagoDAO, ActividadDAO, AsistenciaDAO.*

 *model*

- *Entidades simples en Java (POJOs) para mapear tablas: Evento, Participante, Inscripcion, etc.*

 *service*


- *Reglas de negocio, validaciones y lógica de aplicación.*

 *ui*

- *Formularios creados con NetBeans GUI Builder: FrmEvento, FrmParticipante, FrmInscripcion, etc.*

 *util*

- *ArchivoUtil.java → Manejo de archivos HTML.*
- *Configuracion.java → Ruta fija para salida de reportes y certificados.*

 *resources*

- *Iconos, estilos y plantillas HTML para certificados.*

9 BASE DE DATOS

9.1 MODELO RELACIONAL

- **Evento:** almacena datos de eventos.
- **Participante:** contiene datos personales y de institución.
- **Inscripción:** relaciona participantes y eventos.
- **Pago:** registra pagos por inscripción.
- **Actividad:** actividades dentro de un evento.
- **Asistencia:** registro de asistencia a cada actividad.

Todas las tablas tienen **PK y FK**, con validaciones como:

- Pagos con monto > 0.
- Inscripción única por participante-evento.
- Cupo máximo en actividades.

10 GENERACION DE REPORTES Y CERTIFICADOS

El sistema genera **archivos HTML** en la carpeta configurada (Configuracion.RUTA_SALIDA).

- **Reporte de Participantes:** correo, tipo, nombre completo, institución, estado de inscripción.
- **Reporte de Actividades:** código, evento, título, encargado, hora, cupo, cantidad de inscritos.
- **Reporte de Pagos:** correo, nombre, método de pago, monto.
- **Certificados:** documento HTML personalizado por participante y evento.

11 COMPILACION Y EJECUCION

11.1 DESDE NETBEANS

- Abrir proyecto.
- Botón derecho → Run.

11.2 DESDE LINEA DE COMANDOS

1. *Compilar y empaquetar:*
 2. *mvn clean package*
 3. *Ejecutar el JAR empaquetado:*
 4. *java -jar target/eventos-hyrule-1.0-SNAPSHOT-shaded.jar*
-

12 CONCLUSIONES

*Este manual proporciona la documentación necesaria para que un desarrollador pueda instalar, compilar, ejecutar y mantener el sistema **Eventos Hyrule**. La arquitectura modular facilita la **extensión y evolución del sistema**, asegurando un ciclo de vida de software ordenado y profesional.*

13 DESPEDIDA

ESPERO QUE ESTE MANUAL SEA DE UTILIDAD PARA QUE PUEDAS COMPRENDER DE UNA MEJOR MANERA EL DESARROLLO DEL PROYECTO Y QUE PUEDAS AUMENTAR TUS HABILIDADES EN EL LENGUAJE DE PROGRAMACION JAVA. ¡GRACIAS POR LEER!

“RECUERDA, SI PUEDES IMAGINARLO, PUEDES PROGRAMARLO” (Programación ATS, 2018).