



Aplicación web responsiva para torneos de microfútbol comunales en el municipio
de Soacha

Jorge Enrique Romero Murcia

10892311617

Remberto Jose Prioló Flórez

10892311018

Universidad Antonio Nariño

Programa Especialización e Ingeniería de Software

Facultad de Sistemas

Bogotá, Colombia

2023

**Aplicación web responsiva para torneos de microfútbol comunales en el
municipio de Soacha**

Jorge Enrique Romero Murcia

Remberto Jose Prioló Flórez

Proyecto de grado presentado como requisito parcial para optar al título de:
Especialista en Ingeniería de Software

Director (a):

Dianalin Neme Prada Msc

Universidad Antonio Nariño

Programa Especialización en Ingeniería de Software

Facultad de Sistemas

Bogotá, Colombia

2023

NOTA DE ACEPTACIÓN

El trabajo de grado titulado

_____.

Cumple con los requisitos para optar

Al título de _____.

Firma del Tutor

Firma Jurado

Firma Jurado

Ciudad, Día Mes Año.

Contenido

Pág.

Tabla de contenido

1.	Título.....	11
2.	Resumen.....	12
3.	Abstract	13
4.	Introducción	14
5.	Formulación y Descripción del Problema	16
6.	Objetivo General	19
7.	Objetivos Específicos.....	20
8.	Marco de Referencia	21
	8.1 Estado del arte.....	21
	8.2. Impacto.....	24
	8.3 Componentes de innovación	25
	8.4 Marco Teórico	25
9.	Metodología	29
10.	Proceso del Software	34
	10.1 Requerimientos Funcionales:	36
	10.2 Requerimientos No Funcionales.....	37
	10.3 Diseño y Arquitectura.....	38
	10.3.1 Diagrama de Despliegue	39

10.3.2	Caso de uso Arquitectura relevante	40
10.3.3	Diagrama de secuencia	41
10.3.4	Diagrama de clases	42
10.3.5	Arquitectura de Alto Nivel.....	44
11.	Construcción.....	45
4.	Pruebas	48
a.	De Proceso	49
b.	De Producto	67
c.	De Seguridad.....	72
5.	Instalación y Configuración.....	81
11.	Conclusiones	94
12.	Anexos.....	95
13.	Referencias Bibliográficas.....	96

Tabla de Ilustraciones

Ilustración 1 - Kanban Board	33
Ilustración 2 - Diseño y Arquitectura.....	38
Ilustración 3 - Diagrama de Despliegue.....	39
Ilustración 4 - Caso de Uso Arquitectura Relevante.....	40
Ilustración 5 - Diagrama de secuencia	41
Ilustración 6 - Diagrama de clases	43
Ilustración 7 - Arquitectura de Alto Nivel (MVC)	44
Ilustración 8 - Entorno de Desarrollo Node JS.....	46
Ilustración 9 - Entorno de desarrollo npm.....	46
Ilustración 10 - Backend	47
Ilustración 11 - Llamado a las apis del Backend	48
Ilustración 12 - Pantalla Inicial.....	51
Ilustración 13 - Create First Tournament	52
Ilustración 14 - Register User.....	52
Ilustración 15 - Create User.....	53
Ilustración 16 - Registro Exitoso	53
Ilustración 17 - Confirmación	54
Ilustración 18 - Login Exitoso	54
Ilustración 19 - Opción Creación de Torneo	55
Ilustración 20 - Autenticación de Usuario.....	55
Ilustración 21 - Creación de Torneo	56

Ilustración 22 - Creación de Torneo Exitoso.....	56
Ilustración 23 - Verificación de Creación de Torneo.....	57
Ilustración 24 - Creación de Equipos para el Torneo	57
Ilustración 25 - Registro Equipo.....	58
Ilustración 26 - Verificación de Equipos creados.....	58
Ilustración 27 - Creación de Jugadores y asociarlos a un Equipo	59
Ilustración 28 - Registro de Jugador.....	59
Ilustración 29 - Registro Jugador Exitoso.....	60
Ilustración 30 - Verificación de Creación de Jugador Exitoso.....	60
Ilustración 31 - Módulo de Partidos.....	61
Ilustración 32 - Modulo Pantalla de Partidos	61
Ilustración 33 - Módulo Pantalla Estadísticas Jugadores.....	62
Ilustración 34 - Funcionalidad de Envío de notificaciones a E-mail.	62
Ilustración 35 - Verificación de notificación al E-mail.....	63
Ilustración 36 - Login Exitoso BDD.....	63
Ilustración 37 - Login Init Failed.....	64
Ilustración 38 - Account No Registered.....	64
Ilustración 39 - Incorrect Password	65
Ilustración 40 - Load Test Suite.....	66
Ilustración 41 - Execution Load Test Suite.....	66
Ilustración 42 - Report HTML Load Test Suite	67
Ilustración 43 - Test End To End.....	69

Ilustración 44 - Registerd Failed.....	70
Ilustración 45 - Registered Weak Password.....	70
Ilustración 46 - Test Load Suite Register User.....	71
Ilustración 47 - Reporte HTML Register User Suite	72
Ilustración 48 - Pruebas estáticas Sonarqube	73
Ilustración 49 - Prueba OWAS ZAP.....	75
Ilustración 50 - Prueba OWAS ZAP II.....	75
Ilustración 51 - Prueba OWAS ZAP III.....	76
Ilustración 52 - Pruebas OAS ZAP IV	76
Ilustración 53 - Prueba OWAS ZAP V.....	77
Ilustración 54 - Instalación XAMPP.....	82
Ilustración 55 - Inicio de XAMPP	84
Ilustración 56 - Ubicación Proyecto	85
Ilustración 57 - Proceso Levantamiento de Ambiente Backend.....	86
Ilustración 58 - Comando para levantar Ambiente Back con NPM	87
Ilustración 59 - Confirmación Ambiente Backend Arriba	87
Ilustración 60 - Validación de la BD Arriba	88
Ilustración 61 - Confirmación en el Navegador	89
Ilustración 62 - Levantamiento de Ambiente Frontend	90
Ilustración 63 - Apertura IDE VSC	90
Ilustración 64 - Inicio Terminal IDE	91
Ilustración 65 - Ejecución Comando Levantamiento Frontend desde el IDE.....	92

Ilustración 66 - Verificación Levantamiento Ambiente Frontend.....	92
Ilustración 67 - Acceso a Ambiente.....	93

Tabla de Referencias

Tabla 1 - Tecnologías Lenguajes de Programación - Elaboración Propia	28
Tabla 2 - Tecnologías B.D. - Elaboración Propia.....	28
Tabla 3 - Control de versiones - Elaboración Propia.....	28
Tabla 4 - Herramientas de Desarrollo Web - Elaboración Propia	28
Tabla 5 - Herramientas de seguridad - Elaboración Propia	29
Tabla 6 - Framework de Pruebas y Control de Calidad - Elaboración Propia	29
Tabla 7 - Requerimientos funcionales - Elaboración Propia.....	36
Tabla 8 - Requerimientos No Funcionales - Elaboración Propia	37
Tabla 9 - Tabla de Pruebas de Proceso - Elaboración Propia	50
Tabla 10 - Tabla de Pruebas de Producto - Elaboración Propia	68

1. Título

Aplicación web responsiva para torneos de microfútbol comunales en el municipio de Soacha

2. Resumen

El proyecto se centra en el desarrollo de una aplicación web para la gestión eficiente de torneos de microfútbol en Soacha, Cundinamarca. Utilizando la metodología Kanban y una arquitectura MVC, la aplicación aborda la problemática actual de la gestión manual de torneos, ofreciendo una solución integral. Desde la inscripción de equipos hasta la programación de partidos y el seguimiento de resultados, la aplicación busca mejorar la experiencia tanto para participantes como administradores. La adopción de tecnologías web modernas y un enfoque inclusivo en el desarrollo refuerzan la innovación y la usabilidad. Este proyecto no solo aborda una necesidad local específica, sino que también sienta las bases para futuras mejoras en la gestión de eventos deportivos.

3. Abstract

The project focuses on the development of a web application for the efficient management of micro soccer tournaments in Bogota. Using Kanban methodology and MVC architecture, the application addresses the current problem of manual tournament management, offering a comprehensive solution. From team registration to match scheduling and results tracking, the application seeks to improve the experience for both participants and administrators. The adoption of modern web technologies and an inclusive approach to development reinforce innovation and usability. This project not only addresses a specific local need, but also lays the groundwork for future improvements in sports event management.

4. Introducción

En la era digital actual, donde la tecnología permea cada aspecto de nuestras vidas, la necesidad de soluciones innovadoras que faciliten y optimicen procesos es más evidente que nunca. Este proyecto de grado se embarca en la creación de una aplicación web dedicada a la gestión de torneos de microfútbol en el municipio de Soacha Cundinamarca, Colombia. La iniciativa nace de la observación de una brecha tecnológica en la administración de torneos locales de microfútbol, donde la mayoría de los procesos aún se llevan a cabo de manera manual y desconectada. La aplicación propuesta busca introducir una plataforma moderna que transforme la manera en que se planifican, ejecutan y siguen los torneos de microfútbol en Soacha. Al incorporar tecnologías web avanzadas, la solución no solo tiene como objetivo mejorar la eficiencia operativa, sino también enriquecer la experiencia de los participantes, administradores y seguidores de estos emocionantes eventos deportivos.

En esta tesis, exploraremos en detalle el proceso de desarrollo de la aplicación web, desde la formulación del problema y los objetivos del proyecto hasta la implementación de características clave como la gestión de equipos, la programación de partidos y la automatización de procesos administrativos. Además, se abordarán aspectos cruciales como la seguridad de la aplicación y su capacidad para adaptarse a diversos dispositivos.

Este proyecto no solo busca ser una solución tecnológica eficiente para la gestión de torneos de microfútbol, sino también un testimonio del impacto positivo que la tecnología

puede tener en la mejora de procesos tradicionales. A medida que avanzamos en la era digital, la fusión de la pasión por el deporte y la innovación tecnológica representa una oportunidad emocionante para potenciar y enriquecer las experiencias deportivas a nivel local. Este trabajo se propone ser un paso significativo en esa dirección, explorando cómo una aplicación web puede ser la clave para elevar la calidad y la eficiencia de los torneos de microfútbol en el municipio de Soacha.

5. Formulación y Descripción del Problema

Formulación:

- ¿Cómo se puede mejorar la eficiencia y transparencia en la organización y gestión de los torneos de microfútbol en Soacha, considerando la falta de tecnología y la necesidad de simplificar la inscripción de equipos, la programación de partidos, la gestión de resultados y la comunicación con los participantes?

Descripción del Problema:

Falta de eficiencia en la organización:

- En la organización de campeonatos de microfútbol comunales, hay una deficiencia en la toma de datos manual a lápiz y papel, lo que dificulta la anotación de los datos de todos los participantes al momento de completar los formularios de inscripción de los equipos involucrados. Durante el evento, las estadísticas de los partidos se anotan en una hoja y luego se realiza la transmisión de los resultados de los partidos. Actualmente, los organizadores de torneos de microfútbol en Soacha dependen en gran medida de métodos manuales para programar partidos, registrar equipos y administrar los resultados. Esto requiere mucho tiempo y recursos, lo que con frecuencia lleva a retrasos y errores en la organización de los torneos.

Dificultades en la comunicación:

- Suele ser difícil comunicarse entre los equipos participantes, los árbitros y los organizadores. La falta de una plataforma centralizada para comunicar y difundir información relevante, como cambios en los horarios de los partidos o actualizaciones en las tablas de clasificación, puede causar confusiones y malentendidos. El proceso de sortear los equipos para determinar los rivales y las fechas de los partidos a jugar es evidente debido a la manualidad requerida, lo que puede generar inconsistencias con la información.

Transparencia y equidad:

- Las acusaciones de favoritismo o manipulación por parte de los organizadores pueden surgir de la falta de transparencia en la programación de partidos y la gestión de resultados. Esto afecta negativamente la integridad de los torneos y la confianza de los equipos. Además, no se registran las estadísticas de los encuentros que se llevaron a cabo en los campeonatos previamente programados, lo que impide obtener un registro preciso y confiable de los torneos anteriores. No se divulga la información limitada recopilada durante los torneos.

Experiencia del Usuario Insatisfactoria:

- Debido a la falta de información actualizada y la dificultad para seguir los torneos de microfútbol en Soacha de manera conveniente, tanto los equipos participantes como los espectadores frecuentemente tienen una experiencia insatisfactoria.

Por la problemática mencionada, se buscan alcanzar los objetivos que se detallan a continuación.

6. Objetivo General

Construir un aplicativo web adaptable que permita la gestión de campeonatos de microfútbol comunales en el municipio de Soacha, Cundinamarca.

7. Objetivos Específicos

1. Proporcionar a los organizadores de torneos de microfútbol una herramienta que les permita organizar un campeonato comunal de manera rápida y sencilla.
2. Permitir que los grupos se conformen aleatoriamente con los equipos que participan en el campeonato.
3. Permitir la creación de equipos y el registro de jugadores.
4. Después del partido deportivo, se enviará una notificación de las actualizaciones por correo electrónico del registro de los eventos más importantes, como las tarjetas de cada equipo, los jugadores amonestados, el número de goles y otros detalles.
5. Estadísticas del torneo.

8. Marco de Referencia

8.1 Estado del arte

Como parte de los componentes de investigación para el desarrollo de este proyecto, se realiza una búsqueda en varios sitios web y se encuentran tres aplicaciones que manejan de manera excelente los torneos de fútbol profesional, nacionales e internacionales, esto mejora la comprensión de las aplicaciones de la gestión de campeonatos de fútbol.

Antecedentes de Investigación

Luego se discuten las tres aplicaciones, una aplicación móvil y dos aplicaciones web, que han venido reflejando avances en la gestión de torneos, además de agilizar sus procesos de registros y hacer que la experiencia de los interesados sea más fácil y atractiva. Las aplicaciones que brindan información detallada sobre los pormenores ocurridos durante el desarrollo de los partidos de fútbol profesional y ligas de cada país han ganado importancia en los últimos cinco años. Estas aplicaciones brindan toda esta información de forma profesional y detallada a todos los interesados en tiempo real. Según esta investigación, las aplicaciones web actualmente disponibles facilitan la gestión de torneos de fútbol profesional, pero debido a su alto costo, no son adecuadas para los torneos de microfútbol comunales. Adicionalmente, la mayoría de las personas no tienen acceso a estos aplicativos.

365scores.com

Esta App móvil y web es usada actualmente para mostrar resultados en vivo de ligas profesionales nacionales e internacionales de fútbol, notificando en tiempo real a los fans del fútbol, de cualquier novedad que se esté presentando en el partido de su interés.

“365Scores es el más rápido y más preciso servicio online de resultados en vivo, más de 100 millones de usuarios alrededor del mundo avalan confían en nosotros desde el 2012. Nuestra cobertura de Fútbol incluye las últimas noticias, calendarios, resultados, tablas de posiciones, estadísticas y actualizaciones en vivo de los partidos de las diferentes ligas alrededor del mundo incluyendo Liga BetPlay, Champions League, La Liga, Premier League y Copa Colombia”

ligafansport.com

Este sitio Web creado por “serversports.com” es actualmente usado para la gestión de torneos regionales y nacionales para aficionados. Convirtiéndose en una herramienta para el seguimiento del desarrollo de todas las actividades que se planearon en el torneo.

Es usada por muchos jugadores aficionados para tener el consolidado de los partidos jugados, fechas de próximos partidos, etc. De los distintos torneos organizados a nivel nacional.

“FanSport se compromete en cada uno de nuestros eventos deportivos por poner en primer lugar el bienestar y los intereses de todos nuestros participantes cada fin de semana, ofreciendo una atención personalizada mediante un canal de comunicación con cada uno de los capitanes, cumpliendo los tiempos de programación de cada partido, asegurando cada uno de los beneficios prometidos al inicio del torneo como hidratación para los jugadores, tiempos flexibles que se ajusten a los requerimientos iniciales de cada equipo, cumplimiento del reglamento, acompañamiento de coordinadores en cancha que le brindará atención presencial a los equipos participantes.”

flashscore.co

“La página de marcadores en vivo de fútbol en Flashscore.co ofrece los más recientes resultados de más de 1.000 ligas de fútbol y torneos alrededor del mundo como la (Liga Betplay, la Premier League, La Liga, la Serie A, la Bundesliga, la UEFA Champions League, y también otras ligas como la Liga MX o la Liga de Argentina. Sigue todos los marcadores de fútbol de hoy/esta noche en Flashscore.co y encontrarás las previas de los partidos de fútbol, estadísticas detalladas (remates a puerta, posesión del balón, estadísticas de tiros de esquina, tarjetas amarillas y rojas, faltas, etc.), la comparación de cuotas, los H2H, alineaciones, comentarios en vivo, transferencias, noticias o resúmenes en video.”

En la culminación del análisis del estado del arte, se ha identificado un espacio significativo en la aplicación de soluciones tecnológicas para la gestión de torneos de microfútbol en el contexto específico del municipio de Soacha. La revisión exhaustiva de investigaciones y desarrollos previos resalta la necesidad de una plataforma que integre

eficientemente procesos administrativos, optimice la participación de los equipos y ofrezca una experiencia en línea atractiva para los aficionados. En este sentido, el presente proyecto busca llenar esta brecha mediante el desarrollo de una aplicación web innovadora y responsiva. Al fusionar las mejores prácticas identificadas en el estado del arte con soluciones personalizadas para las características específicas de los torneos de microfútbol en Soacha, estableciendo un nuevo estándar para la gestión moderna de eventos deportivos locales. Este trabajo busca ser un catalizador para la convergencia exitosa de la pasión deportiva y la eficacia tecnológica en el contexto específico de los torneos de microfútbol en Soacha.

8.2. Impacto

Los participantes de un campeonato de microfútbol comunales en Soacha podrán utilizar este aplicativo web para utilizar la tecnología para mejorar la logística, ayudando al organizador del torneo a tener una planificación clara y ágil que permita a los jugadores una inscripción fácil, rápida y sin tantas complicaciones de trámites.

Para lograrlo, la aplicación tendrá funcionalidades que te permitirán tomar datos de equipo, organizar y distribuir a los rivales y recibir notificaciones por WhatsApp sobre las fechas de los partidos programados.

8.3 Componentes de innovación

Las aplicaciones actuales son costosas y están destinadas a torneos profesionales, por lo que la mayoría de la gente no las puede usar. Este aplicativo web tiene como objetivo brindar una gestión simple y enfocada para satisfacer la necesidad de estos campeonatos de microfútbol comunales. La información generada en los torneos de microfútbol amateur del municipio será profesionalizada.

Innovación en Publicidad y Patrocinios:

- Explorar nuevas formas creativas de integrar publicidad y patrocinios en la aplicación sin comprometer la experiencia del usuario, como anuncios interactivos o experiencias publicitarias inmersivas.

8.4 Marco Teórico

“La industria de contenidos digitales se encuentra en el espacio de convergencia de la producción creativa y las tecnologías de la información y las comunicaciones (TIC).” MinTic, 2018 afirma De acuerdo con la ley 1341 de 2009, también conocida como ley de TIC, el Ministerio de Tecnologías de la Información y las Comunicaciones en Colombia tiene la responsabilidad de fomentar el uso y aprovechamiento de las tecnologías de la información y las comunicaciones entre la población, las empresas, el gobierno y otras entidades públicas para apoyar el progreso social, económico y político del país. MinTIC (2014).

Esta aplicación web responsiva ofrece varios beneficios en comparación con las páginas web convencionales, como la apariencia nativa porque la interfaz de usuario y la navegación son muy similares a las de una aplicación nativa. Dado que se apoyan en el almacenamiento de caché del navegador, mejoran la velocidad de respuesta del aplicativo tanto en la carga como en la navegación.

Deporte y tecnología:

- El deporte es una parte importante de la cultura de Soacha, y el microfútbol se ha convertido en una actividad deportiva de base en el municipio. La tecnología ha cambiado mucho la forma en que las personas juegan y siguen deportes en todo el mundo. Un ejemplo de cómo la tecnología puede mejorar la experiencia deportiva y la gestión de eventos es la aplicación web de torneos de microfútbol.

Gestión de torneos deportivos:

- La organización de un torneo deportivo implica planificar los partidos, registrar a los equipos, administrar los resultados y comunicarse con los participantes. Tradicionalmente, estas tareas se han realizado manualmente o

a través de hojas de cálculo y comunicación presencial o telefónica. Sin embargo, en el ámbito deportivo, las aplicaciones web y móviles han simplificado y mejorado estos procesos.

Torneos deportivos en Soacha:

- El microfútbol se juega en una variedad de lugares en Soacha, desde las comunidades locales hasta los colegios y las zonas de vecindario. Los torneos se llevan a cabo en varios niveles, desde eventos casuales hasta competencias más organizadas. La eficiencia, la transparencia y la comunicación han sido problemas para la gestión de estos torneos.

Tecnologías.

Característica	Lenguaje de Programación	Descripción
Back-end	TypeScript	Es un lenguaje de programación desarrollado por Microsoft que extiende JavaScript al agregarle tipos estáticos. Está diseñado para mejorar la calidad y la mantenibilidad del código en proyectos de gran envergadura y para facilitar el desarrollo en equipos grandes.
Framework Front-end	Angular	Es un popular framework de desarrollo de aplicaciones web desarrollado por Google y una comunidad activa de desarrolladores. Está diseñado para crear aplicaciones web robustas, escalables y de una sola página (SPA) y proporciona un conjunto de características poderosas y herramientas que facilitan el desarrollo.
		Es un marco de desarrollo backend que combina conceptos modernos de JavaScript/TypeScript con una arquitectura

Framework de Backend	Nest js	modular y orientada a servicios para facilitar la creación de aplicaciones robustas y mantenibles. Es especialmente adecuado para construir aplicaciones empresariales y APIs RESTful.
-----------------------------	----------------	--

Tabla 1 - Tecnologías Lenguajes de Programación - Elaboración Propia

Característica	Base de Datos	Descripción
B.D.	MySQL	Es un sistema de gestión de bases de datos (DBMS) popular y de código abierto ampliamente utilizado en aplicaciones web y empresariales.

Tabla 2 - Tecnologías B.D. - Elaboración Propia

Característica	Gestor de Versiones	Descripción
Control de versiones	Git	Para el control de versiones del código fuente, lo que facilita la colaboración y el seguimiento de cambios.

Tabla 3 - Control de versiones - Elaboración Propia

Característica	Herramienta	Descripción
Herramientas de Desarrollo Web	Visual Studio Code (VS Code)	Es un popular entorno de desarrollo integrado (IDE) de código abierto desarrollado por Microsoft que se utiliza ampliamente para programar y desarrollar aplicaciones en diversos lenguajes de programación.

Tabla 4 - Herramientas de Desarrollo Web - Elaboración Propia

Herramientas de Seguridad	Descripción
SonarQube	Es una herramienta de revisión de código automática y autogestionada que le ayuda sistemáticamente a entregar Código Limpio. Como elemento central de nuestra solución Sonar, SonarQube se integra en su flujo de trabajo existente y detecta problemas en su código para ayudarle a realizar inspecciones continuas del código de sus proyectos. La herramienta analiza más de 30 lenguajes de programación diferentes y se integra en su pipeline CI y en su plataforma DevOps para garantizar que su código cumple con los estándares de alta calidad.

OWASP ZAP	OWASP ZAP es una herramienta diseñada para realizar pruebas de seguridad automatizadas en aplicaciones web. Se utiliza para encontrar vulnerabilidades comunes de seguridad en aplicaciones web durante el desarrollo y las pruebas de penetración. Zed Attack Proxy (ZAP) es una de las herramientas de seguridad gratuitas más populares del mundo, mantenida activamente por un equipo internacional de voluntarios. Puede ayudarle a encontrar automáticamente vulnerabilidades de seguridad en sus aplicaciones web mientras las desarrolla y prueba. También es una gran herramienta para que los pentesters experimentados la utilicen para realizar pruebas de seguridad manuales.
------------------	--

Tabla 5 - Herramientas de seguridad - Elaboración Propia

Plataformas de Alojamiento (Hosting):

(Jorge y Remberto)

Framework de Pruebas y Control de Calidad	Descripción
Katalon	El proyecto decidió utilizar el Framework de Katalon para ejecutar las pruebas funcionales de manera automatizada porque era la opción más adaptativa al proyecto y aseguró que el software cumpliera con sus criterios de aceptación, reglas comerciales y requerimientos funcionales establecidos durante la fase previa. Su interfaz gráfica permite diseñar casos de prueba, agregar pasos, verificar los resultados y administrar los casos de prueba.

Tabla 6 - Framework de Pruebas y Control de Calidad - Elaboración Propia

9. Metodología

En el marco de este proyecto, se adoptó la metodología Kanban como enfoque principal de gestión y desarrollo. Kanban, con sus raíces en el sistema de producción de Toyota, se ha consolidado como una metodología ágil que prioriza la visibilidad y la mejora continua en los flujos de trabajo. La utilización de tableros Kanban virtuales permitió una

representación visual clara de las tareas, desde la concepción hasta la implementación, lo que facilitó la identificación rápida de cuellos de botella y la asignación eficiente de recursos. Además, la flexibilidad inherente de Kanban se alineó con la naturaleza dinámica de un proyecto en evolución, adaptándose con agilidad a los cambios en los requisitos y prioridades.

Kanban

La metodología Kanban es un enfoque de gestión visual que mejora la eficiencia y la productividad en los proyectos, incluidas las aplicaciones web. Se requiere una metodología que, por su naturaleza, facilite los mecanismos necesarios para identificar y priorizar entregables, tomando como criterio el beneficio aportado, para el desarrollo de la solución expuesta en la formulación del problema. Adicionalmente, esta metodología debe permitir un aprovechamiento adecuado de los recursos humanos, técnicos y tecnológicos, teniendo en cuenta los límites de tiempo. El equipo de trabajo se ha definido de la siguiente manera para llevar a cabo el proyecto actual:

Distribución del equipo:

- **Service Request Manager:** Remberto José Prioló Flórez
- **Service Delivery Manager:** Jorge Enrique Romero Murcia
- **Team Developer:** Jorge Enrique Romero Murcia y Remberto José Prioló Flórez

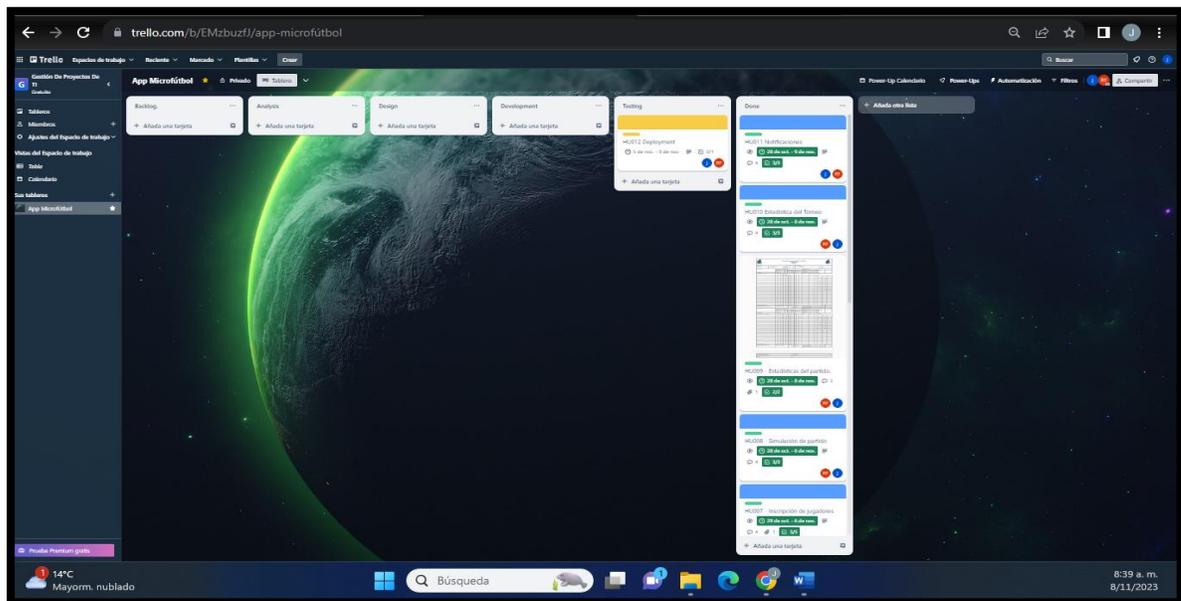
Se adapta el framework de trabajo de acuerdo con la metodología definida para el proyecto, adicionando las ceremonias que se tienen para la implementación del proyecto y sus MVP (Producto mínimo viable) una vez definida la ceremonia (Daily Standup) de 15 minutos para evaluar el estado de las tareas y mantenerse actualizados sobre el progreso del trabajo y abordar posibles problemas. Los requerimientos funcionales serán definidos mediante Historias de Usuario. La elección de la herramienta Trello para la gestión de este proyecto se basó en su interfaz intuitiva, flexibilidad y capacidad para proporcionar una visualización clara del flujo de trabajo. Trello ofrece tableros Kanban virtuales que se alinean perfectamente con la metodología Kanban adoptada para este proyecto, permitiendo la representación visual de las tareas, desde la planificación hasta la implementación. La simplicidad de Trello facilitó la rápida adopción por parte del equipo, permitiendo una fácil colaboración y actualización en tiempo real de los progresos y cambios en las tareas. Además, la integración de Trello con otras herramientas y su accesibilidad desde diversos dispositivos garantizaron una gestión ágil y eficiente del proyecto en todas las etapas de desarrollo. La elección de Trello como herramienta de gestión se fundamenta en su capacidad para fomentar la transparencia, la colaboración y la adaptabilidad, elementos esenciales en el entorno dinámico de un proyecto de desarrollo de software.

Trello:

Es una herramienta de gestión de proyectos que se utiliza comúnmente para implementar la metodología Kanban. Mediante un tablero digital creado en la herramienta Trello se podrá visualizar y dar seguimiento y control a las tareas planteadas en dicho proyecto, permitiendo reducir el esfuerzo con un flujo continuo de actividades. Además, para usar Kanban no se exigen roles con funciones específicas y los cambios se pueden hacer en cualquier momento. La metodología KANBAN el cual propone mostrar gráficamente el flujo de trabajo por columnas, manifestando:

1. Análisis
2. Diseño
3. Desarrollo
4. Pruebas

Ilustración 1 - Kanban Board



Nota: Elaboración propia (2023)

Características Clave:

Cada una de estas características se crea manualmente para las siguientes actividades.

- **Creación de Tareas:** Los miembros del equipo pueden crear tareas directamente en el tablero ágil, asignarlas a personas responsables y agregar detalles relevantes.
- **Listas y Columnas Personalizadas:** Puede configurar listas y columnas en el tablero para representar las etapas del flujo de trabajo de tu proyecto, como "Por Hacer", "En Progreso", "En Revisión" y "Completado". Estas listas son personalizables según tus necesidades.
- **Arrastrar y Soltar:** Los usuarios pueden mover las tareas de una columna a otra simplemente arrastrándolas y soltándolas, lo que facilita el seguimiento del progreso y la gestión del flujo de trabajo.

- **Asignación de Tareas:** Cada tarea puede asignarse a un miembro del equipo para indicar quién es responsable de completarla.
- **Comentarios y Conversaciones:** Los usuarios pueden agregar comentarios a las tareas para discutir detalles, proporcionar actualizaciones o hacer preguntas relacionadas con la tarea.
- **Priorización:** Las tareas pueden priorizarse mediante etiquetas o marcadores para destacar las más importantes o urgentes.

10. Proceso del Software

La inclusión y la apertura son fundamentales en el proceso de desarrollo de software para este proyecto. Reconocer la diversidad de habilidades y perspectivas que aportan los miembros del equipo fomenta un ambiente inclusivo donde cada voz es valorada y respetada. Se establece un compromiso con la transparencia y la comunicación abierta, lo que crea un entorno donde las ideas innovadoras encuentran un terreno fértil para enfrentar los desafíos en colaboración. Este enfoque inclusivo fomenta un equipo unido y fomenta la creatividad y la resiliencia necesarias para el éxito en el proceso de desarrollo de software.

El proceso de desarrollo de software se divide en las siguientes fases principales:

1. Análisis:

- Definición de requisitos.
- Priorización de tareas y funcionalidades.
- Diseño de la estructura general del proyecto.

2. Diseño:

- Diseño de la arquitectura.
- Diseño de la interfaz de usuario.
- Creación de wireframes y prototipos.

3. Desarrollo:

- Implementación de funcionalidades.
- Codificación y programación.
- Integración de sistemas y bases de datos.

4. Pruebas:

- Elaboración de casos de prueba.
- Ejecución de pruebas funcionales y de rendimiento.
- Detección y documentación de errores.

Cada fase se adapta a los principios de Kanban, lo que permite una gestión eficiente y flexible de las tareas y una comunicación transparente entre los miembros del equipo.

10.1 Requerimientos Funcionales:

Requerimiento Funcional	Descripción
Registro de Usuarios	<ul style="list-style-type: none"> • Los usuarios deben poder registrarse en la aplicación proporcionando información básica como nombre, apellido, dirección de correo electrónico y contraseña. • Se debe verificar la validez de la dirección de correo electrónico y garantizar que no se utilice la misma dirección para múltiples cuentas.
Inicio de Sesión	<ul style="list-style-type: none"> • Los usuarios registrados deben poder iniciar sesión utilizando su dirección de correo electrónico y contraseña. • El sistema debe autenticar a los usuarios y permitirles acceder a sus perfiles y funcionalidades relacionadas.
Gestión de Equipos	<ul style="list-style-type: none"> • Los equipos deben poder inscribirse en torneos proporcionando detalles como nombre del equipo, información de contacto y lista de jugadores. • Los equipos deben poder editar su información y agregar o eliminar jugadores de la lista.
Programación de partidos	<ul style="list-style-type: none"> • La aplicación debe generar automáticamente la programación de partidos para los torneos en función de los equipos inscritos y sus disponibilidades. • Los usuarios deben poder ver la programación de partidos y recibir notificaciones sobre los próximos juegos.
Registro de Resultados	<ul style="list-style-type: none"> • Los equipos deben poder registrar los resultados de los partidos, incluyendo detalles como marcador y destacados del juego. • La aplicación debe actualizar automáticamente las clasificaciones y estadísticas en función de los resultados de los partidos.
Estadística y Análisis	<ul style="list-style-type: none"> • La aplicación debe proporcionar estadísticas detalladas sobre el rendimiento de los equipos y jugadores, incluyendo goles, asistencias y otros indicadores relevantes. • Los usuarios deben poder realizar análisis personalizados y ver tablas de clasificación.

Tabla 7 - Requerimientos funcionales - Elaboración Propia

10.2 Requerimientos No Funcionales

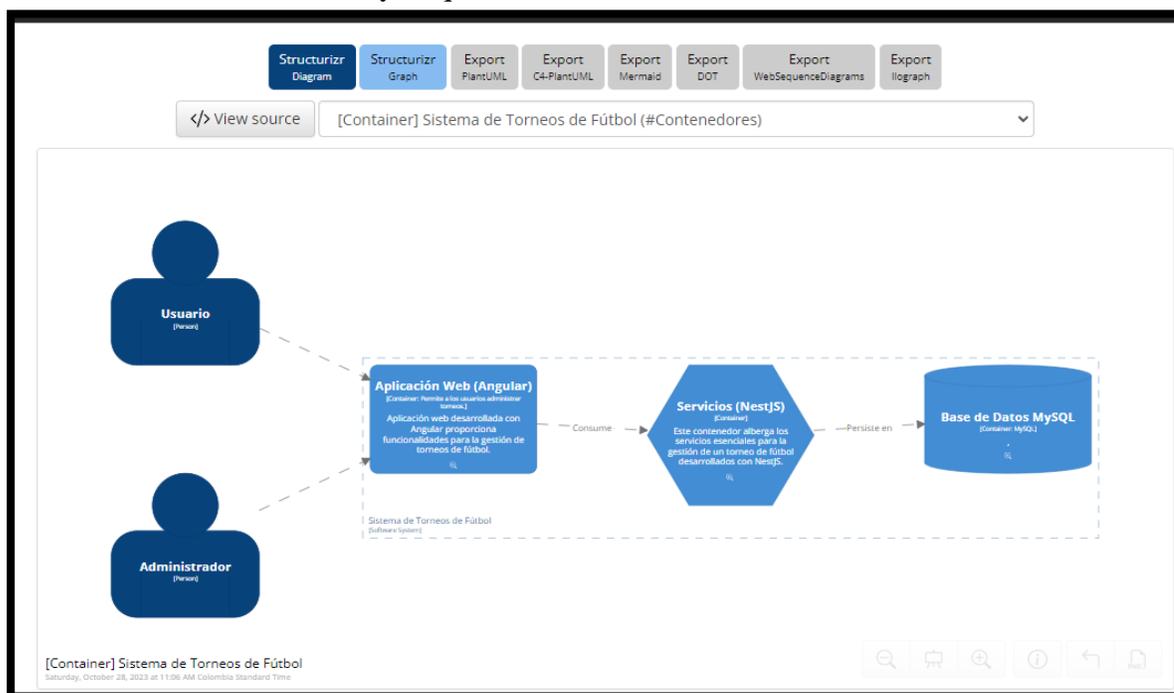
Requerimiento No Funcional	Descripción
Seguridad	<ul style="list-style-type: none"> La aplicación debe implementar medidas sólidas de seguridad para proteger la información de los usuarios y garantizar la privacidad de los datos.
Rendimiento	<ul style="list-style-type: none"> La aplicación debe ser rápida y eficiente, con tiempos de carga mínimos y capacidad de respuesta en todas las funciones.
Escalabilidad	<ul style="list-style-type: none"> La arquitectura de la aplicación debe ser escalable para manejar un crecimiento potencial en el número de usuarios y equipos.
Compatibilidad con Navegadores	<ul style="list-style-type: none"> La aplicación debe ser compatible con una variedad de navegadores y funcionar correctamente en diferentes navegadores web.
Usabilidad	<ul style="list-style-type: none"> La interfaz de usuario debe ser intuitiva y fácil de usar, con una experiencia de usuario agradable.
Disponibilidad	<ul style="list-style-type: none"> La aplicación debe estar disponible y funcionando sin interrupciones la mayor parte del tiempo, con un tiempo de inactividad mínimo programado para mantenimiento.
Cumplimiento Legal	<ul style="list-style-type: none"> La aplicación debe cumplir con las leyes y regulaciones aplicables, incluyendo las relacionadas con la protección de datos y privacidad.
Documentación	<ul style="list-style-type: none"> Debe proporcionarse una documentación completa y actualizada para los usuarios y desarrolladores que describa las características y el funcionamiento de la aplicación.

Tabla 8 - Requerimientos No Funcionales - Elaboración Propia

10.3 Diseño y Arquitectura

En el diseño y arquitectura de este proyecto, se adoptó el patrón Modelo Vista Controlador (MVC) como base estructural. Este enfoque divide la aplicación en tres componentes interconectados: el Modelo, que gestiona la lógica de negocio y la manipulación de datos; la Vista, que se encarga de la presentación y la interfaz de usuario; y el Controlador, que actúa como intermediario, gestionando las interacciones del usuario y coordinando las actualizaciones entre el Modelo y la Vista. Al seguir el paradigma MVC, se logra una separación clara de responsabilidades, facilitando la escalabilidad, el mantenimiento y la evolución de la aplicación de manera eficiente.

Ilustración 2 - Diseño y Arquitectura

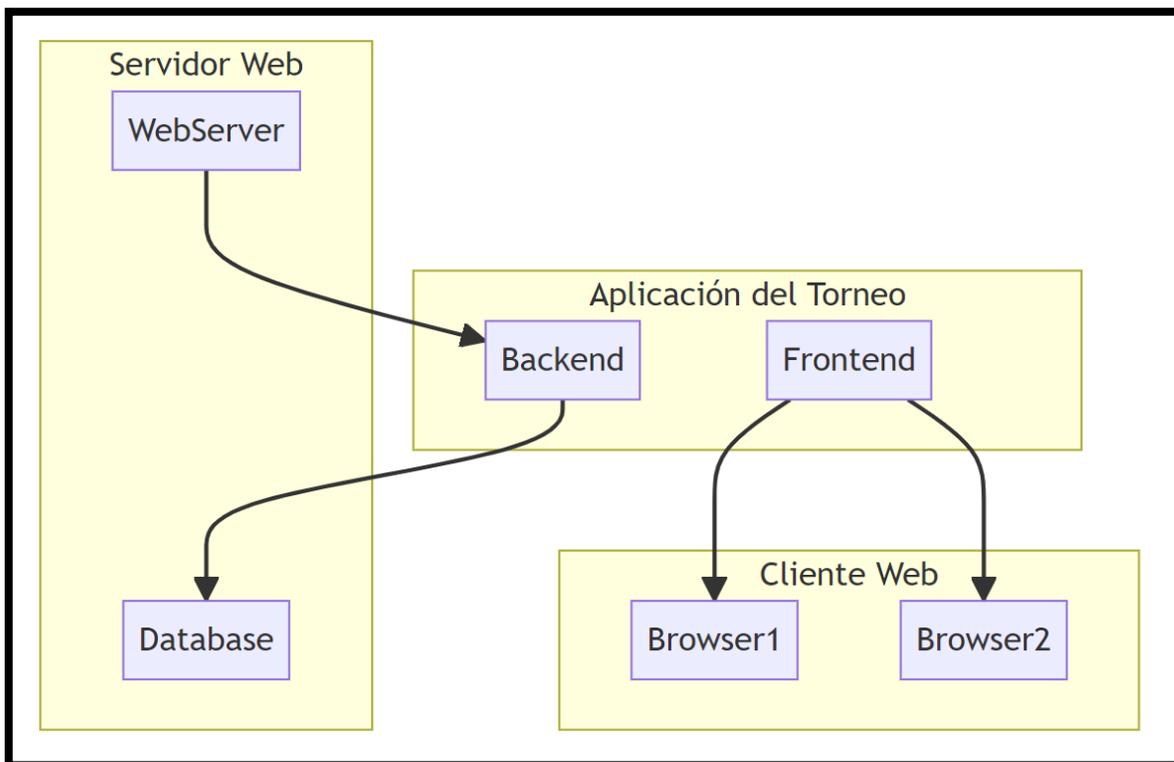


Nota: Elaboración propia (2023)

10.3.1 Diagrama de Despliegue

El diagrama de despliegue de este proyecto proporciona una representación visual clara de la infraestructura técnica necesaria para ejecutar la aplicación. Los nodos del diagrama, que representan los distintos componentes del sistema, incluyen servidores web, bases de datos y servicios externos. Las conexiones entre estos nodos indican las interacciones y comunicaciones necesarias para el funcionamiento adecuado de la aplicación. Este enfoque de diseño permite una comprensión detallada de cómo se distribuyen los recursos y cómo interactúan entre sí, facilitando la planificación y gestión efectiva de la implementación.

Ilustración 3 - Diagrama de Despliegue

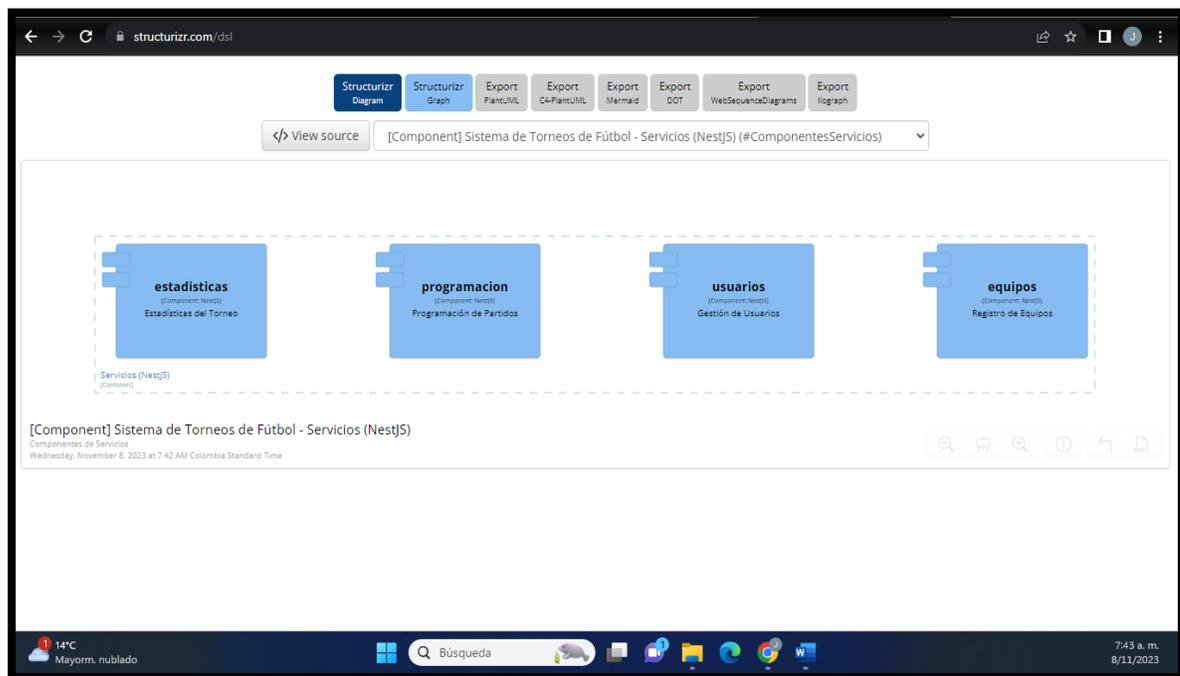


Nota: Elaboración propia (2023)

10.3.2 Caso de uso Arquitectura relevante

En el contexto de la arquitectura de este proyecto, un caso de uso relevante es la interacción entre los diferentes actores y el sistema. Un ejemplo de ello es el caso de uso "Gestión de Torneos", donde el administrador puede crear, modificar y eliminar torneos, mientras que los participantes pueden inscribirse en torneos existentes. Este caso de uso demuestra la integración fluida de los actores con el sistema, permitiendo una gestión eficiente de los torneos a través de una interfaz clara y accesible.

Ilustración 4 - Caso de Uso Arquitectura Relevante

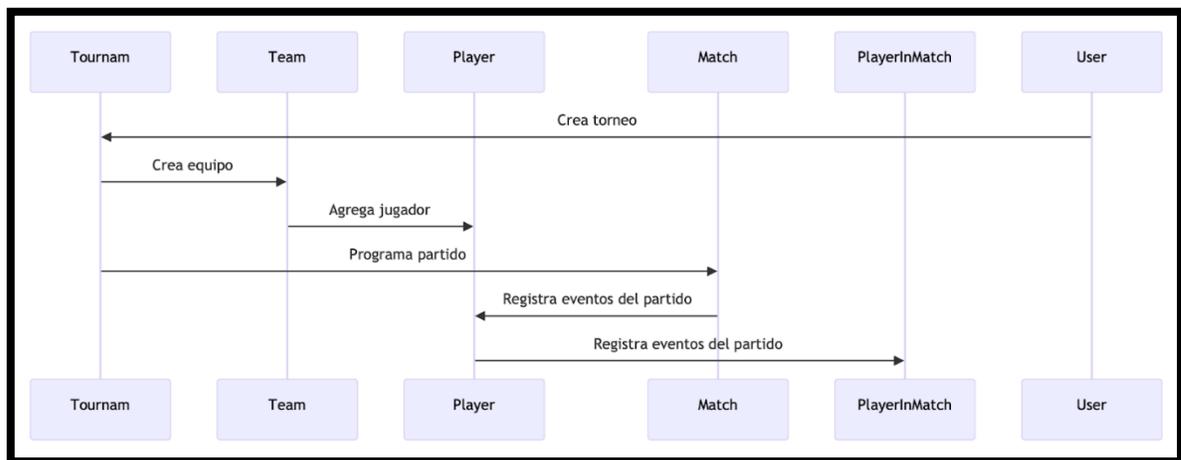


Nota: Elaboración propia (2023)

10.3.3 Diagrama de secuencia

El diagrama de secuencia de este proyecto ofrece una visión detallada de la interacción entre los diferentes objetos y componentes del sistema a lo largo del tiempo. En un escenario específico, como la inscripción de un equipo en un torneo, el diagrama de secuencia muestra las llamadas de métodos y los mensajes intercambiados entre los objetos involucrados, revelando la secuencia precisa de eventos. Esta representación visual facilita la comprensión de los flujos de trabajo y la identificación de posibles puntos de optimización en el proceso de inscripción.

Ilustración 5 - Diagrama de secuencia

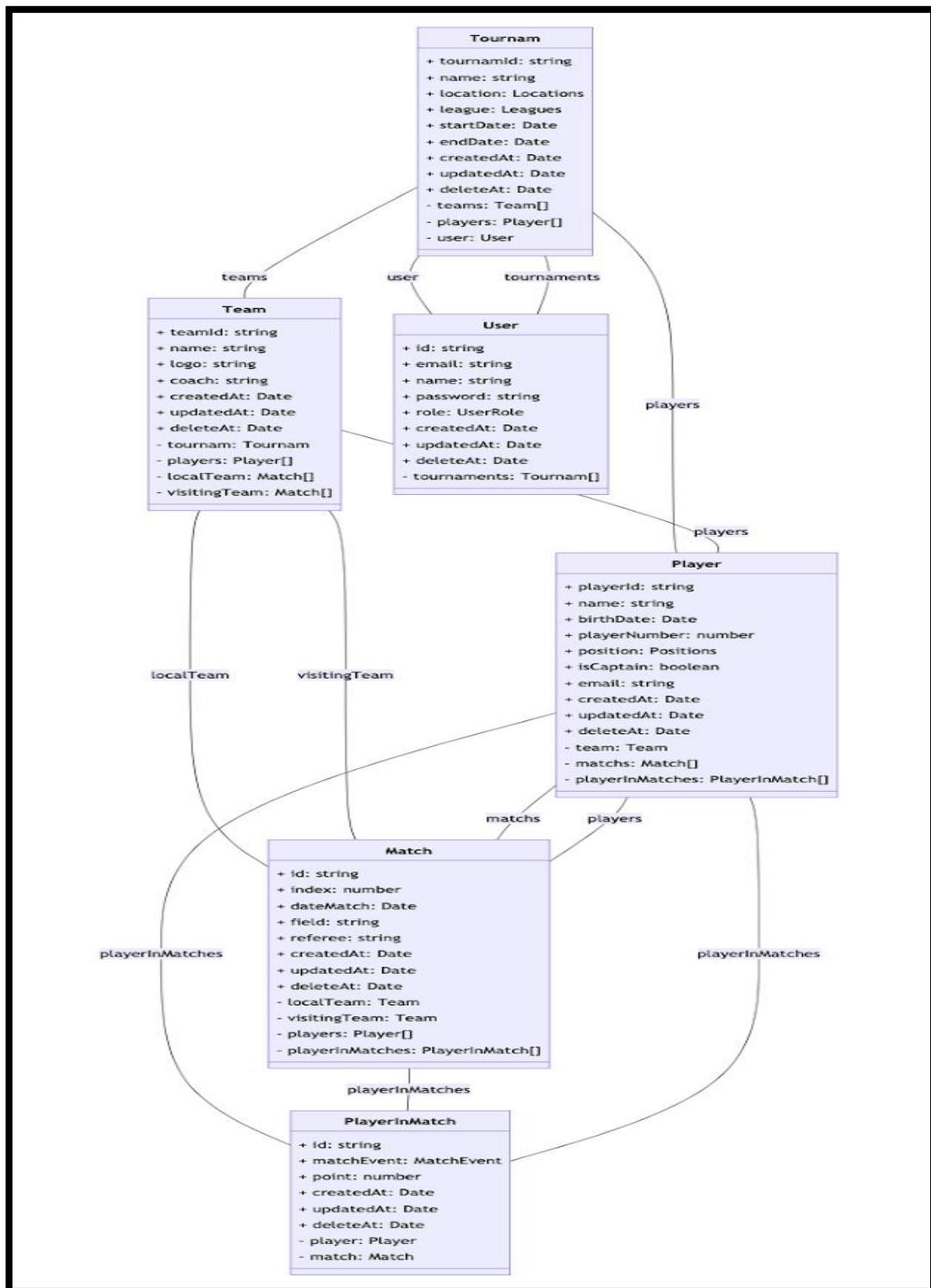


Nota: Elaboración propia (2023)

10.3.4 Diagrama de clases

El diagrama de clases de este proyecto ofrece una representación visual de las estructuras de clases y las relaciones entre ellas en el sistema. Cada clase, identificada con sus atributos y métodos, encapsula la lógica y los datos relacionados. Las asociaciones entre clases ilustran las conexiones y dependencias entre los distintos componentes, proporcionando una visión integral de la arquitectura del sistema. Este diagrama es esencial para comprender la organización de las entidades en el sistema y facilita el diseño y la implementación efectivos.

Ilustración 6 - Diagrama de clases

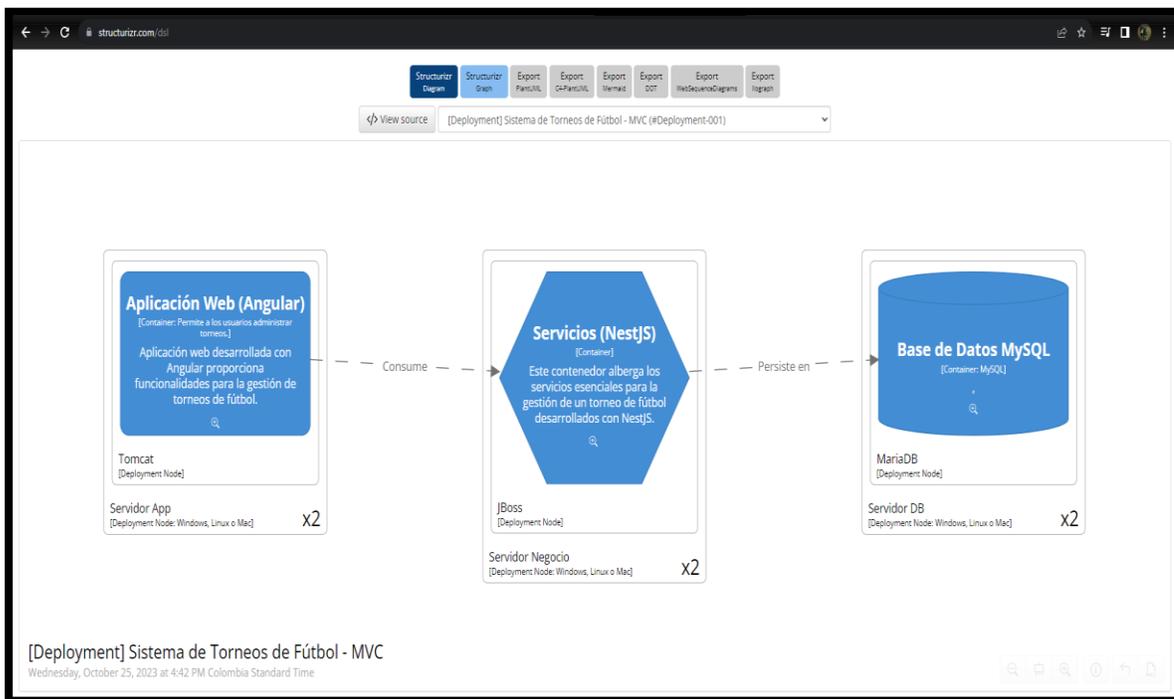


Nota: Elaboración propia (2023)

10.3.5 Arquitectura de Alto Nivel

El Diagrama de Arquitectura de Alto Nivel proporciona una visión panorámica de la estructura general del sistema, identificando los componentes clave y sus interconexiones. En este proyecto, el diagrama destaca las capas fundamentales, como la interfaz de usuario, la lógica de negocio y la capa de datos. Las flechas indican la dirección de las interacciones, ilustrando cómo fluye la información a través de las distintas capas. Este enfoque de alto nivel facilita la comprensión rápida de la arquitectura general del sistema, siendo una herramienta valiosa para los desarrolladores, arquitectos y otros stakeholders.

Ilustración 7 - Arquitectura de Alto Nivel (MVC)



Nota: Elaboración propia (2023)

11. Construcción

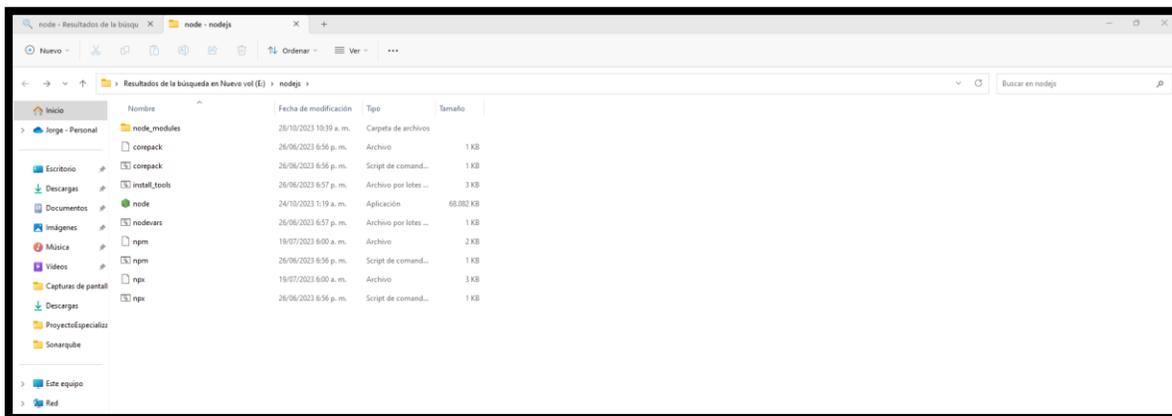
La fase de construcción en este proyecto representa el núcleo del desarrollo, donde se traducen los conceptos y diseños en una aplicación funcional. Durante esta etapa, los desarrolladores trabajan en colaboración, implementando las diversas características definidas en las etapas de diseño. Se lleva a cabo la codificación, la integración de componentes y la resolución de posibles problemas que puedan surgir. La construcción implica una iteración constante y pruebas continuas para garantizar la coherencia y calidad del código. Además, se enfoca en la optimización del rendimiento y la adaptabilidad del sistema a medida que evoluciona. La atención se centra tanto en la funcionalidad como en la eficiencia, asegurando que la aplicación cumpla con los requisitos especificados y proporcione una experiencia de usuario fluida y efectiva.

La fase de construcción del software en este proyecto se realizó de manera integral, con el desarrollo del back-end utilizando NestJS, mientras que el front-end se construyó con Angular. Esta fase comprendió varias etapas clave:

1. Configuración del Entorno de Desarrollo:

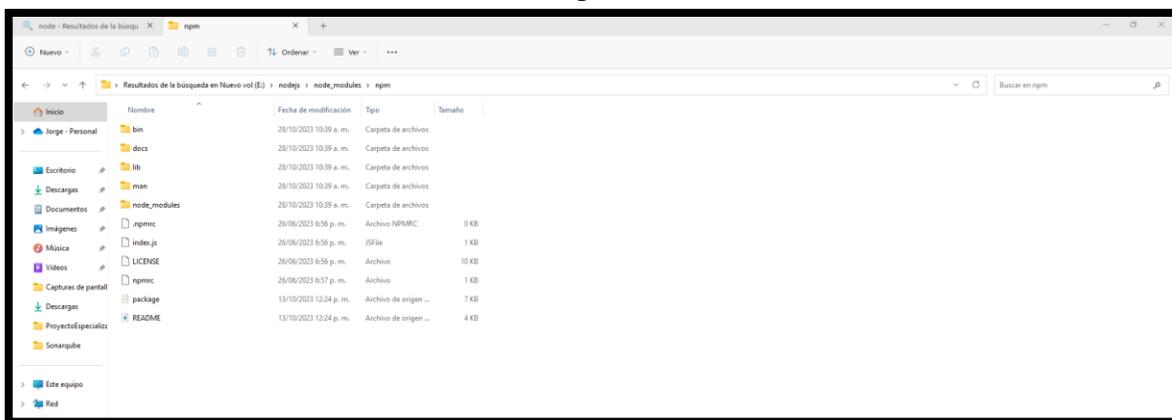
- Se estableció un entorno de desarrollo compatible con TypeScript y NestJS para el back-end, y Angular para el front-end.
- Las dependencias y herramientas necesarias, como Node.js y npm, se instalaron y configuraron.

Ilustración 8 - Entorno de Desarrollo Node JS



Nota: Elaboración propia (2023)

Ilustración 9 - Entorno de desarrollo npm



Nota: Elaboración propia (2023)

2. Desarrollo del Back-end con TypeScript y NestJS:

- Se implementó la lógica de negocio del back-end utilizando el lenguaje de programación TypeScript, aprovechando las capacidades de este superset de JavaScript.

- Se utilizaron los principios de diseño de NestJS para crear controladores, módulos y servicios que gestionaran las funciones del servidor.

Ilustración 10 - Backend

```

1 import { BadRequestException, Injectable, InternalServerErrorException, NotFoundException } from '@nestjs/common';
2 import { InjectRepository } from '@nestjs/typeorm';
3 import { generate as short } from 'short-uuid';
4 import { AuthService } from 'src/auth/auth.service';
5 import { Repository } from 'typeorm';
6 import { Tournament } from './dto';
7 import { Tournament } from './entities/tournament.entity';
8
9 @Injectable()
10 export class TournamentService {
11
12   constructor(
13     @InjectRepository(Tournament)
14     private readonly tournamentRepository: Repository<Tournament>,
15     private readonly authService: AuthService,
16   ) {}
17
18   async createTournament(tournamentDto: Tournament): Promise<Tournament> {
19     const existingUser = await this.authService.findUserById(String(tournamentDto.user));
20     if (!existingUser) {
21       throw new NotFoundException('Usuario con ID: ${tournamentDto.user} no encontrado.');

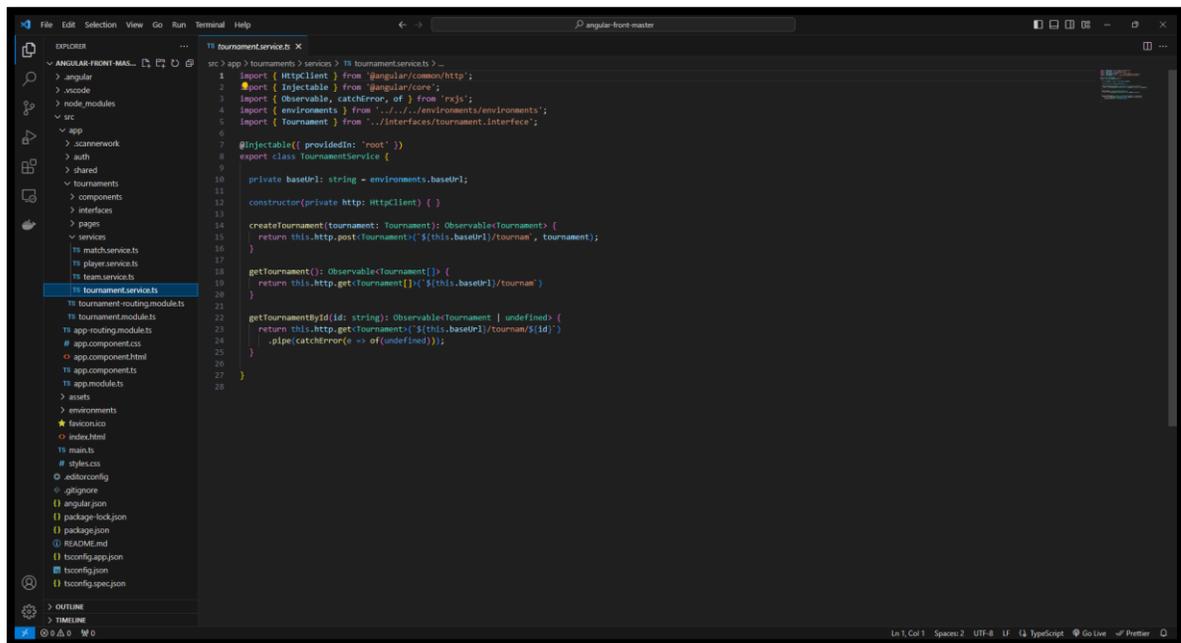
```

Nota: Elaboración propia (2023)

3. Desarrollo del Front-end con Angular:

- Se crearon componentes, servicios y módulos en Angular para la interfaz de usuario.
- Se implementó la lógica de interacción del usuario, la manipulación del DOM y las llamadas a la API del back-end.

Ilustración 11 - Llamado a las apis del Backend



```
src > app > tournaments > services > tournament.service.ts >
1 import { HttpClient } from '@angular/common/http';
2 import { Injectable } from '@angular/core';
3 import { Observable, catchError, of } from 'rxjs';
4 import { environments } from '../../environments/environments';
5 import { Tournament } from '../interfaces/tournament.interface';
6
7 @injectable({ providedIn: 'root' })
8 export class TournamentService {
9
10   private baseUrl: string = environments.baseUrl;
11
12   constructor(private http: HttpClient) {}
13
14   createTournament(tournament: Tournament): Observable<Tournament> {
15     return this.http.post<Tournament>(`${this.baseUrl}/tournament`, tournament);
16   }
17
18   getTournament(): Observable<Tournament[]> {
19     return this.http.get<Tournament[]>(`${this.baseUrl}/tournament`);
20   }
21
22   getTournamentById(id: string): Observable<Tournament | undefined> {
23     return this.http.get<Tournament>(`${this.baseUrl}/tournament/${id}`)
24       .pipe(catchError(e => of(undefined)));
25   }
26 }
27
28
```

Nota: Elaboración propia (2023)

4. Pruebas

La fase de pruebas en este proyecto despliega un enfoque meticuloso para garantizar la calidad y confiabilidad del software desarrollado. Se implementan diversos niveles de pruebas, desde pruebas unitarias hasta pruebas de sistema, para evaluar la funcionalidad, la integridad y el rendimiento de la aplicación. Durante este proceso, se identifican y corrigen posibles errores, se verifica la conformidad con los requisitos establecidos y se asegura la coherencia del sistema en diferentes escenarios. La fase de pruebas no solo busca detectar defectos, sino también validar que la aplicación cumple con las expectativas y necesidades del usuario final, garantizando así un producto final robusto y de alta calidad.

a. De Proceso

Las pruebas de proceso son un conjunto de pruebas funcionales diseñadas para evaluar la ejecución y el flujo de un proceso en un aplicativo web. Para garantizar que los procesos importantes se lleven a cabo de manera adecuada, estas pruebas se enfocan en verificar la interacción entre las diversas funciones y características de la aplicación. El objetivo principal es garantizar la coherencia y eficiencia en la ejecución de las operaciones críticas necesarias para que la aplicación funcione correctamente. Basado en una serie de requisitos que deben cumplirse para poder llevar a cabo las pruebas, así como para determinar si están cumpliendo con las reglas comerciales acordadas inicialmente por parte de las partes interesadas. A continuación, explicaremos cómo se deben llevar a cabo o qué características deben tener las pruebas de proceso.

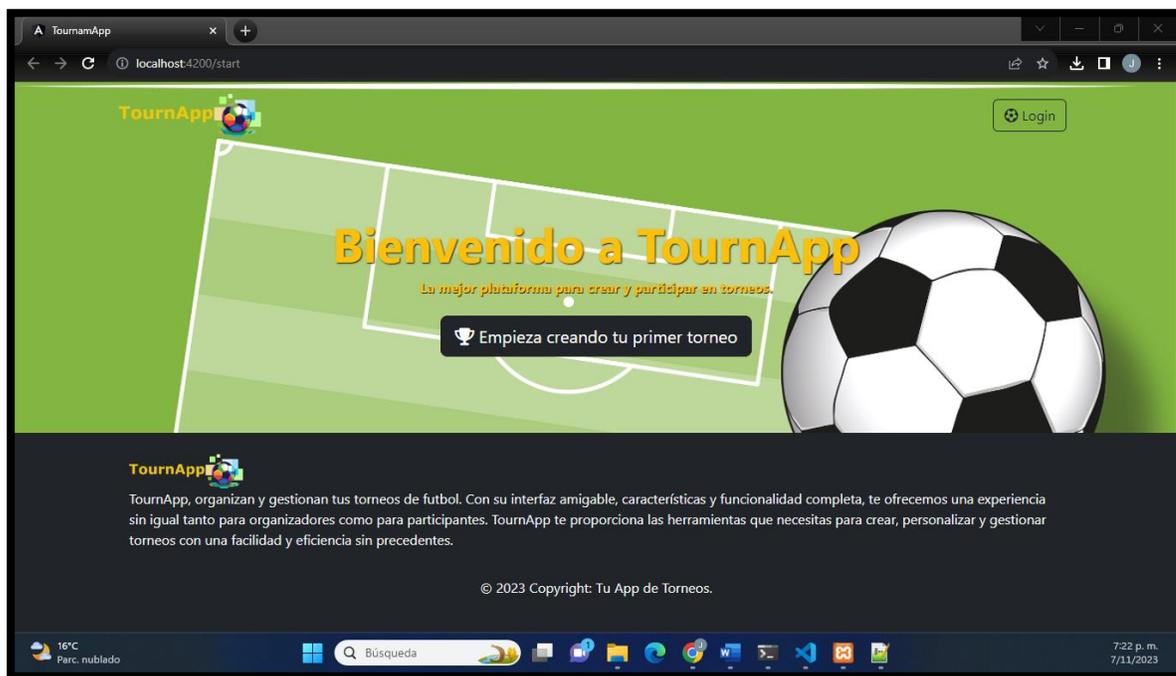
Característica	Descripción
Secuencia de Procesos	<ul style="list-style-type: none"> Las pruebas de proceso verifican que las operaciones y las actividades de la aplicación siguen el flujo lógico previsto.
Interacción de Funcionalidades	<ul style="list-style-type: none"> Se enfocan en cómo las diversas funciones y módulos de la aplicación interactúan entre sí para garantizar la coherencia y la integridad de los datos a lo largo del proceso.
Validación de Datos	<ul style="list-style-type: none"> En cada paso del proceso, verifican la entrada y la salida de datos para garantizar la precisión y la consistencia de los datos.
Manejo de Excepciones	<ul style="list-style-type: none"> Evalúan cómo la aplicación maneja situaciones excepcionales o errores durante la ejecución del proceso, asegurándose de que se proporcionen mensajes de error claros y asegurándose de que la aplicación mantenga un estado coherente.

Eficiencia del Proceso	<ul style="list-style-type: none"> • Evalúan el rendimiento general del aplicativo web durante la ejecución de procesos clave para detectar cuellos de botella o retrasos imprevistos.
Cumplimientos de Requisitos	<ul style="list-style-type: none"> • Verifican que el proceso cumpla con los requisitos funcionales y comerciales de la aplicación.

Tabla 9 - Tabla de Pruebas de Proceso - Elaboración Propia

Para las pruebas de proceso cumplieran con las anteriores condiciones se utilizaron los flujos funcionales y se dará un ejemplo de dos de ellos dónde se verán las evidencias del aplicativo web que fueron probados en el marco del proyecto. Nos basamos en una creación de casos o escenarios de prueba utilizando el framework de Katalon y su entorno gráfico para hacer una descripción de los casos contemplados, agregando una metodología de equipo como es la BDD (Behavior Drive Development). Basándonos que en BDD se enfoca más en una prueba de alto nivel, la prueba funcional o la prueba de aceptación. Para la construcción de los casos de prueba en una posterior mejora del aplicativo se tendrá la base para realizar unas pruebas de regresión y a su vez automatizarlas. Luego se presentará un caso de prueba que contiene cuatro escenarios, dónde se mostrará el marco de trabajo y cómo se adaptó al proyecto para realizar pruebas funcionales automatizadas y manuales.

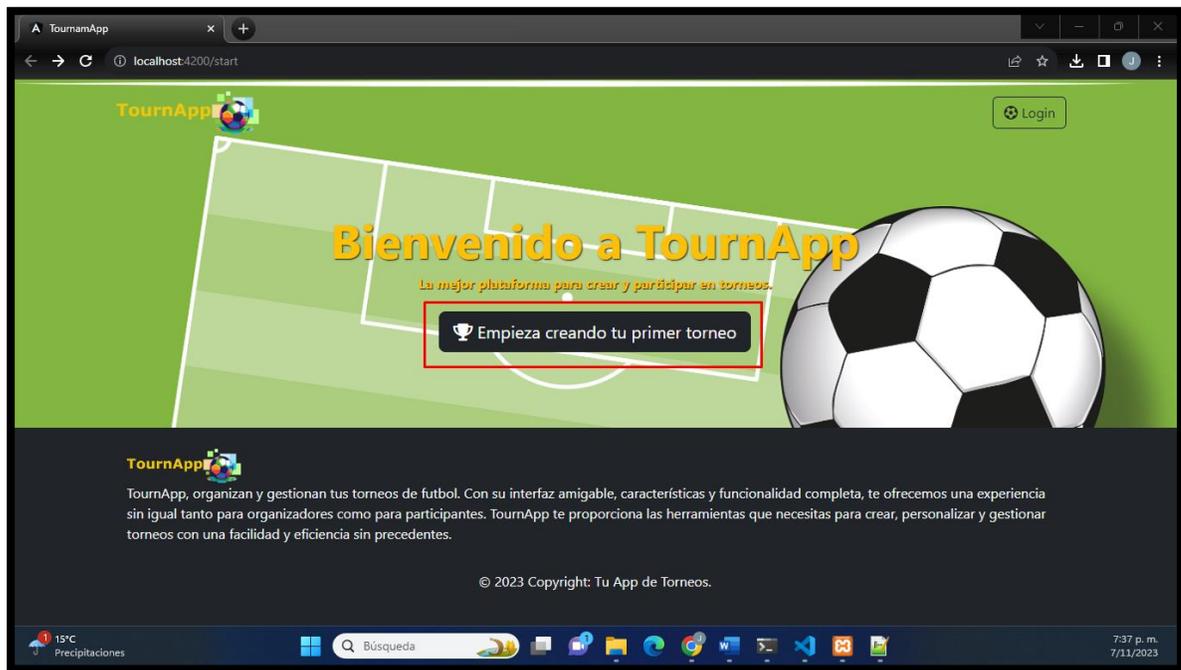
Ilustración 12 - Pantalla Inicial



Nota: Elaboración propia (2023)

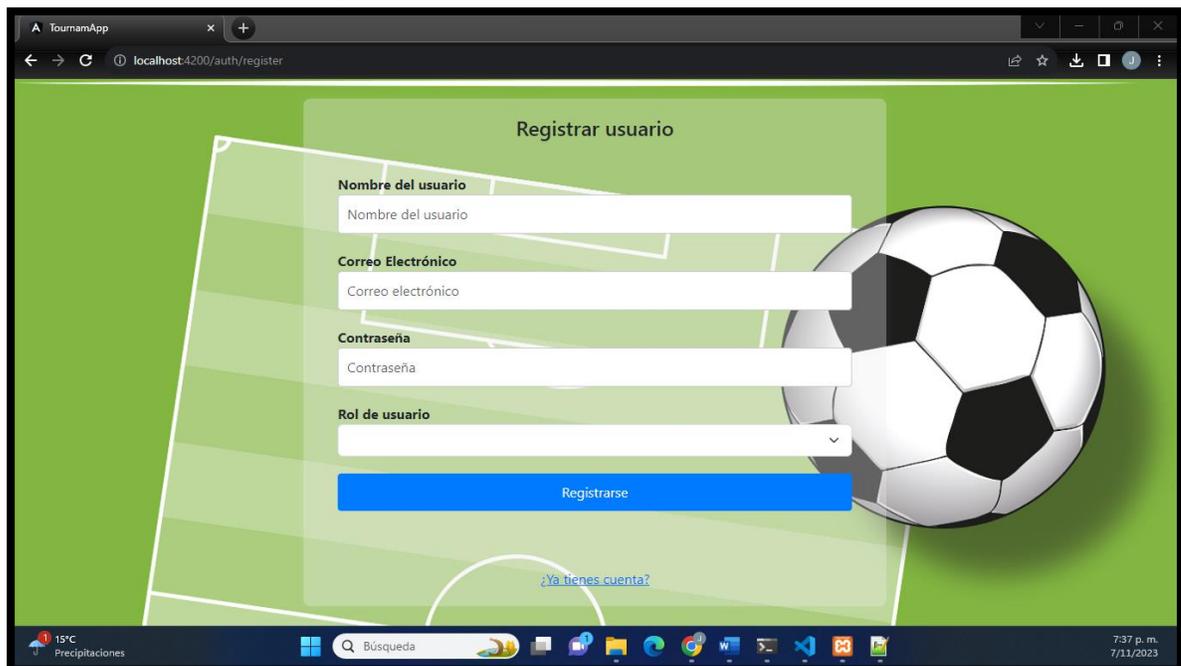
La pantalla inicial tendrá dos opciones. Uno para registrarse como usuario y ver el torneo en el que está participando, y otro para crear un nuevo torneo. Para este ejemplo no se tiene un torneo creado, así que se hará un flujo ent to end para su creación.

Ilustración 13 - Create First Tournament



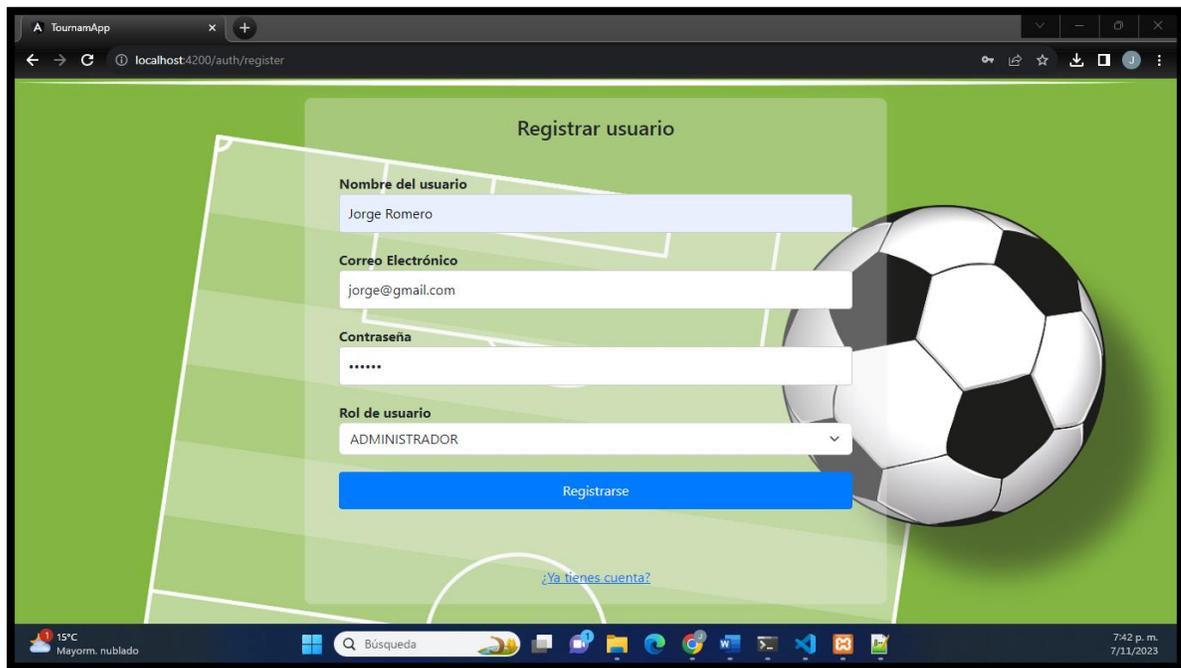
Nota: Elaboración propia (2023)

Ilustración 14 - Register User



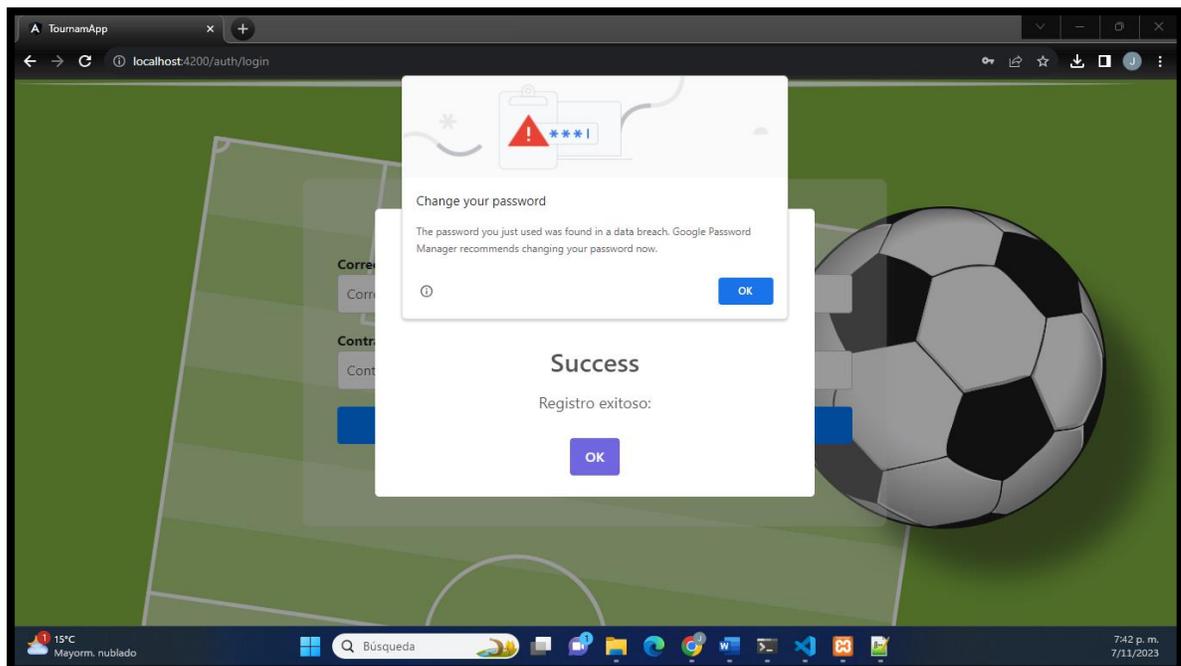
Nota: Elaboración propia (2023)

Ilustración 15 - Create User



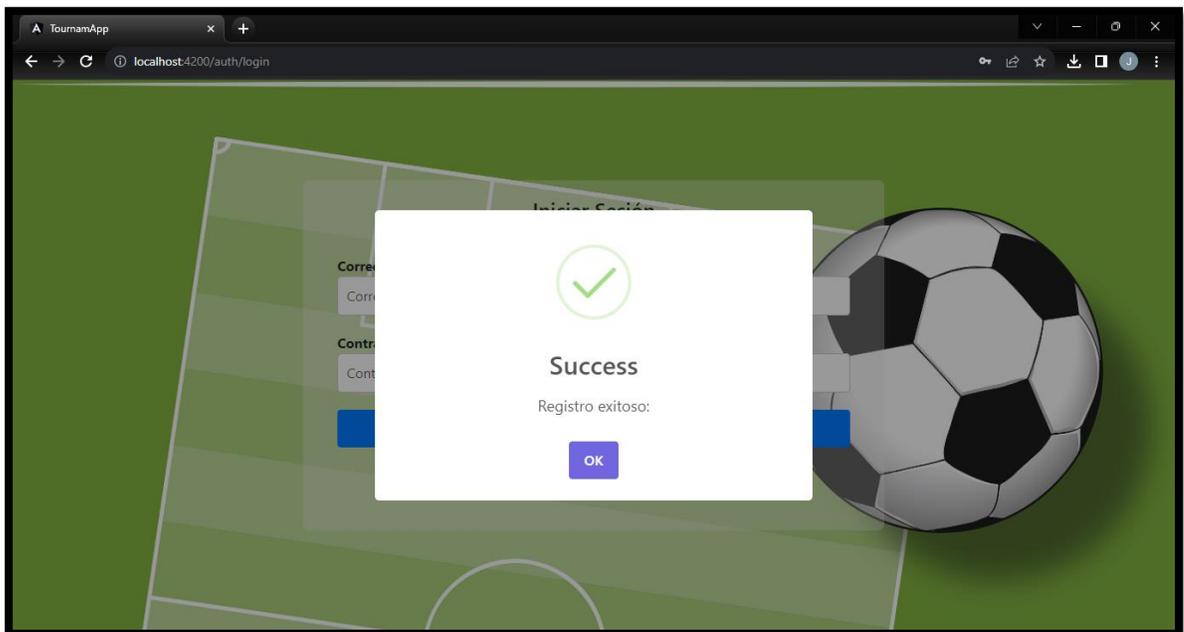
Nota: Elaboración propia (2023)

Ilustración 16 - Registro Exitoso



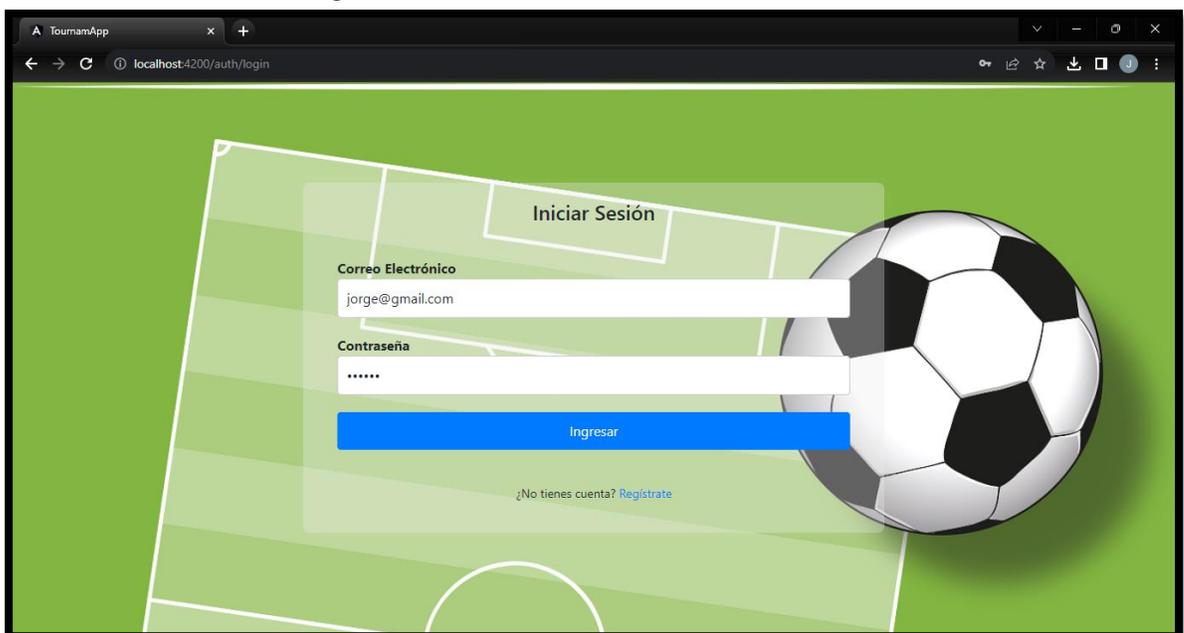
Nota: Elaboración propia (2023)

Ilustración 17 - Confirmación



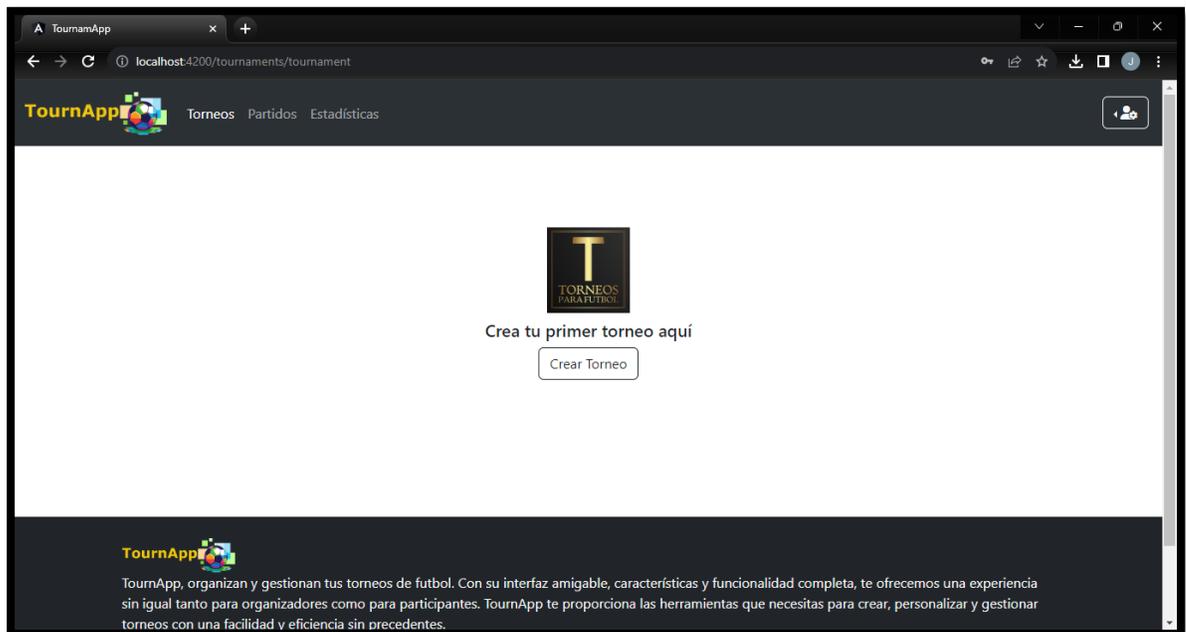
Nota: Elaboración propia (2023)

Ilustración 18 - Login Exitoso



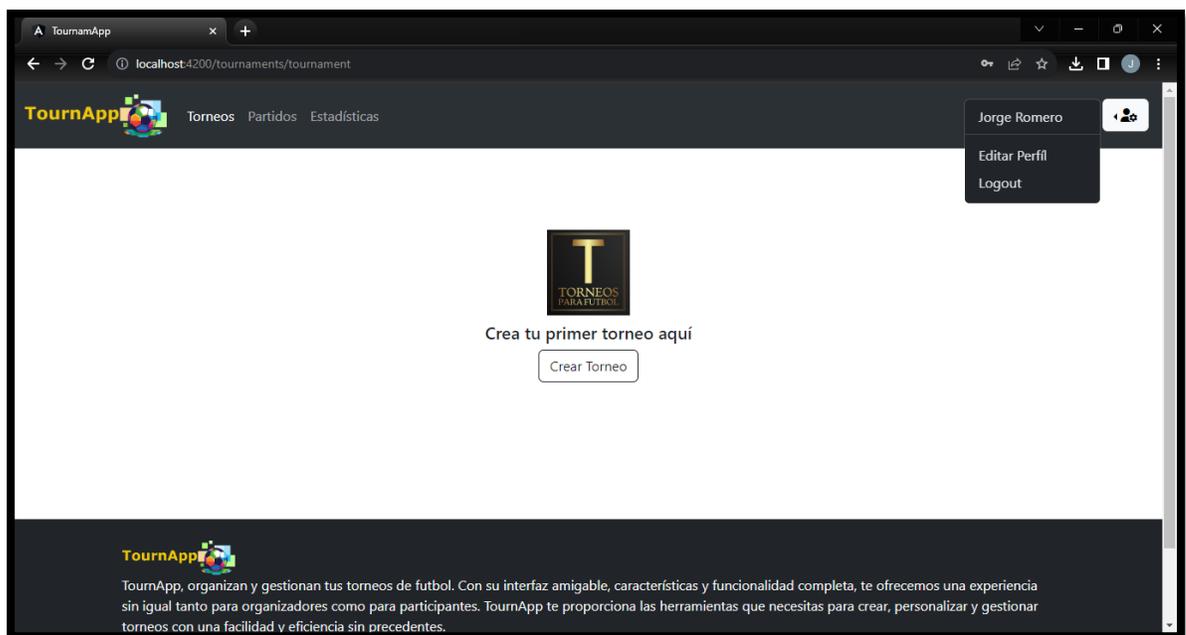
Nota: Elaboración propia (2023)

Ilustración 19 - Opción Creación de Torneo



Nota: Elaboración propia (2023)

Ilustración 20 - Autenticación de Usuario



Nota: Elaboración propia (2023)

Ilustración 21 - Creación de Torneo

Nuevo Torneo

Nombre del Torneo
Torneo Quintanares

Fecha de Inicio
11/07/2023
La fecha de inicio debe ser posterior a la fecha actual.

Fecha de Finalización (opcional)
mm/dd/yyyy

Ubicación
Comuna 4. Cazuca

Liga
VETERANO

Cerrar Crear

Nota: Elaboración propia (2023)

Ilustración 22 - Creación de Torneo Exitoso

Nuevo Torneo

Nombre del Torneo

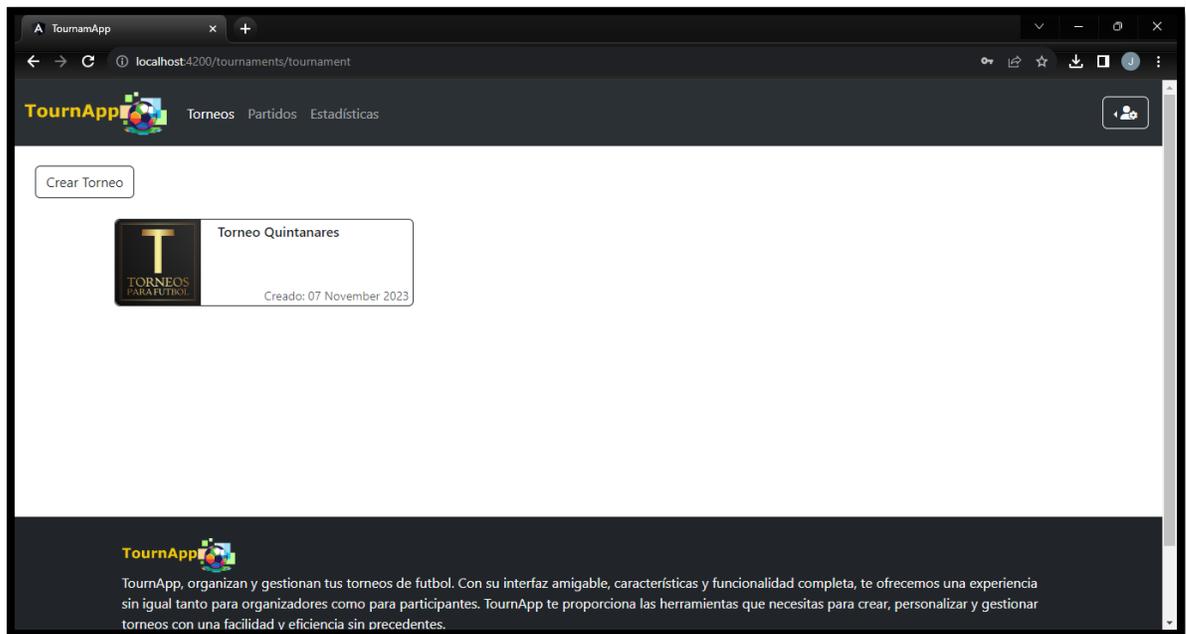
Success
Torneo creado con éxito! Torneo Quintanares

OK

Cerrar Crear

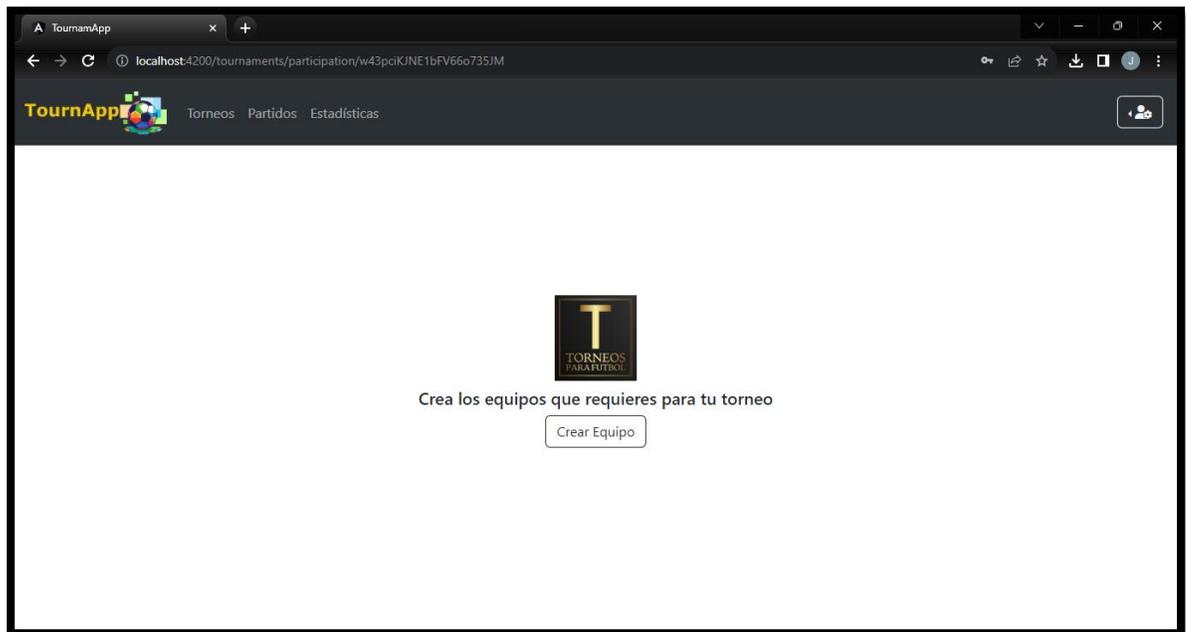
Nota: Elaboración propia (2023)

Ilustración 23 - Verificación de Creación de Torneo



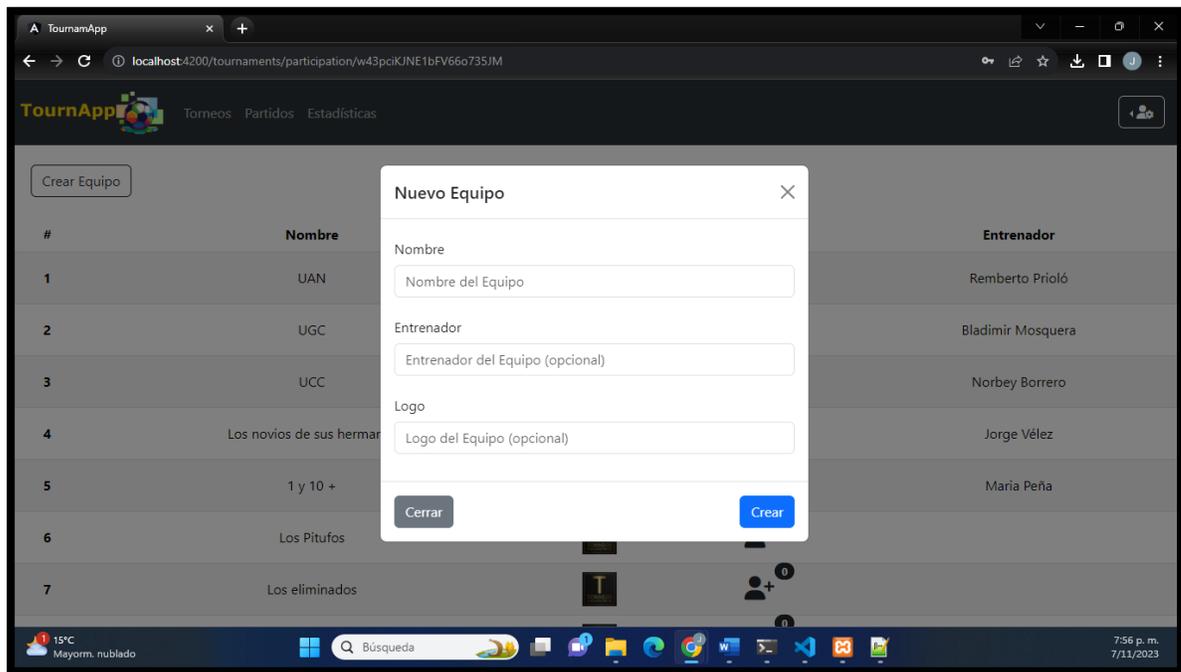
Nota: Elaboración propia (2023)

Ilustración 24 - Creación de Equipos para el Torneo



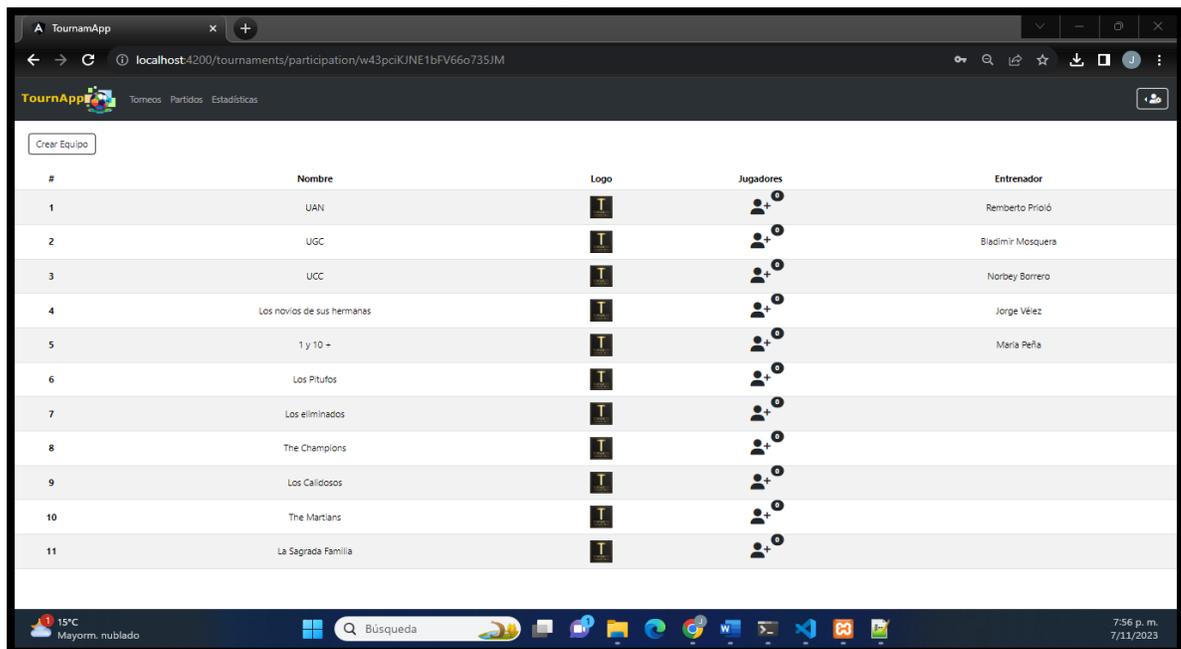
Nota: Elaboración propia (2023)

Ilustración 25 - Registro Equipo



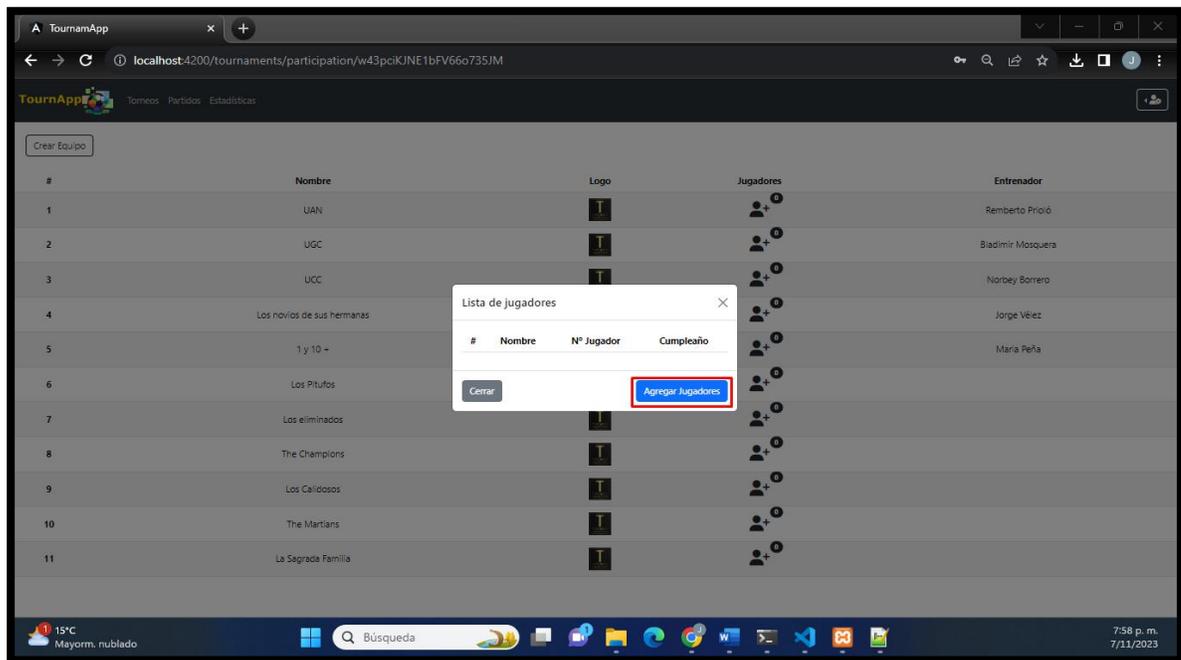
Nota: Elaboración propia (2023)

Ilustración 26 - Verificación de Equipos creados



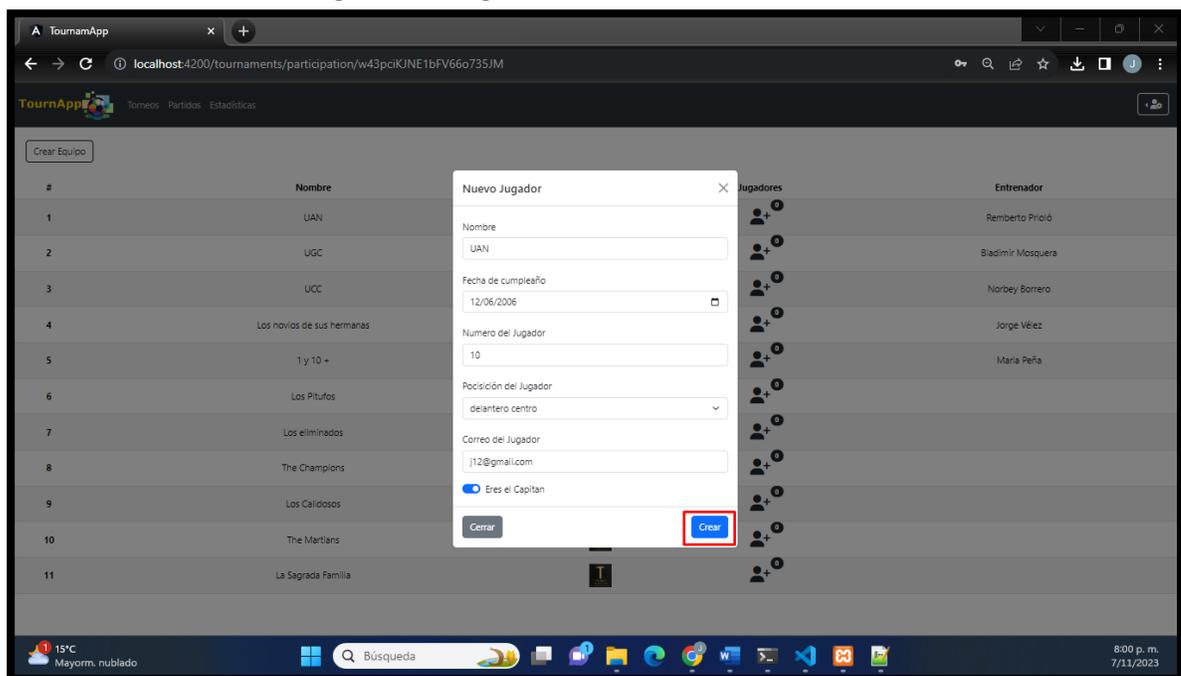
Nota: Elaboración propia (2023)

Ilustración 27 - Creación de Jugadores y asociarlos a un Equipo



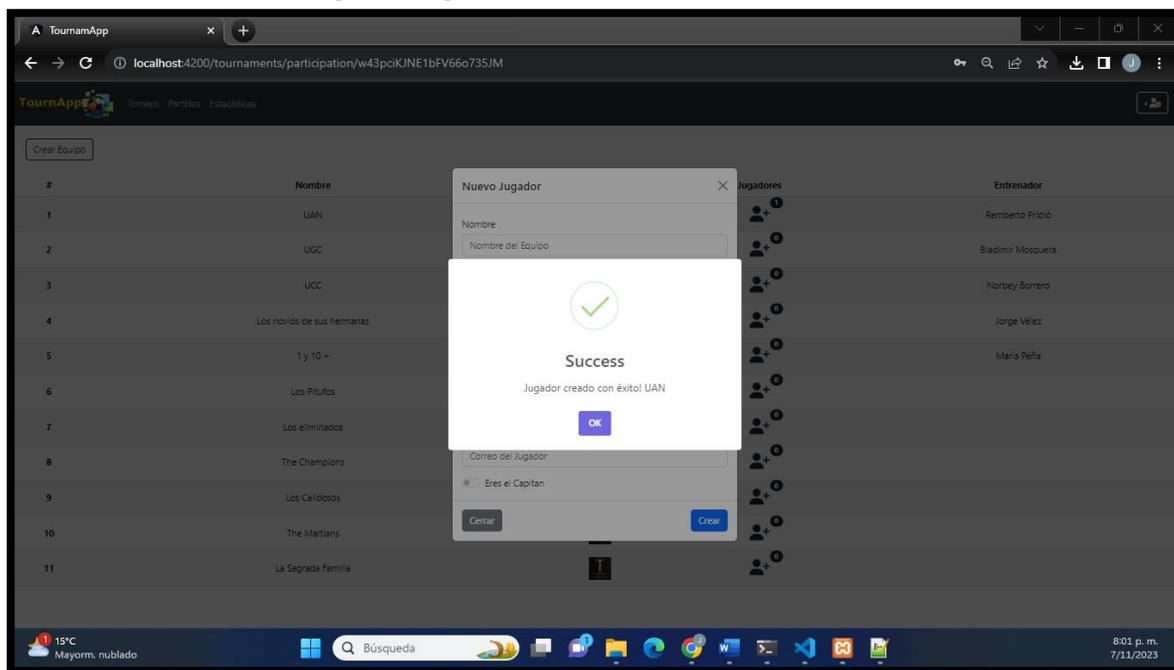
Nota: Elaboración propia (2023)

Ilustración 28 - Registro de Jugador



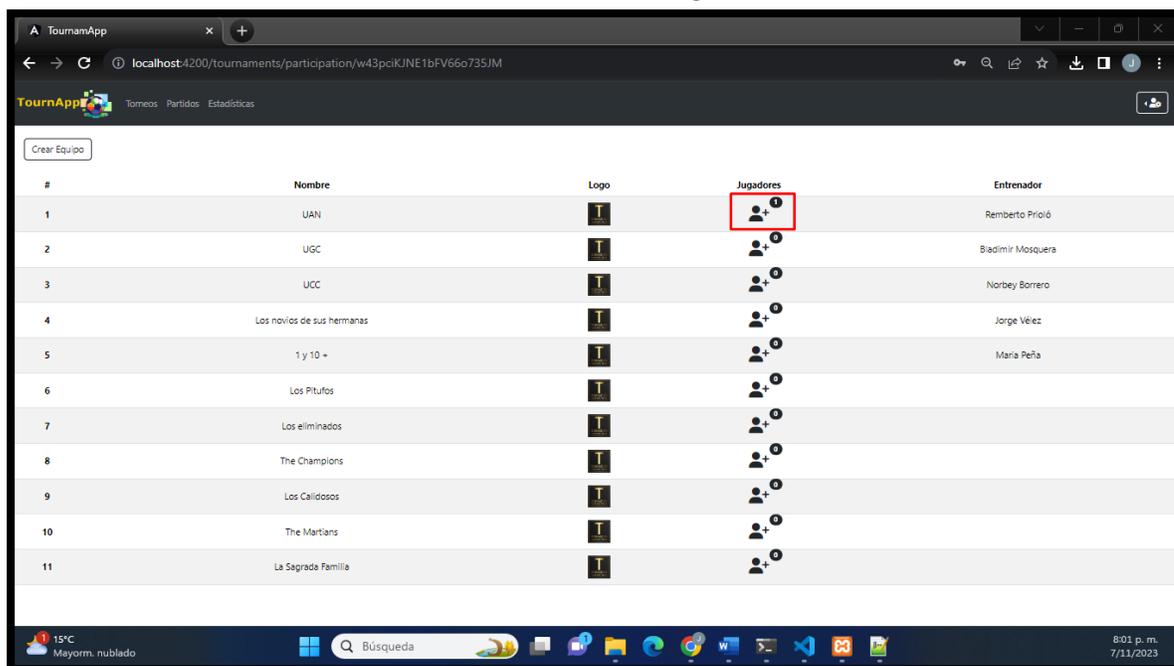
Nota: Elaboración propia (2023)

Ilustración 29 - Registro Jugador Exitoso



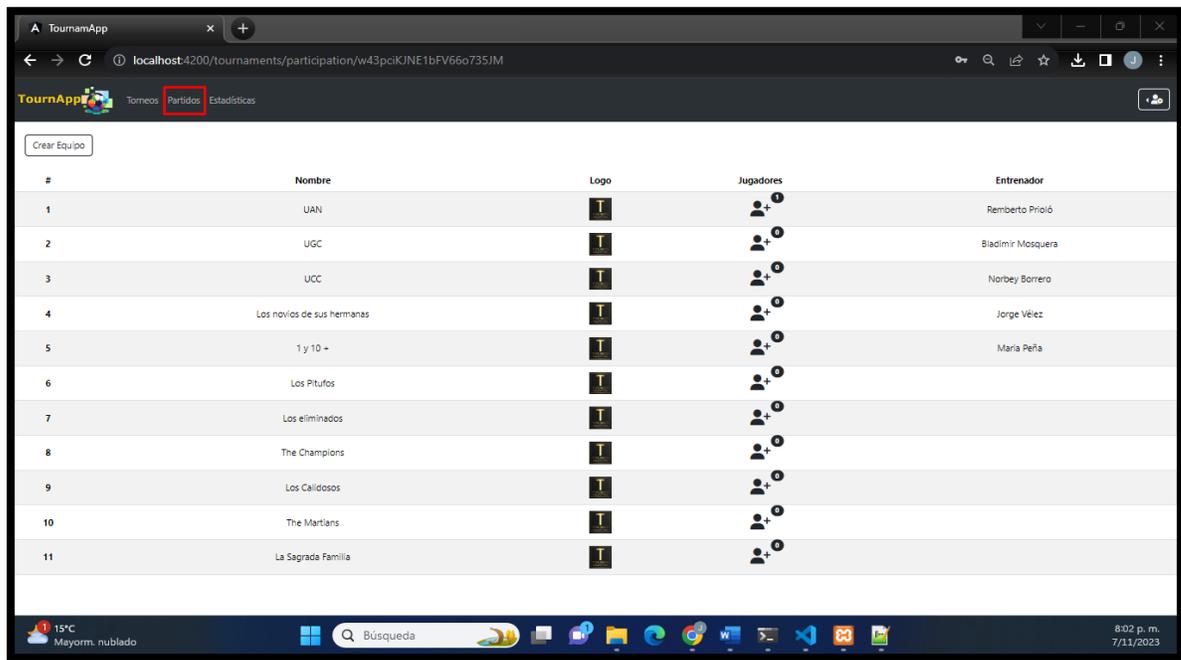
Nota: Elaboración propia (2023)

Ilustración 30 - Verificación de Creación de Jugador Exitoso



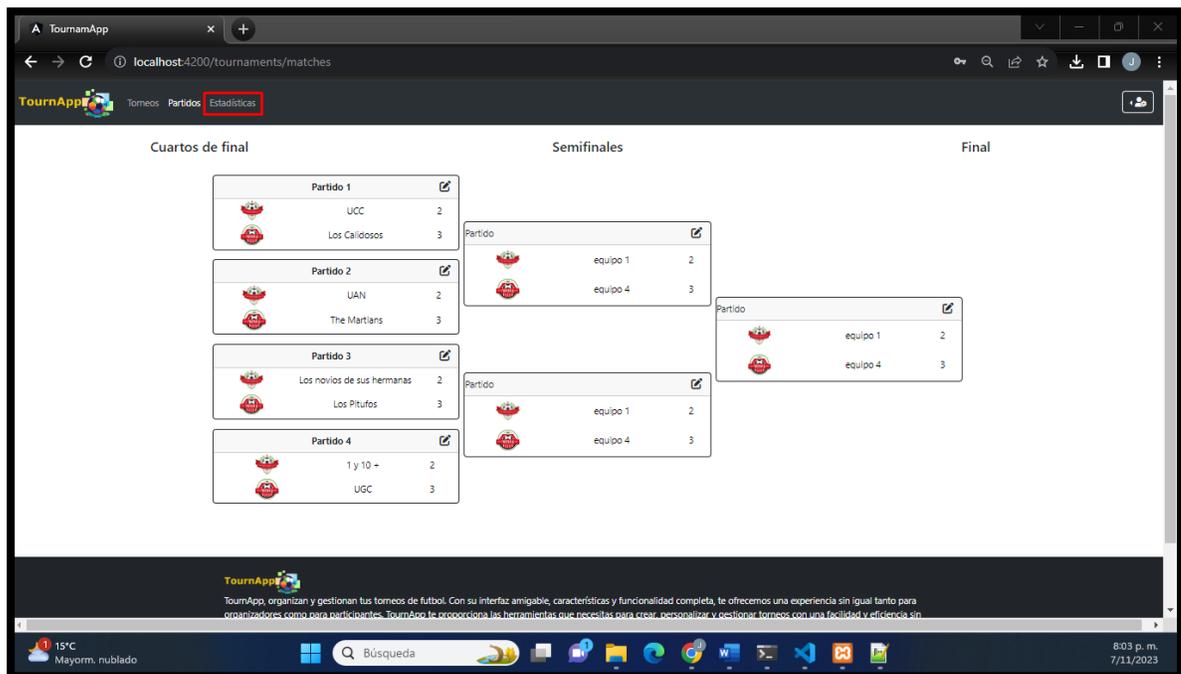
Nota: Elaboración propia (2023)

Ilustración 31 - Módulo de Partidos



Nota: Elaboración propia (2023)

Ilustración 32 - Modulo Pantalla de Partidos



Nota: Elaboración propia (2023)

Ilustración 33 - Módulo Pantalla Estadísticas Jugadores

The screenshot shows a web browser window displaying the 'Estadísticas del Jugador' page. The page has a dark header with the TournApp logo and navigation links. The main content area contains four tables, each with a title and a button to send statistics via email. The tables are:

- Máximos Goleadores** (Green header): Columns: #, Equipo, Jugador, Goles.
- Máximas Tarjetas Rojas** (Red header): Columns: #, Equipo, Jugador, Tarjetas Rojas.
- Máximas Tarjetas Amarillas** (Yellow header): Columns: #, Equipo, Jugador, Tarjetas Amarillas.
- Máximas Tarjetas Azules** (Blue header): Columns: #, Equipo, Jugador, Asistentes.

A button 'Enviar estadísticas por correo' is located in the top right corner of the main content area. The footer contains a TournApp logo and a short description of the application's purpose.

Nota: Elaboración propia (2023)

Ilustración 34 - Funcionalidad de Envío de notificaciones a E-mail.

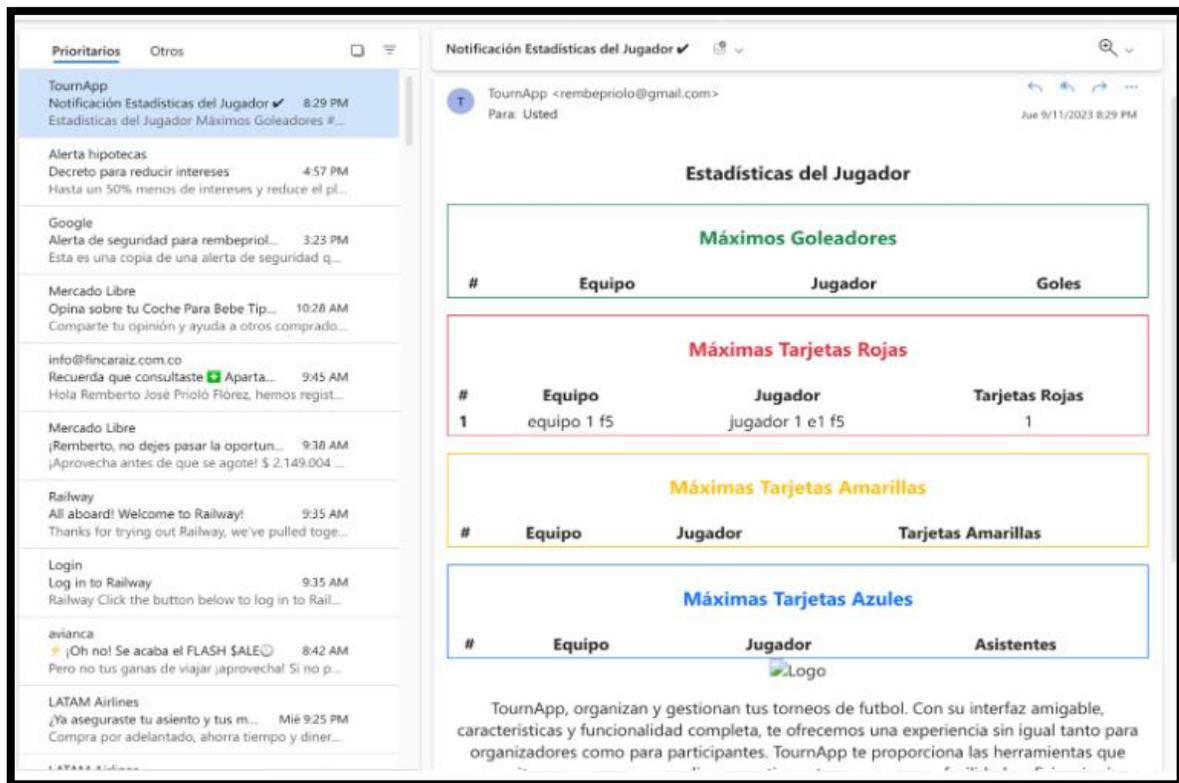
The screenshot shows the same 'Estadísticas del Jugador' page as in Illustración 33, but with a success notification overlay. The notification is a white box with a green checkmark icon and the text:

Success
Los correos se han enviado exitosamente

There is an 'OK' button at the bottom of the notification. The background content is dimmed, showing the same tables and the 'Enviar estadísticas por correo' button.

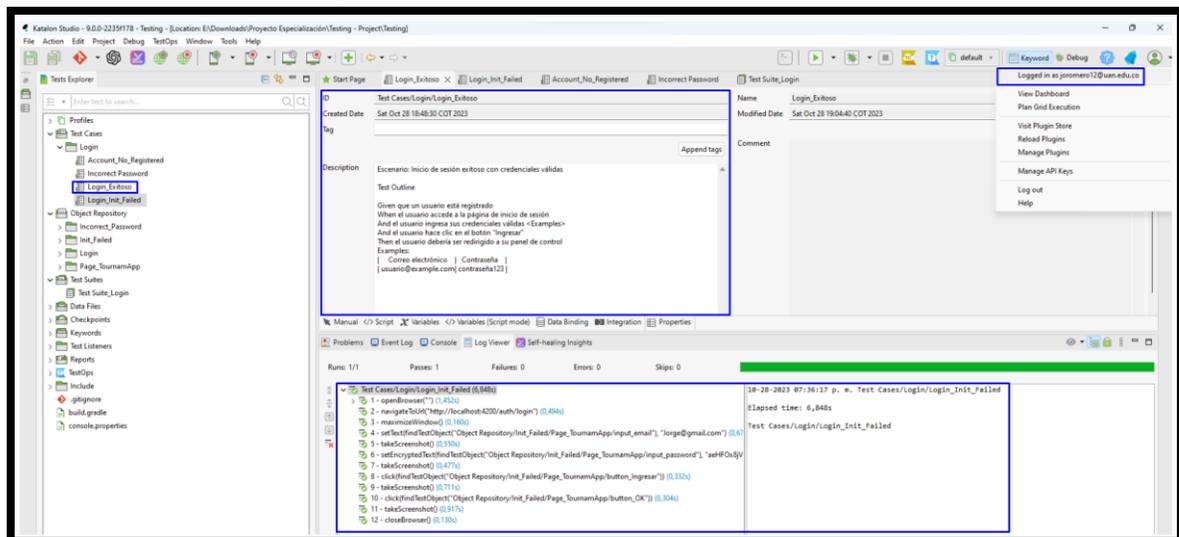
Nota: Elaboración propia (2023)

Ilustración 35 - Verificación de notificación al E-mail



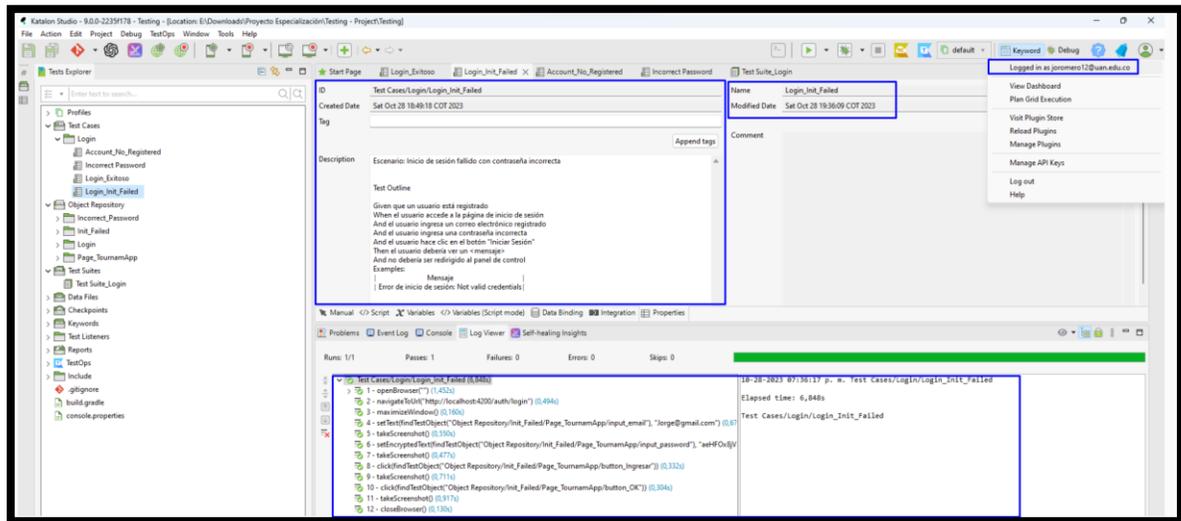
Nota: Elaboración propia (2023)

Ilustración 36 - Login Exitoso BDD



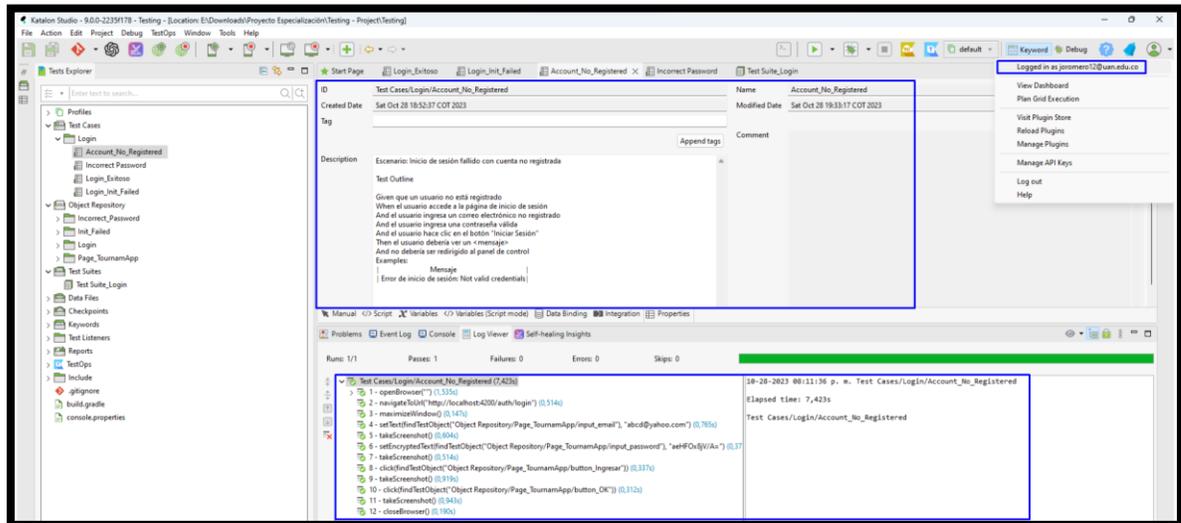
Nota: Elaboración propia (2023)

Ilustración 37 - Login Init Failed



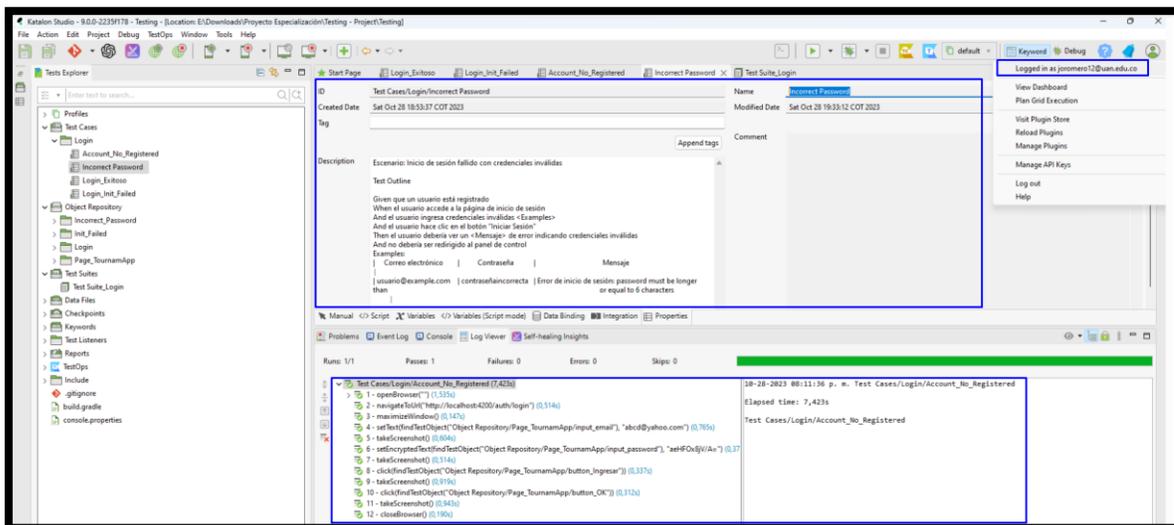
Nota: Elaboración propia (2023)

Ilustración 38 - Account No Registered



Nota: Elaboración propia (2023)

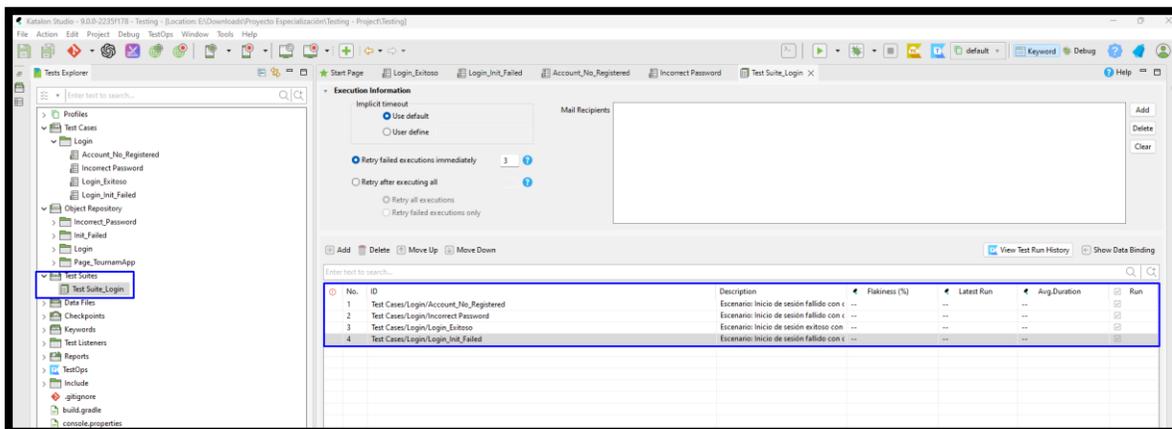
Ilustración 39 - Incorrect Password



Nota: Elaboración propia (2023)

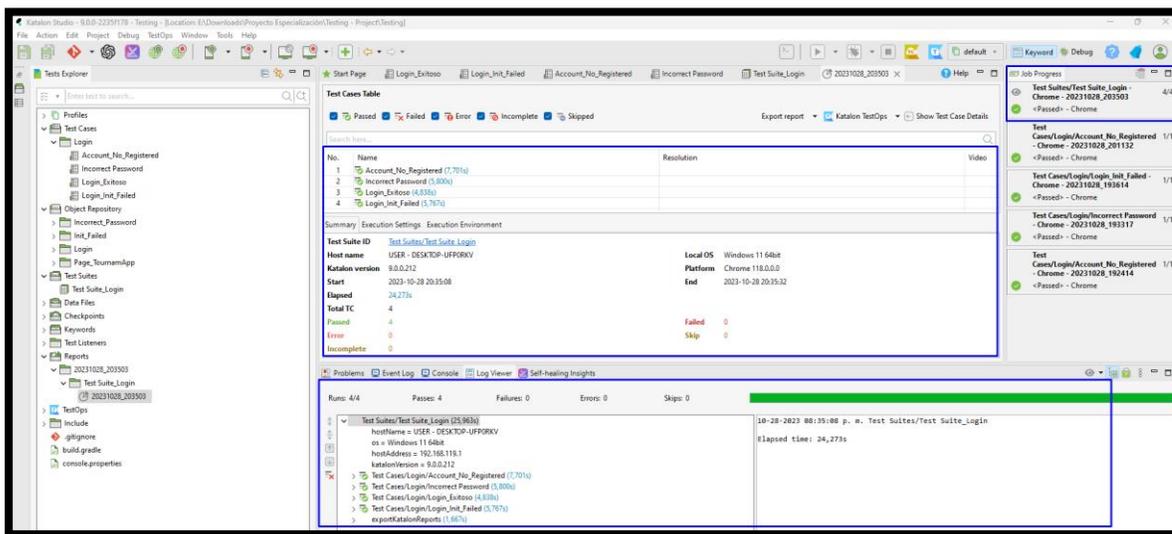
Luego simulamos una prueba de carga utilizando una de las funcionalidades que nos proporcionaba la herramienta y era armar una suite con estos 4 escenarios y se lanzó en una sola ejecución brindando un reporte en un formato HTML, mostrando cada uno de los elementos de tiempo de carga, cuánto se tardó y cuál fue el resultado final de la prueba, adicional en el reporte construimos el script de cada escenario y en cada paso tomara una captura de pantalla para así poder tener un mejor panorama de cada uno de los pasos que se evaluaron.

Ilustración 40 - Load Test Suite



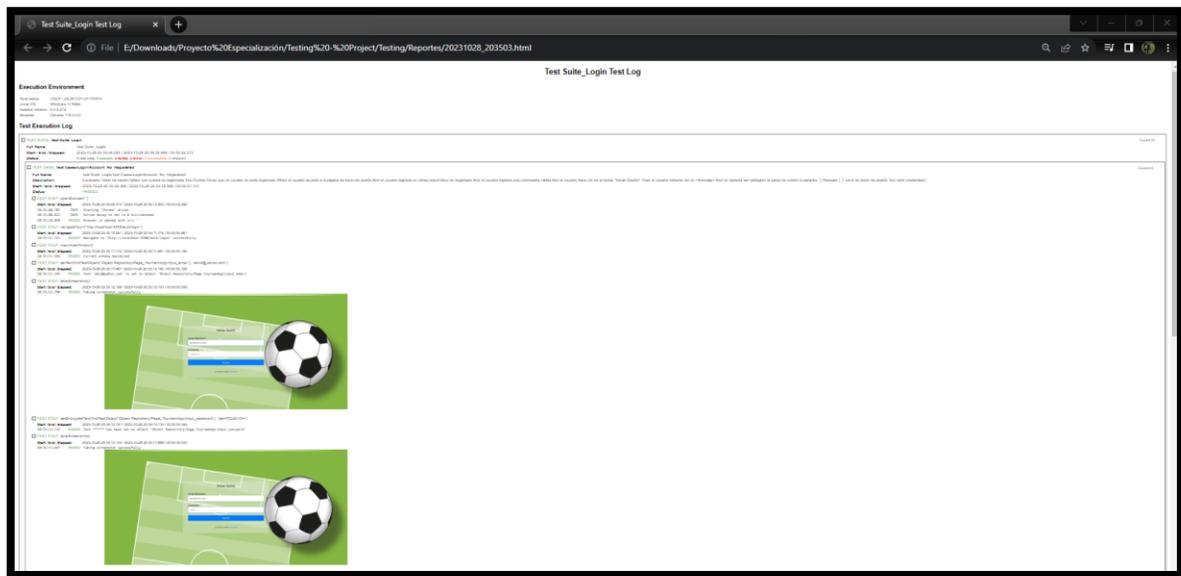
Nota: Elaboración propia (2023)

Ilustración 41 - Execution Load Test Suite



Nota: Elaboración propia (2023)

Ilustración 42 - Report HTML Load Test Suite



Nota: Elaboración propia (2023)

b. De Producto

Las pruebas de producto son un conjunto de acciones diseñadas para evaluar el producto final de un aplicativo web desde la perspectiva del usuario. Estas pruebas tienen como objetivo principal garantizar que la aplicación cumpla con los requisitos funcionales y no funcionales, funcione de manera consistente y brinde una experiencia de usuario satisfactoria. Basándose en una serie de requisitos para poder llevar a cabo las pruebas y determinar si están cumpliendo con el punto de vista de la experiencia de usuario. A continuación, explicaremos cómo se deben realizar o qué características deben tener las pruebas de producto.

Características Principales

Características	Descripción
Validación de Requisitos	<ul style="list-style-type: none"> Las pruebas de producto confirman que la aplicación cumple con todos los requisitos, funcionales y no funcionales, especificados.
Experiencia de Usuario (UX)	<ul style="list-style-type: none"> Para garantizar una experiencia de usuario positiva, se enfocan en evaluar la interfaz de usuario, la navegación y la usabilidad general.
Funcionalidad General	<ul style="list-style-type: none"> Verifican que todas las funciones y características de la aplicación funcionen correctamente y que no haya errores críticos.
Compatibilidad	<ul style="list-style-type: none"> Evalúan la compatibilidad de la aplicación para garantizar un rendimiento consistente en una variedad de navegadores web, dispositivos y sistemas operativos.
Rendimiento	<ul style="list-style-type: none"> Para maximizar el rendimiento, se enfocan en la velocidad de carga, la respuesta del sistema y la eficiencia general.
Seguridad	<ul style="list-style-type: none"> Verifican la seguridad de la aplicación y protegen los datos del usuario contra amenazas potenciales.
Pruebas de Estrés y Carga	<ul style="list-style-type: none"> Evalúan la capacidad de la aplicación para manejar cargas de trabajo elevadas y situaciones de estrés, identificando posibles problemas de rendimiento bajo presión.

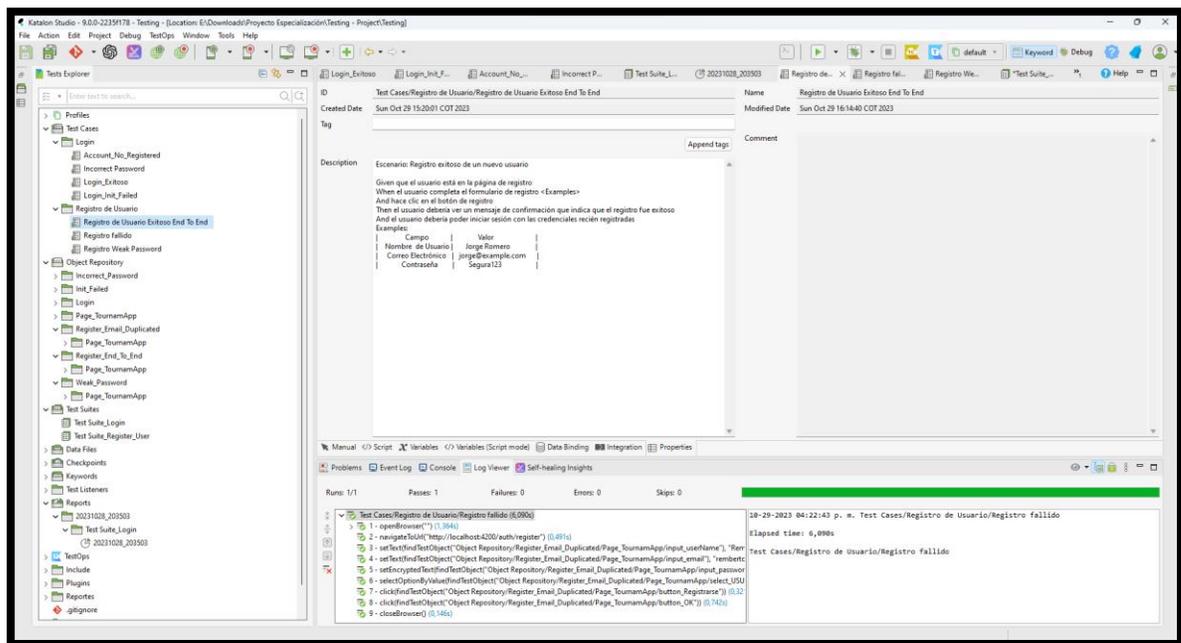
Tabla 10 - Tabla de Pruebas de Producto - Elaboración Propia

Los siguientes flujos se utilizaron para realizar pruebas de producto para el proyecto:

- Verificación de la funcionalidad de inicio de sesión y registro de usuarios.
- Evaluación de la usabilidad de la interfaz de usuario.
- Comprobación de la compatibilidad con varios navegadores y dispositivos.

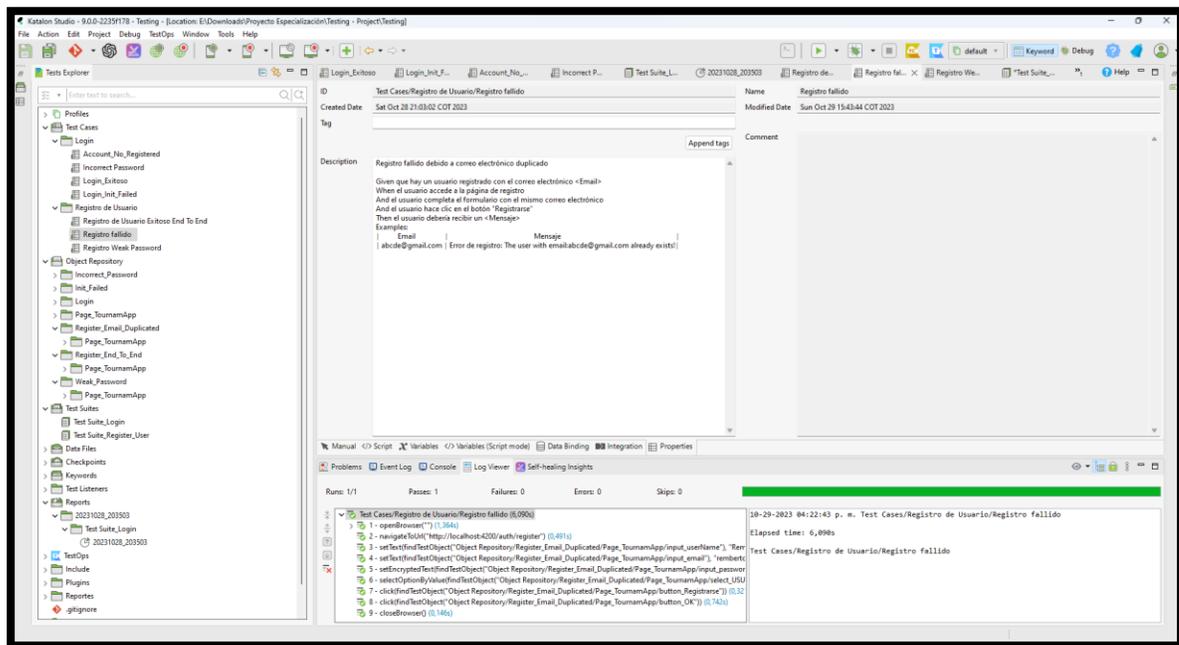
Para las pruebas de producto cumplieran con las anteriores condiciones se utilizaron los flujos funcionales y se dará un ejemplo de una prueba end to end dónde se verán las evidencias del aplicativo web que fueron probados en el marco del proyecto.

Ilustración 43 - Test End To End



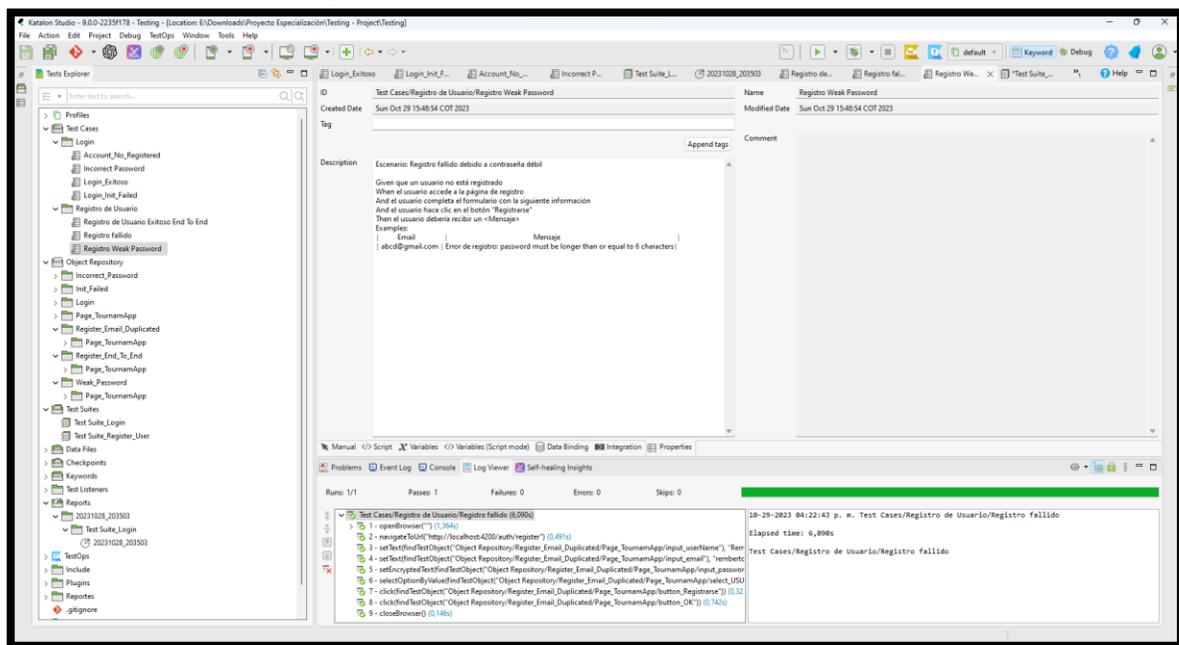
Nota: Elaboración propia (2023)

Ilustración 44 - Registered Failed



Nota: Elaboración propia (2023)

Ilustración 45 - Registered Weak Password



Nota: Elaboración propia (2023)

Utilizando una de las funcionalidades de la herramienta, se realizó una prueba de carga. Construimos una suite con estos tres escenarios y los lanzamos en una sola ejecución. En el reporte, mostramos el tiempo de carga, el tiempo de carga y el resultado final de la prueba. Además, construimos el script de cada escenario y tomamos una captura de pantalla en cada paso para así poder tener una idea de cómo funcionaba la herramienta.

Ilustración 46 - Test Load Suite Register User

No.	ID	Description	Flakiness (%)	Latest Run	Avg Duration	Run
1	Test Case:Registro de Usuario/Registro de Usuario Exitoso End To End	Escenario: Registro exitoso de un nuev	--	--	--	Run
2	Test Case:Registro de Usuario/Registro fallido	Registro fallido debido a correo electr	--	--	--	Run
3	Test Case:Registro de Usuario/Registro Weak Password	Escenario: Registro fallido debido a cor	--	--	--	Run

Run: 1/1 Pasos: 1 Faluros: 0 Erros: 0 Skips: 0

```

Test Cases/Registro de Usuario/Registro fallido (0.096)
  1 - openBrowser(1.164)
  2 - navigateToUrl("http://localhost:4200/auth/register") (0.011)
  3 - setTestFindTestObject("Object Repository/Registrar_Email_Duplicated/Page_TournamApp/input_userName"), "Test"
  4 - setTestFindTestObject("Object Repository/Registrar_Email_Duplicated/Page_TournamApp/input_email"), "rebert@t"
  5 - setEncryptedTestFindTestObject("Object Repository/Registrar_Email_Duplicated/Page_TournamApp/input_password")
  6 - selectOptionByValueFindTestObject("Object Repository/Registrar_Email_Duplicated/Page_TournamApp/select_USU")
  7 - clickFindTestObject("Object Repository/Registrar_Email_Duplicated/Page_TournamApp/button_Registrars") (0.12)
  8 - clickFindTestObject("Object Repository/Registrar_Email_Duplicated/Page_TournamApp/button_OK") (0.742)
  9 - closeBrowser(0.146)
  
```

10-20-2023 04:22:43 p. m. Test Cases/Registro de Usuario/Registro fallido
Elapsed time: 6,096s
Test Cases/Registro de Usuario/Registro fallido

Nota: Elaboración propia (2023)

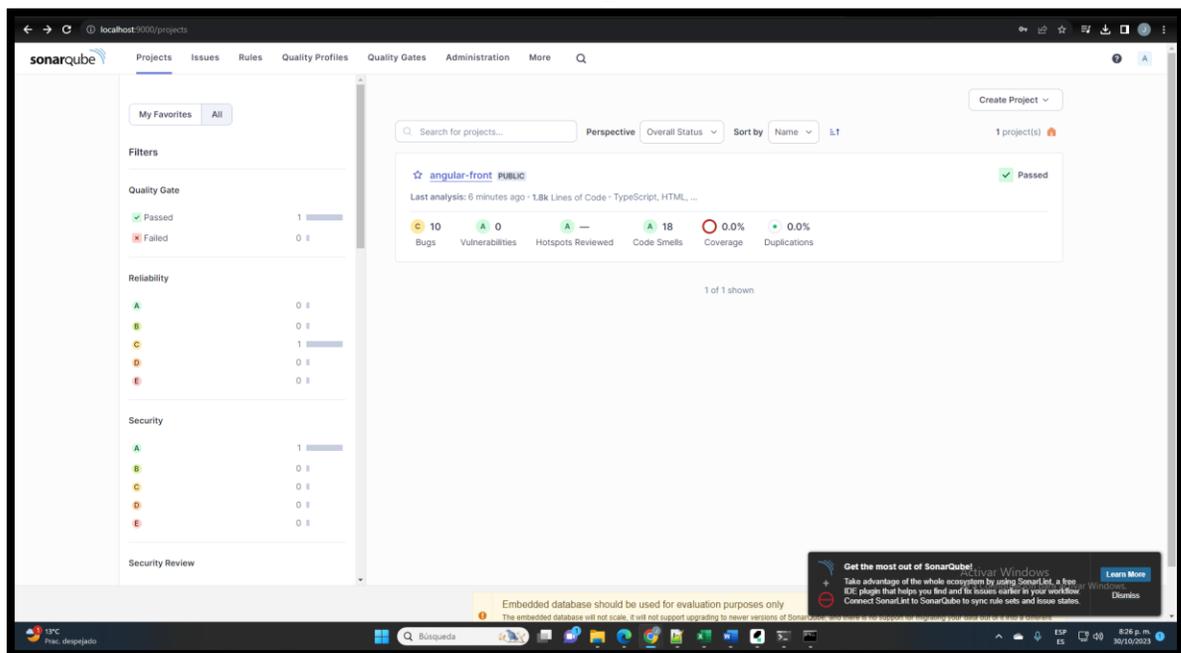
Objetivos de Pruebas de seguridad

- **Identificación de vulnerabilidades:** Identificar y corregir cualquier falla en el aplicativo web que pueda ser utilizada por personas malintencionadas.
- **Prevención de Ataques:** Asegurarse de que las medidas de seguridad sean efectivas para prevenir y mitigar posibles ataques.
- **Protección de Datos:** Garantizar que la información que almacena y procesa la aplicación sea segura y privada.
- **Cumplimiento de Normas:** Garantizar que el aplicativo cumpla con las normas y estándares de seguridad establecidos.

Escaneo de Vulnerabilidades:

- Utilización de herramientas automatizadas para buscar posibles vulnerabilidades en el código y la configuración.

Ilustración 48 - Pruebas estáticas Sonarqube

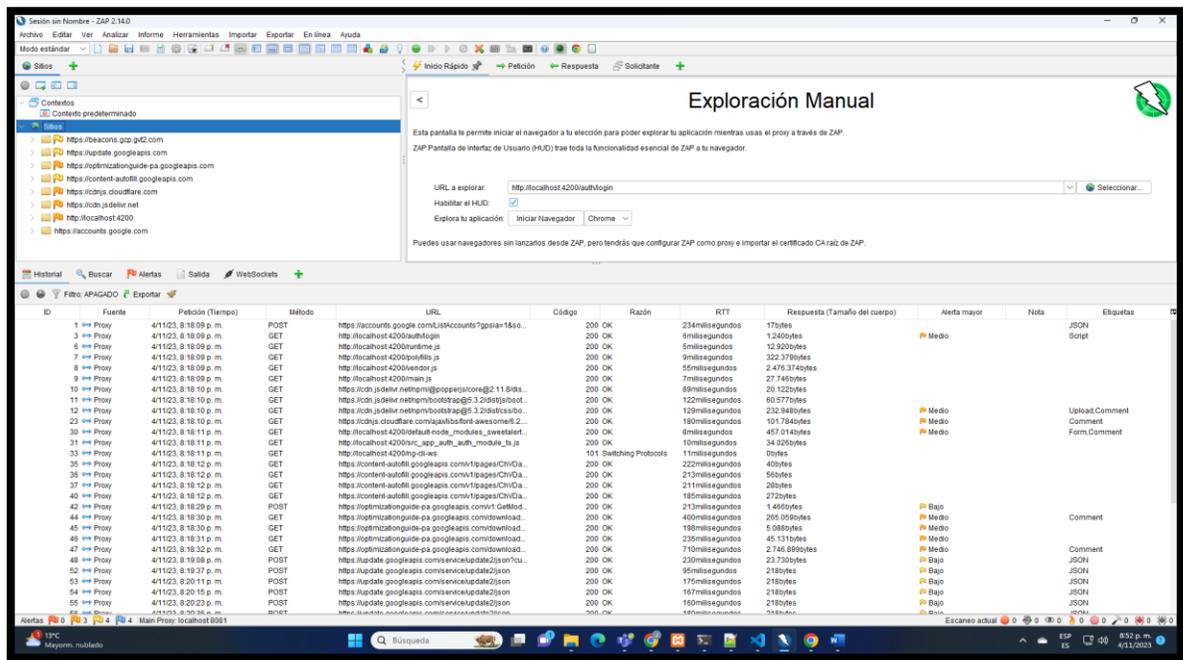


Nota: Elaboración propia (2023)

A continuación, se anexan las evidencias de las pruebas realizadas con la herramienta de OWASP ZAP que se divide en 10 subcategorías las cuáles se mencionan a continuación:

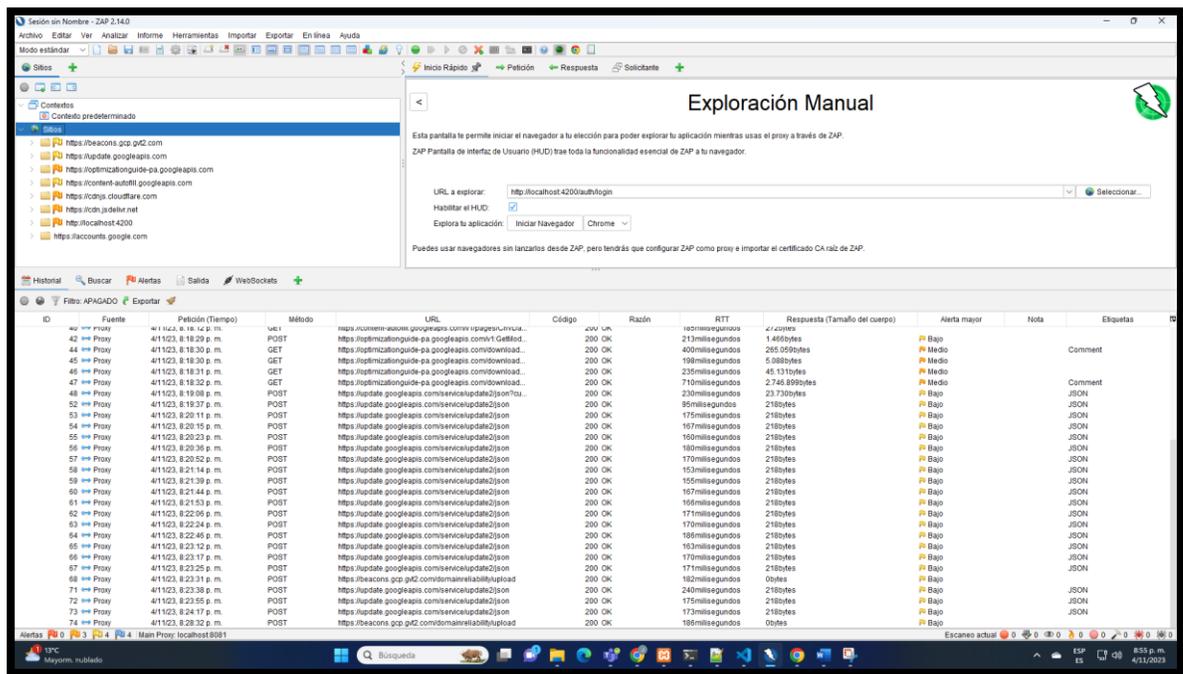
- Recopilación de Información.
- Pruebas de gestión de la configuración.
- Pruebas de lógica de negocio.
- Pruebas de Autenticación.
- Pruebas de autorización.
- Pruebas de gestión de sesiones.
- Pruebas de validación de datos.
- Pruebas de denegación de Servicio.
- Pruebas de Servicio Web.
- Pruebas de AJAX.

Ilustración 49 - Prueba OWAS ZAP



Nota: Elaboración propia (2023)

Ilustración 50 - Prueba OWAS ZAP II



Nota: Elaboración propia (2023)

Ilustración 51 - Prueba OWAS ZAP III

Alerta (11)

Alerta: Cabeecera Content Security Policy (CSP) no configurada (5)

Parámetro: Alerta

Evidencia: CVE ID: 693

WASC ID: 15

Origen: Pasivo (10038 - Cabeecera Content Security Policy (CSP) no configurada)

Referencia de Alerta: 10038-1

Vector de Entrada:

Descripción: La Política de seguridad de contenido (CSP) es una capa adicional de seguridad que ayuda a detectar y mitigar ciertos tipos de ataques, incluidos Cross Site Scripting (XSS) y ataques de inyección de datos. Estos ataques se utilizan para todo, desde el robo de datos hasta la desfiguración del sitio o la distribución de malware. CSP proporciona un conjunto de encabezados HTTP estándar que permiten a los propietarios de sitios web declarar fuentes de contenido aprobadas que los navegadores deberían poder cargar en esa página. Los tipos cubiertos son JavaScript, CSS, marcos HTML, fuentes, imágenes y objetos incrustados como applets de Java, ActiveX, archivos de audio y video.

Otra información:

Solución: Asegúrese de que su servidor web, servidor de aplicaciones, balanceador de carga, etc. esté configurado para establecer la cabecera Content-Security-Policy.

Referencias: https://developer.mozilla.org/en-US/docs/Web/Security/CSP/Introducing_Content_Security_Policy
https://owasp.org/whats-new/owasp.org/whats-new/Content_Security_Policy_Cheat_Sheet.html
<http://www.w3.org/TR/CSP/>

Etiquetas de Alerta:	Clave	Valor
OWASP_2021_A05		https://owasp.org/Top10A05_2021-Security_Misconfiguration/
OWASP_2017_A06		https://owasp.org/www-project-top-ten2017A6_2017-Security_Misconfiguration.html

Nota: Elaboración propia (2023)

Ilustración 52 - Pruebas OAS ZAP IV

Alerta (11)

Alerta: Cabeecera X-Frame-Options (5)

Parámetro: Alerta

Evidencia: CVE ID: 200

WASC ID: 13

Origen: Pasivo (10037 - El servidor divulga información mediante un campo(s) de encabezado de respuesta HTTP "X-Frame-Options")

Vector de Entrada:

Descripción: El servidor de la webaplicación está divulgando información mediante uno o más encabezados de respuesta HTTP "X-Frame-Options". El acceso a tal información podría facilitar a los atacantes la identificación de otros marcos/componentes de los que su aplicación web depende y las vulnerabilidades a las que pueden estar sujetos tales componentes.

Otra información:

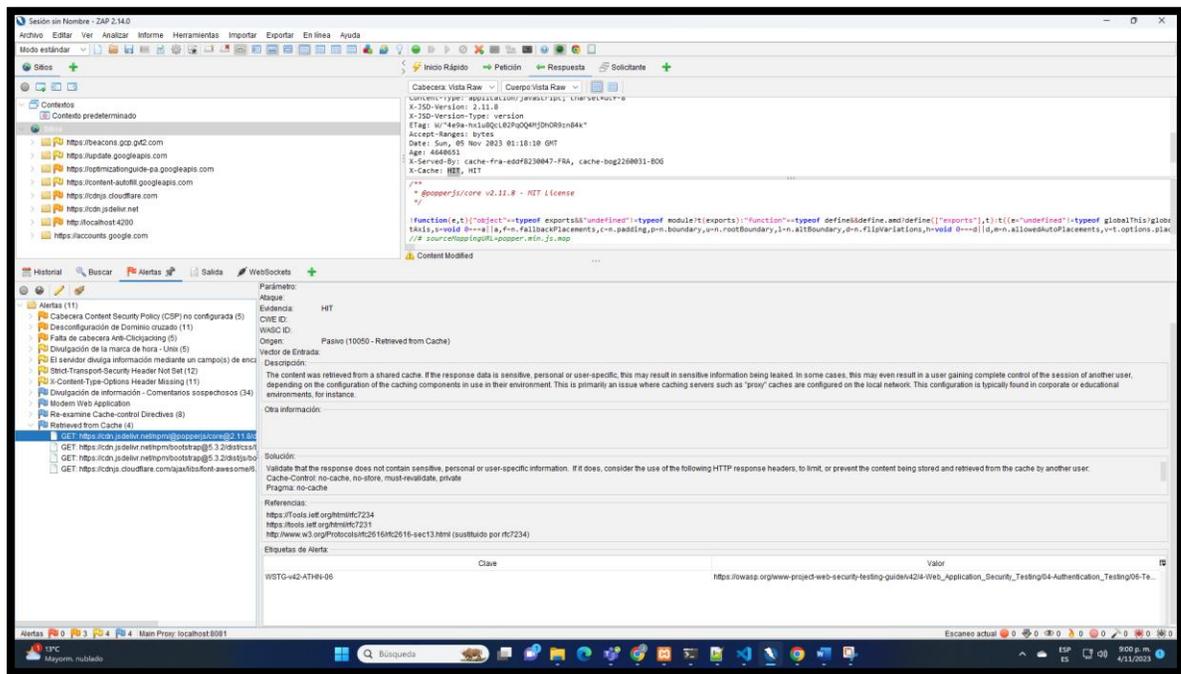
Solución: Asegúrese de que su servidor web, servidor de aplicación, equilibrador de carga, etc. está configurado para suprimir encabezados "X-Frame-Options".

Referencias: <http://blogs.msdn.com/b/ranum/Archive/2013/04/23/Remove-Unwanted-HTTP-Response-Headers.aspx>
<http://www.troyhunt.com/2012/02/shhh-don-t-deje-la-respuesta-headers.html>

Etiquetas de Alerta:	Clave	Valor
OWASP_2021_A01		https://owasp.org/Top10A01_2021-Broken_Access_Control/
WSTG-42-INFD-08		https://owasp.org/www-project-web-security-testing-guide/v424-Web_Application_Security_Testing/01-Information_Gathering/08-Fin...
OWASP_2017_A03		https://owasp.org/www-project-top-ten2017A3_2017-Sensitive_Data_Exposure.html

Nota: Elaboración propia (2023)

Ilustración 53 - Prueba OWAS ZAP V



Nota: Elaboración propia (2023)

Recopilación de la Información:

La primera fase de la evaluación de seguridad se centra en recopilar la mayor cantidad de información posible sobre la aplicación deseada. Un paso necesario es la recopilación de información una prueba de la infiltración. Esta tarea se puede realizar de muchas formas diferentes.

- Spiders, Robots y Crawlers
- Reconocimiento mediante motores de búsqueda.
- Identificación de puntos de entrada de la aplicación.
- Pruebas para encontrar firmas de aplicación web.
- Descubrimiento de aplicaciones.
- Análisis de códigos de error.

Pruebas de gestión de la configuración:

Los análisis de infraestructura o topología de arquitectura con frecuencia pueden revelar información importante sobre una aplicación web. Se pueden obtener datos como por ejemplo el código fuente, los métodos HTTP permitidos, funcionalidades administrativas, métodos de autenticación y configuraciones de la infraestructura.

- Pruebas de SSL/TLS
- Pruebas del receptor de escucha de la BD
- Pruebas de la gestión de la configuración de la Infraestructura.
- Pruebas de la gestión de la configuración de la aplicación.
- Gestión de extensiones de archivo.
- Archivos antiguos, copias de seguridad y sin referencias.
- Interfaces de administración de la Infraestructura y de la aplicación.
- Métodos HTTP y XST.

Comprobación del Sistema de Autenticación:

Autenticar a una persona implica con frecuencia verificar su identidad, mientras que autenticar un objeto puede significar confirmar su procedencia. La autenticación depende de al menos uno de los elementos de autenticación. La autenticación es el proceso de intentar verificar la identidad digital del remitente de una comunicación en seguridad informática.

- Transmisión de credenciales a través de un canal cifrado.
- Enumeración de usuarios.
- Pruebas de diccionario sobre cuentas de usuario o cuentas predeterminadas.
- Pruebas de fuerza bruta.
- Saltarse el sistema de autenticación.
- Comprobar sistemas de recordatorio/restauración de contraseñas vulnerables.
- Pruebas de gestión del caché de navegación y de salida de sesión.

- Pruebas de CAPTCHA.
- Múltiples factores de autenticación.
- Probar por situaciones adversas.

Gestión de sesiones:

La gestión de sesiones abarca todos los controles de usuario, desde la autenticación hasta la salida de la aplicación. Como protocolo sin estados, HTTP responde a las peticiones de los clientes sin enlazarlos entre sí. Es fundamental considerar la seguridad de la aplicación en función de las necesidades y expectativas del proveedor.

Pruebas de Autorización:

El concepto de autorización es permitir el acceso a recursos solo aquellos que tienen permiso para hacerlo. Las pruebas de autorización implican comprender cómo funciona el proceso de autorización y utilizar esa información para evitar el proceso.

Pruebas de lógica de Negocio:

Es necesario considerar métodos no convencionales para detectar fallas en la lógica comercial de una aplicación web multifuncional.

Pruebas de validación de Datos:

La falta de validación adecuada de las entradas del cliente o del entorno de la aplicación es la falla más común en la seguridad de las aplicaciones web. Casi todas las principales vulnerabilidades en aplicaciones son causadas por esta falla, como inyecciones en el

intérprete, ataques locales/Unicode, sobre el sistema de archivos y desbordamientos de búfer.

Pruebas de Denegación de Servicio:

El tipo más común de ataque de denegación de servicio (Dos) se utiliza en una red para hacer que otros usuarios válidos no puedan comunicarse con un servidor. Un ataque DoS de red se basa en un usuario malicioso inundando una máquina objetivo con suficiente tráfico para que no pueda manejar la cantidad de peticiones que recibe. Los ataques de denegación de servicio distribuidos (DDoS) se producen cuando un usuario malicioso utiliza una gran cantidad de máquinas para inundar de tráfico una sola máquina objetivo.

Pruebas de servicio WEB:

Las aplicaciones emergentes como los servicios web y la arquitectura orientada a servicios (SOA) están permitiendo que las empresas interoperen y crezcan a un ritmo sin precedentes. Los clientes de servicios web generalmente no son frontales web, sino otros servidores. Los servicios web están expuestos a la red como cualquier otro servicio; sin embargo, pueden usarse con HTTP, FTP, SMTP o con cualquier otro protocolo de transporte. Las vulnerabilidades en servicios web son similares a otras vulnerabilidades como la inyección SQL, revelación de información, etc, pero también tienen vulnerabilidades de XML.

Pruebas de AJAX:

Las técnicas AJAX pueden mejorar significativamente la experiencia de uso de los usuarios de las aplicaciones web. Sin embargo, desde el punto de vista de la seguridad, las aplicaciones AJAX son más vulnerables a los ataques que las aplicaciones web convencionales, ya que a veces se desarrollan con más enfoque en lo que se puede hacer que en lo que debería hacer. Además, las aplicaciones AJAX son más complicadas porque se procesan tanto en el lado del cliente como en el lado del servidor.

5. Instalación y Configuración

El proceso de despliegue se llevó a cabo de manera metódica y eficiente. Después de completar el desarrollo y las pruebas pertinentes, la nueva funcionalidad fue implementada en el entorno de producción de manera escalonada, minimizando cualquier impacto potencial en la disponibilidad del sistema. La coordinación estrecha entre el equipo de desarrollo y el equipo de operaciones garantizó una transición suave, y las métricas de rendimiento y monitoreo se establecieron para asegurar el correcto funcionamiento de la nueva funcionalidad de manera local. Este enfoque cuidadoso de despliegue aseguró una experiencia de usuario continua y sin contratiempos al introducir mejoras específicas en la aplicación.

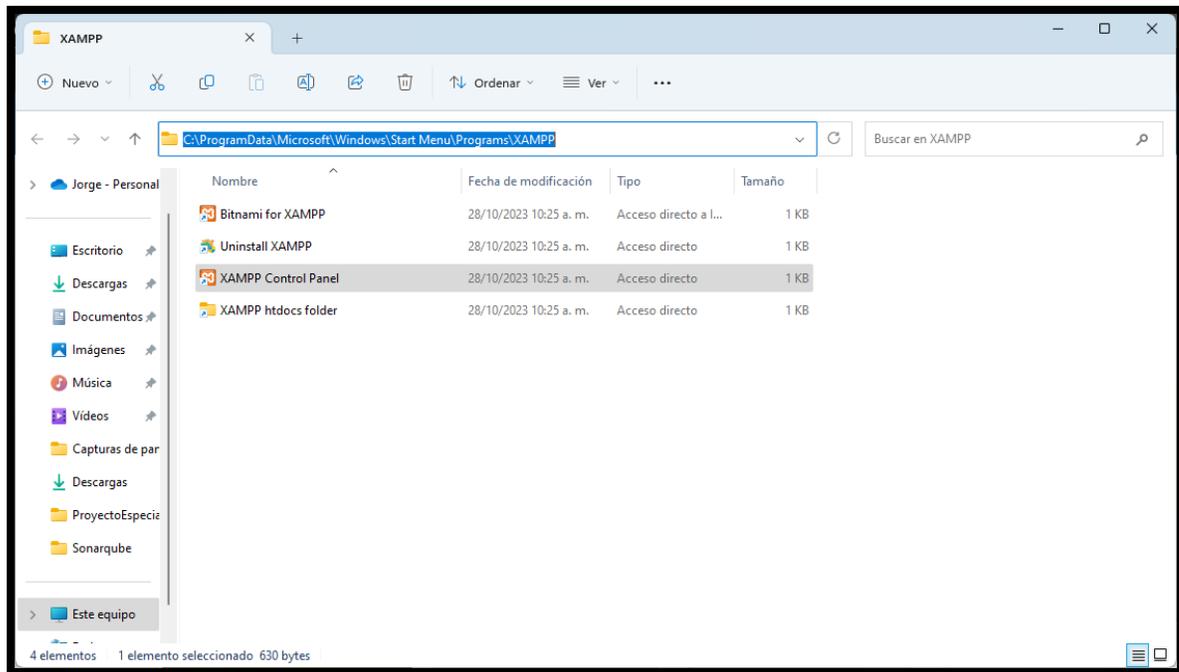
La fase de despliegue local utilizando XAMPP implica configurar y ejecutar el entorno de desarrollo en una máquina local antes de implementar la aplicación en un servidor en vivo.

Aquí hay un resumen de los pasos básicos para la fase de despliegue local con XAMPP:

1. Instalación de XAMPP:

- Descarga e instalación de la herramienta XAMPP desde el sitio oficial.

Ilustración 54 - Instalación XAMPP



Nota: Elaboración propia (2023)

2. Configuración del Proyecto:

- Se realiza el alojamiento de la aplicación en el directorio "htdocs" de la instalación de XAMPP. Esto será la raíz del servidor local.

3. Configuración de la Base de Datos:

- Utiliza phpMyAdmin, incluido en XAMPP, para crear la base de datos necesaria para tu aplicación.

- Importa la estructura de la base de datos desde tus archivos SQL.

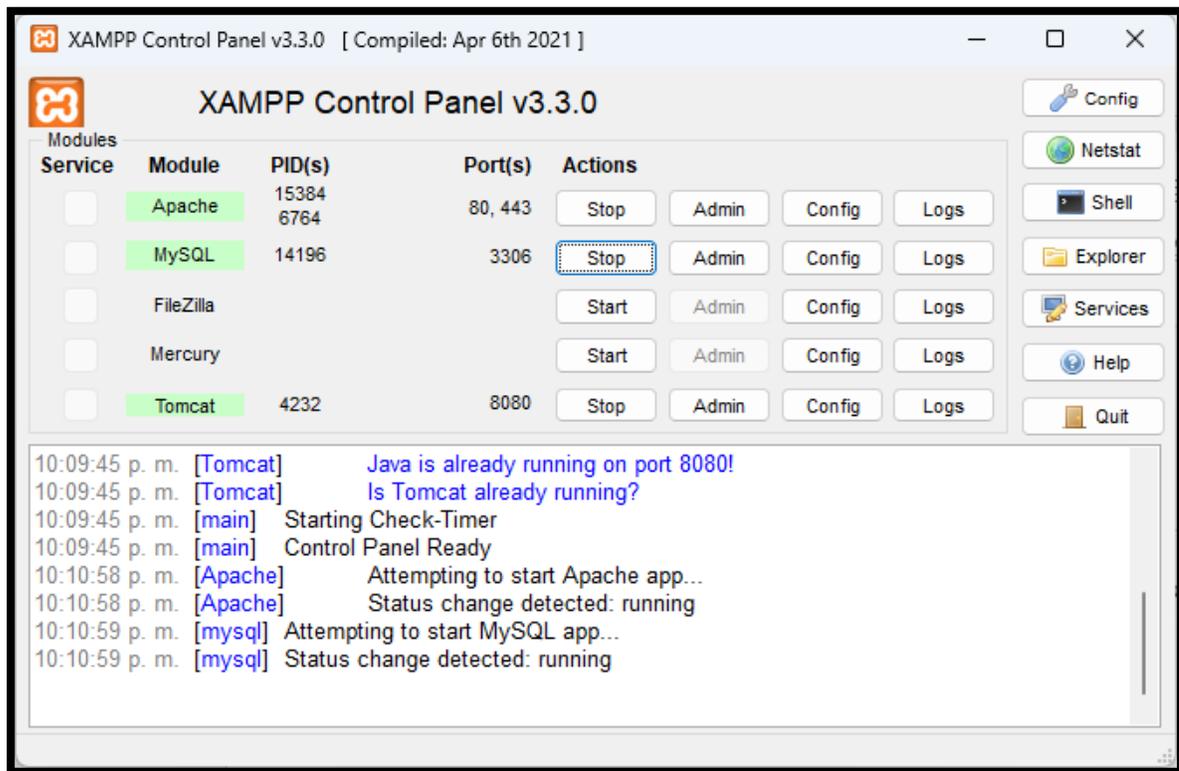
4. Configuración del Archivo de Configuración:

- Asegúrate de que el archivo de configuración de tu aplicación (por ejemplo, `config.php`) refleje la configuración de la base de datos local, con el nombre de usuario y la contraseña correctos.

5. Inicio de Servidores:

- Inicia los servidores Apache y MySQL desde el panel de control de XAMPP.

Ilustración 55 - Inicio de XAMPP

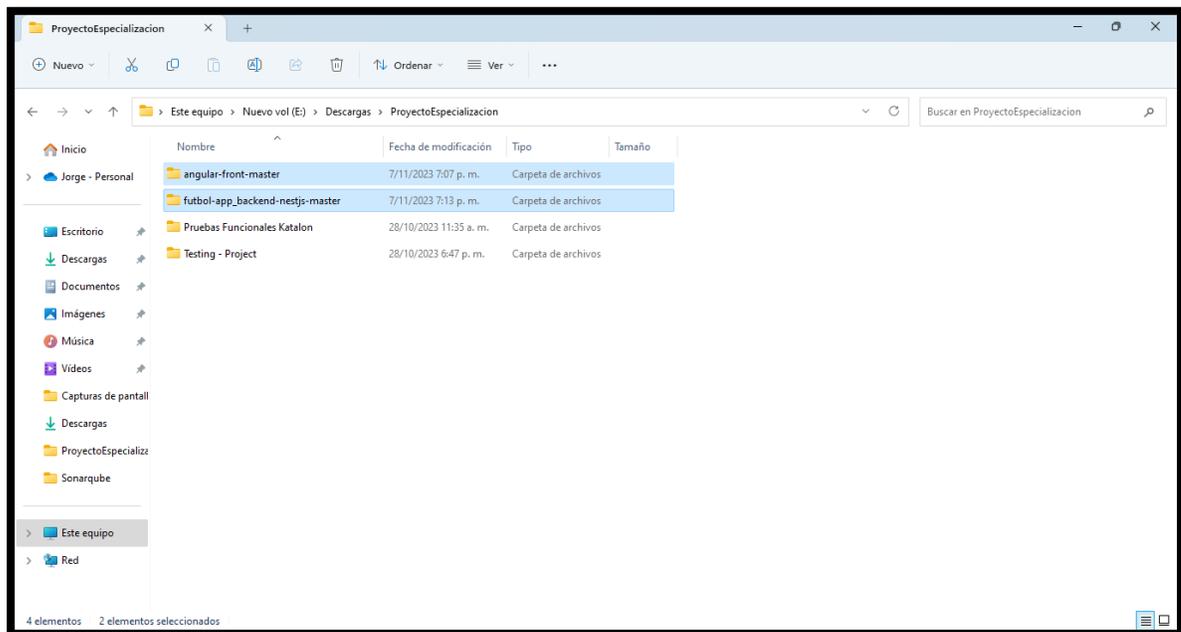


Nota: Elaboración propia (2023)

6. Inicializar o levantamiento de los ambientes Front y Back

- Se va a la raíz dónde se encuentran alojados los dos frentes.

Ilustración 56 - Ubicación Proyecto

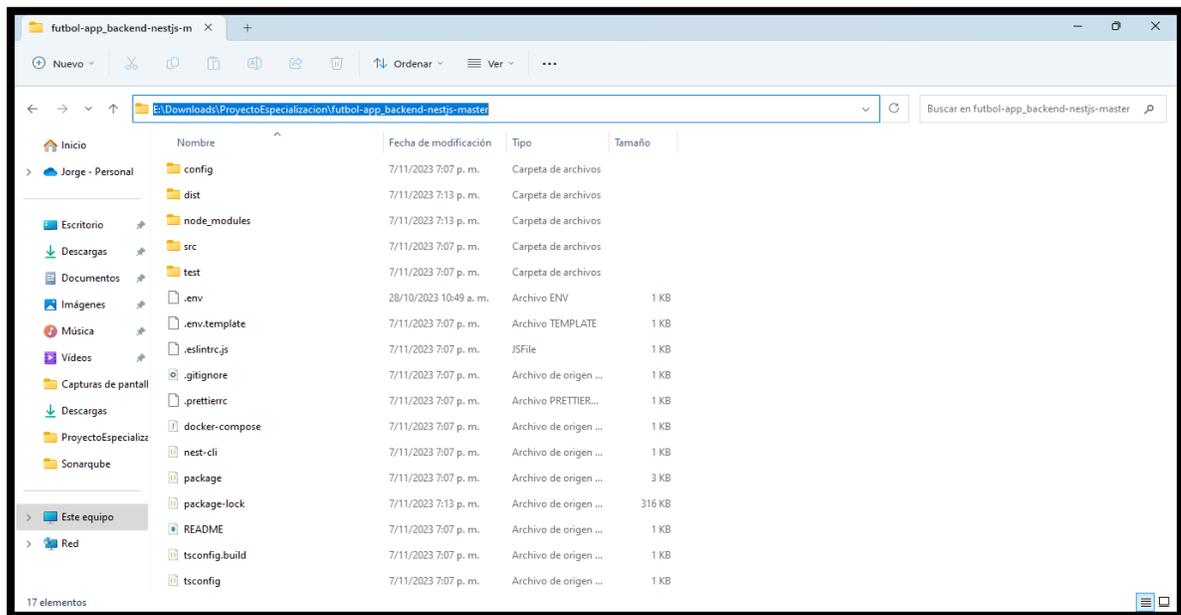


Nota: Elaboración propia (2023)

7. Levantamiento Back:

- Abre la carpeta dónde se encuentra localizado el proyecto y se ubica en el directorio raíz y abre la consola (cmd).

Ilustración 57 - Proceso Levantamiento de Ambiente Backend



Nota: Elaboración propia (2023)

- Abre el DOS desde la carpeta raíz y ejecutamos el siguiente comando “Npm run start:dev” para levantar automáticamente el ambiente back.

Ilustración 58 - Comando para levantar Ambiente Back con NPM

```

C:\WINDOWS\system32\cmd. X + v
Microsoft Windows [Versión 10.0.22621.2428]
(c) Microsoft Corporation. Todos los derechos reservados.

E:\Downloads\ProyectoEspecializacion\futbol-app_backend-nestjs-master>Npm run start:dev
> nest-backend@0.0.1 start:dev
> nest start --watch

```

Nota: Elaboración propia (2023)

Ilustración 59 - Confirmación Ambiente Backend Arriba

```

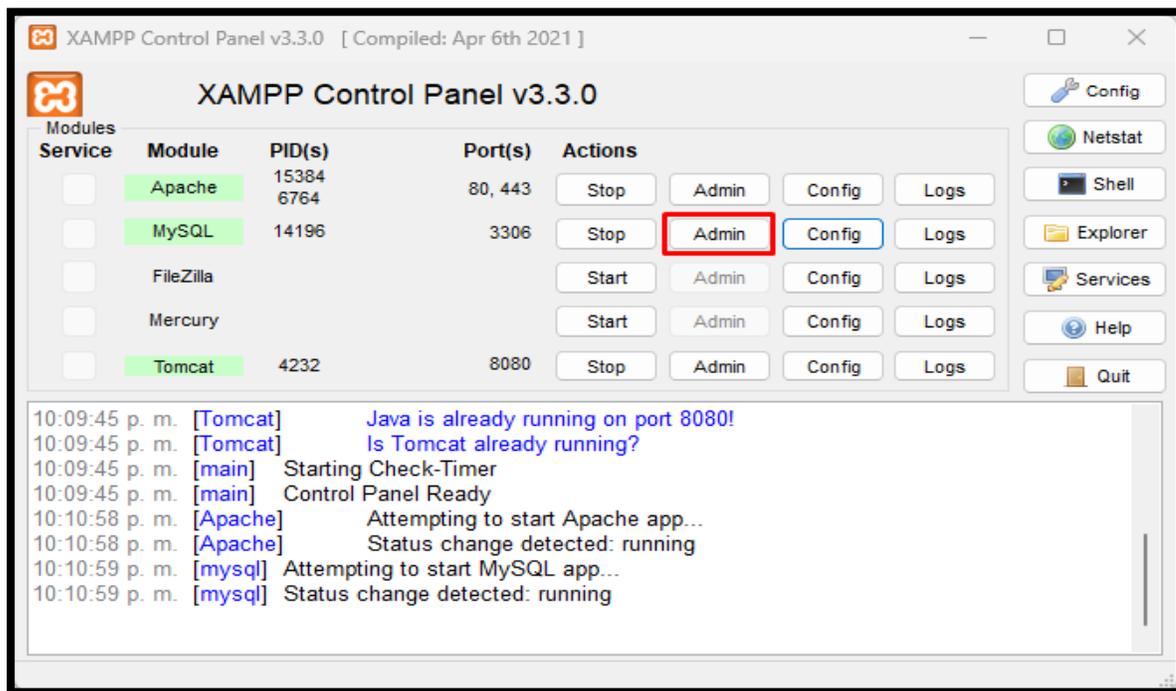
C:\WINDOWS\system32\cmd. X + v
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/auth/roles, GET} route +2ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/auth/register, POST} route +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/auth/login, POST} route +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/auth, GET} route +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/auth, GET} route +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/auth, PUT} route +1ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/auth/:id, DELETE} route +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/auth/restore/:id, PATCH} route +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RoutesResolver] TeamController {/team}: +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/team, POST} route +1ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/team, GET} route +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/team/:id, GET} route +1ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/team/:id, PUT} route +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/team/:id, DELETE} route +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/team/restore/:id, PATCH} route +1ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RoutesResolver] TournamentController {/tournament}: +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/tournament, POST} route +1ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/tournament/leagues, GET} route +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/tournament/locations, GET} route +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/tournament, GET} route +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/tournament/:id, GET} route +0ms
[nest] 4772 - 09/11/2023, 10:30:40 p. m. LOG [RouterExplorer] Mapped {/tournament/:id, PUT} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/tournament/:id, DELETE} route +1ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/tournament/restore/:id, PATCH} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RoutesResolver] PlayerController {/player}: +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/player/positions, GET} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/player, POST} route +1ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/player, GET} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/player/:id, PUT} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/player/:id, DELETE} route +1ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/player/restore/:id, PATCH} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RoutesResolver] ParticipationController {/participation}: +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/participation, POST} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/participation, GET} route +1ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/participation/:id, GET} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/participation/:id, PUT} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/participation/:id, DELETE} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/participation/restore/:id, PATCH} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RoutesResolver] MatchController {/match}: +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/match, POST} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/match, GET} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/match/:id, GET} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/match/:id, PUT} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/match/:id, DELETE} route +1ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/match/restore/:id, PATCH} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RoutesResolver] PlayerInMatchController {/player_in_match}: +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/player_in_match/randomId, POST} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/player_in_match, POST} route +0ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/player_in_match, GET} route +1ms
[nest] 4772 - 09/11/2023, 10:30:41 p. m. LOG [RouterExplorer] Mapped {/player_in_match/:id, GET} route +0ms

```

Nota: Elaboración propia (2023)

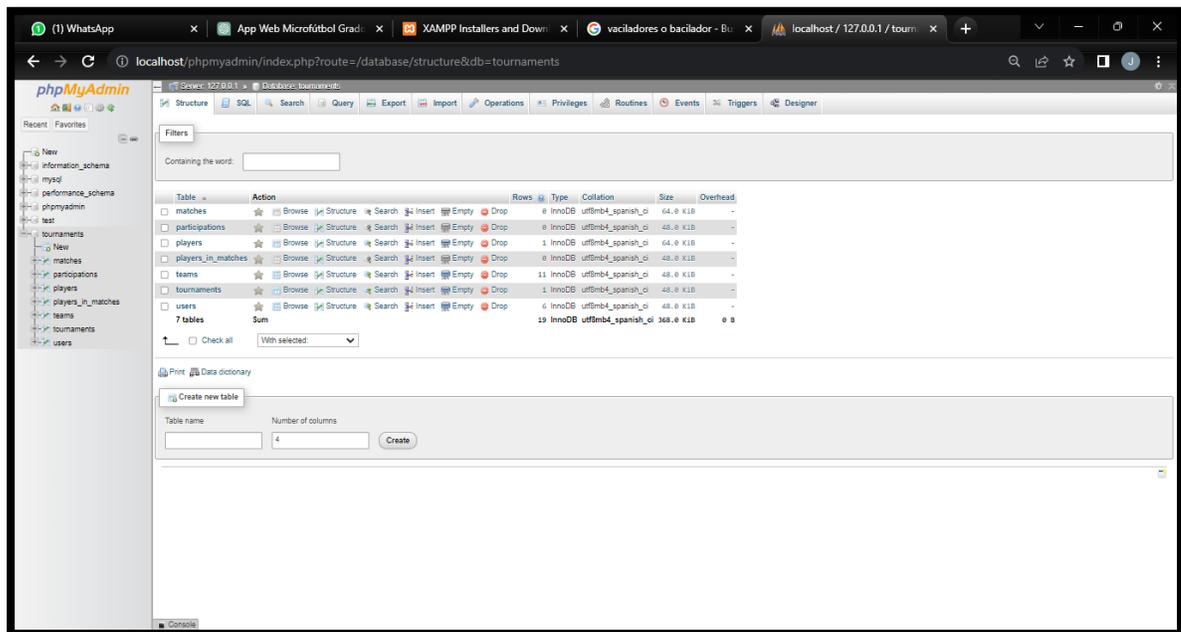
- Luego se da clic en el XAMPP en la opción de MySQL en la opción de **Admin** y se confirma el levantamiento del back y el navegador se abre automáticamente la BD.

Ilustración 60 - Validación de la BD Arriba



Nota: Elaboración propia (2023)

Ilustración 61 - Confirmación en el Navegador

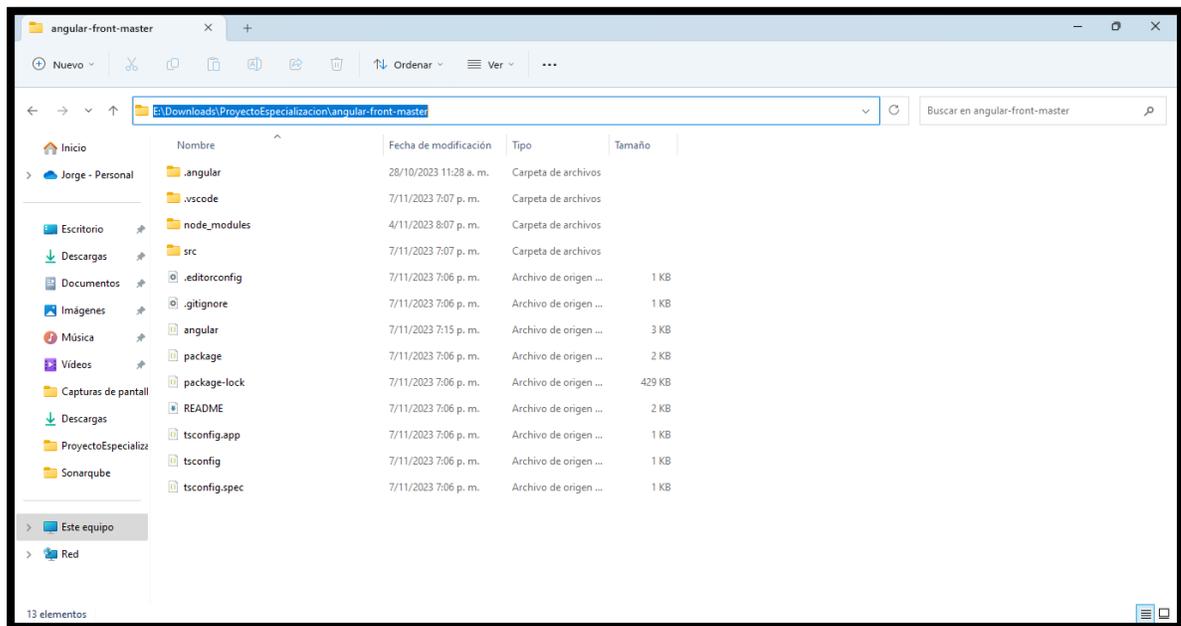


Nota: Elaboración propia (2023)

8. Levantamiento Front:

- Abre la carpeta dónde se encuentra localizado el proyecto y se ubica en el directorio raíz y abre la consola (cmd).

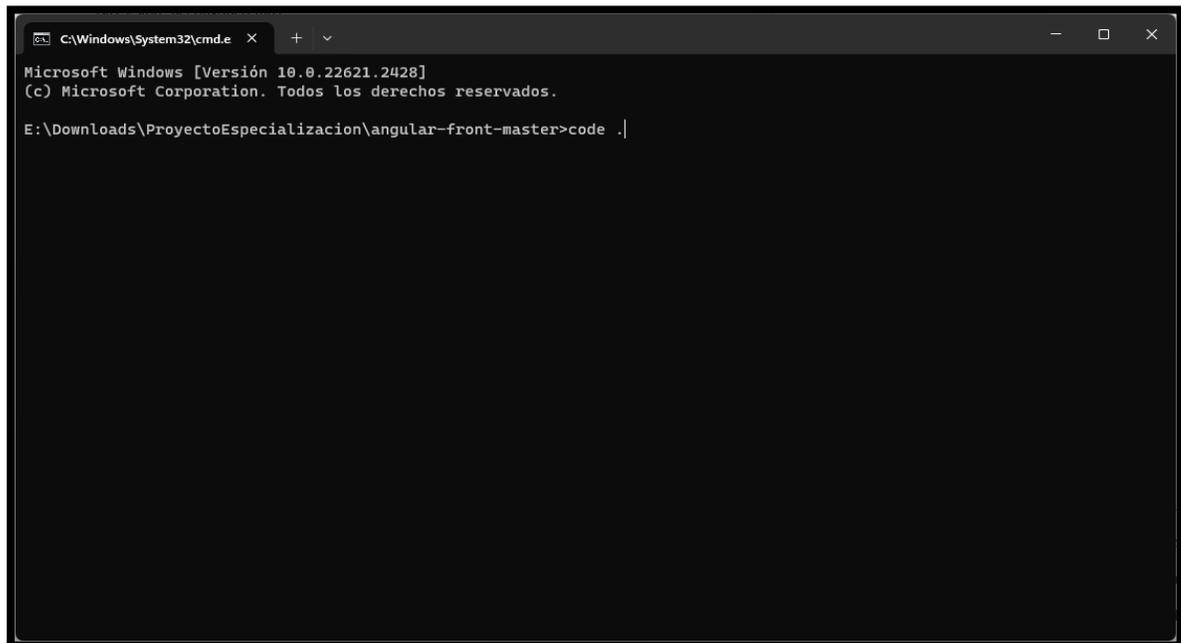
Ilustración 62 - Levantamiento de Ambiente Frontend



Nota: Elaboración propia (2023)

- Se escribe el comando “code .”

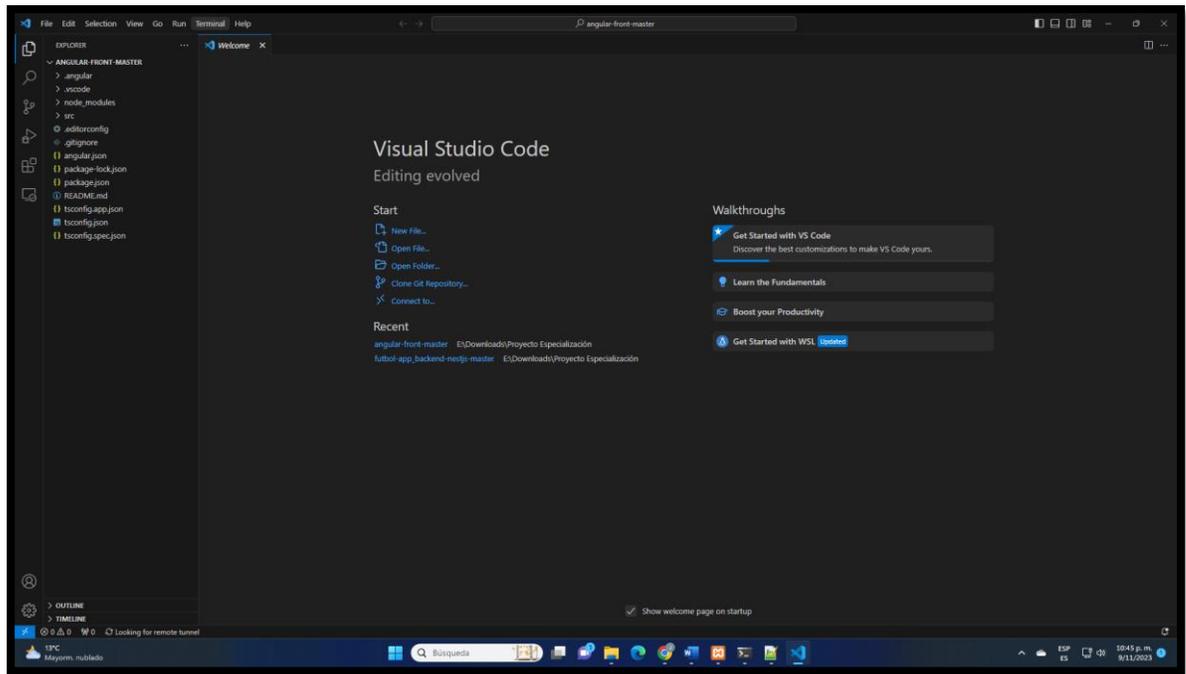
Ilustración 63 - Apertura IDE VSC



Nota: Elaboración propia (2023)

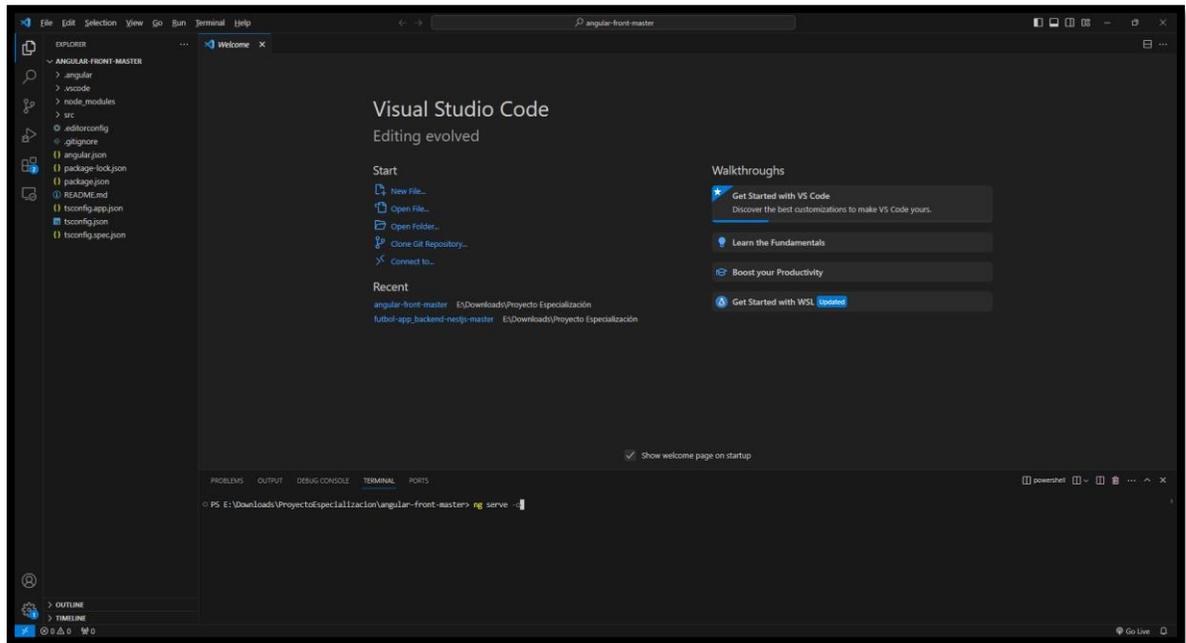
- Se abre la herramienta de Visual studio code y abrimos una terminal para ejecutar el comando “ng serve -o”

Ilustración 64 - Inicio Terminal IDE



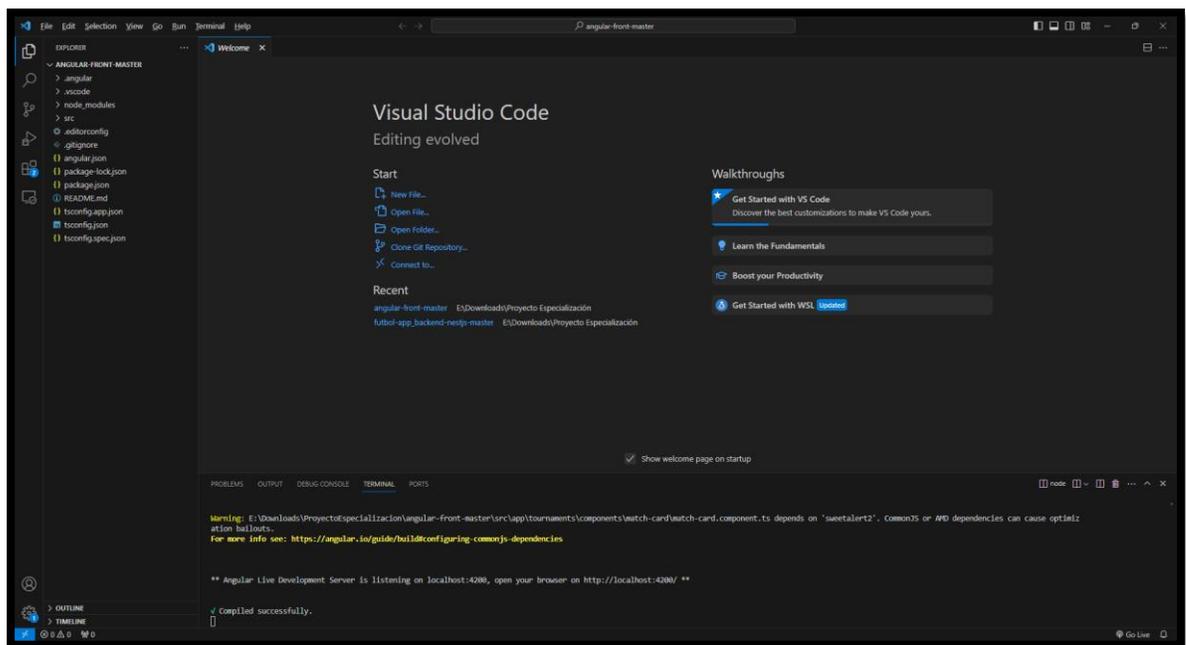
Nota: Elaboración propia (2023)

Ilustración 65 - Ejecución Comando Levantamiento Frontend desde el IDE



Nota: Elaboración propia (2023)

Ilustración 66 - Verificación Levantamiento Ambiente Frontend

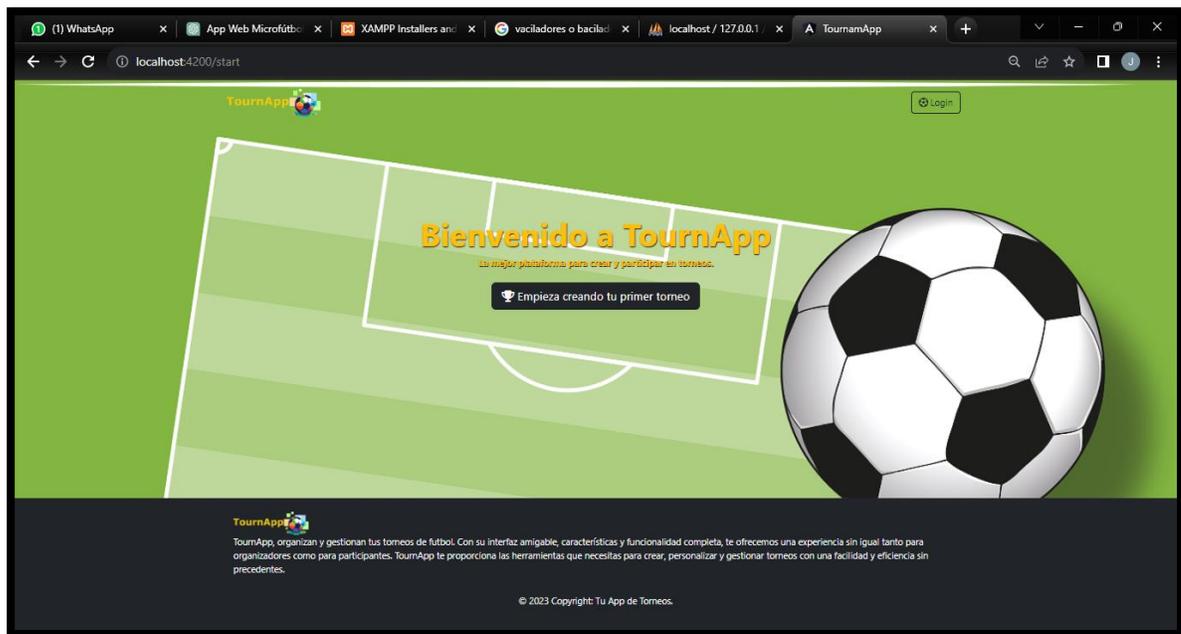


Nota: Elaboración propia (2023)

9. Acceso a la Aplicación:

- Abre un navegador web y accede a tu aplicación utilizando la URL local, por ejemplo, <http://localhost:4200/start>.

Ilustración 67 - Acceso a Ambiente



Nota: Elaboración propia (2023)

10. Pruebas Locales:

- Ejecución de pruebas exhaustivas para asegurar de que la aplicación funciona correctamente en el entorno local.

11. Conclusiones

Este proyecto de desarrollo de una aplicación web responsiva para la gestión de torneos de microfútbol en el municipio de Soacha Cundinamarca, representa un avance significativo hacia la modernización y eficiencia en la administración de eventos deportivos locales. La aplicación, diseñada bajo la metodología Kanban y respaldada por una arquitectura MVC, busca superar las limitaciones actuales en la gestión de torneos al ofrecer una plataforma integral, desde la inscripción de equipos hasta la programación de partidos y el seguimiento de resultados. La adopción de tecnologías web modernas y un enfoque inclusivo en el desarrollo subrayan nuestro compromiso con la innovación y la experiencia del usuario. Este proyecto no solo aborda una necesidad específica en el contexto del municipio, sino que también sienta las bases para futuras mejoras y adaptaciones en la gestión de eventos deportivos a nivel local. La combinación de tecnología, metodología ágil y un enfoque centrado en el usuario resalta la relevancia y el impacto positivo que esta aplicación puede tener en la comunidad deportiva del departamento.

12. Anexos

13. Referencias Bibliográficas

<https://docs.sonarsource.com/sonarqube/latest/>

<https://www.zaproxy.org/>

<https://owasp.org/>

<https://katalon.com/web-testing>

<https://trello.com/teams/remote-team-management>

<https://www.atlassian.com/es/agile/kanban/boards>

<https://www.365scores.com/es-mx>

<https://ligafansport.com/>

<https://www.flashscore.co/>

<https://code.visualstudio.com/docs>

<https://angular.io/docs>

<https://www.typescriptlang.org/docs/>

<https://structurizr.com/dsl>

https://www.mintic.gov.co/portal/715/articles-6398_Ley_1341_2009.pdf

<https://dev.mysql.com/doc/>

<https://git-scm.com/>

<https://docs.nestjs.com/>

<https://kanban.university/wp-content/uploads/2021/11/The-Official-Kanban->

[Guide_Spanish_A4.pdf](#)

<https://academy.seguridadcero.com.pe/blog/escaneo-vulnerabilidades->

[autom%C3%A1tico-OWASP-ZAP](#)

<https://www.federico-toledo.com/introduccion-al-testing-de-seguridad/>

https://owasp.org/www-pdf-archive/Gu%C3%ADa_de_pruebas_de_OWASP_ver_3.0.pdf

<https://trello.com/b/EMzbuzfJ/app-microf%C3%BAtbol>

<https://www.apachefriends.org/index.html>