



Universidad Nacional de Loja

***FACULTAD DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES
NO RENOVABLES***

Carrera de Ingeniería en Sistemas/Computación

Programación Orientada a Objetos

DOCENTE:

Coronel Romero Edison Leonardo

ESTUDIANTE:

Miguel Ángel Luna Yunga

Estefanía Carolina Torres Balcázar

Brian Fernando Aguiñaga Loarte

Abel Alejandro Mora López

Darwin Andrés Sarango Soto

CICLO: Segundo "A"

PERIODO: ABRIL - OCTUBRE 2023

Introducción:

Este informe abarca el proceso completo de desarrollo de una aplicación web diseñada específicamente para gestionar las Competencias Deportivas de la Universidad Nacional de Loja. Desde el principio, se enfocó en una planificación detallada y un diseño minucioso para asegurar que la implementación fuera fluida y que la interfaz de usuario resultara intuitiva para los usuarios finales. Una parte fundamental de este proceso involucró la creación de un diagrama de clases como etapa inicial de diseño, el cual sirvió como base para la posterior implementación.

La necesidad de una aplicación web para gestionar las competencias deportivas de la universidad surgió como respuesta a la creciente demanda de una herramienta centralizada que facilitara la organización, registro y seguimiento de los eventos deportivos. Antes de abordar el desarrollo, se reconoció la importancia de establecer un enfoque sólido que garantizara la efectividad y el éxito del proyecto.

La implementación de la aplicación se llevó a cabo siguiendo el diseño del diagrama de clases. Se utilizaron tecnologías web modernas, incluyendo HTML, CSS, Python, Frameworks y algunas herramientas extra para la creación del sitio web.

Objetivos del Proyecto:

- Diseñar e implementar una aplicación web que permita la inscripción y gestión eficiente de competencias deportivas.
- Mejorar la visibilidad y accesibilidad de la información relacionada con las competencias para los estudiantes y el personal involucrado.
- Proporcionar una plataforma que fomente la participación activa en actividades deportivas y promueva la vida saludable dentro del campus.

Herramientas Utilizadas:

1. ChatGPT

ChatGPT es un modelo de lenguaje desarrollado por OpenAI que utiliza inteligencia artificial para generar respuestas coherentes y contextuales a partir de una entrada de texto. Puede ser utilizado para crear interacciones conversacionales naturales en una aplicación web. Puede proporcionar recomendaciones, respuestas a preguntas frecuentes y mantener conversaciones fluidas con los usuarios. Este recurso se ha empleado para brindar soporte y resolver dificultades que surgían durante el proceso de desarrollo de nuestro sitio web. Su capacidad para comprender el contexto y generar respuestas relevantes ha contribuido significativamente a agilizar la detección y corrección de problemas en la programación de nuestro sitio web.

2. GitHub

GitHub es una plataforma de desarrollo colaborativo que facilita el control de versiones y la colaboración en proyectos de software. Esta nos permitió como equipos trabajar juntos en el desarrollo de código fuente, realizar seguimiento de problemas y gestionar nuestro proyecto de manera efectiva. GitHub también es una excelente herramienta para revisar y aprobar cambios que se hayan realizado nuestro equipo en el código.

3. GitHub Copilot

GitHub Copilot es una innovadora extensión que aprovecha el potencial de la inteligencia artificial para optimizar el flujo de trabajo de los desarrolladores al escribir código. Esta herramienta se ha convertido en un aliado valioso al programar, al ofrecer sugerencias de código

en tiempo real, autocompletar fragmentos de código y brindar explicaciones detalladas sobre el funcionamiento de cada componente.

Es fundamental destacar que, si bien GitHub Copilot proporciona un soporte significativo, no reemplaza la necesidad de la intervención humana en el proceso de desarrollo. Los desarrolladores aún deben proporcionar la lógica y la dirección esencial para garantizar un rendimiento óptimo. En otras palabras, GitHub Copilot actúa como un colaborador inteligente que acelera la creación de código al proporcionar sugerencias perspicaces, pero la creatividad y el juicio humano siguen siendo esenciales para garantizar la calidad y la eficacia del producto final.

4. YouTube

Se usó YouTube porque nos ofrece una amplia variedad de tutoriales y recursos para el aprendizaje y la formación en el desarrollo web. Los creadores de contenido publican videos sobre programación, diseño web, manejo de herramientas y tecnologías específicas, lo que facilita el aprendizaje autodidacta y la adquisición de nuevas habilidades, con las cuales hemos puesto en práctica para la realización de nuestro sitio web.

La decisión de emplear YouTube como recurso clave se basó en su capacidad para proporcionar instrucción visual y práctica en temas relevantes. Los videos publicados por estos creadores de contenido ofrecen una guía concreta y detallada, lo que ha sido fundamental para implementar y materializar nuestro sitio web. La diversidad de enfoques presentados en la plataforma ha sido esencial para encontrar soluciones creativas a los desafíos que surgieron durante el proceso de desarrollo.

5. Frameworks como Bootstrap y Django

Bootstrap es un popular framework de diseño web front-end que proporciona una base sólida y preestablecida de componentes, estilos y diseños. Facilita la creación de interfaces de usuario atractivas y responsivas, reduciendo la necesidad de escribir código CSS y JavaScript desde cero.

Django, por otro lado, es un framework de desarrollo web en Python que se centra en la eficiencia y la facilidad de uso. Proporciona herramientas para la creación de aplicaciones web de alta calidad y con características avanzadas, como manejo de bases de datos, autenticación de usuarios y enrutamiento.

La combinación de estos marcos resultó fundamental para optimizar el proceso de creación de nuestro sitio web. Al implementar Bootstrap y Django, se eliminó la necesidad de crear componentes desde cero, ya que estos se encontraban predefinidos y listos para su empleo. Esta decisión de emplear ambos marcos fue para agilizar y mejorar la calidad del desarrollo web al proporcionar una base estable y práctica sobre la cual construir y expandir nuestra plataforma en línea.

6. API de Autenticación de Cuentas de Google

Las APIs de autenticación de cuentas de Google permiten a los usuarios iniciar sesión en una aplicación utilizando sus credenciales de Google. Esto simplifica el proceso de registro e inicio de sesión para los usuarios, ya que no necesitan crear nuevas cuentas. Además, proporciona un nivel adicional de seguridad y confianza, ya que Google se encarga de la autenticación.

7. PyCharm

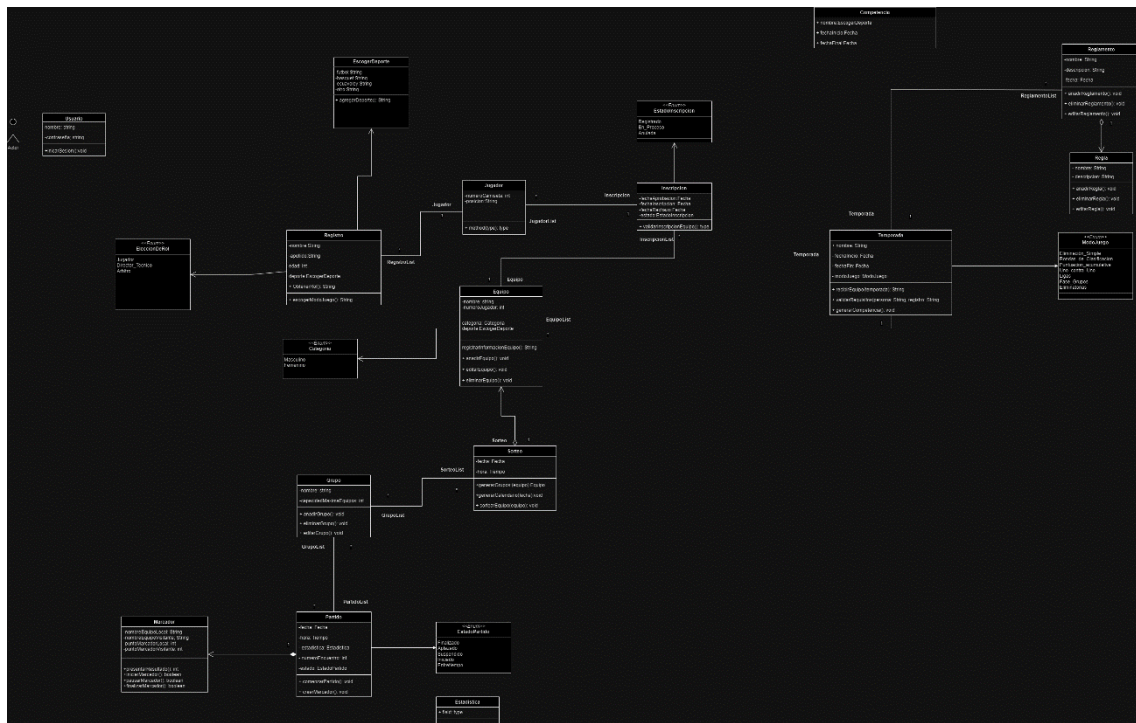
PyCharm es un entorno de desarrollo integrado (IDE) específicamente diseñado para programación en Python. Ofrece características avanzadas como resaltado de sintaxis, depuración, refactorización y control de versiones. PyCharm facilita la escritura de código Python

La elección de este IDE fue motivada por su poderosa utilidad al momento de redactar código. PyCharm no solo provee un ambiente de desarrollo cómodo y robusto, sino que también optimiza el flujo de trabajo al facilitar una serie de características que respaldan la productividad y la calidad del código generado, elevando así la eficacia y eficiencia en la creación de componentes backend para aplicaciones web.

Visual Paradigm es una herramienta de modelado visual que ayuda en el diseño y la planificación de proyectos. Es especialmente útil en el proceso de diseño de la arquitectura de una aplicación web. Proporciona herramientas para crear diagramas de flujo, diagramas de clases y otros modelos visuales que facilitan la comprensión y comunicación de las ideas del proyecto.

Metodología Utilizada:

Creación del Diagrama UML:



El proceso de diseño UML involucró la identificación de casos de uso clave, la definición de las clases necesarias y las relaciones entre ellas. A continuación, la explicación del mismo:

Comenzamos con el proceso de registro del usuario. El usuario proporcionará un nombre de usuario y una contraseña para acceder a la aplicación. Una vez que ingrese estos datos, se le presentará la opción de seleccionar un rol: Jugador, Director Técnico o Árbitro. Si elige "Jugador", procederá a registrar jugadores que participarán en competencias individuales. Para ello, se interactúa con la clase "Registro". En esta etapa, el usuario debe ingresar obligatoriamente nombres, apellidos, edad, el deporte seleccionado y el rol. En cuanto al deporte, las opciones incluyen Fútbol, Básquetbol o Vóley, o cualquier otro deporte proporcionado por el usuario. Continuando con la situación del jugador, se le requerirá registrar su nombre, apellido, edad, número de camiseta y posición. Estos datos se someterán a evaluación mediante la clase "Inscripción", que determinará el estado de la inscripción: "Registrado", "En Proceso" o "Anulado". Este mismo proceso es aplicable a los equipos. ¿Y qué sucede con los equipos? Estos son accedidos por el Director Técnico. Debe ingresar el nombre del equipo, el deporte en el que desea que participe su equipo y la información de los jugadores. Además, deberá especificar si su equipo es masculino o femenino en términos de categoría. Una vez que se han registrado todos los equipos, se accede a la clase "Grupo", donde se definen los grupos y se establece una capacidad máxima. Por ejemplo, si se permite un máximo de 4 equipos por grupo, cada nuevo equipo inscrito debe incorporarse a un grupo existente. Esto nos lleva a la clase "Sorteo", que organiza los equipos en grupos y genera los enfrentamientos entre ellos. La clase "Partido" incluye aspectos como la fecha y hora aleatoria para cada equipo, así como un estado que puede ser "Finalizado", "Aplazado", "Suspendido", etc. La función de la clase "Sorteo" también permite al usuario seleccionar el equipo ganador, brindando un mensaje de felicitación si la elección es correcta. La clase "Marcador" entra en juego para definir los goles o puntos obtenidos por los equipos, además de establecer el equipo local y el visitante. En relación con la explicación, toda la información registrada de los jugadores y los equipos se almacena en la clase "Temporada". Su propósito es presentar esta información y permitir la selección del modo de juego para los grupos generados. Las opciones de modo de juego incluyen "Eliminatorias", "Uno contra Uno", "Ligas", entre otros. Además, la temporada incluye un reglamento que detalla las reglas para todos los partidos. Estas reglas pueden ser modificadas o se pueden agregar nuevas reglas por parte del Árbitro.

Desarrollo aplicación escritorio Java:

La herramienta usada para el software sobre Competencias Deportivas fue IntelliJ IDEA este es un entorno de desarrollo integrado (IDE) de renombre creado por JetBrains, diseñado para potenciar la eficiencia y la creatividad en el proceso de desarrollo de software. Con su amplio conjunto de herramientas y características avanzadas, IntelliJ IDEA se ha convertido en una opción preferida para programadores en todo el mundo. Ya sea que estés trabajando en proyectos Java, Kotlin, Scala u otros lenguajes populares, este IDE ofrece un editor inteligente, capacidades de depuración intuitivas, refactorización automatizada y una integración fluida con diversas tecnologías. Con una comunidad activa y una diversidad de extensiones, IntelliJ IDEA se adapta a las necesidades individuales de los desarrolladores, acelerando el ciclo de desarrollo y ayudando a transformar conceptos en código de manera más eficiente.

IntelliJ IDEA, desarrollado por JetBrains, es un entorno de desarrollo integrado (IDE) que ha ganado una reputación sólida y un seguimiento apasionado en la comunidad de desarrollo de software. Su popularidad radica en una serie de ventajas clave que lo han establecido como una herramienta esencial para programadores de todo el mundo. Una de las principales ventajas de IntelliJ IDEA es su enfoque en la productividad. El IDE está equipado con un editor inteligente

que ofrece un resaltado de sintaxis altamente preciso, autocompletado inteligente y sugerencias contextuales mientras escribes código. Esta funcionalidad no solo acelera el proceso de codificación, sino que también ayuda a los desarrolladores a evitar errores sintácticos y a mantener un código más limpio y coherente. Otra característica distintiva es su poderosa capacidad de depuración. IntelliJ IDEA proporciona una experiencia de depuración sin problemas con características como la inspección de variables en tiempo real, seguimiento de pila detallado y la posibilidad de establecer puntos de interrupción con facilidad. Estas herramientas facilitan la identificación y resolución de problemas en el código, lo que agiliza el proceso de desarrollo y reduce el tiempo dedicado a la depuración. La refactorización automatizada es otra área en la que IntelliJ IDEA brilla. Cambiar la estructura de un código puede ser una tarea complicada, pero este IDE simplifica el proceso al permitir a los desarrolladores realizar refactorizaciones complejas con solo unos pocos clics. Esto ayuda a mantener el código limpio y mantenible a medida que evoluciona el proyecto. IntelliJ IDEA es ampliamente conocido por su excelente soporte para una variedad de lenguajes y tecnologías. Además del soporte excepcional para Java, ofrece compatibilidad con Kotlin, Scala, Groovy y otros lenguajes populares. También proporciona integración con sistemas de control de versiones como Git, así como con herramientas de construcción como Gradle y Maven. Esto permite a los desarrolladores trabajar con sus herramientas y tecnologías preferidas sin problemas, lo que contribuye a un flujo de trabajo más eficiente y sin interrupciones. La inspección de código automática es una característica esencial que ayuda a mantener la calidad del código en un nivel alto. IntelliJ IDEA analiza automáticamente el código en busca de posibles problemas, errores y mejoras sugeridas, lo que permite a los desarrolladores abordar estos problemas antes de que se conviertan en problemas mayores. Esto fomenta las buenas prácticas de codificación y contribuye a la creación de software más confiable y estable. Otra razón fundamental detrás de la popularidad de IntelliJ IDEA es su comunidad activa y su ecosistema de complementos. Los desarrolladores pueden acceder a una amplia variedad de complementos y extensiones creados por la comunidad, lo que les permite personalizar el IDE según sus necesidades específicas. Esta flexibilidad hace que IntelliJ IDEA sea adecuado para una variedad de proyectos y entornos de desarrollo, desde aplicaciones de escritorio hasta aplicaciones web y móviles.

En conclusión, IntelliJ IDEA ha demostrado ser mucho más que un simple entorno de desarrollo integrado. Gracias a su enfoque en la productividad, su conjunto de características avanzadas y su amplia gama de capacidades, se ha ganado el estatus de herramienta esencial para programadores de diversas disciplinas. Desde su editor inteligente y su capacidad de depuración intuitiva hasta su soporte integral para una variedad de lenguajes y tecnologías, IntelliJ IDEA ha demostrado su valía al acelerar el proceso de desarrollo, mejorar la calidad del código y brindar un ambiente flexible y personalizable para los desarrolladores. Con una comunidad activa y un ecosistema de complementos en constante crecimiento, este IDE continúa evolucionando para adaptarse a las necesidades cambiantes de la industria del software. En última instancia, la adopción generalizada de IntelliJ IDEA es un testimonio de su efectividad y su capacidad para impulsar la innovación y la excelencia en el mundo del desarrollo de software.

Luego de haber presentado sobre la app usada para la programación en Java para el software de Seguimiento De Competencias Deportivas toca presentar el programa:

Toda gira en torno a una clase denominada Registro, esta clase "Registro" es un componente crucial dentro de un sistema de gestión deportiva, ya que desencadena una serie de acciones esenciales para la inscripción, administración y seguimiento de equipos, jugadores y reglas en el contexto de competencias deportivas. Esta clase encapsula múltiples funciones que permiten a los usuarios realizar diversas operaciones, desde registrar jugadores individuales hasta inscribir equipos completos, y visualizar reglas y partidos planificados. Una de las principales

funcionalidades proporcionadas por la clase "Registro" es la capacidad de registrar jugadores individuales en temporadas específicas. A través de la función `registrarJugadores(Scanner scanner)`, los usuarios pueden proporcionar la información necesaria de los jugadores, que luego se valida y se agrega a la lista de jugadores registrados. Esta función también permite la administración de grupos de jugadores, asegurando que se inscriban en la temporada correcta y facilitando la organización de competencias individuales. Otra característica importante es la inscripción de equipos en las temporadas deportivas mediante la función `realizarRegistroEquipos(Scanner scanner)`. Los usuarios pueden ingresar detalles sobre los equipos y sus respectivos jugadores, lo que desencadena la validación y posterior inscripción de equipos en la lista de equipos inscritos. Además, esta función realiza la generación automática de grupos basados en la capacidad de cada grupo para asegurar una distribución equitativa de equipos en las competencias. La función `realizarRegistro(Scanner scanner)` guía a los usuarios a través de diferentes roles, como jugador, director técnico o árbitro, y proporciona opciones correspondientes a cada rol. Para los árbitros, se ofrece la posibilidad de ver y gestionar reglas, incluida la visualización de todas las reglas, reglas específicas, así como la adición y modificación de reglas existentes. Esto garantiza que los reglamentos estén actualizados y disponibles para todos los involucrados.

Por otro lado, el rol de jugador desencadena una serie de acciones y funcionalidades específicas que están diseñadas para facilitar la inscripción y la participación individual en competencias deportivas. Al seleccionar el rol de jugador en la función `realizarRegistro(Scanner scanner)`, se activa un conjunto de pasos que permiten a los usuarios inscribirse como jugadores individuales en una temporada determinada. A continuación, se detallan las acciones y funcionalidades desencadenadas por el rol de jugador en la clase "Registro":

-Elección de Rol: Jugador

Al seleccionar el rol de jugador en la función `realizarRegistro(Scanner scanner)`, se inicia el proceso específico para la inscripción de jugadores individuales.

-Verificación de Temporada: Visualización Opcional

Se presenta la opción de visualizar la temporada en curso a través de la función `temporada.verTemporada()`. Esta acción permite a los jugadores ver información relevante sobre la temporada actual, como detalles de inicio y finalización, modo de juego y otros detalles relevantes.

-Visualización de Jugadores en Temporada: Opcional

En caso de que el jugador decida ver la información de la temporada actual, se le brinda la opción de visualizar a todos los jugadores inscritos en la temporada mediante la función `jugador.mostrarInformacion()`.

-Registro de Jugadores Individuales: `registrarJugadores(Scanner scanner)`

La acción principal desencadenada por el rol de jugador es el registro individual en la temporada. La función `registrarJugadores(Scanner scanner)` guía a los jugadores a través del proceso de inscripción, solicitando y validando información personal y detalles relacionados con la participación en la competición.

-Validación de Inscripción: `Inscripcion.validarInscripcionDelJugador()`

Después de que el jugador proporcione su información, se realiza una validación de inscripción para asegurarse de que los datos sean correctos y cumplan con los requisitos establecidos. Esto se lleva a cabo utilizando la clase `Inscripcion` y su método `validarInscripcionDelJugador()`.

-Actualización de Jugadores Registrados: `jugadoresRegistradosEnTemporada`

Una vez que la inscripción del jugador es validada con éxito, el jugador se agrega a la lista de jugadores registrados en la temporada actual. Esto se realiza mediante la actualización de la lista `jugadoresRegistradosEnTemporada`.

-Generación de Temporada: `temporada`

Si es necesario, se genera una instancia de la clase `Temporada` para la temporada actual. Esta instancia contiene detalles como el nombre de la temporada, la fecha de inicio y finalización, y los jugadores inscritos.

Visualización Adicional: Jugadores Inscritos y Grupos

Después de completar el registro, los jugadores tienen la opción de ver la información actualizada de la temporada y los jugadores inscritos. También pueden visualizar los grupos a los que han sido asignados mediante la función `grupoJugadores.mostrarJugadoresEnGrupo()`.

Ahora bien, el rol de Director Técnico desencadena un conjunto de acciones y funcionalidades específicas diseñadas para facilitar la inscripción y la administración de equipos en competiciones deportivas. Al seleccionar el rol de Director Técnico en la función `realizarRegistro(Scanner scanner)`, se activa un proceso que permite a los usuarios crear y gestionar equipos en una temporada específica. A continuación, se detallan las acciones y funcionalidades desencadenadas por el rol de Director Técnico en la clase "Registro":

-Elección de Rol: Director Técnico

Al seleccionar el rol de Director Técnico en la función `realizarRegistro(Scanner scanner)`, se inicia el proceso específico para la inscripción y gestión de equipos.

-Verificación de Temporada: Visualización Opcional

Similar al proceso del rol de jugador, se presenta la opción de visualizar la temporada actual a través de la función `temporada.verTemporada()`. Esto permite a los Directores Técnicos obtener información relevante sobre la temporada en curso.

-Visualización de Equipos en Temporada: Opcional

Los Directores Técnicos pueden optar por visualizar información sobre los equipos inscritos en la temporada actual. Esto se logra mediante la función `temporada.mostrarEquiposEnTemporada()`.

-Registro de Equipos: `realizarRegistroEquipos(Scanner scanner)`

La función principal desencadenada por el rol de Director Técnico es el registro y la inscripción de equipos en la temporada. Los Directores Técnicos pueden ingresar detalles sobre el equipo y sus jugadores, que luego son validados y registrados en la lista de equipos inscritos.

-Generación de Grupos: `grupoActual.agregarEquipo()`

Una vez que un equipo es inscrito con éxito, se agrega a un grupo correspondiente a través de la función `grupoActual.agregarEquipo()`. Esto asegura que los equipos sean distribuidos en grupos equilibrados para las competiciones.

-Generación de Temporada: `temporada`

Al igual que en el proceso del rol de jugador, si es necesario, se genera una instancia de la clase Temporada para la temporada actual. Esto incluye detalles sobre la temporada y los equipos inscritos.

-Visualización Adicional: Equipos Inscritos y Grupos

Después de inscribir los equipos, los Directores Técnicos pueden optar por ver la información actualizada sobre la temporada, los equipos inscritos y la distribución en grupos. Esto se logra a través de las funciones `temporada.verTemporada()` y `grupoJugadores.mostrarJugadoresEnGrupo()`.

En resumen, este software de gestión deportiva, desarrollado en Java, abarca un conjunto esencial de funciones que revolucionan la inscripción, organización y seguimiento de competiciones deportivas. Este sistema se convierte en un aliado invaluable para jugadores, directores técnicos y árbitros, permitiéndoles ejecutar una serie diversa de operaciones personalizadas en el ámbito deportivo. El proceso de inscripción de jugadores individuales en temporadas específicas se convierte en un viaje guiado, incluyendo validación de datos y asignación a grupos, garantizando una participación estructurada y eficaz en competiciones individuales. Además, el rol del Director Técnico cobra vida al permitir la inscripción y organización de equipos en temporadas, optimizando la equidad y competitividad a través de la distribución ponderada en grupos. Asimismo, los árbitros se benefician de herramientas diseñadas especialmente para visualizar y gestionar reglas, asegurando que el conjunto de regulaciones se mantenga actualizado y accesible para todos los involucrados. En esencia, este software emerge como un pilar fundamental en la promoción y mejora de la experiencia deportiva, al dotar a los usuarios con capacidades innovadoras que enriquecen cada etapa del proceso competitivo.

Anexos Switch Director_Tecnico:

```
Ingrese su nombre de usuario:
Persona1
Ingrese su contraseña:
Arbitro
¡Sesión iniciada correctamente!
¡Advertencia! Escoger el rol de jugador es para equipos individuales.
¡Si usted escoge el rol de Director_Tecnico es para generar un equipo.
¡Si usted escoge el rol de Arbitro es para ver las reglas de los deportes sean grupales o individuales!
Escoja su rol (Jugador, Director_Tecnico, Arbitro):
Director_Tecnico
Desea visualizar la temporada? (Si/No):
Si
No hay temporada actualmente.
Deportes Disponibles:
1. Futbol
2. Basquet
3. Voley
4. Otro
Ingrese el número del deporte que desea escoger:
1
Ingrese la categoria del equipo (masculino o femenino): masculino
Recuerde que al momento de ingresar los datos del jugadores estos tengan una edad menos de 40 años y que el número de camiseta sea único y no repetido.
GRACIAS POR SU COMPRENSION
```

```
Ingrese el nombre del equipo: Barcelona
Ingrese la cantidad de jugadores que conforman el equipo: 2
Ingrese los datos del Jugador 1:
Nombre: Abel
Apellido: Mora
Edad: 19
Número de camiseta: 1
Posición: Defensa
Ingrese los datos del Jugador 2:
Nombre: Leonel
Apellido: Messi
Edad: 37
Número de camiseta: 10
Posición: Delantero
Registro en proceso. Espere un momento...
¡Equipo inscrito correctamente!
¿Desea ingresar otro equipo? (Si/No):
Si
Deportes Disponibles:
1. Futbol
2. Basquet
3. Voley
4. Otro
Ingrese el número del deporte que desea escoger:
```

```
Ingrese el número del deporte que desea escoger:
1
Ingrese la categoría del equipo (masculino o femenino): masculino
Recuerde que al momento de ingresar los datos del jugadores estos tengan una edad menos de 40 años y que el número de camiseta sea único y no repetido.
GRACIAS POR SU COMPRENSION
Ingrese el nombre del equipo: Real Madrid
Ingrese la cantidad de jugadores que conforman el equipo: 2
Ingrese los datos del Jugador 1:
Nombre: Lucas
Apellido: Madrid
Edad: 36
Número de camiseta: 10
Posición: Centrocampista
Ingrese los datos del Jugador 2:
Nombre: Cristiano
Apellido: Ronaldo
Edad: 37
Número de camiseta: 7
Posición: Delantero
Registro en proceso. Espere un momento...
¡Equipo inscrito correctamente!
¿Desea ingresar otro equipo? (Si/No):
No
Registro de equipos finalizado.
Equipos ingresados:
```

Registro de Equipos Finalizado.

Equipos ingresados:

- Barcelona
- Real Madrid

Temporada: Temporada 2023

Fecha de inicio: 10/08/2023

Fecha de finalización: 31/12/2023

¿Desea ver la temporada actualizada? (Si/No):

Si

Desea visualizar la temporada nuevamente? (Si/No):

Si

Modo de Juego de la Temporada: ELIMINATORIAS

Equipos ingresados:

Nombre del equipo: Barcelona

Deporte: Futbol

Categoría: masculino

Jugadores:

- Nombre: Abel, Edad: 19, Número de camiseta: 1, Posición: Defensa
- Nombre: Leonel, Edad: 37, Número de camiseta: 10, Posición: Delantero

Nombre del equipo: Real Madrid

Deporte: Futbol

Categoría: masculino

Jugadores:

- Nombre: Lucas , Edad: 36, Número de camiseta: 10, Posición: Centrocampista
- Nombre: Cristiano, Edad: 37, Número de camiseta: 7, Posición: Delantero

¿Desea ver los grupos? (Si/No):

¿Desea ver los grupos? (Si/No):

Si

En unos momentos se definirán los grupos. Espere un momento...

Grupo 1:

- Barcelona
- Real Madrid

Los Partidos se definirán muy pronto.....

El partido iniciará a las 03:17:22 el día 13/07/2023

- Barcelona vs Real Madrid (Duelo Legendario) estado: Iniciado

Usted que equipo cree que va a ganar? (Barcelona o Real Madrid):

Barcelona

Lo lamento pero ganó Real Madrid con un marcador de 2 - 1

¿Desea ingresar otro rol? (Si/No):

No

Registro finalizado.

Switch Jugador:

```
Ingrese su nombre de usuario:
Persona1
Ingrese su contraseña:
marita
;Sesión iniciada correctamente!
;Advertencia! Escoger el rol de jugador es para equipos individuales.
;Si usted escoge el rol de Director_Tecnico es para generar un equipo.
;Si usted escoge el rol de Arbitro es para ver las reglas de los deportes sean grupales o individuales!
Escoja su rol (Jugador, Director_Tecnico, Arbitro):
Jugador
Desea visualizar la temporada ? (Si/No):
Si
No hay temporada actualmente.
Ingrese los datos del Jugador:
Nombre: Abel
Apellido: Mora
Edad: 19
Al momento de ingresar otro deporte considere que este sea un deporte individual
Escoja el deporte:
Deportes Disponibles:
1. Karate
2. Natacion
3. Golf
```

```
3. Golf
4. Otro
1
Número de camiseta : 1
Posición: Ninguno
Registro en proceso. Espere un momento...
;Jugador inscrito correctamente!
¿Desea registrar otro jugador? (Si/No):
Si
Ingrese los datos del Jugador:
Nombre: Elías
Apellido: Poma
Edad: 21
Al momento de ingresar otro deporte considere que este sea un deporte individual
Escoja el deporte:
Deportes Disponibles:
1. Karate
2. Natacion
3. Golf
4. Otro
1
Número de camiseta : 2
Posición: Ninguno
Registro en proceso. Espere un momento...
;Jugador inscrito correctamente!
```

```

¿Desea registrar otro jugador? (Si/No):
No
Temporada: Temporada 2023
Modo de Juego de la Temporada: FASE_GRUPOS
Fecha de inicio: 10/08/2023
Fecha de finalización: 31/12/2023
Desea visualizar la temporada nuevamente? (Si/No):
Si
Jugadores en la temporada:
- Nombre: Abel, Edad: 19, Deporte: Karate, Posición: Ninguno
- Nombre: Elias, Edad: 21, Deporte: Karate, Posición: Ninguno
Desea visualizar los grupos? (Si/No):
Si
Jugadores en el Grupo A:
- Abel
- Elias
En unos momentos se definirán los grupos. Espere un momento...
Jugadores en el Grupo A:
- Abel
- Elias
Los Enfrentamientos se definirán muy pronto.....
El enfrentamiento iniciará a las 10:53:28 el día 28/08/2023
Abel vs Elias (Partido del Siglo) - Estado: Finalizado
Abel 41 puntos - Elias 60 puntos

```

Switch Arbitro:

```

¿Desea ingresar otro rol? (Si/No):
Si
¡Advertencia! Escoger el rol de jugador es para equipos individuales.
¡Si usted escoge el rol de Director_Tecnico es para generar un equipo.
¡Si usted escoge el rol de Arbitro es para ver las reglas de los deportes sean grupales o individuales!
Elija su rol (Jugador, Director_Tecnico, Arbitro):
Arbitro
Usted ha elegido el rol de Árbitro.
Seleccione una opción:
1. Ver todas las reglas
2. Ver regla específica
3. Agregar nueva regla
4. Modificar regla existente
2
Ingrese el índice de la regla que desea ver:
4
Regla 4: No realizar entradas o faltas peligrosas que pongan en riesgo la integridad de otros jugadores.
¿Desea ingresar otro rol? (Si/No):
Si
¡Advertencia! Escoger el rol de jugador es para equipos individuales.
¡Si usted escoge el rol de Director_Tecnico es para generar un equipo.
¡Si usted escoge el rol de Arbitro es para ver las reglas de los deportes sean grupales o individuales!
Elija su rol (Jugador, Director_Tecnico, Arbitro):
Arbitro

```

```
Arbitro
Usted ha elegido el rol de Árbitro.
Seleccione una opción:
1. Ver todas las reglas
2. Ver regla específica
3. Agregar nueva regla
4. Modificar regla existente
4
Ingrese el índice de la regla que desea modificar:
1
Regla 1: Respetar las decisiones del árbitro.
Ingrese la nueva descripción de la regla:
Todjo jugador debe seguir las reglas del Director
Regla 1: Todjo jugador debe seguir las reglas del Director modificada correctamente.
¿Desea ingresar otro rol? (Si/No):
```

Desarrollo Página web en Python:

Siguiendo el diseño detallado en el diagrama UML, procedimos a convertir las clases y relaciones previamente delineadas en elementos concretos en el código Python. Para este propósito, aprovechamos la potencia y flexibilidad que ofrece el framework Django. Este enfoque metodológico resultó invaluable para el desarrollo fluido y coherente de la aplicación web de gestión de Competencias Deportivas de la Universidad Nacional de Loja.

La conversión de las clases y relaciones del diagrama UML en clases y métodos en Python a través de Django permitió una transición coherente y estructurada del diseño a la implementación. El framework Django ofrece un entorno completo y bien integrado que facilita la creación de aplicaciones web robustas y escalables. Las clases del diagrama UML se tradujeron en modelos de Django, estableciendo así la estructura y las propiedades de las entidades clave en la base de datos.

La elección de Django también nos brindó una serie de ventajas, como la generación automática de la interfaz de administración, que simplifica la gestión y el mantenimiento de los datos. Además, el enfoque basado en vistas y URL de Django se alineó perfectamente con las rutas y acciones definidas en el diagrama de clases.

Diseño de Interfaz de Usuario (UI):

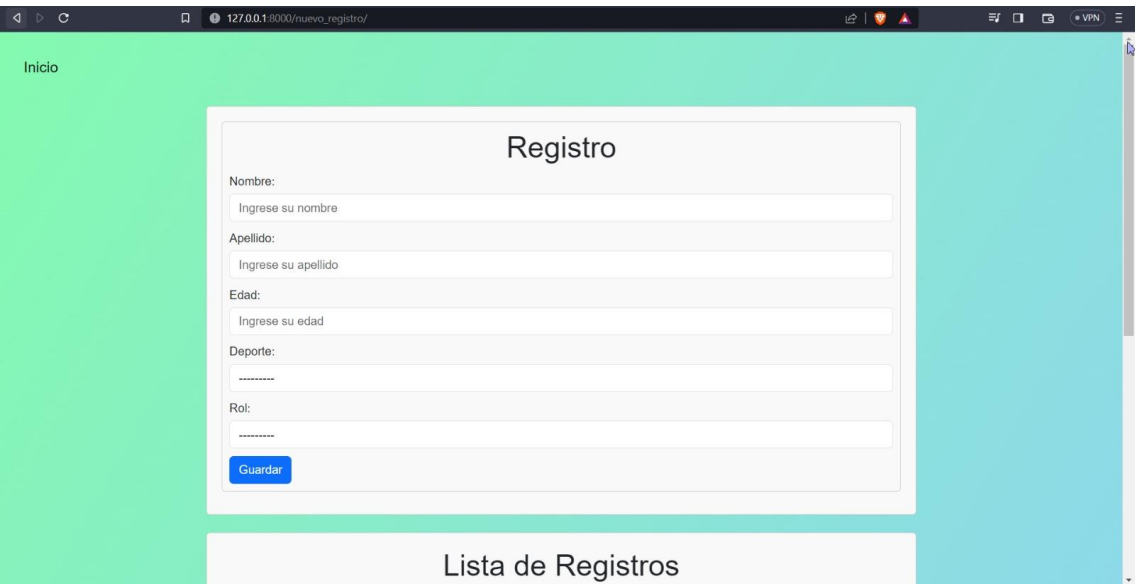
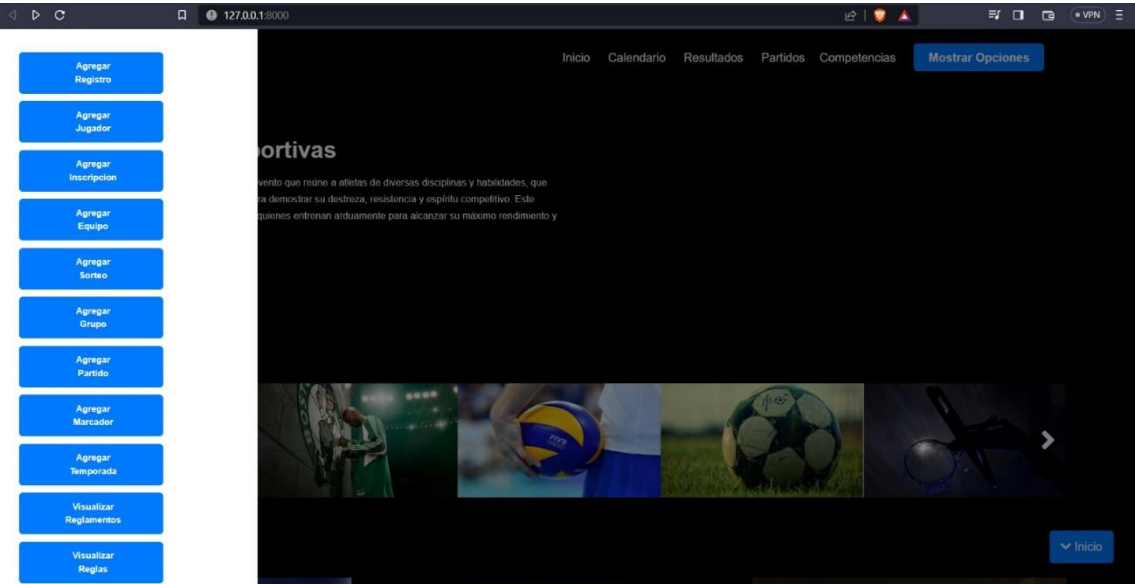
El diseño de la interfaz de usuario se realizó siguiendo estrechamente las directrices establecidas en el diagrama UML, en el cual cada clase se alineó con secciones específicas en la interfaz. Para la materialización de esta visión, se emplearon las herramientas poderosas que Python, HTML y CSS ofrecen, también se usaron los elementos proporcionados por Bootstrap, un framework front-end ampliamente reconocido.

La concepción de la interfaz de usuario se apoyó en el aprovechamiento eficiente de las capacidades interconectadas del diagrama UML. Cada entidad y relación plasmada en el diagrama tuvo su equivalente en términos de diseño en la interfaz. Esto permitió una correspondencia visual clara entre los componentes lógicos y su representación en pantalla.

En el proceso de desarrollo, HTML y CSS fueron los pilares fundamentales para la creación de una interfaz estéticamente agradable y fácil de usar. Sin embargo, para llevar el diseño un paso más allá y agilizar el proceso, se incorporó Bootstrap. Este conjunto de herramientas front-end ofreció una serie de componentes preestablecidos y estilos responsivos que simplificaron significativamente la creación de páginas web visualmente atractivas.

Los elementos de Bootstrap, como los menús desplegables, las tarjetas informativas y los formularios preestilizados, se implementaron con facilidad para dar vida a la interfaz. Además, la naturaleza responsive de Bootstrap aseguró que la interfaz se adaptara de manera elegante a diferentes tamaños de pantalla, mejorando la experiencia del usuario tanto en dispositivos móviles como en ordenadores de escritorio.

Anexos:



Inicio

Registrar Jugador

Registro:

Número de camiseta:

Ingrese su número de camiseta

Posición:

Ingrese su posición

Guardar

Lista de Jugadores

Nombre:

[Lionel Messi Cuccitini](#)

Acciones:

Editar

Eliminar

Inicio

Registrar Inscripción

Fecha de solicitud:

Ingrese su fecha de solicitud

Jugador:

Guardar

Lista de Inscripciones

Nombre:

[Lionel Messi Cuccitini](#)

Acciones:

Editar

Eliminar

[Robert Lewandowski](#)

Editar

Eliminar

Inicio

Registrar Equipo

Nombre:

Ingrese el nombre del Equipo

Número de jugadores:

Ingrese su número de jugadores

Categoría:

Inscripciones Registradas:

Lionel Messi Cuccitini

Robert Lewandowski

Cristiano Ronaldo

Mohamed Salah

Robert C...

Guardar

Lista de Equipos



Inicio

Registrar Marcador

Nombre de equipo local:

Nombre de equipo visitante:

Punto de marcador local:

Punto de marcador visitante:

[Guardar](#)

Lista de Marcadores

Nombre:	Acciones:
Barcelona FC vs Real Madrid	Editar Eliminar

Volver

Reglamento:

Nombre:	Descripcion:	Fecha de Creación:	Lista de Reglas:
Reglamento N°1	Reglas Generales para los Deportes	Aug. 15, 2023	Regla N°1 Regla N°2 Regla N°3 Regla N°4 Regla N°5 Regla N°6 Regla N°7 Regla N°8 Regla N°9 Regla N°10
Reglamento N°2	Contenido del reglamento de Futbol	Aug. 15, 2023	Regla N°11 Regla N°12
Reglamento N°3	Contenido del reglamento de Voleibol	Aug. 15, 2023	Regla N°13 Regla N°20
Reglamento N°4	Contenido del reglamento de Baloncesto	Aug. 15, 2023	Regla N°14 Regla N°16
Reglamento N°5	Contenido del reglamento de Balonmano	Aug. 15, 2023	Regla N°15
Reglamento N°6	Contenido del reglamento de Hockey	Aug. 15, 2023	Regla N°17
Reglamento N°7	Contenido del reglamento de Rugby	Aug. 15, 2023	Regla N°18
Reglamento N°8	Contenido del reglamento de Futbol Americano	Aug. 15, 2023	Regla N°19

Volver

Regla:

Nombre:	Descripcion:
Regla N°1	"Respetar las decisiones del árbitro"
Regla N°2	"Mantener un comportamiento deportivo y respetuoso en todo momento"
Regla N°3	"No usar lenguaje ofensivo ni gestos provocativos hacia oponentes, árbitros o público"
Regla N°4	"No realizar entradas o faltas peligrosas que pongan en riesgo la integridad de otros jugadores"
Regla N°5	"Evitar el juego brusco y desleal que pueda causar daño a los rivales"
Regla N°6	"Respetar las líneas de juego y no salir del campo durante el juego"
Regla N°7	"No utilizar objetos o accesorios que puedan causar daño o ventaja injusta"
Regla N°8	"No interferir en el juego del oponente o en la ejecución de tiros o saques"
Regla N°9	"Mantener el fair play y aceptar las decisiones arbitrales sin discutir"
Regla N°10	"Mantener un ambiente de juego seguro para todos los participantes"
Regla N°11	"Fuera de Juego: En deportes como el fútbol, un jugador se encuentra en posición de fuera de juego si está más cerca de la línea de gol rival que el balón y el penúltimo defensor al recibir el balón. Se sanciona con un tiro libre indirecto para el equipo contrario"
Regla N°12	"Tiro Penal: En deportes como el fútbol, se otorga un tiro penal por faltas cometidas dentro del área de penalti. El jugador atacante ejecuta un tiro libre directo desde el punto de penalti hacia la portería"
Regla N°13	"Bloqueo en Deportes de Red: En deportes como el voleibol, los bloqueadores no pueden tocar el balón antes o durante el ataque del oponente. Se permite bloquear si el balón no ha cruzado la red"
Regla N°14	"Líneas de Saque: En deportes como el baloncesto, los jugadores deben respetar las líneas de saque al realizar saques laterales o desde la línea de falta"

Django administration

WELCOME, BRIAN VIEW SITE / CHANGE PASSWORD / LOG OUT

Site administration

AUTHENTICATION AND AUTHORIZATION

Groups

+ Add

Change

Users

+ Add

Change

DAWNBEM

Equipos

+ Add

Change

Grupos

+ Add

Change

Inscripciones

+ Add

Change

Jugadores

+ Add

Change

Marcadores

+ Add

Change

Partidos

+ Add

Change

Registros

+ Add

Change

Reglamentos

+ Add

Change

Reglas

+ Add

Change

Revision inscripciones

+ Add

Change

Sorteos

+ Add

Change

Temporadas

+ Add

Change

Recent actions

My actions

Miguel Luna

Inscripcion

Bernardo Silva

Inscripcion

Redi Gonzales

Inscripcion

Bernardo Silva

Inscripcion

Redi Gonzales

Inscripcion

Barcelona Fc vs Real Madrid

Partido

Mohamed Salah

Inscripcion

Bernardo Silva

Inscripcion

Redi Gonzales

Inscripcion

Cristiano Ronaldo

Inscripcion

Django administration

WELCOME, BRIAN VIEW SITE / CHANGE PASSWORD / LOG OUT

Home » Dawnbem » Equipos

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups

+ Add

Users

+ Add

DAWNBEM

Equipos

+ Add

Grupos

+ Add

Inscripciones

+ Add

Jugadores

+ Add

Marcadores

+ Add

Partidos

+ Add

Registros

+ Add

Reglamentos

+ Add

Reglas

+ Add

Revision inscripciones

+ Add

Sorteos

+ Add

Temporadas

+ Add

Select equipo to change

ADD EQUIPO

Action:

Go

0 of 6 selected

☐

EQUIPO

☐

Chesela FC

☐

Liverpool

☐

Manchester City

☐

Dormunt

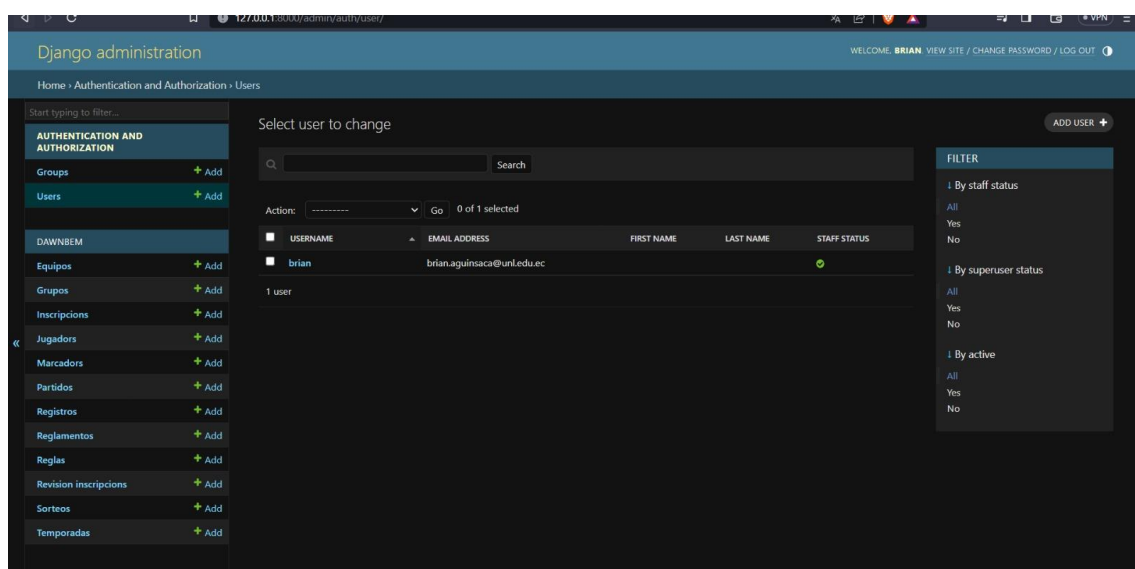
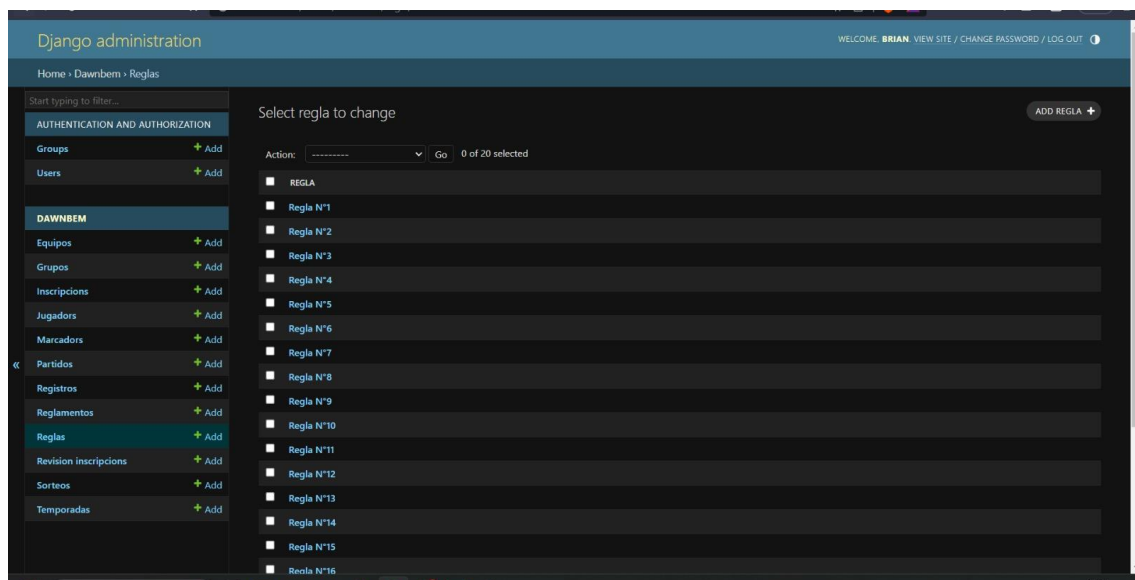
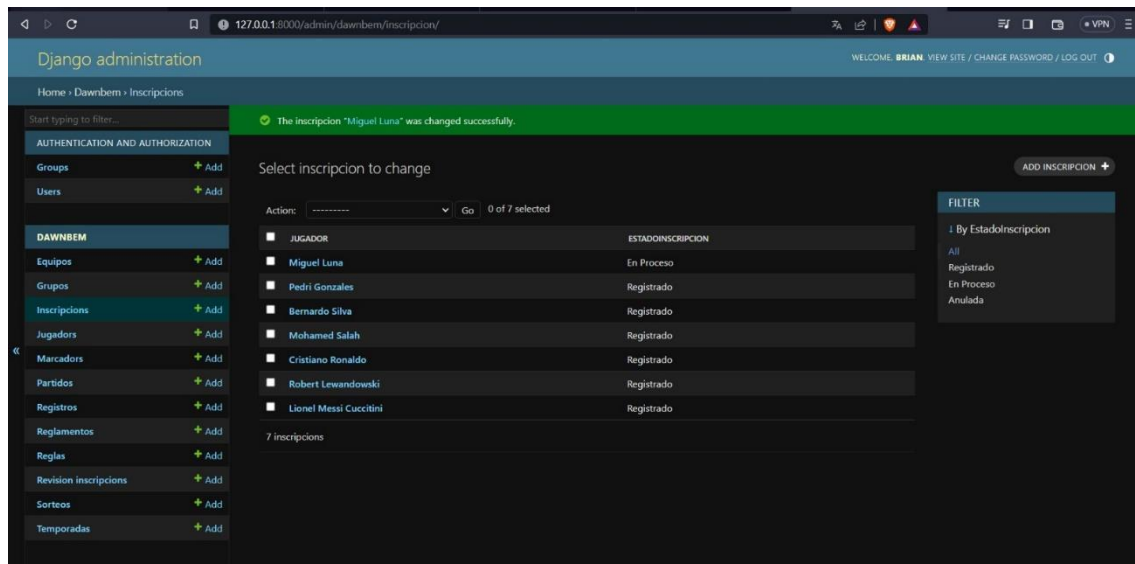
☐

Real Madrid

☐

Barcelona FC

6 equipos



Uso de API-Google

En nuestro proyecto de Competencias Deportivas. Fue un desafío, pero finalmente lo conseguimos.

Entonces, estábamos trabajando en este proyecto que requería que los usuarios pudieran iniciar sesión usando sus cuentas de Google. Queríamos hacerlo fluido y sin problemas. Así que, después de un poco de investigación, nos dimos cuenta de que necesitábamos hacer uso de la API de Google para autenticación. Aquí está el resumen de lo que hicimos:

Paso 1: Configurar las Credenciales en Google Dev Console

Primero, tuvimos que dirigirnos a la Consola de Desarrolladores de Google y configurar un proyecto allí. Fue sencillo, solo tuvimos que crear una nueva aplicación y obtener nuestras credenciales, como el ID de cliente y el secreto. Esto es como la llave que nos permitiría acceder a las cuentas de Google de los usuarios.

Paso 2: Integración con Django usando Allauth

En nuestro proyecto Django, utilizamos la biblioteca Allauth para gestionar la autenticación. Allauth hace que todo este proceso sea menos doloroso. Lo conectamos con las credenciales que obtuvimos en el paso anterior. Entonces, cada vez que un usuario hacía clic en "Iniciar sesión con Google", Allauth hacía magia y dirigía al usuario a la página de inicio de sesión de Google.

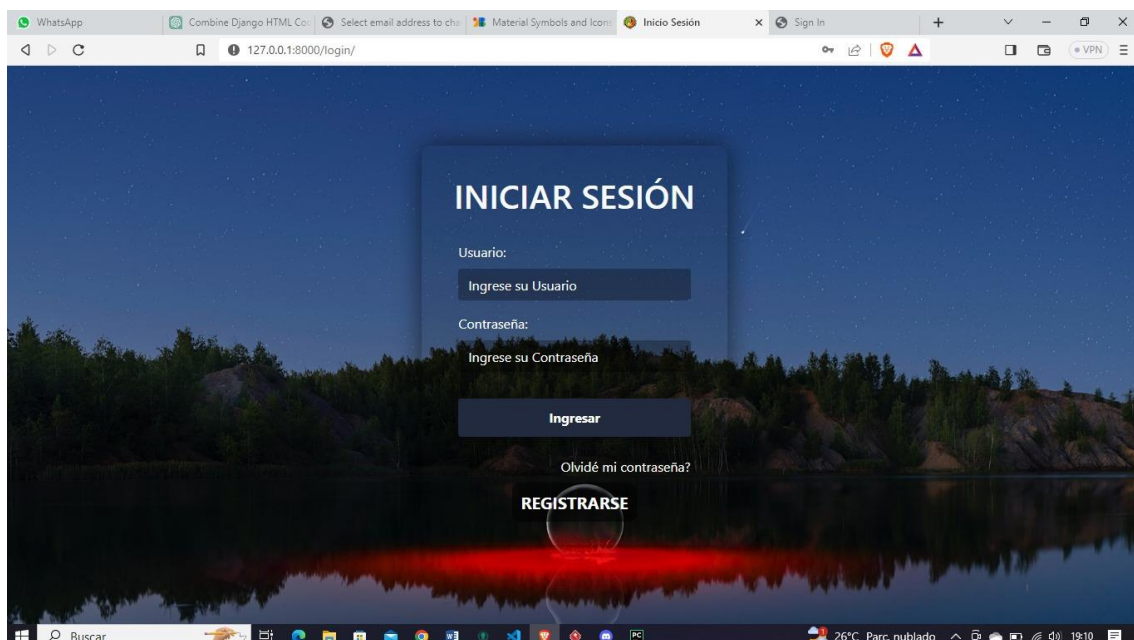
Paso 3: Manejo de Errores y Mensajes Personalizados

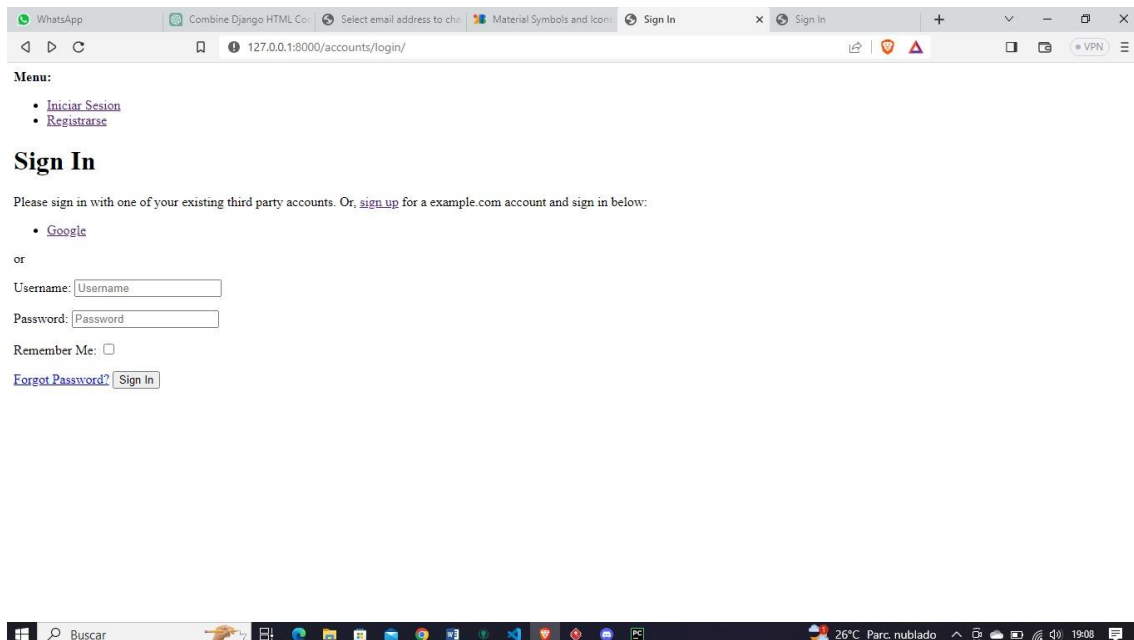
Ah, aquí es donde las cosas se pusieron un poco más interesantes. En el camino, nos topamos con algunos errores (¡quién no lo hace?). Por ejemplo, a veces los usuarios tenían problemas para iniciar sesión debido a algún error en la API de Google.

Después de un poco de sudor y trabajo en equipo, finalmente lo conseguimos. Los usuarios podían iniciar sesión con sus cuentas de Google sin problemas. Ahorró tiempo y esfuerzo a nuestros usuarios, lo que es genial.

Así que, en resumen, usar la API de Google para nuestro proyecto fue una montaña rusa emocionante. Aprendimos sobre cómo configurar credenciales, integrarlas en Django, manejar errores y realizar pruebas exhaustivas. Definitivamente, fue un desafío que valió la pena y nos ayudó a mejorar nuestras habilidades de desarrollo.

Anexos Api de Google:





Contribución a la Comunidad Universitaria:

La aplicación desarrollada puede servir como un ejemplo concreto de cómo la tecnología puede optimizar la gestión de competencias deportivas en el ámbito universitario. Puede ser una fuente de inspiración y referencia para futuros proyectos académicos que busquen abordar desafíos similares en la universidad.

Conclusiones:

El diseño previo a través del diagrama UML demostró ser esencial para comprender y visualizar la arquitectura del sistema antes de la implementación. El uso de PyCharm y las herramientas de desarrollo web permitieron una implementación más eficiente y organizada. Se identificaron áreas de mejora, como la implementación de autenticación de usuarios, para futuras actualizaciones.

En la era actual del desarrollo de aplicaciones web, contar con herramientas como ChatGPT, GitHub, Copilot, YouTube, frameworks como Bootstrap y Django, APIs de autenticación de cuentas de Google y PyCharm puede marcar la diferencia en la eficiencia, la calidad y el éxito general de un proyecto. Estas herramientas abarcan desde la colaboración en equipo y el aprendizaje autodidacta hasta la generación de código, la creación de interfaces atractivas y el desarrollo eficiente en Python. Utilizar estas herramientas de manera estratégica puede agilizar el proceso de desarrollo y mejorar la experiencia del usuario final.

En conclusión, la fusión de Python, HTML, CSS y la potencia de Bootstrap permitió transformar la visión del diagrama UML en una interfaz de usuario tangible y atractiva. La cuidadosa correspondencia entre clases y secciones, junto con los elementos modernos y flexibles de Bootstrap, resultó en una experiencia de usuario cohesiva y agradable que se alinea con las expectativas contemporáneas en el diseño de páginas web.

APP MOVIL

Kodular Creator es una plataforma de desarrollo de aplicaciones móviles que se basa en la programación visual y sin código. Fue diseñada para permitir a personas sin experiencia en

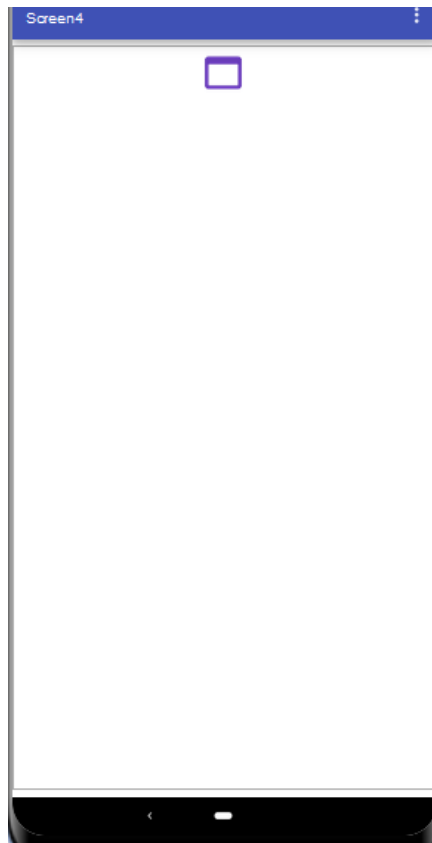
programación crear aplicaciones para dispositivos Android de manera sencilla y eficiente. A continuación, te proporcionaré información sobre las funcionalidades y características principales de Kodular Creator:

1. **Interfaz de Arrastrar y Soltar:** Una de las características más destacadas de Kodular Creator es su enfoque en la programación visual. Los usuarios pueden crear aplicaciones arrastrando y soltando componentes en una interfaz gráfica intuitiva en lugar de escribir código.
2. **Componentes Predefinidos:** Kodular ofrece una amplia variedad de componentes predefinidos que los usuarios pueden utilizar para construir sus aplicaciones. Estos componentes incluyen botones, etiquetas, imágenes, listas, bases de datos y muchos más.
3. **Bloques Lógicos:** Aunque Kodular está diseñado para ser sin código, utiliza bloques lógicos para permitir la personalización y la lógica en las aplicaciones. Los usuarios pueden conectar estos bloques para definir el comportamiento y las acciones de sus aplicaciones.
4. **Conexión a Bases de Datos:** Kodular permite la integración con bases de datos en tiempo real, lo que permite a los desarrolladores crear aplicaciones con características como inicio de sesión de usuarios, almacenamiento de datos y sincronización en tiempo real.
5. **Diseño Responsivo:** Las aplicaciones creadas en Kodular son automáticamente responsivas, lo que significa que se adaptarán automáticamente a diferentes tamaños de pantalla y dispositivos, como teléfonos y tabletas.
6. **Pruebas en Tiempo Real:** Los usuarios pueden ver cómo se ve y funciona su aplicación en tiempo real utilizando un emulador de Android incorporado en la interfaz de Kodular.
7. **Aplicaciones Nativas:** Las aplicaciones creadas con Kodular son aplicaciones nativas de Android, lo que significa que se ejecutan en el sistema operativo Android sin depender de navegadores web.
8. **Extensibilidad:** Kodular permite a los desarrolladores avanzados crear sus propias extensiones personalizadas, lo que les brinda la flexibilidad de agregar funcionalidades adicionales a sus aplicaciones.
9. **Comunidad y Soporte:** Kodular tiene una comunidad activa en línea donde los usuarios pueden hacer preguntas, compartir sus creaciones y obtener soporte de otros miembros de la comunidad y del equipo de Kodular.
10. **Distribución de Aplicaciones:** Una vez que se crea una aplicación en Kodular, los usuarios pueden exportar y distribuir sus aplicaciones a través de la tienda Google Play Store u otras plataformas de distribución de aplicaciones.

Kodular Creator es una herramienta poderosa para crear aplicaciones móviles de manera rápida y sencilla, lo que la hace ideal para personas que desean desarrollar aplicaciones sin tener experiencia en programación. Sin embargo, es importante tener en cuenta que, si bien Kodular permite crear aplicaciones sofisticadas, las aplicaciones más complejas y personalizadas pueden requerir un enfoque de desarrollo tradicional con código personalizado.



- En esta sección tenemos la parte inicial, es decir la “BIENVENIDA A NUESTRA APP”



- En esta sección estará conectada con la pagina web

Referencias:

- 1) Otto, M., Thornton, J., & Bootstrap contributors. *Bootstrap*. Getbootstrap.com. Recuperado el 13 de agosto de 2023, de <https://getbootstrap.com/>
- 2) Londoño, P. (2023). *Qué es Python, para qué sirve y cómo se usa (+ recursos para aprender)*. Hubspot.es. <https://blog.hubspot.es/website/que-es-python>
- 3) González, F. (2005). *¿Qué Es Un Paradigma? Análisis Teórico, Conceptual Y Psicolingüístico Del Término*. Investigación y posgrado, 20(1), 13–54. http://ve.scielo.org/scielo.php?script=sci_abstract&pid=S1316-00872005000100002&lng=es&nrm=iso&tlng=es
- 4) *Descargar PyCharm: IDE de Python para desarrolladores profesionales*. (s/f). JetBrains. Recuperado el 13 de agosto de 2023, de <https://www.jetbrains.com/es-es/pycharm/download/>