



**Universidad**  
**Europea** MADRID

PROYECTO INTEGRADOR

1DAW

DataCode++



Farías Morales, Daniel Rafael.

Gonzalez Garrote, Daniel.

Moreno García, Ignacio.

## Índice

Resumen.....	2
1. Introducción .....	3
2. Objetivos .....	4
Objetivos generales del proyecto.....	4
Objetivos específicos del Proyecto (vinculados a capacidades de los módulos). ....	4
3. Tecnologías utilizadas .....	5
4. Desarrollo e implementación .....	6
Modelo E-R.....	6
Modelo Relacional y Normalización. ....	6
Diseño de la Interfaz de las ventanas.....	8
Diseño del logo. ....	14
Diagrama de casos de uso. ....	15
Diagrama de clases. ....	16
5. Metodología.....	20
Roles.....	20
Eventos de Scrum .....	20
Aplicación Práctica en DataCode .....	20
Enlaces del proyecto en la nube.....	23
6. Resultados y conclusiones.....	24
7. Trabajos futuros.....	26
Anexos.....	28
Anexo I – Listado de requisitos de la aplicación .....	28
Especificaciones del Hardware. ....	28
Especificaciones del Software.....	29
Anexo II – Guía de uso de la aplicación .....	30

## Resumen

El proyecto integrador "DataCode++" desarrollado por Daniel Rafael Farías Morales, Daniel González Garrote e Ignacio Moreno García, dentro del contexto del Ciclo Formativo de Desarrollo de Aplicaciones Web, aborda la creación de una aplicación para gestionar partidas de rol. La aplicación se enfoca en la creación, modificación y visualización de personajes, así como en la gestión de partidas dirigidas por un GameMaster. Se utilizaron tecnologías como Eclipse para el desarrollo, MySQL para la gestión de bases de datos, y herramientas organizativas como Trello y GitHub para el trabajo en equipo.

Los objetivos del proyecto incluyen la gestión eficiente del tiempo, la autonomía en el trabajo colaborativo, la innovación y actualización en el ámbito laboral, la resolución de conflictos y la toma de decisiones individuales. Además, se buscan capacidades específicas en bases de datos, programación y entornos de desarrollo.

El proyecto se desarrolló siguiendo una metodología ágil, como Scrum, con una planificación detallada, representada en Sprint Backlog y Burndown chart. Se coordinó el equipo para finalizar el trabajo de manera eficiente.

Los resultados obtenidos permiten una gestión efectiva de personajes y partidas, con una interfaz gráfica intuitiva y funcionalidades diferenciadas según el rol de usuario. Como conclusiones, se destaca el cumplimiento de los objetivos y la mejora continua como equipo.

Como trabajos futuros, se plantea la implementación de mejoras adicionales en la aplicación, aprovechando el conocimiento adquirido y la retroalimentación recibida. Las palabras clave incluyen "desarrollo de aplicaciones web", "gestión de partidas de rol", "Scrum" y "colaboración en equipo".

## 1. Introducción

En el presente proyecto, denominado DataCode++, surge como una iniciativa conjunta para integrar los conocimientos adquiridos en distintos módulos de un Ciclo Formativo en el ámbito de la programación, bases de datos y entornos de desarrollo. Este documento presenta una visión detallada del proyecto, desde sus objetivos generales y específicos hasta las tecnologías utilizadas, el desarrollo e implementación del sistema, la metodología empleada, los resultados obtenidos, las conclusiones del grupo y las posibles mejoras futuras.

DataCode++ tiene como propósito principal proporcionar a los integrantes del grupo la oportunidad de gestionar y realizar el trabajo asignado en relación con las cargas propias del Ciclo Formativo, haciendo especial énfasis en la planificación del tiempo, la consulta adecuada a los compañeros, la innovación y actualización en el ámbito laboral, la gestión efectiva de conflictos, la resolución de problemas y la toma de decisiones dentro de su competencia.

El proyecto se estructura en torno a objetivos específicos vinculados a los módulos de bases de datos, programación y entornos de desarrollo. Se emplean tecnologías como Eclipse para el desarrollo de la aplicación, MySQL para la gestión de la base de datos, Trello para la gestión del trabajo en equipo y GitHub para la colaboración en el desarrollo del software.

El desarrollo e implementación del proyecto abarca desde la creación de modelos entidad-relación y relacionales hasta el diseño de la interfaz de usuario y la elaboración de diagramas de casos de uso y clases. Todo ello se lleva a cabo siguiendo una metodología ágil, con roles definidos como Product Owner, equipo y Scrum Máster, y eventos clave como sprint, reuniones de planificación y revisiones de sprint.

A lo largo de este documento, se detallan los resultados obtenidos, las conclusiones del grupo sobre el trabajo realizado y las posibles mejoras futuras que podrían implementarse con más tiempo o conocimiento del lenguaje. Se incluyen también anexos con listados de requisitos de la aplicación, especificaciones del hardware y software utilizados, y una guía de uso de la aplicación para facilitar su comprensión y utilización por parte de los usuarios.

## 2. Objetivos

### **Objetivos generales del proyecto.**

Mediante este trabajo se pretende que los integrantes del grupo consigan:

- Gestionar y realizar el trabajo del área asignada, en relación con las cargas de trabajo propias del Ciclo Formativo (planificación del tiempo).
- Realizar consultas a la persona adecuada y saber respetar la autonomía de los compañeros.
- Mantener el espíritu de innovación y actualización en el ámbito de su trabajo, buscando habilidades para la sinergia de equipos y liderazgo.
- Gestionar eficazmente los conflictos que se puedan producir, mediando y contribuyendo al establecimiento de un ambiente de trabajo agradable y actuando de forma respetuosa y tolerante.
- Resolver problemas y tomar decisiones individuales, siguiendo las normas y procedimientos establecidos, definidos dentro del ámbito de su competencia.

### **Objetivos específicos del Proyecto (vinculados a capacidades de los módulos).**

#### **Bases de datos.**

- Crear bases de datos definiendo su estructura y las características de sus elementos según el modelo relacional.
- Diseñar modelos lógicos normalizados interpretando diagramas entidad/relación.
- Realizar el diseño físico de bases de datos utilizando asistentes, herramientas gráficas y el lenguaje de definición de datos.
- Consultar y modificar la información almacenada utilizando asistentes, herramientas gráficas y el lenguaje de manipulación de datos.

#### **Programación.**

- Realizar el diseño de clases necesarias para seguir el patrón MVC.
- Implementar cada una de las clases para lograr su objetivo: clases de interfaz gráfica pertenecientes a la vista, clases que representen el modelo y clases encargadas de la lógica de la aplicación pertenecientes al control.
- Desarrollar una aplicación que gestione información almacenada en bases de datos relacionales identificando y utilizando mecanismos de conexión.

Esta tarea permitirá alcanzar la competencia completa para desarrollar componentes software en lenguajes de programación orientados a objetos.

#### **Entornos de Desarrollo.**

- Realizar el análisis y el diseño de cualquier aplicación empleando técnicas UML.
- Documentar aplicaciones.
- Gestionar las diferentes versiones de un software y el trabajo colaborativo.
- Realizar pruebas de testeo sobre los programas.

- Seguir metodologías ágiles para el desarrollo y planificación de programas.

### **3. Tecnologías utilizadas**

Para este apartado se recomienda una extensión de 2 a 3 páginas. Enumeración de las tecnologías, herramientas y lenguajes de programación utilizados durante el desarrollo del proyecto. Enumeración de las herramientas organizativas para favorecer el trabajo en equipo.

#### **Eclipse.**

Es una plataforma de software compuesto por un conjunto de herramientas de programación de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores

#### **MySQL.**

Es un sistema de administración de bases de datos relacionales. Es un software de código abierto desarrollado por Oracle. Se considera como la base de datos de código abierto más utilizada en el mundo.

#### **Trello.**

Es una herramienta flexible para la gestión del trabajo, con la que los equipos pueden diseñar planes, colaborar en proyectos, organizar flujos de trabajo y hacer un seguimiento del progreso de una manera visual, productiva y gratificante

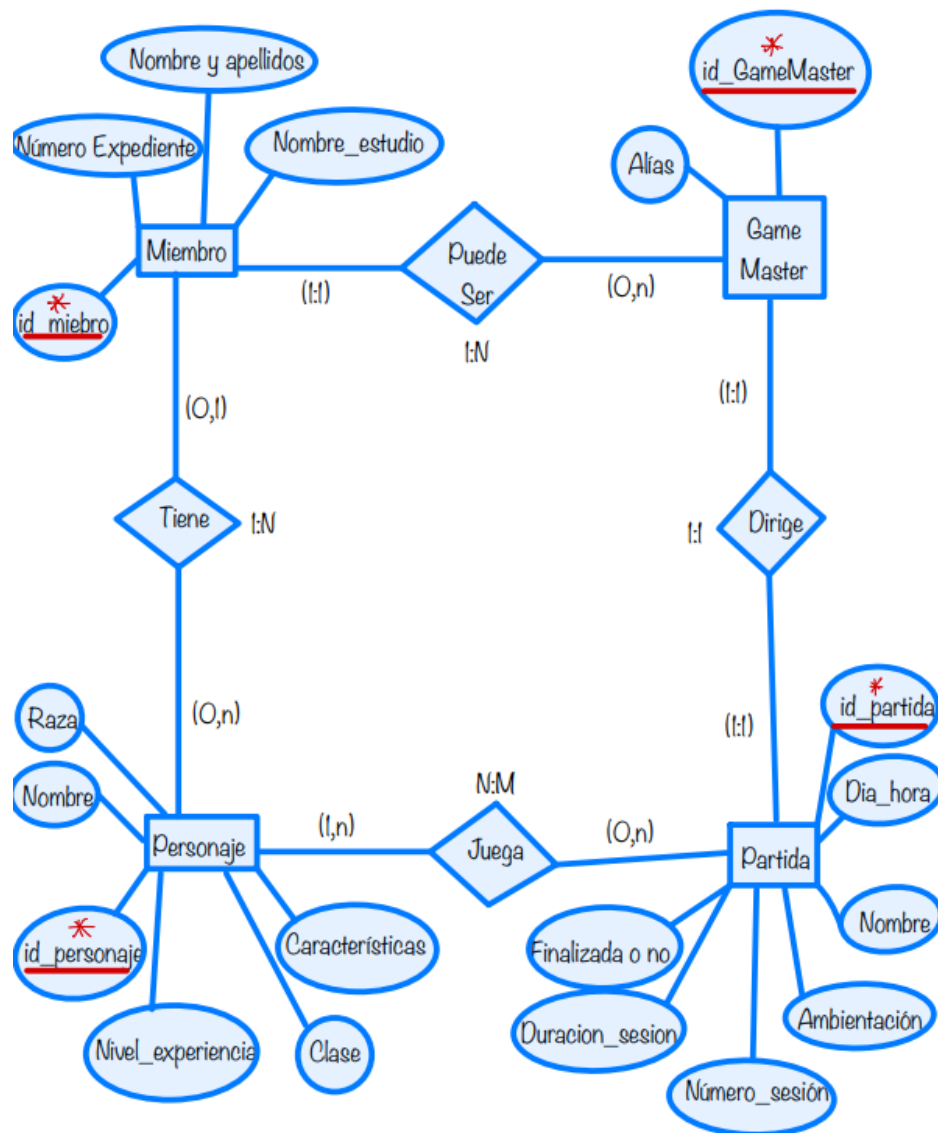
#### **GitHub.**

Es una forja para alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente de programas de ordenador. El software que opera GitHub fue escrito en Ruby on Rails. Desde enero de 2010, GitHub opera bajo el nombre de GitHub, Inc.

- -MySQL: Para la gestión de la base de datos del sistema
- -GitHub: Para la gestión de versiones del código fuente y colaboración en el desarrollo del software

## 4. Desarrollo e implementación

### Modelo E-R.



En base al juego de rol se ha realizado la creación de un modelo entidad relación para el comienzo del proyecto, en el cual tenemos cuatro (4) relaciones.

1. **Miembro:** el cual representa cual quiere individuo que participe en el juego.
2. **Personaje:** es la representación de un miembro en el juego.
3. **Game Máster:** es un miembro del juego y su rol es dirigir una partida de rol.
4. **Partida:** realizara la acción principal del juego, puede tener múltiples personajes.

### Modelo Relacional y Normalización.

#### 1. Modelo Relacional.

En base al modelo entidad-relación se debe crear el modelo relacional para poder comenzar con la creación de las tablas de la base de datos. Para ello

tendremos cinco (5) tablas, de las cuales cuatro (4) son de las entidades y una es de una relación. Las tablas con sus atributos son los siguientes:

**Tabla Miembro.**

- id\_miembro. (Clave Primaria).
- Nombre y apellidos.
- Número\_Expediente.
- Nombre\_Estudio

**Tabla GameMaster.**

- id\_GameMaster. (Clave Primaria).
- Alias.
- \*id\_miembro\* (Clave foránea, referencia a la tabla Miembro).

**Tabla Personaje.**

- id\_personaje. (Clave Primaria).
- Nombre.
- Raza
- Nivel\_experiencia
- Clase.
- \*id\_miembro\* (Clave foránea, referencia a la tabla Miembro).

**Tabla Partida.**

- id\_partida. (Clave Primaria).
- Nombre.
- Dia\_hora.
- Número\_sesión.
- Ambientación.
- Finalizada.
- Duración\_sesión.
- \*id\_GameMaster\* (Clave foránea, referencia a la tabla GameMaster).

**Tabla Juega.**

- id\_personaje. (Clave Primaria).
- id\_partida. (Clave Primaria).
- Caracteristicas\_personaje.



En esta última tabla obtenemos dos claves primarias, por lo cual hace referencia a distintas tablas, las cuales son la tabla personaje que alberga la clave foránea id\_personaje, y la tabla partida que contiene la clave foránea id\_partida.

## 2. Normalización.

En el modelo relacional que se ha descrito, existen distintas razones por las cuales se ha optado por no normalizar, esas razones son:

- a. Normalizar estas tablas podría resultar un diseño más complejo y consultas más complicadas, lo que podría afectar el rendimiento del sistema y la facilidad de uso para los usuarios.
- b. Una estructura de tablas menos normalizada puede ser más fácil de entender y mantener, especialmente para usuarios menos familiarizados con el modelo relacional de base de datos. Esto puede ser importante si la prioridad es la facilidad de uso.

La normalización es una técnica valiosa para garantizar la integridad y eficiencia de una base de datos. De igual manera explicaremos porque no es necesario realizar las tres (3) primeras formas de normalización.

1. **Primera forma (1 FN):** esta primera forma indica que cada columna de la tabla debe contener un único valor y no múltiples valores. En dichas tablas no hay columnas que puedan albergar más de un valor.
2. **Segunda forma (2 FN):** la segunda forma de normalización establece que una tabla debe estar en 1FN y que cada columna que no sea clave solo dependa de la clave primaria y no de otras columnas adicionales. Cada tabla tiene una clave primaria y todas las columnas no clave están relacionadas funcionalmente con la clave primaria respectiva.
3. **Tercera forma (3 FN):** en ella establece que una tabla debe estar en 2FN y que no debe haber dependencias transitivas entre las columnas no clave, es decir, que, si una columna depende de otra que a su vez depende de la clave primaria, entonces la columna debería trasladarse a otra tabla.

## **Diseño de la Interfaz de las ventanas.**

En la aplicación de DataCode++, se ha creado diez (10) interfaces gráficas fundamentales que facilitarán la ejecución del programa. Estas interfaces solamente son el primer diseño, es decir, puede que resulte no ser el resultado final. A continuación, se detallará el contenido y las funciones disponibles en cada una de estas ventanas.

### 1. Ventana "Inicio Sesión".

Al ejecutar el programa, lo primero que se encuentra es el inicio de sesión, donde el usuario puede interactuar para ingresar con su nombre de usuario y contraseña. Además, puede seleccionar entre las opciones "Jugador" y "Game Master" para acceder a la aplicación, cuando el usuario pulse en el botón de "Entrar".

Dependiendo del tipo de usuario seleccionado, las opciones disponibles en las ventanas siguientes variarán.



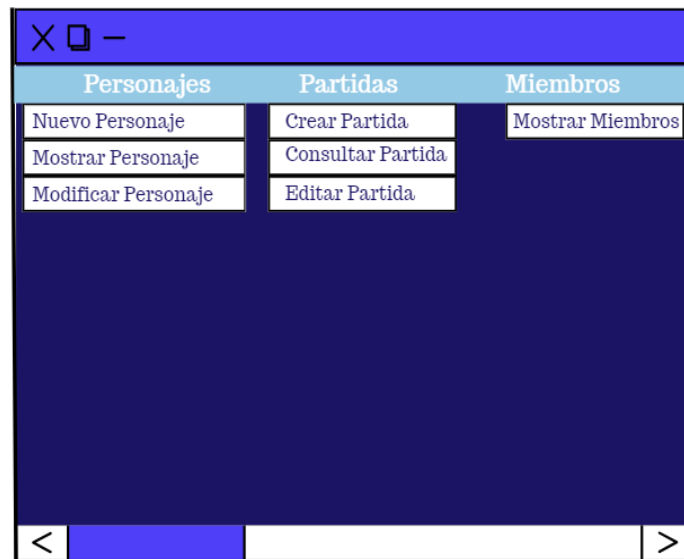
## 2. Ventana GameMaster.

En la ventana de inicio de sesión la aplicación te preguntara como deseas entrar como usuario o como GameMaster. Pero si no tienes un alias creado como GameMaster el sistema te mostrara una ventana nueva en la cual podrás rellenar con tu alias favorito.



## 3. Menú Superior.

El menú exhibe todas las opciones disponibles con el que el usuario o GameMaster puede interactuar, en ella solo se verán tres (3) opciones, de las cuales son desplegables en la interfaz. Cada una de estas opciones dirige al usuario a la ventana correspondiente, permitiéndole ejecutar la función deseada.



#### 4. Ventana Menú Principal.

En la ventana del menú principal, se encuentra una breve introducción al juego, además aparecerá el logo en la parte izquierda de la pantalla, y también el menú antes mencionadas anteriormente.



#### 5. Ventana Mostrar Personajes.

En esta ventana el usuario o GameMaster podrá visualizar un listado con los personajes que tiene en el juego. Además, se incluyen tres botones, eliminar, aceptar y consultar las stats, de los personajes.



#### 6. Ventana Nuevo Personaje.

En ella, como su nombre lo dice, podrás crear nuevos personajes. Para ello la venta te indica que datos necesitara para crear el personaje. Simplemente se deberán rellenar cada campo con la información que el sistema indique y presionando el botón de aceptar, se crea el personaje.

The screenshot shows a window titled 'Nuevo Personaje' with a blue header bar containing three tabs: 'Personajes', 'Partidas', and 'Miembros'. The 'Personajes' tab is active. Below the header, the text 'Nuevo Personaje:' is centered. The form contains five input fields, each preceded by a label: 'ID Personaje:', 'Nombre Personaje:', 'Raza:', 'Clase:', and 'Nivel de experiencia:'. At the bottom right of the form is a button labeled 'Aceptar'. The window has a standard Windows-style title bar with a close button (X), a maximize button, and a minus button (-). At the bottom of the window is a status bar with a left arrow, a blue progress bar, and a right arrow.

#### 7. Ventana Modificar Personaje.

Para esta ventana, se podrá modificar las características de los personajes, si es que en algún momento el usuario se ha equivocado a la hora de introducir la información requerida para la creación de un personaje o, si el usuario desea modificar algún campo en específico.

#### 8. Ventana Crear Partida.

El usuario podrá ver la pantalla de “Crear Partida”, pero solamente el GameMaster será el encargado de rellenar los campos necesarios para que esa partida se cree. Simplemente es rellenar los datos y presionar el botón de “Crear Partida”.

#### 9. Ventana Editar Partida.

De una manera similar a la ventana anterior, en esta ventana se podrá editar las partidas. Es importante recordar, que solo podrá rellenar los campos el GameMaster.

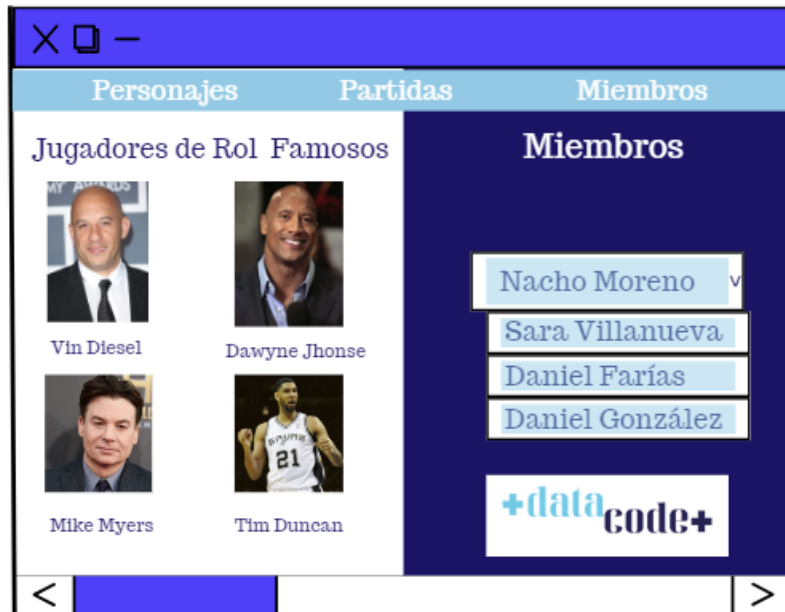
### 10. Venta Consultar Partida.

Y la última opción en partida es la de “Consultar Partida”, en esta ventana se visualizará un listado con todas las partidas, incluyendo toda la información relevante que lleva una partida. Para esta opción se puede visualizar tanto el usuario común como el GameMaster.

Identificador	Nombre	Día	Nº Sesión	Ambientación	Finalizada (Si/No)	Duración	ID Game Master
1	La búsqueda del anillo	2024-04-01	1	Tierra Media	No	4	99
2	Galaxia en guerra	2024-04-02	1	Galaxia	No	3	99
3	Las sombras de la montaña	2024-04-03	1	Montañas	No	5	99
4	La magia perdida	2024-04-04	1	Bosque encantado	No	4	99
5	El rescate de la princesa	2024-04-05	1	Castillo	No	3	99
6	Intriga en la corte	2024-04-06	1	Palacio real	No	5	99
7	La ciudad maldita	2024-04-07	1	Ciudad abandonada	No	4	99
8	El laberinto de la muerte	2024-04-08	1	Laberinto	No	6	99

### 11. Ventana Miembros del Club.

En ella se muestra en el lado derecho un listado con todos los miembros que hacen parte de la aplicación, y en la parte izquierda un cartel con famosos que también juegan rol.



### Diseño del logo.



### Colores.

1. Azul Oscuro: Suele asociarse con la confianza, la profesionalidad y la seriedad. Puede transmitir estabilidad y fiabilidad, lo que es importante para muchas empresas.
2. Azul Celeste: Este tono de azul más claro puede evocar calma, tranquilidad y frescura. También puede sugerir una sensación de apertura y claridad.
3. Blanco: El blanco a menudo se asocia con la pureza, la limpieza y la simplicidad. También puede sugerir espacio, claridad y neutralidad.

En conjunto, esta combinación de colores podría transmitir un mensaje de confianza y profesionalismo (azul oscuro), junto con una sensación de calma y

frescura (azul celeste) y una imagen de limpieza y claridad (blanco). Dependiendo del tipo de negocio y su público objetivo, esta combinación de colores podría ser adecuada para transmitir una variedad de mensajes, desde la fiabilidad hasta la modernidad o la frescura. Es importante considerar el contexto y la audiencia específica al interpretar el significado de los colores en un logo comercial.

"DataCode++" como nombre del proyecto integrador, que abarca los módulos de programación, bases de datos y entornos de desarrollo, sugiere una combinación de dos elementos principales: "Data" y "Code".

1. Data (Datos): Este término hace referencia a la información que se maneja en el proyecto, lo que sugiere un enfoque en la gestión, manipulación y almacenamiento de datos. Esto podría implicar el uso de bases de datos para almacenar información de manera estructurada y accesible.
2. Code (Código): Este término implica una relación con la programación y el desarrollo de software. Sugiere que el proyecto involucra la creación y manipulación de código informático, lo que puede abarcar desde la escritura de aplicaciones hasta la implementación de algoritmos y la creación de interfaces de usuario.

El proyecto "DataCode" sugiere que se centra en la intersección entre la gestión de datos y el desarrollo de software. Esto podría implicar la creación de aplicaciones que manipulan datos de diversas fuentes, el diseño de sistemas de gestión de bases de datos, o el desarrollo de herramientas para análisis y visualización de datos.

En resumen, el nombre "DataCode" transmite la idea de un proyecto que combina la manipulación de datos con la programación y el desarrollo de software.

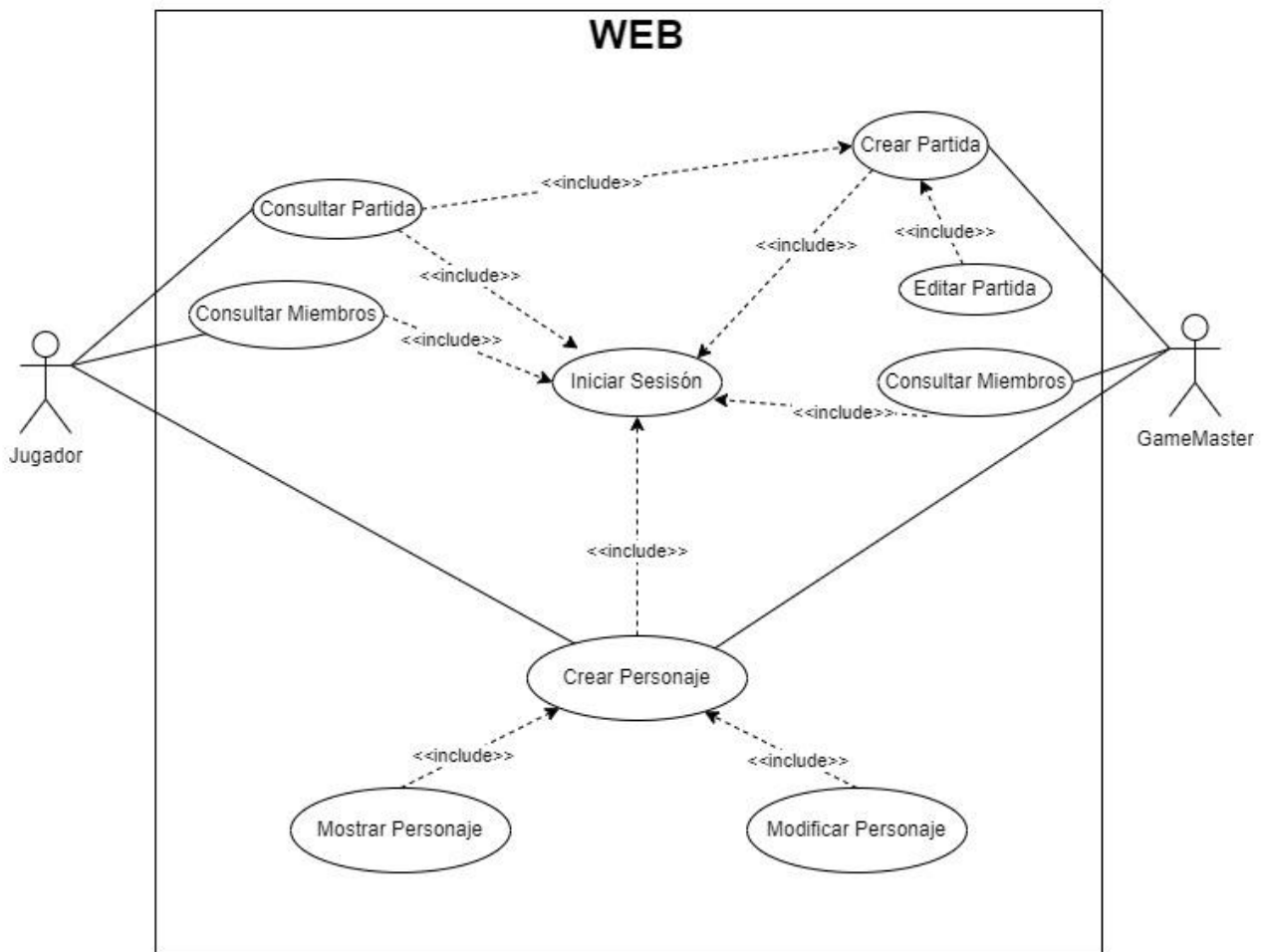
### **Diagrama de casos de uso.**

Se ha realizado el diagrama de casos de uso, en función a las diferentes acciones que pueden realizar los personajes durante el transcurso del juego de rol. Existen dos formas de poder interactuar con la aplicación y son:

1. Como Jugador, que el mismo puede crear personajes, consultar miembros y consultar partidas.
2. Como GameMaster, que puede crear personajes, al igual que el Jugador, que, a su vez, puede mostrar personajes y modificar dichos personajes (indicado con relación "include" en el diagrama).



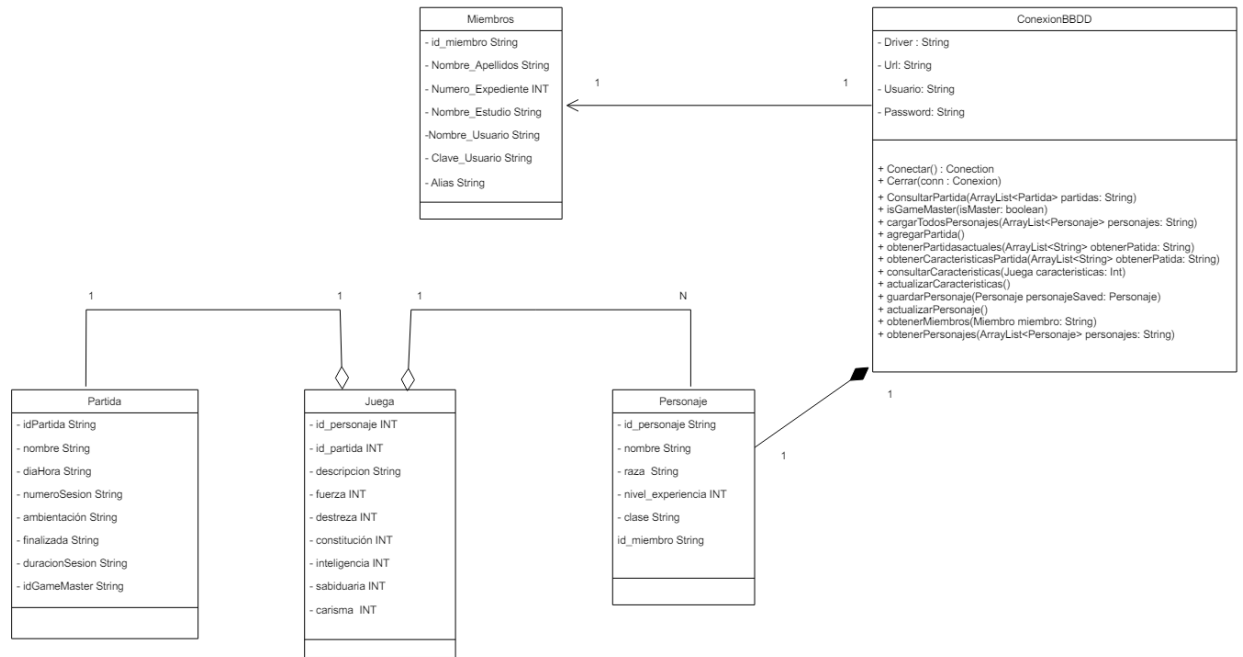
Tanto Jugador como GameMaster, además de poder crear personajes (lo que incluye mostrar y modificar dichos personajes) pueden iniciar sesión. Todas las



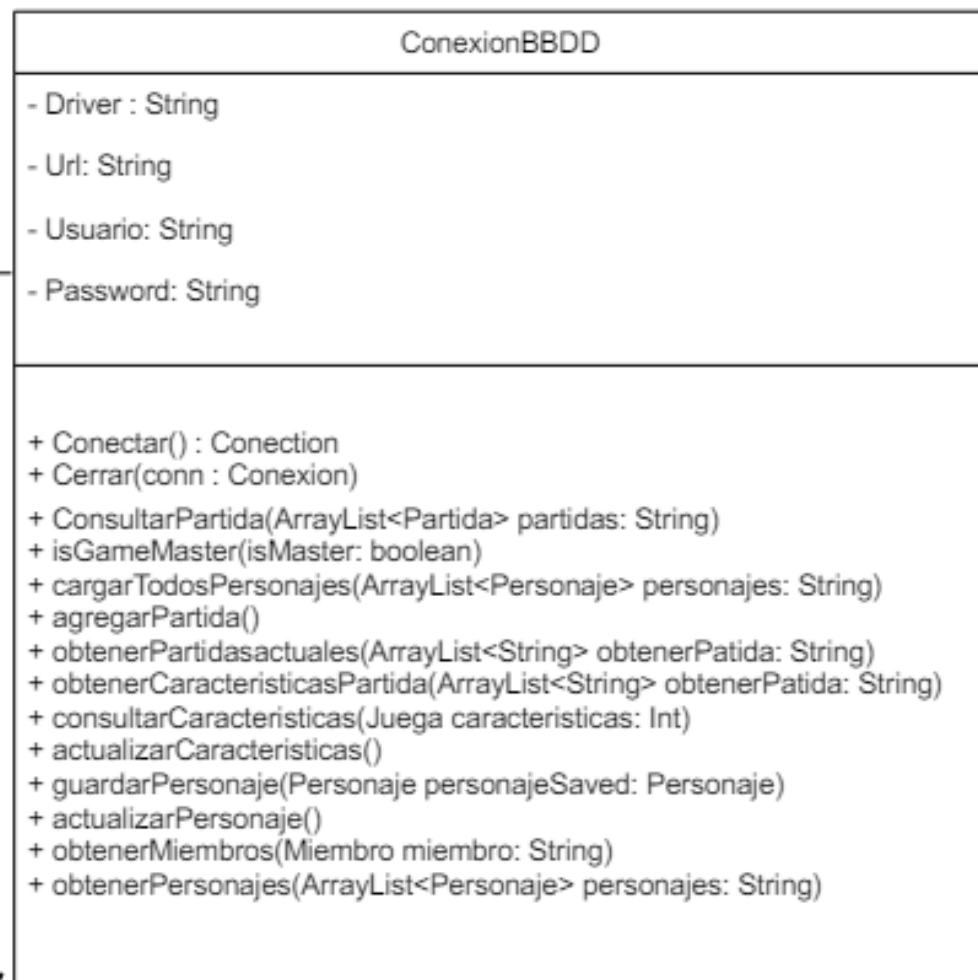
acciones están incluidas en Iniciar sesión, ya que, para realizar cualquier acción, primero debemos iniciar sesión, es decir, si no iniciamos sesión en el juego, no podemos realizar nada más, como por ejemplo crear partidas o editarlas.

### Diagrama de clases.

El diagrama muestra la estructura de clases y métodos de una aplicación de gestión de partidas de rol (RPG) en línea, junto con sus atributos y relaciones. Este diagrama proporciona una visión general de la estructura y funcionalidad de la aplicación de gestión de partidas de rol en línea. Cada clase representa una entidad principal en el sistema y encapsula su comportamiento y datos asociados.



## ■ ConexiónBBDD.



## ▪ Miembros

Miembros
- id_miembro String
- Nombre_Apellidos String
- Numero_Expediente INT
- Nombre_Estudio String
-Nombre_Usuario String
- Clave_Usuario String
- Alias String

## ▪ Partida

Partida
- idPartida String
- nombre String
- diaHora String
- numeroSesion String
- ambientación String
- finalizada String
- duracionSesion String
- idGameMaster String

## ▪ Juega

Juega
- id_personaje INT
- id_partida INT
- descripcion String
- fuerza INT
- destreza INT
- constitución INT
- inteligencia INT
- sabiduria INT
- carisma INT

▪ **Personaje**

Personaje
- id_personaje String
- nombre String
- raza String
- nivel_experiencia INT
- clase String
id_miembro String

## 5. Metodología

### Roles

1. Product Owner: En el proyecto DataCode, los roles de Product Owner son desempeñados por Sara e Irene. Su responsabilidad principal es definir y priorizar las funcionalidades del producto, representando así los intereses y necesidades del cliente.
2. Team: El equipo está compuesto por Ignacio Moreno, Daniel Farias y Daniel González. Son los encargados de desarrollar las funcionalidades definidas por el Product Owner durante cada sprint, garantizando la entrega de un producto de calidad.
3. Scrum Máster: Daniel Farias asume el rol de Scrum Máster, siendo responsable de asegurar que el equipo comprenda y aplique los principios de Scrum correctamente. Facilita la resolución de problemas y elimina los obstáculos que puedan afectar el progreso del equipo.

### Eventos de Scrum

1. Sprint: Los sprints en el proyecto DataCode tienen una duración de dos semanas cada uno. Durante este período, el equipo se compromete a completar una cantidad específica de trabajo, basado en las prioridades definidas por el Product Owner.
2. Reunión de Planificación del Sprint: Antes de comenzar cada sprint, el equipo se reúne con el Product Owner para planificar el trabajo a realizar. Utilizando herramientas como la aplicación Trello, se definen las tareas y se asignan a los miembros del equipo, teniendo en cuenta la capacidad de trabajo y las prioridades del Product Owner.
3. Revisiones de Sprint: Al final de cada sprint, se lleva a cabo una reunión de revisión. En el caso de DataCode, estas revisiones se realizan cada dos lunes. Durante esta reunión, el equipo presenta el trabajo completado al Product Owner, recibe feedback sobre dicho proyecto y realiza ajustes según sea necesario para futuros sprints.

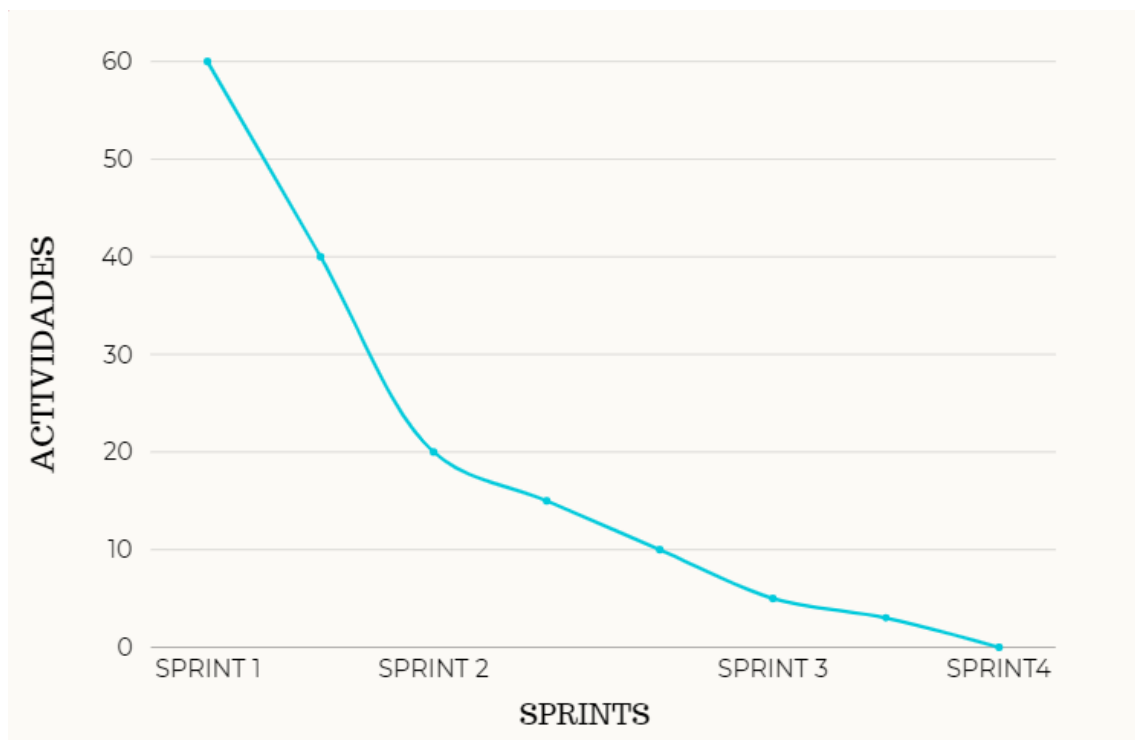
### Aplicación Práctica en DataCode

Durante el proyecto DataCode, Scrum ha proporcionado un marco de trabajo ágil que ha permitido al equipo adaptarse rápidamente a los cambios y prioridades del cliente. La comunicación efectiva entre el Product Owner y el equipo, facilitada por el

Scrum Máster, ha garantizado una alineación constante con los objetivos del proyecto.

La estructura de los sprints de dos semanas ha proporcionado un ritmo constante de entrega, lo que ha permitido al equipo mantenerse enfocado en metas alcanzables a corto plazo. Además, la aplicación de reuniones de planificación y revisiones de sprint ha fomentado la transparencia y la colaboración entre todos los miembros del equipo.

En resumen, la implementación de la metodología Scrum en el proyecto DataCode ha demostrado ser efectiva para gestionar la complejidad de integrar múltiples asignaturas en un proyecto de desarrollo de software. Ha permitido una entrega iterativa y continua de valor, asegurando la satisfacción del cliente y la calidad del producto final.



En este diagrama Burndown chart, se representa las actividades que se han realizado en cada uno de los diferentes sprints, indicando así una actividad que va disminuyendo hasta llegar a 0 que es cuando se completa el trabajo y por tanto la entrega de la actividad. A continuación, se explica detalladamente que tipo de

Sprint #1:

Este sprint marca el inicio del proyecto DataCode y se alinea con el evento de la Reunión de Planificación del Sprint. Durante esta reunión, el equipo se compromete a completar las tareas definidas por el Product Owner para este período.

Las actividades como la presentación del módulo, análisis y diseño de requisitos, y la planificación detallada se corresponden con la responsabilidad del Product Owner de definir y priorizar las funcionalidades del producto.

El uso de herramientas como GitHub y Trello para la planificación y seguimiento del trabajo refleja la adopción de prácticas ágiles para facilitar la colaboración y la transparencia en el equipo.

#### Sprint #2:

Durante este sprint, el equipo se centra en la creación del proyecto y el diseño de la interfaz, así como en el desarrollo de la base de datos. Estas actividades están alineadas con el compromiso del equipo de entregar un incremento potencialmente entregable al final de cada sprint.

La realización del diagrama de casos de uso y el diseño del logo también reflejan la colaboración entre el Product Owner y el equipo para asegurar que el producto final satisfaga las necesidades y expectativas del cliente.

#### Sprint #3:

En este sprint, el equipo se enfoca en la generación del diagrama de clases y la creación de las clases correspondientes al modelo y al control del sistema. Estas actividades demuestran el compromiso del equipo de desarrollar un producto de calidad, siguiendo las mejores prácticas de diseño y arquitectura de software.

#### Sprint #4:

El último sprint del proyecto se dedica al refinamiento del programa, la realización de pruebas y la integración de todos los elementos. Estas actividades reflejan el enfoque de Scrum en la entrega continua de valor y la mejora continua del producto a lo largo del tiempo.

La realización del manual de usuario en la wiki de GitHub en este sprint demuestra el compromiso del equipo de garantizar una buena experiencia de usuario y facilitar la adopción del producto por parte de los usuarios finales.

En resumen, el diagrama de Burndown chart, muestra cómo las actividades de cada sprint en el proyecto DataCode están alineadas con los principios y eventos clave de la metodología Scrum, permitiendo una gestión efectiva del proyecto y una entrega iterativa y continua de valor al cliente. La implementación de Scrum ha facilitado la colaboración, la transparencia y la adaptación a los cambios y prioridades del cliente a lo largo del proyecto.

### **Enlaces del proyecto en la nube.**

En dicho apartado encontraras los enlaces que te llevaran al proyecto para el cual se utilizó GitHub, además se incluye la organización de los sprint en Trello (en el apartado de especificaciones del software podrás encontrar que es GitHub y Trello).

- **GitHub:**

<https://github.com/danielfariasmo/DataCode->

- **Trello:**

<https://trello.com/invite/b/uKghl6wk/ATTI9ba53f18fd762247954989021efac6c16A65D15E/datacode>



## 6. Resultados y conclusiones

El proyecto se enfocó en alcanzar tanto objetivos generales como específicos. En términos generales, buscábamos mejorar nuestra capacidad de gestionar cargas de trabajo, consultar de manera efectiva, fomentar la innovación y habilidades de resolución de conflictos, todo dentro de un marco de respeto y tolerancia. Específicamente, nos propusimos alcanzar habilidades en áreas como bases de datos, programación y entornos de desarrollo, para desarrollar componentes de software y aplicaciones de manera efectiva.

En relación con los objetivos generales del proyecto, hemos logrado gestionar nuestras cargas de trabajo de manera más eficiente, respetando la autonomía de los compañeros y promoviendo un ambiente de trabajo positivo. Además, hemos fortalecido nuestras habilidades de resolución de conflictos y toma de decisiones, lo que contribuye a un ambiente laboral más armonioso y productivo.

En cuanto a los objetivos específicos relacionados con bases de datos, hemos adquirido la capacidad de crear bases de datos, diseñar modelos lógicos y físicos, y consultar y modificar información almacenada de manera efectiva. Estas habilidades son fundamentales para el desarrollo de aplicaciones que requieran interacción con bases de datos relacionales.

En el ámbito de la programación, hemos desarrollado habilidades para diseñar clases siguiendo el patrón MVC, implementar estas clases de manera efectiva y desarrollar aplicaciones que gestionen información almacenada en bases de datos relacionales. Estas habilidades son esenciales para el desarrollo de componentes de software orientados a objetos y para la creación de aplicaciones robustas y eficientes.

En cuanto a los entornos de desarrollo, hemos aprendido a realizar análisis y diseño de aplicaciones utilizando técnicas UML, documentar aplicaciones de manera adecuada, gestionar versiones de software y realizar pruebas de testeo. Además, hemos seguido metodologías ágiles para el desarrollo y planificación de programas, lo que nos ha permitido adaptarnos de manera más efectiva a los cambios y requerimientos del proyecto.

En resumen, el proyecto nos ha permitido adquirir habilidades y conocimientos fundamentales en áreas clave como bases de datos, programación y entornos de

desarrollo, que son esenciales para el desarrollo de componentes de software y aplicaciones eficientes. Además, hemos aprendido a trabajar de manera colaborativa utilizando herramientas y tecnologías modernas, lo que nos ha permitido alcanzar nuestros objetivos de manera efectiva.

## 7. Trabajos futuros

En el caso de contar con más tiempo o un mayor conocimiento del lenguaje de programación y las tecnologías utilizadas, el proyecto DataCode++ podría beneficiarse de diversas mejoras y ampliaciones en diferentes áreas. A continuación, se detallan algunas de las próximas mejoras que podrían implementarse:

### 1. Mejoras en la interfaz de usuario:

- Refinamiento del diseño de las ventanas y elementos de la interfaz para mejorar la experiencia del usuario.
- Implementación de animaciones y transiciones suaves para hacer la navegación más atractiva y fluida.
- Personalización de temas y estilos para permitir a los usuarios ajustar la apariencia según sus preferencias.

### 2. Funcionalidades adicionales:

- Incorporación de funcionalidades de mensajería o chat en tiempo real entre los jugadores y el GameMaster durante las partidas.
- Desarrollo de un sistema de notificaciones para informar a los usuarios sobre eventos importantes, como la creación de nuevas partidas o cambios en los personajes.

### 3. Optimización del rendimiento y escalabilidad:

- Implementación de técnicas de optimización de código y consultas de base de datos para mejorar la velocidad y eficiencia de la aplicación, especialmente cuando se trabaja con grandes volúmenes de datos.
- Evaluación y selección de soluciones de alojamiento y almacenamiento en la nube para garantizar la escalabilidad y disponibilidad del sistema ante un aumento en el número de usuarios o la carga de trabajo.

### 4. Integración de herramientas de análisis y seguimiento:

- Incorporación de herramientas de análisis de datos para recopilar información sobre el uso de la aplicación, el comportamiento de los usuarios y el rendimiento del sistema.
- Implementación de paneles de control y tableros de visualización para presentar de manera clara y concisa los datos recopilados, permitiendo al equipo tomar decisiones informadas sobre futuras mejoras y actualizaciones.

### 5. Expansión de la comunidad y funciones sociales:

- Desarrollo de funcionalidades de redes sociales dentro de la aplicación, como perfiles de usuario, seguimiento de amigos y grupos de interés, para fomentar la interacción y la participación de la comunidad.
- Organización de eventos especiales, torneos o competiciones dentro de la aplicación para promover la participación y el compromiso de los usuarios.

#### **6. Integración de sistemas de pago y monetización:**

- Incorporación de opciones de monetización, como suscripciones premium, compras dentro de la aplicación o publicidad, para generar ingresos y financiar el desarrollo continuo del proyecto.
- Implementación de un sistema de recompensas o puntos para incentivar la participación de los usuarios y aumentar la retención.

Estas son solo algunas de las posibles mejoras que podrían considerarse para el proyecto DataCode en el futuro. La implementación de estas mejoras dependerá de los recursos disponibles, las necesidades de los usuarios y los objetivos del equipo de desarrollo.

## Anexos

### Anexo I – Listado de requisitos de la aplicación

#### Especificaciones del Hardware.

Para el desarrollo del proyecto, se ha optado por utilizar las herramientas y aplicaciones integradas en el entorno Eclipse, que ofrece una plataforma sólida y versátil para el diseño y programación de la aplicación. Además, se ha empleado MySQL como sistema de gestión de bases de datos, aprovechando su eficiencia para el manejo de grandes volúmenes de datos.

- Especificaciones para Eclipse.

Como se menciona en el apartado de especificaciones del software se utilizará la herramienta de Eclipse para la elaboración de la aplicación. Esta elección permite diseñar una interfaz dinámica, facilitando la interacción del usuario con la aplicación y mejorando la experiencia de usuario final.

Eclipse proporciona herramientas integradas para el desarrollo de código, depuración y prueba, optimizando así el flujo de trabajo y permitiendo la implementación de funcionalidades clave de la aplicación. Además, su amplia variedad de plugins ofrece la flexibilidad de incorporar funcionalidades adicionales y personalizar el entorno de desarrollo según nuestras necesidades específicas.

Los requisitos mínimos y recomendados para el uso de Eclipse son los siguientes:

- SO: XP
- Procesador: Core 2 Duo.
- Memoria: 2 GB de RAM
- Gráficos: 9800 GT, o superior
- Almacenamiento: 1 GB de espacio disponible

Requisitos recomendados:

- SO: XP
- Procesador: Core i5
- Memoria: 4 GB de RAM
- Gráficos: 280 GTX
- Red: Conexión de banda ancha a Internet
- Almacenamiento: 2 GB de espacio disponible
- Especificaciones para MySQL.

De igual forma, para el almacenamiento y gestión de datos de la aplicación, se ha integrado MySQL como sistema de gestión de bases de datos. Ya que ofrece una solución fiable y escalable para manejar grandes volúmenes de datos de manera eficiente, garantizando la integridad y seguridad de la información almacenada.

Los requerimientos básicos para el uso de MySQL son los siguientes:

- 512 Mb de memoria RAM.

- 1024 Mb máquina virtual.
- 1 GB de espacio de disco duro.
- Sistema operativo: Windows, Linux y Unix.
- Arquitectura del sistema 32/64 bit.
- Protocolo de red TCP/IP.

Es importante señalar que, si las especificaciones del equipo son superiores a las indicadas, se podrá hacer uso de MySQL sin ningún tipo de problema e incluso mejorando el rendimiento del sistema.

### **Especificaciones del Software.**

Para el análisis y diseño del software del proyecto, así como su planificación detallada, es necesario seguir unas pautas y un proceso estructurado en referencia al software necesario:

#### Características Generales.

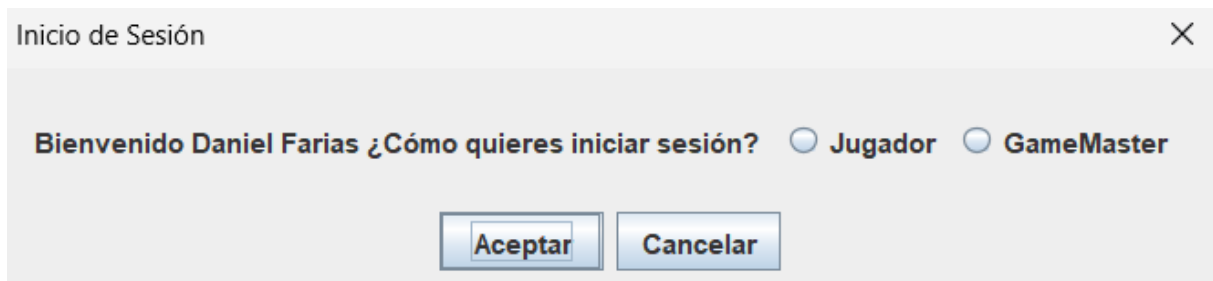
1. Gestión de usuarios:
  - La aplicación debe permitir a los usuarios registrarse e iniciar sesión con un usuario y contraseña.
  - Es importante destacar entre usuarios “jugadores” y “usuarios” GameMaster (GM), ya que cada rol, llevará unos permisos específicos y diferentes.
2. Gestión de Personajes:
  - Crear, Leer, Actualizar, y Eliminar personajes.
  - Cada personaje debe tener un identificador único, nombre, jugador asociado, raza, clase, nivel de experiencia y características como fuerza, destreza, constitución, inteligencia, sabiduría y carisma.
3. Gestión de partidas:
  - Cada partida debe tener al igual que los personajes, un identificador único, nombre, GameMaster asociado, ambientación, lista de personajes-participantes, día y hora del juego, duración de la sesión, número de sesión y estado de la partida (en curso/ terminada)
  - Interfaz de Usuario:
  - Pantalla de inicio de sesión.
  - Menú con opciones de consultas, altas, bajas y modificación para los usuarios.
  - Permisos diferenciados para los distintos usuarios:
    - Los jugadores pueden editar y eliminar sus propios personajes, crear nuevos personajes y consultar cualquier personaje en el sistema.
    - Los jugadores pueden consultar cualquier partida, pero no editarla.
    - Los GM pueden consultar cualquier partida y editar todas aquellas que esté dirigiendo.
    - Los GM pueden crear nuevas partidas

## Anexo II – Guía de uso de la aplicación

Al iniciar la aplicación lo primero que se verá es una pantalla de inicio de sesión, el cual deberás rellenar con tu nombre de usuario y la contraseña.



Luego pulsarás entrar y el sistema te preguntara como deseas entrar, hay dos opciones como gameMaster o como jugador.



### Jugador:

Al escoger esta opción, se abrirá una ventana con una pequeña introducción y también un menú principal que tendrá 5 apartados distintos en el cual podrás navegar en ellos.



En el apartado de **personajes**, podrás:

- Crear nuevos personajes, simplemente debes rellenar el nombre que deseas tener y rellenar los campos solicitados.

- Mostrar personajes, el cual tendrás una lista con todos tus personajes y en el podrás eliminarlos si deseas.



- Editar personaje, es similar a la de crear personaje, pero en ella deberás seleccionar el personaje que desees modificar y cambiarle las características que desees.

En el apartado **partidas**, podrás:

- Consultar partidas, en el cual puedes visualizar todas las partidas disponibles y en el botón de información de partidas al seleccionar una partida, podrás ver más información de ella y poder unirte a esas partidas seleccionando el personaje con el que desees jugar.

**Consultar Partida**

Nombre	Día y hora de Creación	Duración Sesión	Ambientación	Finalizada (Si/No)
Galaxia en guerra	2024-05-22 15:00:00	3	Galaxia	No
Las sombras de la montaña	2024-05-22 21:00:00	5	Montañas	No
La magia perdida	2024-05-24 09:00:00	4	Bosque encantado	No
El rescate de la princesa	2024-05-24 12:00:00	3	Castillo	No
Intriga en la corte	2024-06-06 20:00:00	5	Palacio real	No
La ciudad maldita	2024-05-22 10:45:00	4	Ciudad abandonada	No
El laberinto de la muerte	2024-05-23 19:25:00	6	Laberinto	No
La expedición al misterio	2024-05-22 18:00:00	4	Selva Amazónica	No

**Información de la partida**

**+datacode+**

---

**La magia perdida**

**ID Partida** 4  
**Máximo de Jugador...** 4 horas.  
**GameMaster** 99

Nombre	Raza	Clase	Nivel
Drako el Sabio	Draconido	Mago	4

**Personajes**

Personaje: Thranduil. [Experiencia: 2]

**Unirme a la partida**

En la pestaña **jugar**, podrás:

- Jugar, que en el seleccionaras el personaje y la partida que deseas modificar las características del personaje, simplemente al cambiarlo debes dar actualizar y el sistema te indicara que se ha actualizado y si finalizas te llevara a la pestaña partidas actuales.

**Personajes**   **Partidas**   **Miembros**   **Jugar**

**Personaje**

Personaje: Thranduil. [Experiencia: 1]

**Partida**

El viaje al centro de la Tierra

Carisma: 55

Fuerza: 75

Destreza: 65

Constitución: 85

Sabiduría: 55

Inteligencia: 60

**Actualizar**   **Finalizar Sesión**

**Mensaje**

Se ha actualizado el personaje Thranduil, en la partida El viaje al centro de la Tierra

**Aceptar**

- Partidas actuales, solamente aparece una tabla con el nombre de la partida en las que estas con tus personajes que incluyen en esa partida y las características de este.

**Partidas Actuales**

Nombre Partida	Finalizada (Si/No)	Nombre Per.	Fuerza	Destreza	Constitución	Inteligencia	Sabiduría	Carisma
La ciudad maldita	No	Thranduil	44	50	62	49	66	89
La búsqueda del tesoro perdido	No	Thranduil	30	40	40	90	85	65
La leyenda del unicornio	No	Thranduil	1	83	24	29	58	16
La conquista del espacio	No	Thranduil	53	99	36	20	98	99
El viaje al centro de la Tierra	No	Thranduil	75	65	85	60	70	55
Las sombras de la montaña	No	Han Solo	47	32	76	88	56	74
La maldición del faraón	No	Han Solo	65	85	55	40	35	60
La venganza del pirata	No	Han Solo	95	50	90	35	30	40

**+data code+**

En la pestaña **miembros**, podrás:

- Visualizar una lista con todos los miembros que pertenecen a la comunidad.



### GameMaster:

Al escoger esta opción, se abrirá una ventana con una pequeña introducción y también un menú principal que tendrá 2 apartados distintos en el cual podrás navegar en ellos.



En la pestaña **miembros**, podrás:

- Visualizar una lista con todos los miembros que pertenecen a la comunidad. Es igual que la de jugador.

En la pestaña **partidas**, podrás:

- Crear partida, simplemente deberes rellenar los datos que te solicita el programa.

The screenshot shows a web application window titled 'DataCode++: dani'. It has a navigation bar with 'Partidas' and 'Miembros' tabs, and a home icon. The main form is titled 'Crear Partida' and contains the following fields:

- Nombre de la partida:** A text input field.
- Ambientación:** A text input field.
- Día y hora de la semana:** A text input field with the value '2024-05-19 00:32'.
- Duración de la sesión (horas):** A text input field.
- Número de la sesión:** A text input field.
- Crear Partida:** A blue button at the bottom right.

- Editar partida, es similar a la anterior pero aquí deberás escoger una partida ya creada.

The screenshot shows a web application window titled 'DataCode++: dani'. It has a navigation bar with 'Partidas' and 'Miembros' tabs, and a home icon. The main form is titled 'Editar Partida' and contains the following fields:

- Nombre de la partida:** A dropdown menu.
- Ambientación:** A text input field.
- Día y hora de la semana:** A text input field with the value '2024-05-19 00:32'.
- Duración de la sesión (horas):** A text input field.
- Número de la sesión:** A text input field.
- ¿Está en curso o terminada?:** Radio buttons for 'Si' and 'No'.
- Editar Partida:** A blue button at the bottom center.

- Consultar partida, podrás ver todas las partidas que hayas creado y en información de la partida veras todos los personajes que participan y modificarle el nivel que desees.

DataCode++: dani

Partidas Miembros 


## Consultar Partida

Nombre	Día y hora de Creación	Duración Sesión	Ambientación	Finalizada (Si/No)
La guerra de los magos	2024-06-02 18:00:00	5	Reino de Fantasia	Si
Busqueda del tesoro	2024-05-18 14:44:00	4	Castillos	No

Información de la partida

**+datacode+**

DataCode++: dani

Partidas Miembros 

## Busqueda del tesoro

**ID Partida** 21  
**Máximo de Jugador...** 4 horas.  
**GameMaster** 100

Nombre	Raza	Clase	Nivel
--------	------	-------	-------

Subir de Experien...