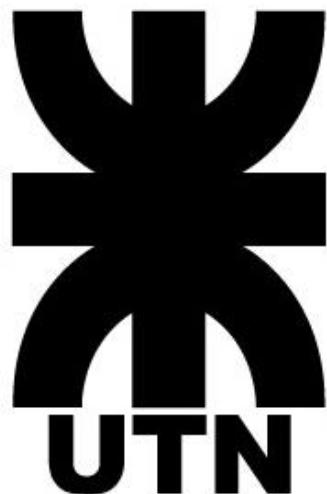


# **Primer Informe de Práctica Profesional Supervisada**

## **Título del Proyecto:**

*Plataforma de Gestión de Inscripciones para el Torneo Esports "Leyendas del Sur"*



**Estudiante:** Manuel Morullo

**Legajo:** 31110

**Carrera:** Ingeniería en Sistemas de Información

**Organización:** Secretaría de Asuntos Universitarios (SAU) de la Universidad Tecnológica Nacional – Facultad Regional La Plata (UTN FRLP).

**Tutor Académico:** Martín Ricardo Jorge

**Supervisor de Campo:** Rafael Valenzuela

**Fecha:** 22/7/2025

---

## **1. Introducción al Proyecto:**

El presente proyecto consiste en el desarrollo de una plataforma web destinada a gestionar las inscripciones para el torneo de League of Legends denominado "Leyendas del Sur". Este torneo es organizado por la Secretaría de Asuntos Universitarios (SAU) de la Universidad Tecnológica Nacional – Facultad Regional La Plata (UTN FRLP).

El torneo, de carácter abierto a todo el público tiene como propósito fundamental fomentar la participación en el ámbito de los esports, tanto dentro de la comunidad universitaria como a nivel internacional. Esta iniciativa busca brindar un espacio en el que los jugadores puedan competir, desarrollar sus habilidades y disfrutar de una experiencia enriquecedora dentro de un entorno competitivo y organizado.

La plataforma desarrollada busca automatizar y centralizar el registro de usuarios, la conformación de equipos y la gestión integral del proceso de inscripción. Esto incluye la carga y verificación de comprobantes de pago, así como la gestión interna de los equipos. Adicionalmente, se implementa un módulo administrativo que permite al equipo organizador — compuesto por integrantes de la SAU— revisar la documentación presentada, aprobar o rechazar equipos participantes, y personalizar de forma dinámica el contenido visible en el sitio web, asegurando una gestión eficiente y flexible del evento.

---

## **2. Descripción General de la Plataforma:**

### **2.1. Objetivos del Sistema:**

La plataforma "Leyendas del Sur" persigue los siguientes objetivos principales:

- **Centralizar la Información:** Unificar todos los datos relacionados con jugadores, equipos, estados de inscripción y resultados del torneo en un único sistema.
  - **Agilizar el Proceso de Inscripción:** Simplificar y automatizar el flujo de registro de usuarios y equipos, minimizando el trabajo manual y reduciendo posibles errores administrativos.
  - **Mejorar la Experiencia del Usuario:** Proporcionar una interfaz intuitiva y fácil de usar tanto para los jugadores, que podrán gestionar sus inscripciones y equipos, como para el staff administrativo.
  - **Facilitar la Administración del Torneo:** Ofrecer al equipo organizador herramientas para la revisión, aprobación y gestión de inscripciones, así como el control del contenido del sitio web.
- 

### **2.2. Roles de Usuarios**

La plataforma está diseñada para interactuar con dos roles principales de usuarios:

- **Jugador:**

- Son los participantes del torneo, provenientes de la comunidad universitaria o externa.

- Se registran en la plataforma para crear sus cuentas personales.
  - Pueden crear un nuevo equipo (convirtiéndose en capitán) o unirse a uno existente a través de invitaciones.
  - Gestionan la información de su perfil y, si son capitanes, la composición y detalles de su equipo.
  - Son responsables de subir los comprobantes de pago requeridos para la inscripción del equipo.
  - Pueden consultar en todo momento el estado de la inscripción y aprobación de su equipo.
  - Pueden editar la información de su perfil.
- **Staff (Administrativo):**
    - Integrantes de la Secretaría de Asuntos Universitarios (SAU) a cargo de la organización del torneo.
    - Acceden a un panel administrativo con funcionalidades exclusivas de gestión.
    - Revisan la documentación y los comprobantes de pago presentados por los equipos.
    - Tienen la autoridad para aprobar o rechazar las solicitudes de inscripción de equipos.
    - Gestionan y personalizan el contenido dinámico visible en el sitio web público, como clips de Twitch o anuncios.
    - Pueden habilitar o deshabilitar funcionalidades clave del sistema, como el período de inscripciones o la capacidad de los jugadores para editar sus perfiles.

---

### 2.3. Tecnologías Utilizadas

El desarrollo de la plataforma se apoya en las siguientes tecnologías principales:

- **Backend:**
  - **Python:** Lenguaje de programación principal para la lógica de negocio.
  - **Django Framework:** Framework web de alto nivel que facilita el desarrollo rápido y seguro, incluyendo su ORM (Object-Relational Mapper) para la interacción con la base de datos.
- **Base de Datos:**
  - **Desarrollo:** **SQLite**, una base de datos ligera y autónoma, ideal para entornos de desarrollo y pruebas.
  - **Producción:** **MariaDB**, un sistema de gestión de bases de datos relacionales (SGBD) robusto y de código abierto, para almacenar toda la información estructurada del torneo en el entorno de producción.

- **Frontend:**
    - Se utilizará una combinación de **HTML**, **CSS** y **JavaScript** para la construcción de la interfaz de usuario, posiblemente complementado con el sistema de plantillas de Django.
  - **Herramientas de Diseño y Modelado:**
    - **dbdiagram.io:** Utilizado para la creación visual y la documentación del esquema de la base de datos.
- 

### **3. Análisis y Diseño (Primera Etapa):**

Durante la primera etapa del proyecto, se ha llevado a cabo una fase de análisis y diseño, fundamental para sentar las bases de la plataforma. Este proceso incluyó la definición detallada de las interacciones de los usuarios mediante casos de uso y la elaboración del esquema de la base de datos que soportará el sistema.

#### **3.1. Casos de Uso**

Los siguientes casos de uso describen las funcionalidades que la plataforma debe ofrecer, desde la perspectiva de los diferentes actores del sistema.

##### **Casos de Uso del Jugador:**

- **CU001 - Registrarse en la Plataforma**
  - **Actor Principal:** Jugador
  - **Descripción:** Permite a un nuevo jugador crear una cuenta en la plataforma para acceder a las funcionalidades del torneo.
- **CU002 - Crear un Equipo**
  - **Actor Principal:** Jugador (como Capitán)
  - **Descripción:** Permite a un jugador crear un nuevo equipo para el torneo, asignándose a sí mismo como capitán.
- **CU003 - Gestionar Miembros del Equipo**
  - **Actor Principal:** Jugador (como Capitán del Equipo)
  - **Descripción:** Permite al capitán del equipo eliminar miembros de su equipo.
- **CU004 - Aceptar/Rechazar Invitación de Equipo**
  - **Actor Principal:** Jugador
  - **Descripción:** Permite a un jugador responder a una invitación recibida para unirse a un equipo, uniéndose al equipo o declinando la oferta.
- **CU005 - Consultar Estado de la inscripción del Equipo**
  - **Actor Principal:** Jugador

- **Descripción:** Permite al jugador verificar el estado actual de la inscripción y aprobación de su equipo en el torneo (Aprobada, Rechazada, En Revisión).
- **CU006 - Editar Perfil de Jugador**
  - **Actor Principal:** Jugador
  - **Descripción:** Permite al Jugador actualizar su información personal (ej. teléfono, Telegram, foto) en la plataforma, si la edición está habilitada por el staff.
- **CU007 - Realizar Pago de Inscripción**
  - **Actor Principal:** Jugador
  - **Descripción:** Permite al Jugador ver el instructivo de pago de la inscripción y adjuntar el o los comprobantes (en formato de imagen o archivo comprimido) del pago realizado para la inscripción al torneo.
- **CU008 - Ver Resultados de la Competencia**
  - **Actor Principal:** Jugador
  - **Descripción:** Permite al jugador ver información de los enfrentamientos entre equipos ya pasados y también próximos por haber (fechas, emparejamientos, resultados).

#### **Casos de Uso del Staff (Administrativo):**

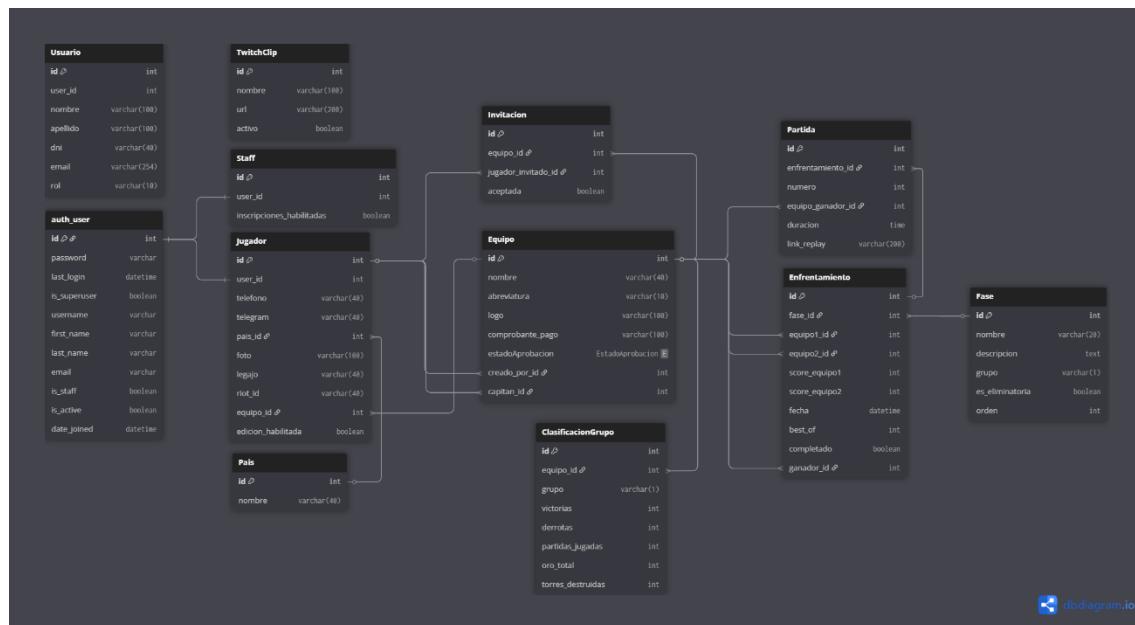
- **CU009 - Iniciar Sesión en el Panel Administrativo**
  - **Actor Principal:** Staff
  - **Descripción:** Permite a un miembro del Staff acceder al área restringida de administración para gestionar el torneo.
- **CU010 - Revisar Solicitud de Inscripción de Equipo**
  - **Actor Principal:** Staff
  - **Descripción:** Permite al Staff acceder y examinar la documentación (comprobante de pago) presentada por un equipo para su inscripción.
- **CU011 - Aprobar/Rechazar Equipo**
  - **Actor Principal:** Staff
  - **Descripción:** Permite al Staff cambiar el estado de aprobación de un equipo a "Aprobado" o "Rechazado" después de su revisión.
- **CU012 - Gestionar Contenido Dinámico del Sitio Web**
  - **Actor Principal:** Staff
  - **Descripción:** Permite al Staff añadir, editar, activar o desactivar elementos de contenido visible en la página pública, como clips de Twitch.
- **CU013 - Habilitar/Deshabilitar Edición de Perfiles de Jugador**

- **Actor Principal:** Staff
- **Descripción:** Permite al Staff controlar globalmente si los jugadores tienen la capacidad de modificar sus propios perfiles personales.
- **CU014 - Habilitar/Deshabilitar Inscripciones**
  - **Actor Principal:** Staff
  - **Descripción:** Permite al Staff abrir o cerrar oficialmente el período durante el cual los equipos pueden inscribirse en el torneo.
- **CU015 – Crear Enfrentamientos y Fases del Torneo**
  - **Actor Principal:** Staff
  - **Descripción:** Permite al Staff crear las fases del torneo, y los enfrentamientos entre los equipos.
- **CU016– Cargar Resultados de un enfrentamiento**
  - **Actor Principal:** Staff
  - **Descripción:** Permite al Staff cargar el resultado de los enfrentamientos, así como determinar el equipo ganador.

### 3.2. Diseño de Base de Datos

Se ha diseñado un esquema de base de datos relacional robusto para sustentar todas las funcionalidades de la plataforma. Este diseño se basa en los modelos de Django, teniendo en cuenta las relaciones entre las entidades y la eficiencia en el almacenamiento y recuperación de datos.

#### 3.2.1. Diagrama de base de datos:



### 3.2.2. Explicación de Tablas y Relaciones:

- **auth\_user (User)**: Tabla estándar de Django para la autenticación de usuarios. Contiene la información básica de la cuenta (username, password, email).
- **Pais**: Una nueva tabla que permite gestionar y relacionar los jugadores con su país de origen de manera estandarizada.
- **Jugador**: Representa a un participante del torneo.
  - **Herencia**: Integra directamente los campos nombre, apellido, dni, email y rol de la clase abstracta Usuario definida en los modelos de Django.
  - **Relaciones**: user\_id (relación uno a uno con auth\_user) y pais\_id (clave foránea a Pais). También tiene una clave foránea equipo\_id para vincular al jugador con su equipo.
- **Staff**: Representa al personal organizador del torneo.
  - **Herencia**: Similar a Jugador, hereda los campos de la clase abstracta Usuario.
  - **Relaciones**: user\_id (relación uno a uno con auth\_user).
- **Equipo**: Almacena los datos de los equipos inscritos.
  - **Campos clave**: nombre (único), abreviatura, logo, comprobante\_pago, y estadoAprobacion que utiliza el Enum definido.
  - **Relaciones**: creado\_por\_id y capitán\_id (claves foráneas a Jugador).
- **Invitacion**: Registra las invitaciones de equipos a jugadores.
  - **Relaciones**: equipo\_id (clave foránea a Equipo) y jugador\_invitado\_id (clave foránea a Jugador).
- **TwitchClip**: Contiene información sobre clips de Twitch para mostrar en la plataforma.
- **Fase**: Define las diferentes etapas del torneo (ej., Fase de Grupos, Cuartos de Final).
- **Enfrentamiento**: Representa un partido o una serie de partidas entre dos equipos en una fase específica.
  - **Relaciones**: fase\_id (clave foránea a Fase), equipo1\_id y equipo2\_id (claves foráneas a Equipo).
- **Partida**: Detalla cada partida individual dentro de un Enfrentamiento (especialmente útil para series "Best of").
  - **Relaciones**: enfrentamiento\_id (clave foránea a Enfrentamiento).
- **ClasificacionGrupo**: Guarda las estadísticas de los equipos dentro de cada grupo.
  - **Relaciones**: equipo\_id (clave foránea a Equipo). Incluye un índice único para asegurar que un equipo solo tenga una entrada por grupo.

---

#### **4. Actividades Desarrolladas en el Período**

Durante el período comprendido en este primer informe, las principales actividades desarrolladas se centraron en la definición del alcance del proyecto, el diseño de la estructura de datos, la configuración inicial del entorno de desarrollo y el desarrollo de los modulos principales del sistema. Las tareas específicas incluyen:

- **Familiarización y Análisis de Requerimientos:**
  - Estudio detallado de la descripción del proyecto "Leyendas del Sur", sus objetivos y funcionalidades.
  - Análisis de los roles de usuario (Jugador, Staff) y sus interacciones con el sistema.
  - Investigación sobre las funcionalidades específicas del torneo (inscripciones, gestión de equipos, fases de juego).
- **Diseño de Base de Datos Lógica y Física:**
  - Identificación de todas las entidades de negocio (Jugadores, Equipos, Enfrentamientos, etc.) y sus atributos relevantes.
  - Definición de las relaciones entre las entidades (uno a uno, uno a muchos).
  - Modelado del esquema de la base de datos utilizando dbdiagram.io, traduciendo los modelos de Django a un diagrama relacional.
  - Resolución de la representación de la herencia de modelos abstractos (como Usuario), asegurando que sus campos se integren correctamente en las tablas concretas (Jugador, Staff).
  - Incorporación de la entidad País como una tabla separada para una mejor normalización y gestión de la información geográfica.
  - Definición de tipos enumerados para estados de aprobación.
- **Configuración del Entorno de Desarrollo:**
  - Instalación y configuración del entorno de desarrollo Python y Django.
  - Creación del proyecto Django base y las aplicaciones necesarias (apps).
  - Definición inicial de los modelos en models.py de acuerdo con el diseño de la base de datos.
  - Generación y aplicación de las primeras migraciones de base de datos para crear el esquema en el SGBD local.
  - Configuración del panel de administración de Django (admin.py) para permitir la gestión básica de las entidades principales.
- **Desarrollo de modulos:**
  - **Módulo de Registro de Usuarios:**

- Implementación de las funcionalidades para que nuevos jugadores puedan crear una cuenta en la plataforma.
  - Desarrollo de los formularios y la lógica de validación para la captura de datos de registro (nombre, apellido, DNI, email, Riot ID, etc.)
  - Integración con el sistema de autenticación de usuarios de Django.
- **Módulo de Creación de Equipos:**
    - Desarrollo de la interfaz y lógica para que un jugador registrado pueda crear un nuevo equipo.
    - Implementación de la funcionalidad para asignar al jugador creador como capitán del equipo.
    - Configuración de la carga inicial de comprobantes de pago de inscripción.
  - **Módulo de Modificación Dinámica de Contenido:**
    - Implementación de la gestión de contenido dinámico de la página principal (ej., clips de Twitch).
    - Desarrollo de la interfaz administrativa para que el Staff pueda añadir, editar y controlar la visibilidad de dicho contenido.
  - **Módulo de Gestión de Competencia (Resultados, Fases, Enfrentamientos) y su Visualización:**
    - Diseño e implementación de los modelos de datos para fases, enfrentamientos, partidas y clasificación de grupos.
    - Desarrollo de la lógica para el registro de resultados de enfrentamientos y partidas.
    - Implementación de interfaces (jugador/staff) para la gestión y visualización de la estructura de la competencia y sus resultados.