

Universidad de San Buenaventura

Facultad de Ingeniería

Tec. Desarrollo de Software

(Proyecto Integrador)

Proyecto

PROGRAMMAS Y

Integrantes:

Ruben Duran

Jorge Agudelo

Cristian Camilo Mesa

Hector Fabio Ramirez

Juan Felipe Quinche

Introducción

El presente documento tiene como propósito registrar y documentar el proceso, avance y cambios del proyecto integrador.

El proyecto integrador es un producto entregable en la carrera de Tecnología de Desarrollo de Software, el proyecto tiene que estar focalizado en los diferentes puntos que se visualicen en los 6 semestres de la carrera, pasando desde bases de datos, back-end, front-end y la utilidad de las metodologías de creación y progreso de proyectos.

La documentación del proyecto menciona y profundiza todas aquellas partes que conforman el mismo, explica los puntos importantes que se presentaron durante la realización del proyecto tales como el lenguaje de programación utilizado, el gestor de base de datos que se implementa, los diagramas UML implementados, entre otras cosas.

Además de documentar el progreso que se haga en el proyecto también se hará hincapié en la organización del equipo responsable y en las herramientas que se utilicen para la ayuda, mejora y cambios en el proyecto a nivel técnico.

Índice

- Explicación del proyecto
- Entorno de desarrollo
 - Java Web
 - Server
 - Eclipse
 - Framework
 - Reac
- Gestor de BD
 - Base de Datos
 - MySQL
 - Tablas
 - Modelo Relacional
- Back y Front End
 - Diagrama de clases
 - Explicación del código fuente
 - Explicación del orden de la codificación
 - Herramientas para el Front
 - Herramientas para el Back

- Metodologías
 - Scrum
 - Diagramas UML
 - Diagrama de casos de uso
 - Diagrama de estado
 - Diagrama de actividades

- Repositorios y respaldos
 - Google Drive
 - GitHub
 - GitBook

Explicación del Proyecto Integrador

En la Universidad de San Buenaventura, facultad de Ingeniería, carrera Tecnología en Desarrollo de Software. El proyecto integrador consta de una aplicación y/o programa funcional en donde se evidencie que los conceptos de las diferentes materias vistas a lo largo de los 6 semestres cursados en la carrera obtengan el conocimiento concreto, critico y eficiente en los estudiantes haciendo que los mismos deban crear un programa web, Software o aplicación móvil utilizando conocimientos, estrategias o herramientas disponibles para su elaboración.

Ya explicado ese punto pasamos a presentar a detalle nuestro proyecto integrador, este mismo, consta de un programa web enfocado al estudio y auto formación de las personas interesadas en el mundo de la programación, nuestro programa se identifica con el nombre de “PROGRAMMASY”, este mismo será un programa web completamente funcional en el cual se podrá visualizar un catálogo de Cursos disponibles para los usuarios, estos Curso serán de la temática de lenguajes de programación (Java, Python, C#, etc.) donde los usuarios podrán inscribirse si dicho Curso esta disponible, los usuarios pueden certificarse al completar cualquier Curso donde estén inscritos, además, también, los mismos usuarios podrán participar en Eventos los cuales serán lanzados en fechas específicas, los Eventos recompensan a los usuarios que participen con su nivel de competencia, este nivel de competencia se puede visualizar tanto en el perfil de los usuarios como en su texto de pie debajo de su icono de perfil, a parte de eso existe un apartado denominado Versus, en este apartado se habilita un entorno de desarrollo para los usuarios donde estos mismos pueden retar o ser retados en un Usuario Versus Usuario en donde se realizara una batalla amistosa en donde los dos

Usuarios pueden codificar en el lenguaje de programación que la plataforma disponga, este apartado de Versus también se relaciona con los Eventos, aunque el programa es de auto formación también cuenta con una herramienta que contacta a un tutor experto en los campos de programación con los usuarios de los cursos.

La principal población o el conjunto de personas a los cuales se quiere llegar con la plataforma de manera objetiva son todas aquellas personas que se vean inmersas en el universo de desarrollo de sistemas, ya sea, aquellos o aquellas que se quieren enfocar en Back o en el Front End, Full Stacks, Bases de datos, entre otros.

Aunque esa persona se encuentre en el rango de Junior, Semi Senior o Senior, podrá registrarse e ingresar a la plataforma de manera fácil y eficiente disponiendo de todo el contenido que la plataforma posee.

Entorno de Desarrollo

El entorno de desarrollo se especifica a continuación todos los ítems utilizados para la elaboración del programa:

Lenguaje de Programación – Java Web

Para nuestro programa decidimos utilizar el lenguaje de programación Java, pero utilizaremos la rama de este lenguaje enfocado a las aplicaciones webs. Java Web es un lenguaje de programación de alto nivel que ha sido diseñado para desarrollar aplicaciones web dinámicas y escalables. Es una plataforma muy popular y ampliamente utilizada en todo el mundo debido a sus características únicas y su capacidad para crear aplicaciones seguras, robustas y confiables. En este ensayo, vamos a discutir las características, conceptos, enfoques y funciones útiles del lenguaje de programación Java Web.

Características de Java Web: Portabilidad: Java Web es un lenguaje de programación independiente de la plataforma, lo que significa que una vez que se escribe un código en Java Web, se puede ejecutar en cualquier sistema operativo.

Orientación a objetos: Java Web es un lenguaje de programación orientado a objetos, lo que significa que permite al programador crear objetos que tienen atributos y métodos. Esto permite la modularidad y la reutilización del código, lo que a su vez facilita el mantenimiento del código.

Seguridad: Java Web tiene un enfoque riguroso en cuanto a la seguridad. La plataforma Java Web incorpora características de

seguridad, como el control de acceso y la autenticación, que permiten a los programadores crear aplicaciones web seguras y confiables.

Interoperabilidad: Java Web tiene una capacidad inherente de interoperabilidad, lo que significa que puede trabajar fácilmente con otros sistemas y lenguajes de programación.

Conceptos de Java Web: Servlets: Los servlets son componentes Java que se utilizan para procesar solicitudes de clientes en aplicaciones web. Los servlets son muy útiles para crear aplicaciones web dinámicas.

JSP: Java Server Pages (JSP) es una tecnología que permite a los desarrolladores de aplicaciones web crear páginas web dinámicas que se pueden generar a partir de datos almacenados en una base de datos.

Beans: Los beans son componentes de software que se utilizan para almacenar y procesar datos en aplicaciones web. Los beans son muy útiles para crear aplicaciones web escalables y flexibles.

Enfoque de Java Web: Java Web se centra en la creación de aplicaciones web dinámicas y escalables que sean seguras y confiables. La plataforma Java Web se basa en una arquitectura de tres capas que permite una fácil integración y desarrollo de aplicaciones web. El enfoque principal de Java Web es proporcionar una plataforma sólida y confiable para el desarrollo de aplicaciones web.

Funciones útiles de Java Web: Desarrollo de aplicaciones web escalables: Java Web es una plataforma que permite el desarrollo de aplicaciones web escalables y flexibles.

Integración fácil: Java Web se integra fácilmente con otras tecnologías y sistemas, lo que facilita la creación de aplicaciones web complejas.

Seguridad: Java Web es una plataforma segura que incorpora características de seguridad avanzadas que permiten a los programadores crear aplicaciones web seguras y confiables.

Interoperabilidad: Java Web es una plataforma que puede trabajar fácilmente con otros sistemas y lenguajes de programación, lo que permite la creación de aplicaciones web que se integran sin problemas con otros sistemas y tecnologías.

En conclusión, Java Web es un lenguaje de programación orientado a objetos independiente de la plataforma que se utiliza para el desarrollo de aplicaciones web escalables y seguras. Java Web tiene un enfoque riguroso en cuanto

Server – GlassFish

GlassFish es un servidor de aplicaciones Java de código abierto desarrollado por Oracle. Este servidor de aplicaciones proporciona una plataforma de desarrollo y ejecución para aplicaciones Java que es altamente escalable y de alto rendimiento.

Características de GlassFish: Escalabilidad: GlassFish es altamente escalable y puede manejar grandes volúmenes de solicitudes de aplicaciones sin afectar el rendimiento. Modularidad: GlassFish se compone de módulos, lo que significa que los desarrolladores pueden agregar o quitar módulos según sea necesario para satisfacer los requisitos de la aplicación.

Compatibilidad: GlassFish es compatible con una amplia gama de sistemas operativos y plataformas, lo que lo hace altamente versátil.

Funcionalidades de GlassFish: Despliegue de aplicaciones: GlassFish permite a los desarrolladores desplegar aplicaciones Java EE de manera fácil y rápida.

Gestión de recursos: GlassFish gestiona de manera eficiente los recursos, lo que permite una mejor utilización de los mismos.
Monitorización de la aplicación:

GlassFish proporciona herramientas de monitorización de la aplicación que permiten a los desarrolladores identificar y solucionar problemas de rendimiento de la aplicación.

Ventajas de GlassFish: Licencia de código abierto: GlassFish es un servidor de aplicaciones de código abierto, lo que significa que es gratuito y los desarrolladores pueden personalizar y modificar el código según sus necesidades.

Soporte de Oracle: GlassFish está respaldado por Oracle, lo que significa que los desarrolladores tienen acceso a soporte técnico y actualizaciones de seguridad.

Escalabilidad: GlassFish es altamente escalable y puede manejar grandes volúmenes de solicitudes de aplicaciones sin afectar el rendimiento, lo que lo hace ideal para aplicaciones empresariales.

Eclipse – Spring

Eclipse IDE es un entorno de desarrollo integrado ampliamente utilizado por desarrolladores de software en todo el mundo. Fue desarrollado inicialmente por IBM en el año 2001 y luego donado a la Fundación Eclipse, una organización de código abierto. La historia de Eclipse se remonta a un proyecto interno de IBM llamado "Plataforma para Herramientas de Desarrollo Componente Extensible" (en inglés, "Platform for Extensible Component Development Tools" o PECT), el cual fue lanzado como código abierto bajo el nombre de Eclipse en noviembre de 2001. Desde entonces, ha evolucionado y se ha convertido en una de las herramientas de desarrollo más populares en la comunidad de programadores.

Una de las principales ventajas de Eclipse es su naturaleza altamente modular y extensible. Permite a los desarrolladores personalizar su entorno de trabajo según sus necesidades y preferencias. Además,

ofrece soporte para una amplia gama de lenguajes de programación, incluyendo Java, C/C++, Python, PHP y muchos más. Esta flexibilidad lo convierte en una opción atractiva para desarrolladores de diferentes disciplinas.

La eficacia de Eclipse en el mercado se ve respaldada por su amplia adopción en la industria del software. Grandes compañías de renombre utilizan Eclipse como su entorno de desarrollo principal. Algunos ejemplos de estas compañías incluyen IBM, Oracle, Google, Adobe, Twitter, Amazon y SAP. Estas organizaciones confían en Eclipse debido a su robustez, escalabilidad y capacidad para manejar proyectos de gran envergadura.

Por otro lado, Spring Framework es un marco de desarrollo de aplicaciones Java que ha ganado una gran popularidad en la comunidad de desarrollo. Fue creado por Rod Johnson y lanzado en el año 2003. Spring se centra en la inversión de control y la programación orientada a aspectos, proporcionando un enfoque modular y flexible para el desarrollo de aplicaciones empresariales.

Una de las ventajas clave de Spring Framework en relación con Eclipse es su excelente integración con el IDE. Eclipse ofrece un conjunto de complementos y herramientas específicas para Spring, lo que facilita el desarrollo, la depuración y la administración de aplicaciones basadas en Spring. Estos complementos proporcionan soporte para la configuración de Spring, la inyección de dependencias, la administración de transacciones y muchas otras características del marco.

La historia de Spring está estrechamente relacionada con la evolución de la plataforma Java Enterprise Edition (Java EE) y su enfoque inicial en la simplificación del desarrollo empresarial. A lo largo de los años, Spring ha crecido y se ha convertido en un ecosistema de proyectos y

herramientas relacionados. Además de su compatibilidad con Eclipse, Spring ha ganado una reputación sólida debido a su enfoque modular, su amplio conjunto de características y su capacidad para abordar una amplia gama de necesidades de desarrollo empresarial.

Cuando se combinan Eclipse IDE y Spring Framework, los desarrolladores tienen a su disposición una poderosa combinación para la creación de proyectos web con Java. Eclipse proporciona un entorno de desarrollo sólido y altamente personalizable, mientras que Spring ofrece un marco de desarrollo ágil y eficiente. Los desarrolladores pueden aprovechar las características de Spring, como la inyección de dependencias, la administración de transacciones, la seguridad y el manejo de excepciones, para construir aplicaciones

webs escalables y de alta calidad. Algunos ejemplos de proyectos web famosos desarrollados con Java utilizando Eclipse y Spring incluyen el sistema de gestión de contenido Alfresco, el sistema de comercio electrónico Hybris de SAP y el portal comunitario Liferay. Estas aplicaciones destacan por su robustez, escalabilidad y eficiencia, gracias a la combinación de Eclipse y Spring.

Gestor Base de Datos

Motor BD – MySQL

Nuestro motor de base de datos para el proyecto será MySQL en resumen utilizaremos este gestor por los siguientes motivos:

Escalabilidad: MySQL es altamente escalable y puede manejar grandes volúmenes de datos y transacciones, lo que lo hace ideal para proyectos que necesiten expandirse en el futuro.

Rendimiento: MySQL es conocido por su rendimiento rápido y eficiente, lo que permite la rápida consulta y gestión de grandes conjuntos de datos.

Disponibilidad: MySQL es un SGBD de código abierto y gratuito, lo que lo hace fácilmente disponible para cualquier proyecto, y tiene una amplia comunidad de desarrolladores que ofrecen soporte y actualizaciones frecuentes.

Seguridad: MySQL proporciona una amplia gama de herramientas de seguridad para garantizar la protección de datos sensibles, como la encriptación y autenticación de usuarios, la verificación de contraseñas y la auditoría de accesos.

Flexibilidad: MySQL puede ser utilizado en una amplia gama de proyectos, incluyendo aplicaciones web, móviles y de escritorio, y es compatible con una amplia variedad de lenguajes de programación.

Integración: MySQL se integra bien con otras herramientas y tecnologías de desarrollo, como PHP, Java, Ruby on Rails, Node.js, entre otras, lo que lo hace ideal para proyectos en los que se necesiten diferentes tecnologías y herramientas.

En resumen, MySQL es una excelente opción para proyectos de cualquier tamaño que necesiten un SGBD confiable, escalable, rápido,

seguro y flexible, con amplia disponibilidad y con una gran comunidad de desarrolladores detrás de él.

Base de Datos – Tablas – DER

La base de datos para Programmasy la hemos creado en MySQL, no es una base de datos con muchos atributos, pero es hecha a medida de lo que necesitamos, es decir que solo creamos lo que era necesario para el programa, ni más ni menos.

A continuación, presentamos el Scrib, algunas capturas de su implementación, el modelo entidad relación y el modelo relacional:

Especificaciones de las tablas:

Tabla: Usuarios

- UsseID INT PRIMARY KEY
- UsseName VARCHAR (80)
- Email VARCHAR (100)
- Phone VARCHAR (50)
- EstadoID INT
- MatricID INT
- RegistrarID INT

Tabla: Estados

- EstadoID INT PRIMARY KEY
- StatusUsse VARCHAR (20)

Tabla: Cursos

- CoursesID INT PRIMARY KEY

- CourseName VARCHAR (50)
- CourseDescription VARCHAR (100)
- CategoryID INT
- LevelID INT

Tabla: Matriculación

- MatricID INT PRIMARY KEY
- Enrolled INT
- CoursesID INT

Tabla: Niveles

- LevelID INT PRIMARY KEY
- LevelCourse VARCHAR (10)

Tabla: Registro

- RegistrarID INT PRIMARY KEY
- NumParticipants INT
- EventsID INT

Tabla: Eventos

- EventsID INT PRIMARY KEY
- EventName VARCHAR (100)
- EventDescription VARCHAR (200)
- EventDate DATE
- VersusID INT

Tabla: Versus

- VersusID INT PRIMARY KEY
- VersusParticipants VARCHAR (100)

- VersusCode CHAR (30)
- VersusTime TIME

Tabla: Categorías

- CategoryID INT PRIMARY KEY
- CategoryName VARCHAR (100)
- SuppliersID INT

Tabla: Proveedores

- SuppliersID INT PRIMARY KEY
- SuppliersName VARCHAR (100)
- ProductName VARCHAR (100)
- CategoryProduct VARCHAR (100)

Tabla: Publicidad

- PublicID INT PRIMARY KEY
- PublicDescription VARCHAR (100)
- Payment FLOAT
-

SCRIP:

```
create database Programasy;  
use Programasy;
```

```
create table Usuarios(  
  UsseID INT PRIMARY KEY, UsseName VARCHAR (80), Email VARCHAR  
  (100), Phone VARCHAR (50), StatusID INT, MatricID INT, RegistrarID INT  
);
```

```
create table Estados(  
  EstadoID INT PRIMARY KEY, EstadoName VARCHAR (100), EstadoDescription  
  VARCHAR (255), EstadoStatusID INT, EstadoMatricID INT, EstadoRegistrarID  
  INT);
```



```
StatusID INT PRIMARY KEY, StatusUsser VARCHAR (20)
);
```

```
create table Cursos(
CoursesID INT PRIMARY KEY, CourseName VARCHAR
(50),CourseDescription VARCHAR (100), CategoryID INT, LevelID INT,
);
```

```
create table Matriculados(
MatricID INT PRIMARY KEY, Enrolled INT, CoursesID INT
);
```

```
create table Niveles(
LevelID INT PRIMARY KEY, LevelCourse VARCHAR (10)
);
```

```
Create table Categorias(
CategoryID INT PRIMARY KEY, CategoryName VARCHAR (100),
SuppliersID INT
);
```

```
Create table Registro(
RegistrarID INT PRIMARY KEY, ParticipannNumber INT, EventsID INT
);
```

```
create table Eventos(
EventsID INT PRIMARY KEY, EventName VARCHAR (100),
EventDescription VARCHAR (200), EventDate DATE, VersusID INT
);
```

```
create table Versus(
VersusID INT PRIMARY KEY, VersusParticipants VARCHAR (100),
VersusCode CHAR (3), VersusTime TIME
);
```

```
create table Proovedores(
SuppliersID INT PRIMARY KEY, SuppliersName VARCHAR (100),
ProductName VARCHAR (100), ProductDescription VARCHAR (100)
);
```

```
Create table Publicidad(
PublicID INT PRIMARY KEY, PublicDescription VARCHAR (100), Payment
FLOAT
);
```

Implementación DER:

DER PROG



