

SISTEMA DE INFORMACION ORIENTADO A LA WRB VU PARA TORNEOS

FABIAN ANDRES BENAVIDES VALENCIA
MIGUEL ANGEL CAMARGO CAMPOS

CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS
FACULTAD DE INGENIERIA
DEPARTAMENTO DE INFORMÁTICA Y ELECTRÓNICA
PROGRAMA DE TECNOLOGÍA EN INFORMÁTICA
BOGOTA D. C.
2013 – II

SISTEMA DE INFORMACION ORIENTADO A LA WRB VU PARA TORNEOS

FABIAN ANDRES BENAVIDES VALENCIA
MIGUEL ANGEL CAMARGO CAMPOS

Trabajo de grado
para optar al título de
Tecnólogo en informática

Asesor:
Ing.

CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS
FACULTAD DE INGENIERIA
DEPARTAMENTO DE INFORMÁTICA Y ELECTRÓNICA
PROGRAMA DE TECNOLOGÍA EN INFORMÁTICA
BOGOTA D. C.
2013 - II

Página de aceptación

Bogotá D.C Enero 2014

Nota de aceptación

Firma jurado

Firma jurado

Firma jurado

Página de agradecimientos

- Agradezco principalmente a mi mamá por apoyarme en este proceso, a mis amigos por confiar en mí y siempre estar a mi lado en los momentos en que los necesitaba, a mis profesores que no se dieron por vencidos aun cuando no era el mejor estudiante, gracias a todos ellos en estos momentos soy la personas que soy, una persona capaz de salir adelante y no detenerse ante los obstáculos que se presenten

Fabián Andrés Benavides valencia

- Primero que todo quiero agradecer a Dios que me dio salud y mucha fortaleza para afrontar este reto, quiero agradecer a mi familia que nunca me abandono y siempre me apoyo en cada paso que di durante este largo camino, quiero agradecer a todos los profesores que nos compartieron tantos conocimientos y tantas experiencias profesionales para aprender de ellas, quiero agradecer a mis compañeros que aportaron de alguna u otra forma para que yo no me rindiera a pesar de los problemas

Miguel Ángel Camargo Campos

Tabla de Contenido

Pag

1 INTRODUCCIÓN.....	4
1.1 Definición del Problema:	5
1.2 Alcance:.....	5
1.3 Justificación:	6
1.4 Objetivos:.....	6
1.4.1 Objetivo general:	6
1.4.2 Objetivos específicos:.....	6
2. Modelo De Desarrollo	8
3 Análisis Y Diseño.....	14
3.1 Definición de requerimientos	14
3.1.1 Requerimientos funcionales:	14
3.1.2 Requerimientos no funcionales.....	15
3.2 Descripción del Sistema propuesto	15
3.3 Diseño del sistema propuesto	17
3.3.1 Diagramas estáticos	17
Diagrama de Clases:	17
Diagramas MER:	23
Diagrama MER Torneo	23
3.3.2 Diagramas Dinámicos	25
Diagrama Casos de uso:.....	25
Diagramas de secuencia.....	28
4 Glosario	30
5. Conclusiones	32
BIBLIOGRAFIA.....	33

Tabla de tablas.

Pag

Tabla N° 1 Sprint Módulo Igu. 11
Tabla N° 2: Sprint Módulo Núcleo. 11
Tabla N° 3: Sprint Módulo Base De Datos..... 12
Tabla N° 4: Sprint Módulo Seguridad. 12
Tabla N° 5: Módulo Torneo. 13

Tabla de Diagramas.

Pag

Diagrama 1: Módulo núcleo.....	18
Diagrama 2: Módulo baseDatos	19
Diagrama 3: Módulo seguridad.....	20
Diagrama 4: Módulo igu	21
Diagrama 5: Modulo torneos	22
Diagrama 6: Modulo de Inscripción.....	23
Diagrama 7 Modelo entidad relacion	24
Diagrama 8 Caso de uso Crear Torneo.....	25
Diagrama 9 caso de uso actualizar torneo.....	25
Diagrama 10 caso de uso configurar torneo.....	26
Diagrama 11 Caso de uso Inscribir Torneo	26
Diagrama 12 caso de uso configurar noticias.	27
Diagrama 13 caso de uso logueo.	27
Diagrama 14 caso de uso consultar torneo.	28
Diagrama 15 Secuencia crear Torneo.....	29
Diagrama 16 Secuencia Modificar Torneo.....	29
Diagrama 17 Secuencia Crear Equipo	29

1 INTRODUCCIÓN

En este tiempo de grandes avances tecnológicos, en los que el acceso a internet cada día es más fácil y hay muchos más lugares desde donde se puede acceder a toda la información que se encuentra en línea. Internet ha abierto la puerta a muchas facilidades y nuevas formas para manejar la información, para esto hay muchas de que las personas puedan encuentre la mejor manera de interactuar con este mundo tan inmenso y saber utilizarlo de la mejor forma, es por esta razón que se ha planteado la necesidad que la corporación universitaria Minuto de Dios (Uniminuto) de crear un sistema web donde se administren algunas actividades de los estudiantes.

Así, tomando el departamento de vida universitaria de Uniminuto, se planeó crear un sistema de información orientado a la web que permita crear y administrar los torneos internos de la misma.

El proyecto está compuesto por varios módulos que se complementan entre sí, con esto los usuarios; que se dividen entre administradores los cuales llevan el control de los torneos como los marcadores y las fechas y los usuarios que usan el sistema para acceder a la información que se les pueda brindar y/o para registrarse en algún torneo, podrá acceder de forma fácil y rápida a esta parte que compone las opciones que Uniminuto ofrece a sus estudiantes y trabajadores.

Para el diseño de estos módulos se han utilizado los diagramas estándar de UML2 (Unified Modeling Language) con los cuales se analiza el funcionamiento e interacción entre el usuario/aplicación, y la estructura del mismo.

También se desarrollaron los módulos a partir del lenguaje de programación PHP acompañado de JavaScript aprovechando algunas librerías de jQuery para mejorar la parte visual del sistema.

Ya que uno de los factores más importantes de un sistema es la información se creó una base de datos en el motor PostgreSQL, con este se garantiza la seguridad de los datos y también que el sistema funcione de forma correcta y eficaz como se espera.

Como modelos de desarrollo se usará una metodología llamada scrum, ya que al ser una metodología rápida y ágil permite a los desarrolladores tener un mejor control sobre cada módulo y saber cuándo se encuentra listo para proseguir con el siguiente y de esta misma forma permite al cliente final ver el proceso o los resultados que se van obteniendo del desarrollo del sistema.

1.1 Definición del Problema:

La Universidad Minuto de Dios cuenta con el Departamento de Vida Universitaria, en este se manejan asuntos que no tienen que ver con la carga académica de los estudiantes tales como, actividades de esparcimiento, torneos interuniversitarios, clases transversales, pin pon voleibol baloncesto futbol dibujo guitarra batería.

Este departamento cuenta con un mal desarrollo tecnológico, con respecto a otras universidades, como por ejemplo el sistema de inscripciones a los torneos internos de Universidad de los Andes¹. Actualmente la manera en la cual se realizan los procesos de consulta, ver un campeonato de algún deporte que maneje la universidad, se tiene que hacer un trámite muy complicado, primero se debe dirigir hasta la sede de bienestar universitario hablar con la persona que este encargada del trámite que vamos a hacer, esperando que este esa persona después si es el caso de inscribir en algún torneo pedir la planilla llenarla entregarla esperar que no se pierda, eso es algo que se puede evitar.

Todos estos trámites son muy tediosos para la mayoría de los estudiantes que al no encontrar una mejor manera de poder acceder a estos recursos de la universidad optan por ignorarlos y así se pierden muchas oportunidades que brinda la universidad. La desorganización, y la falta de información es un serio problema al querer realizar un torneo o un evento deportivo, ya que sin comunicación suficiente, las personas probablemente no le den importancia a esta clase de actividades.

1.2 Alcance:

Este sistema de información se desarrolló para ser utilizado por los estudiantes, administrativos y docentes de la sede principal de la Corporación Universidad Minuto de Dios, para la inscripción de los torneos internos de futbol, fúsala, ping-pong, baloncesto, vóleibol, que se realizan de la misma.

Se mostraran las fechas, los resultados y las estadísticas, tales como goleadores y tabla de posiciones, de cada uno de los torneos anteriormente mencionados.

En este sistema de información se puede consultar las fechas y los horarios de los encuentros de cada torneo, estas fechas son configuradas por el administrador de

¹ <http://decanaturadeestudiantes.uniandes.edu.co/index.php/es/torneos>

la página y del mismo modo el administrador puede organizar los enfrentamientos de los equipos en cada torneo

1.3 Justificación:

Este proyecto se enfoca en facilitar la forma cómo los estudiantes acceden a la información de vida universitaria y de esta manera aumentar la participación en los torneos, que muchas veces los estudiantes pasan por alto.

Las ventajas que se tiene al realizar este proyecto, es un acceso más directo, dinámico y llamativo para los miembros de la comunidad universitaria, además de que es un proceso más ordenado y seguro que el usado actualmente.

Por otro lado se busca un avance tecnológico ya que la manera en la que los estudiantes se tienen que inscribir a los torneos es inadecuada y en muchos casos molesta, sin contar que la información no es guardada de forma segura y es muy fácil su pérdida.

También cabe resaltar que en estos tiempos donde se vive un avance tecnológico muy notable y apresurado, ya no hay espacio para que la inscripción de torneos y de cursos de una universidad como la minuto de Dios, se haga de forma manual sin un sistema organizado y adaptado a las necesidades de los estudiantes.

1.4 Objetivos:

1.4.1 Objetivo general:

Diseñar y desarrollar un sistema de información orientado a la web que permita mejorar los procesos relacionados a los torneos internos que realiza bienestar universitario en la sede principal de Uniminuto, donde se requiera el uso de información de estudiantes, administrativos, profesores, etc.

1.4.2 Objetivos específicos:

- 1).Diseñar un modelo relacional de la Base de Datos que permita almacenar toda la información que se necesita para realizar los torneos.
- 2). Diseñar a partir del Modelo Vista Controlador una aplicación web que permita la interacción entre el Departamento y los interesados en los servicios ofrecidos por este.

3).Desarrollar un prototipo que permita hacer pruebas de implementación

2. Modelo De Desarrollo

La metodología de desarrollo aplicada es Scrum, la cual pertenece al grupo de metodologías ágiles y enfoca su prioridad en la obtención de resultados y satisfacción del cliente, reduciendo muchos procesos que utilizan las metodologías pesadas.

Control de evolución del proyecto

Scrum controla de forma práctica y adaptable el avance del proyecto, y para esto utiliza las siguientes prácticas de la gestión ágil.

- **Revisión de las Iteraciones:**

Al finalizar cada iteración se lleva a cabo una revisión con todas las personas implicadas en el proyecto. Este es el periodo máximo que se tarda en reconducir una desviación en el proyecto o en las circunstancias del producto.

Según el modelo Scrum se ha decidido para este proyecto hacer 5 iteraciones y de esta manera desarrollar los 5 módulos que se necesitan para que éste pueda tener una entrega funcional y lista para hacer las pruebas necesarias.

- **Desarrollo incremental:**

Durante el proyecto, las personas implicadas no trabajan con diseños o abstracciones. El desarrollo incremental implica que al final de cada iteración se dispone de una parte del producto operativa que se puede inspeccionar y evaluar.

Al finalizar cada iteración o ciclo que se propuso para este desarrollo se evaluará los incrementos en la funcionalidad final del sistema y si el módulo en verdad está listo y con esto pasar al siguiente sprint y seguir el desarrollo

- **Desarrollo evolutivo:**

Los modelos de gestión ágil se emplean para trabajar en entornos de incertidumbre e inestabilidad de requisitos. Intentar predecir en las fases iniciales cómo será el producto final, y sobre dicha predicción desarrollar el diseño y la arquitectura del producto no es realista, porque las circunstancias obligarán a remodelarlo muchas veces.

Para qué predecir los estados finales de la arquitectura o del diseño si van a estar cambiando. En Scrum se toma a la inestabilidad como una premisa, y se adoptan técnicas de trabajo para permitir esa evolución sin degradar la calidad de la arquitectura que se irá generando durante el desarrollo.

El desarrollo Scrum va generando el diseño y la arquitectura final de forma evolutiva durante todo el proyecto. No los considera como productos que deban realizarse en la primera “fase” del proyecto.
(El desarrollo ágil no es un desarrollo en fases)

- **Visión general del proceso:**

Scrum denomina “sprint” a cada iteración de desarrollo y recomienda realizarlas con duraciones de 30 días. El sprint es por tanto el núcleo central que proporciona la base de desarrollo iterativo e incremental.

Los elementos que conforman el desarrollo Scrum son

Las reuniones:

- **Planificación de sprint:** Jornada de trabajo previa al inicio de cada sprint en la que se determina cuál va a ser el trabajo y los objetivos que se deben cumplir en esa iteración.
- **Reunión diaria:** Breve revisión del equipo del trabajo realizado hasta la fecha y la previsión para el día siguiente.
- **Revisión de sprint:** Análisis y revisión del incremento generado.

Los elementos

- **Pila del producto:** lista de requisitos de usuario que se origina con la visión inicial del producto y va creciendo y evolucionando durante el desarrollo.
- **Pila del sprint:** Lista de los trabajos que debe realizar el equipo durante el sprint para generar el incremento previsto.
- **Incremento:** Resultado de cada sprint.

Los roles:

Scrum clasifica a todas las personas que intervienen o tienen interés en el desarrollo del proyecto en: propietario del producto, equipo, gestor de Scrum (también Scrum Manager o Scrum Master) y “otros interesados”.

Los tres primeros grupos (propietario, equipo y gestor) son los responsables del proyecto, los que según la comparación siguiente (y sin connotaciones peyorativas) serían los “cerdos”; mientras que el resto de interesados serían las gallinas.

Cerdos y gallinas.

Esta metáfora ilustra de forma muy gráfica la diferencia de implicación en el proyecto entre ambos grupos:

Una gallina y un cerdo paseaban por la carretera.

La gallina dijo al cerdo: “Quieres abrir un restaurante conmigo”.

El cerdo consideró la propuesta y respondió: “Sí, me gustaría. ¿Y cómo lo llamaríamos?”.

La gallina respondió: “Huevos con beicon”.

El cerdo se detuvo, hizo una pausa y contestó:

“Pensándolo mejor, creo que no voy a abrir un restaurante contigo. Yo estaría realmente comprometido, mientras que tu estarías sólo implicada”.

En este proyecto solo trabajaran dos desarrolladores, para distribuir el trabajo se propuso que para cada módulo uno de los dos fuera el Scrum Manager y de esta manera tener un mayor control sobre la distribución del mismo. De este modo vemos a la facultad como los implicados o “gallinas” que son las personas quienes están interesadas en nuestro desarrollo.

Sprint:

Para este proyecto se tuvieron en cuenta y analizaron los requerimientos necesarios, los cuales nos dieron como resultado que se necesitan crear 5 sprint los cuales están destinados a desarrollar los 5 módulos necesarios para que el resultado del proyecto sea una prueba funcional del mismo

Los sprints que se realizaron fueron los siguientes

Sprint 1	Módulo Igu	12 de Septiembre	23 de Septiembre
	Objetivo: hacer la plantilla de la página y las hojas de estilo para darle forma		
	Criterios de Aceptación: se tiene una interfaz gráfica amigable que se puede implementar en todas las vista del proyecto		
	Tareas o Actividades: <ul style="list-style-type: none"> • Crear el código HTML que sirva como plantilla para la aplicación. • Crear la hoja de estilo para darle mejorar la interfaz con el usuario • Implementar el código HTML junto con la hoja de estilos para comprobar que funcione correctamente y la aplicación tenga la apariencia que se espera 		

Tabla N° 1 Sprint Módulo Igu.

Sprint 2	Módulo núcleo	24 de Septiembre	05 de Octubre
	Objetivo: crear el modulo que le da la estructura a la aplicación.		
	Criterios de Aceptación: los controladores creados cargan la vista correspondiente.		
	Tareas o Actividades: <ul style="list-style-type: none"> • Crear un controlador. • Crear una vista. • Verificar que el controlador creado este implementado la vista correcta. 		

Tabla N° 2: Sprint Módulo Núcleo.

Sprint 3	Modulo baseDatos	06 de Octubre	12 de Octubre
	<p>Objetivo: crear la conexión con la base de datos</p>		
	<p>Criterios de Aceptación: la aplicación de estar conectada con la base de datos , se debe realizar consultas a la base de datos y mostrar información en la interfaz del usuario.</p>		
	<p>Tareas o Actividades:</p> <ul style="list-style-type: none"> • realizar la conexión a la base de datos. • Comprobar que la conexión no tenga fallas. • Realizar consultas reales a la base de datos. 		

Tabla N° 3: Sprint Módulo Base De Datos.

Sprint 4	Modulo seguridad	13 de Octubre	23 de Octubre
	<p>Objetivo: configurar la aplicación para que verifique si los usuarios que ingresan son clientes o administradores</p>		
	<p>Criterios de Aceptación: crear dos usuarios uno con el rol de usuario y otro con el rol de administrador que al ingresar en la aplicación cargue una vista diferente por cada uno</p>		
	<p>Tareas o Actividades:</p> <ul style="list-style-type: none"> • Crear un usuario con el rol de usuario • Crear un usuario con el rol de administrador • Cargar una vista para el usuario • Cargar una vista para el administrador 		

Tabla N° 4: Sprint Módulo Seguridad.

Sprint 5	Modulo Torneo	24 de Octubre	04 de Noviembre
	<p>Objetivo: El objetivo de este Sprint construir el modulo que permita al administrador crear torneos</p>		
	<p>Criterios de Aceptación: Debe permitir al administrador crear un nuevo torneo eligiendo su categoría y horario, en este modulo también se pueden ver los torneos ya creados y permitir eliminarlos o editarlos.</p>		
	<p>Tareas o Actividades:</p> <ul style="list-style-type: none"> • Hacer el diagrama de clases. • Creación del modelo y el controlador. • Conceptualización de la vista para la creación de torneos • Crear la vista para permitir la creación de torneos • Probar que los torneos queden registrados en la base de datos. 		

Tabla N° 5: Módulo Torneo.

3 Análisis Y Diseño

3.1 Definición de requerimientos

Teniendo en cuenta los alcances de este proyecto, se plantean los siguientes requerimientos, los cuales son necesarios para lograr el funcionamiento ideal de la aplicación.

3.1.1 Requerimientos funcionales:

Según el análisis de las necesidades del cliente se determino que debe hacer el sistema, a partir de esto se identifican los siguientes requerimientos funcionales.

- La aplicación debe permitir logearse con la contraseña y el usuario de génesis
- Solo los usuarios que estén registrados en la base de datos de génesis pueden tener acceso a la aplicación
- La aplicación debe manejar roles para los administradores y para los clientes
- La aplicación debe permitir al administrador configurar los torneos y los eventos como lo desee
- También debe consolidar toda la información de los torneos
- La aplicación debe permitir a un Usuario inscribir un equipo en la categoría que desea.
- El cliente que tenga registrado un equipo puede hacerle modificaciones hasta un tiempo pre establecido
- El programa debe permitir subir archivos de diferente extensión como imágenes o documentos PDF para ser utilizados en lo que se necesite y se requiera
- Las tablas de los torneos deben actualizarse de forma automática después de que se ingresen los resultados de todos los encuentros

3.1.2 Requerimientos no funcionales

Teniendo en cuenta los requerimientos funcionales del proyecto se necesitan los siguientes requerimientos no funcionales para lograr que el proyecto funcione

- La página debe ser rápida en su navegación, y debe ser segura a la hora de enviar datos.
- La página debe cumplir los estándares web de seguridad para prevenir ataques al sistema.
- Se debe tener un estándar para que la página sea funcional en todos los navegadores.
- Se necesita un servidor en donde este instalado PHP 5.3, MySQL y un dominio para poder alojar la sistema web.
- Para el funcionamiento ideal de este sistema se necesita una conexión autorizada al LDAP de la corporación universitaria minuto de dios.
- Para en funcionamiento el sistema es necesario contar con un computador que tenga conexión a internet ya sea a través de una conexión física o una conexión inalámbrica.
- Dos computadores con un IDE de desarrollo y un servidor local apache que entienda el lenguaje PHP para poder desarrollar la aplicación.
- Un dominio ya sea .edu.co o .com.co para que los estudiantes tenga acceso al sistema de información desde el lugar que lo necesite.

3.2 Descripción del Sistema propuesto

El Sistema de información VU es una aplicación web que permite optimizar los procesos que realiza bienestar universitario, tales como la administración de los

torneos internos de la universidad, además programación de partidos, resultados de los encuentros, convocatoria.

El proyecto está dividido en módulos los cuales cumplen funciones específicas, para llevar a cabo el objetivo general del sistema. Se dividió el total del sistema en 5 módulos los cuales son:

Igu: Es el modulo donde se crean todas las vistas necesarias para que los usuarios finales administren la información de la mejor forma.

Núcleo: Es el modulo encargado de hacer la comunicación entre los datos y la interfaz gráfica, lleva la información hasta la base de datos y también muestra las consultas solicitadas desde la interfaz gráfica.

Base de datos: Se encarga de manejar los datos que son ingresados en el sistema, del mismo modo almacena la información necesaria para ser consultada cuando sea necesario o se haga alguna modificación de la misma.

Seguridad: Ya que el sistema usa dos roles los cuales son Administrador y usuario se creó este módulo para que cada rol al entrar al sistema pueda tener acceso solo a la parte que esta diseñada para él.

Torneo: Este es el modulo que administra todo lo que se puede en el torneo, es el modulo donde se crean los torneos, se hacen las fechas de los encuentros, se calculan las tablas de posiciones y se muestra los resultados al usuario.

El sistema VU sigue el patrón Modelo Vista Controlador (MVC) su desarrollo hace uso del paradigma de programación orientada a componentes, se implementaron los lenguajes de programación PHP, JavaScript Y MySQL también se usó el lenguaje HTML para manejar la interfaz gráfica.

Este sistema de información está dividido para dos clases de usuarios, **los administradores** que pueden modificar la información que se encuentra disponible para los demás usuarios y **los usuarios comunes** que solo acceden al sistema para consultar la información.

Los administradores tienen acceso al sistema de la misma forma que los usuarios comunes, deben digitar su ID y su contraseña para autenticarse, una vez dentro del sistema pueden tener acceso a las categorías de cada torneo. En cada una de estas categorías el administrador va a poder ver cuáles son los equipos que se encuentran registrados así como sus integrantes.

La función primordial del administrador es ingresar al sistema para crear los torneos de cada categoría, darle un nombre a cada torneo, ingresar las fechas en las que se realizarán los enfrentamientos y los resultados de los mismos.

Los usuarios comunes deben ingresar al sistema con sus usuarios y contraseñas de génesis, una vez estén autenticados es en el sistema podrán ver en cada

categoría los torneos que están disponibles y si lo desean pueden crear un equipo; la forma para crear dicho equipo consiste en que el capitán del equipo inscribe a todos los jugadores que lógicamente también deben pertenecer a Uniminuto ya que se debe colocar el número del ID para el registro, luego de esto el capitán selecciona el nombre del equipo y un escudo que los represente.

Los usuarios comunes también pueden ingresar al sistema para verificar la fecha en la que deben jugar, pueden ver la tabla de posiciones de su torneo e igualmente saber cuál es el mayor anotador de su categoría.

3.3 Diseño del sistema propuesto

El diseño del sistema se va a fundamentar en un diseño estructural o vista estática del sistema, además en un diseño funcional o vista dinámica del sistema. Se va a trabajar aplicando el lenguaje UML (Lenguaje Unificado de Modelamiento).

Para el desarrollo de modelo estructural se utilizarán los diagramas de Clases y Objetos, en el modelo funcional serán implementados el diagrama de secuencia y los escenarios sugeridos por la metodología implementada en los cuales se muestran cómo se utilizará la página con su equivalente a los casos de uso.

3.3.1 Diagramas estáticos

Con estos diagramas se representarán los componentes como entidades, objetos, clases, etc, que se van a implementar en los aspectos necesarios en el desarrollo. También se utilizan para especificar como debe estar estructurado el sistema, para esto se utilizaron los siguientes diagramas.

Diagrama de Clases:

Este módulo se encarga de administrar los controladores y asociarlos a la clase página, la cual se encarga de administrar la localización de las vistas del sistema, adicional a eso creo un objeto de tipo seguridad para el manejo de seguridad del sistema.

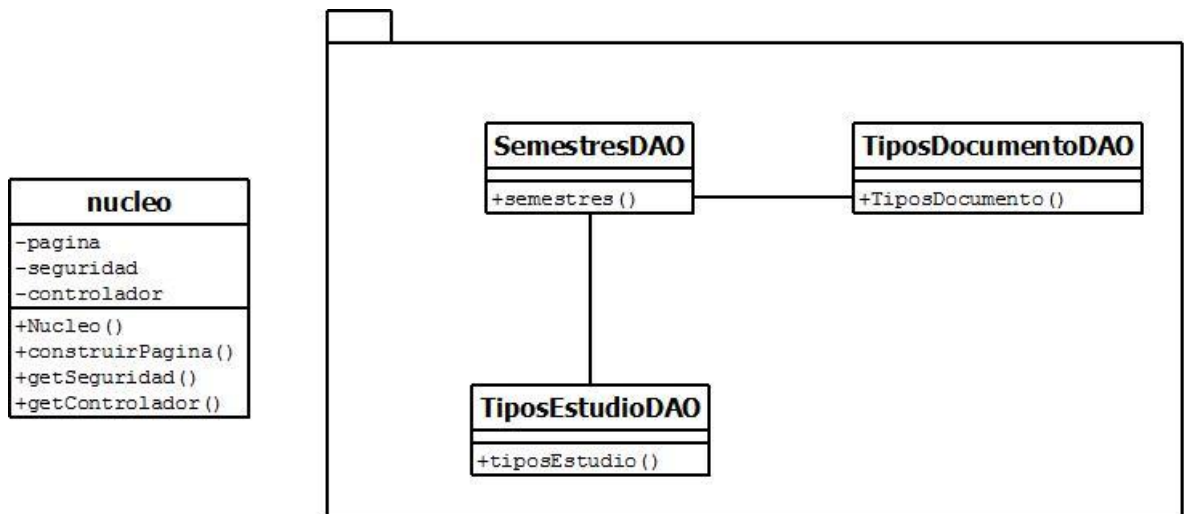


Diagrama 1: Módulo núcleo

Este es el diagrama del módulo baseDatos el cual se de crear la conexión a la base de datos, y de ejecutar las consultas y procesos que se hacen en la base de datos como inserciones de datos.

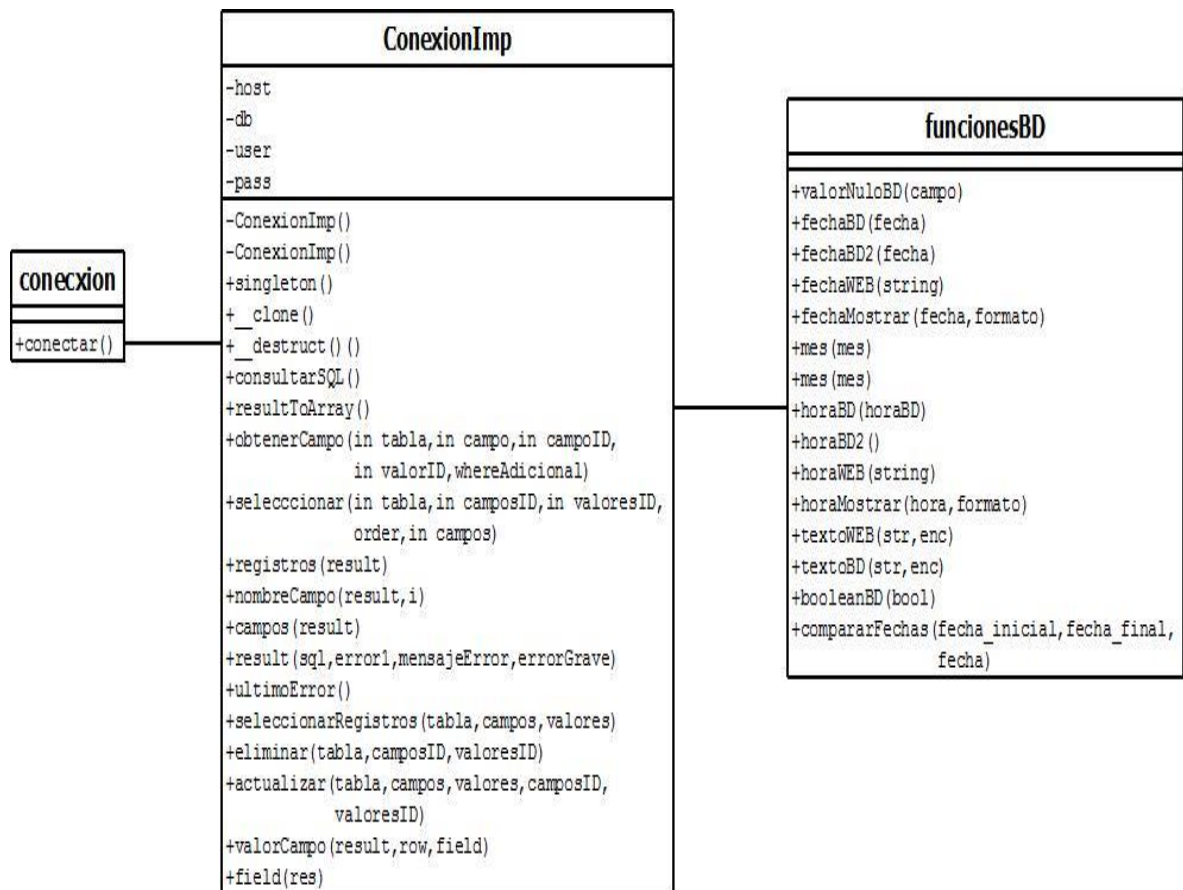


Diagrama 2: Módulo baseDatos

En el módulo de seguridad se administran los permisos para que la aplicación reconozca que tiempo de usuario está ingresando, ya sea administrador o usuario y así cargar los permisos correspondientes.

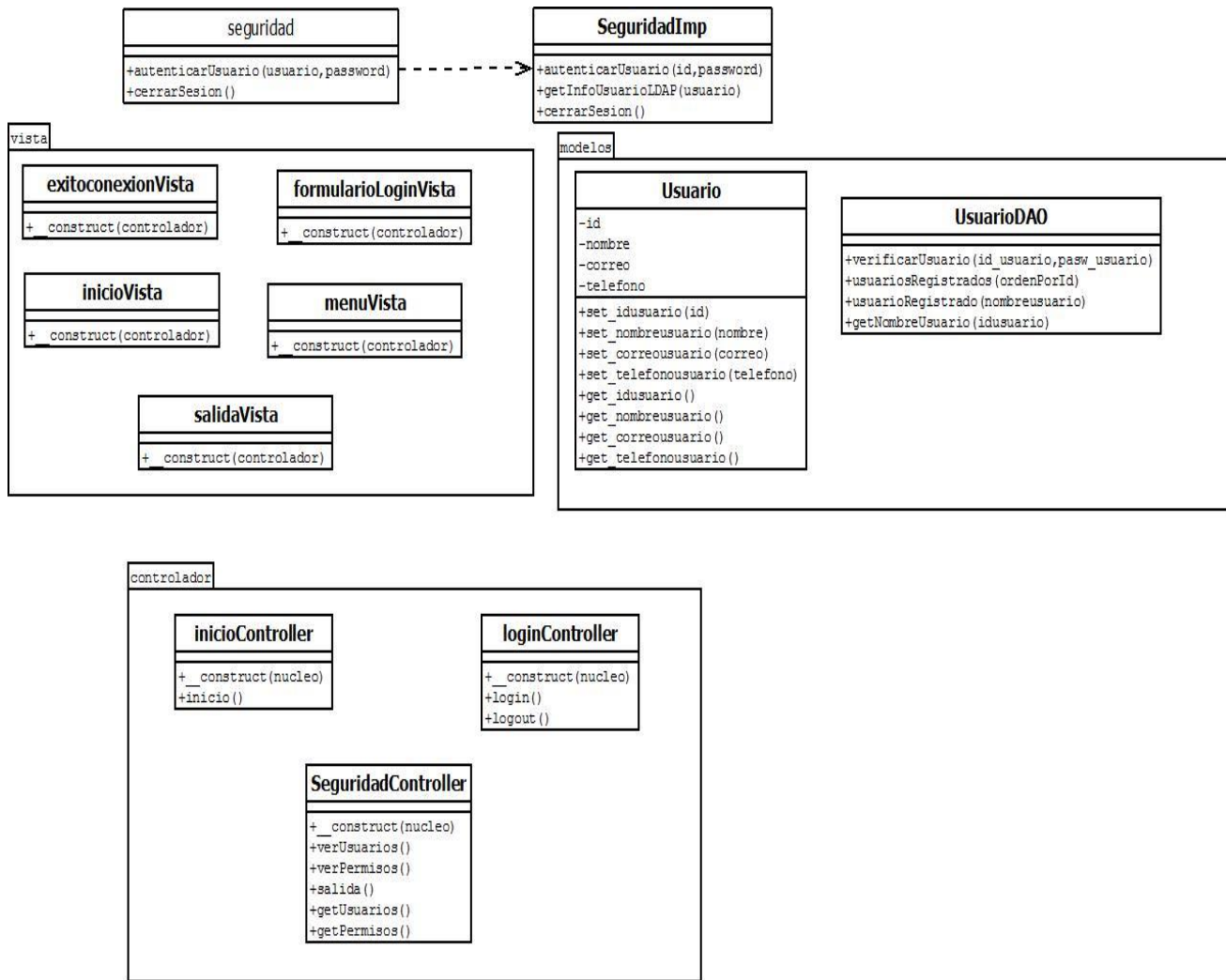


Diagrama 3: Módulo seguridad

Este es el diagrama del módulo igu el cual se encargará de crear la vista para la interacción con el usuario.

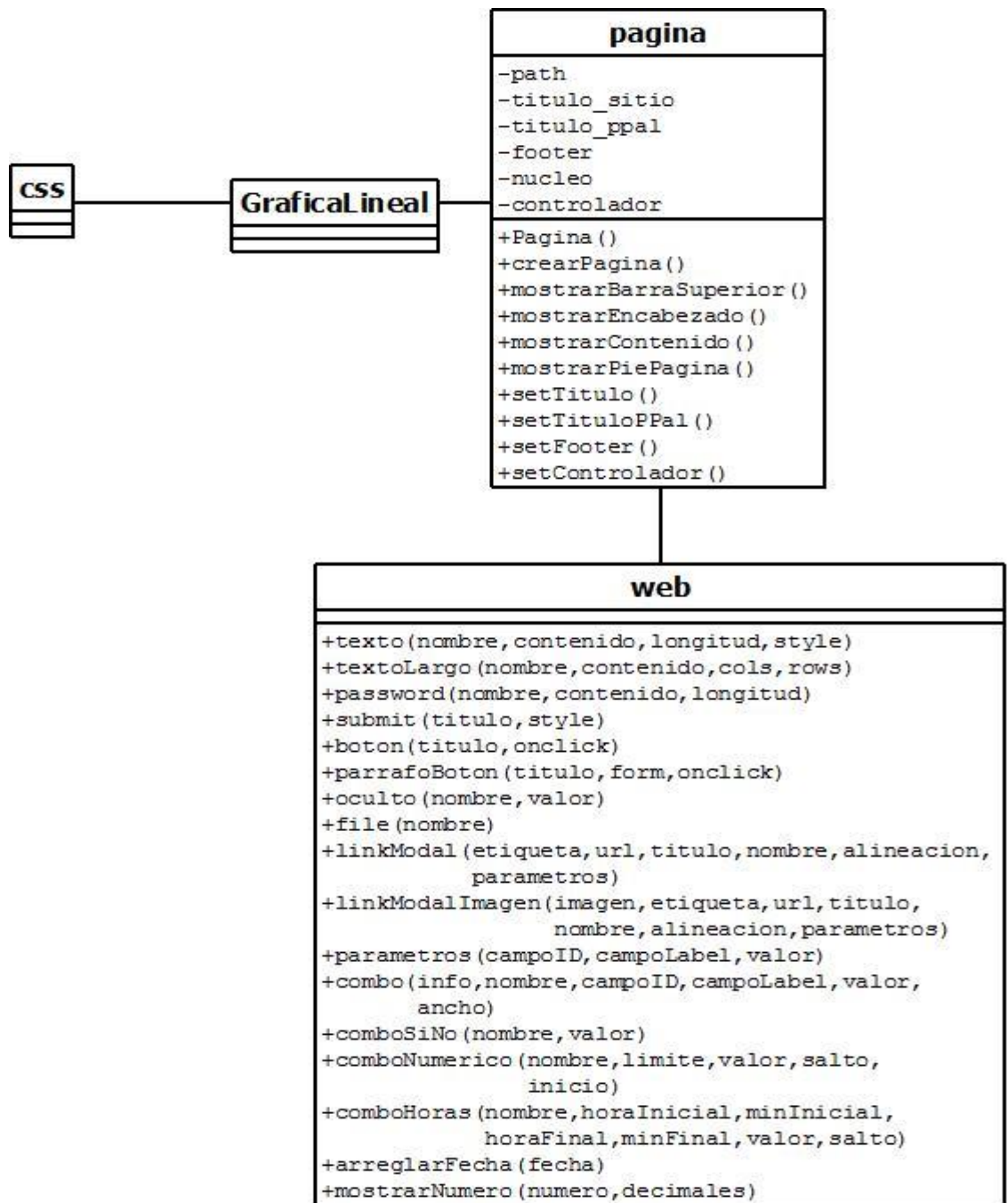


Diagrama 4: Módulo igu

El modelo de torneos permite al administrador del sitio crear los torneos para que los demás usuarios se puedan registrar y participar.

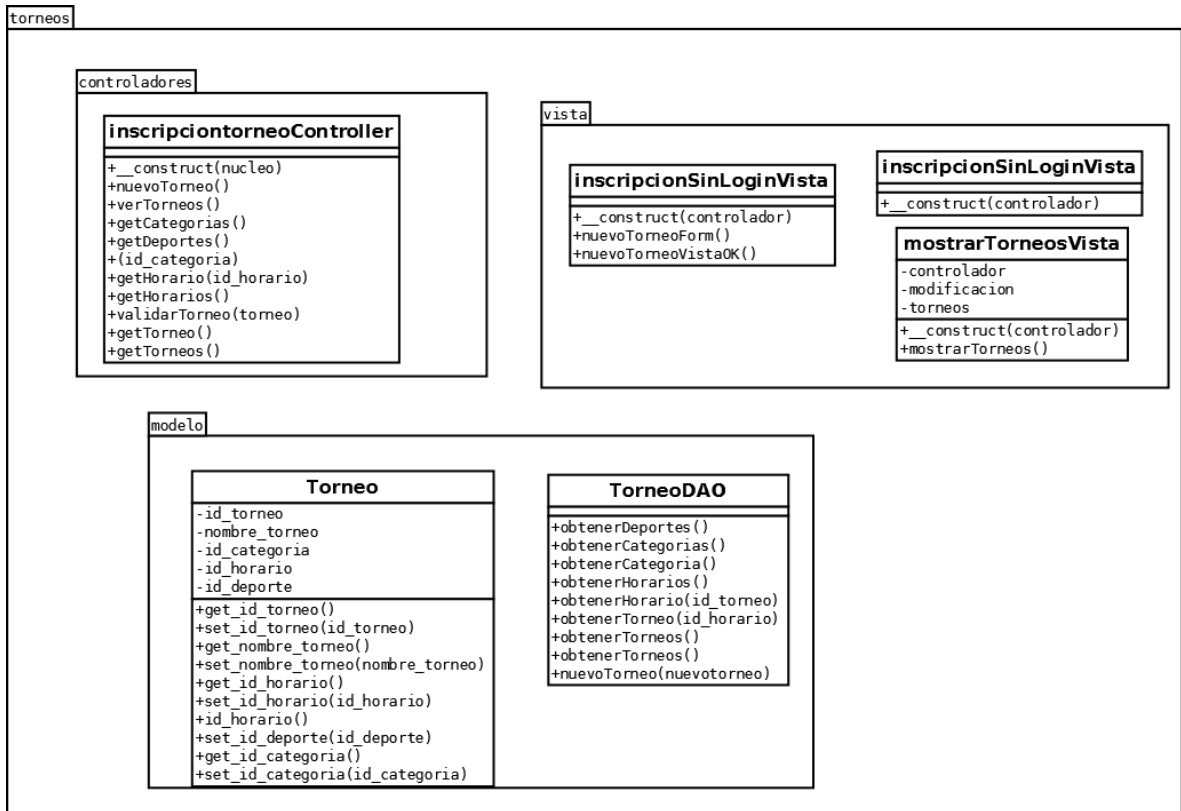


Diagrama 5: Modulo torneos

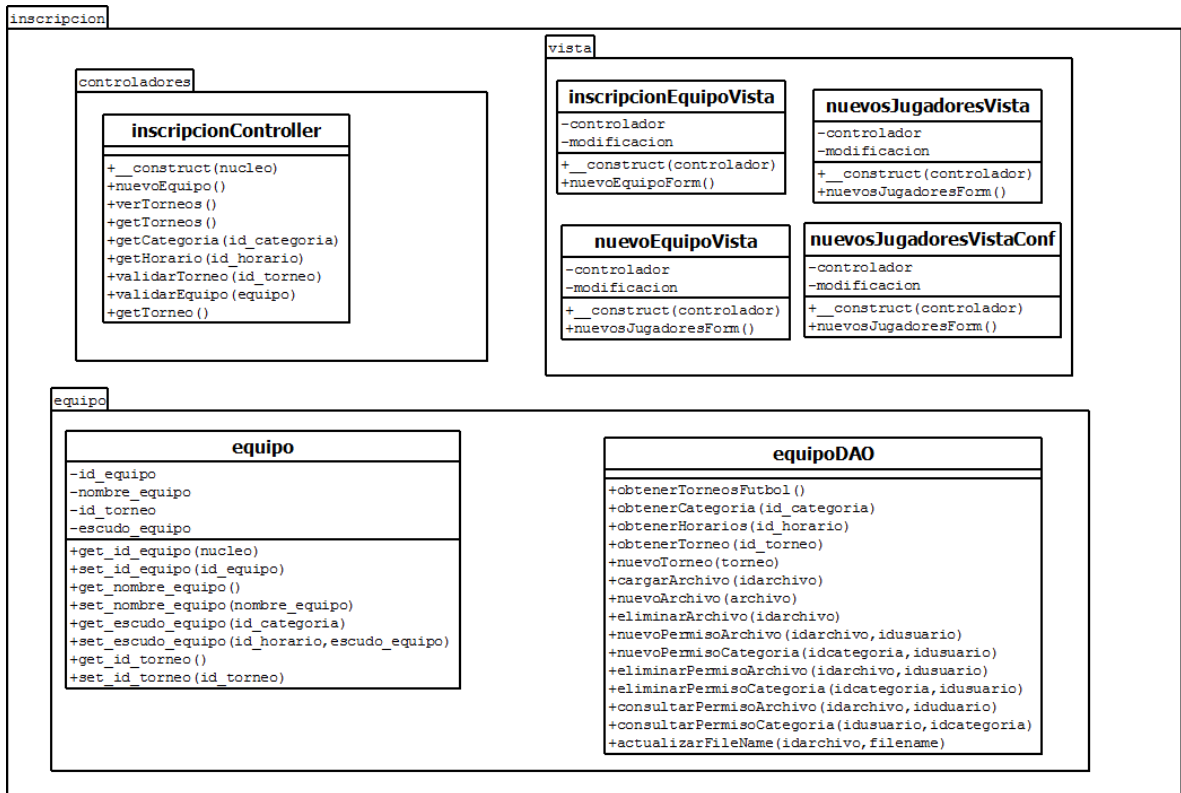


Diagrama 6: Modulo de Inscripción

Diagramas MER:

Diagrama MER Torneo

En el diagrama Modelo Entidad Relación de torneos están todas las tablas que se utilizan en la organización y desarrollo de los torneos

Este modelo consiste en 8 tablas empezando con la de torneos, en la cual se guarda el nombre del torneo la categoría a la que este pertenece y los horarios en los que se juegan estos datos se obtienen de las tablas Categoría y Horario, también se encuentra la tabla equipo que tiene el nombre del equipo y el torneo en el cual está participando dicho equipo, la tabla equipo tiene asociada la tabla jugador donde está el nombre del jugador y el equipo en el que está inscrito, y la tabla Registro_partido donde queda registrado el equipo que jugó y la fecha del partido que se obtiene de la tabla fecha, por último se tiene la tabla gol en donde queda registrado el minuto en el que se marcó, el jugador que lo marco y el partido donde se marcó.

En el siguiente diagrama podemos ver la relación de las tablas que se implementan en el modelo de la base de datos necesarios para el correcto funcionamiento del sistema

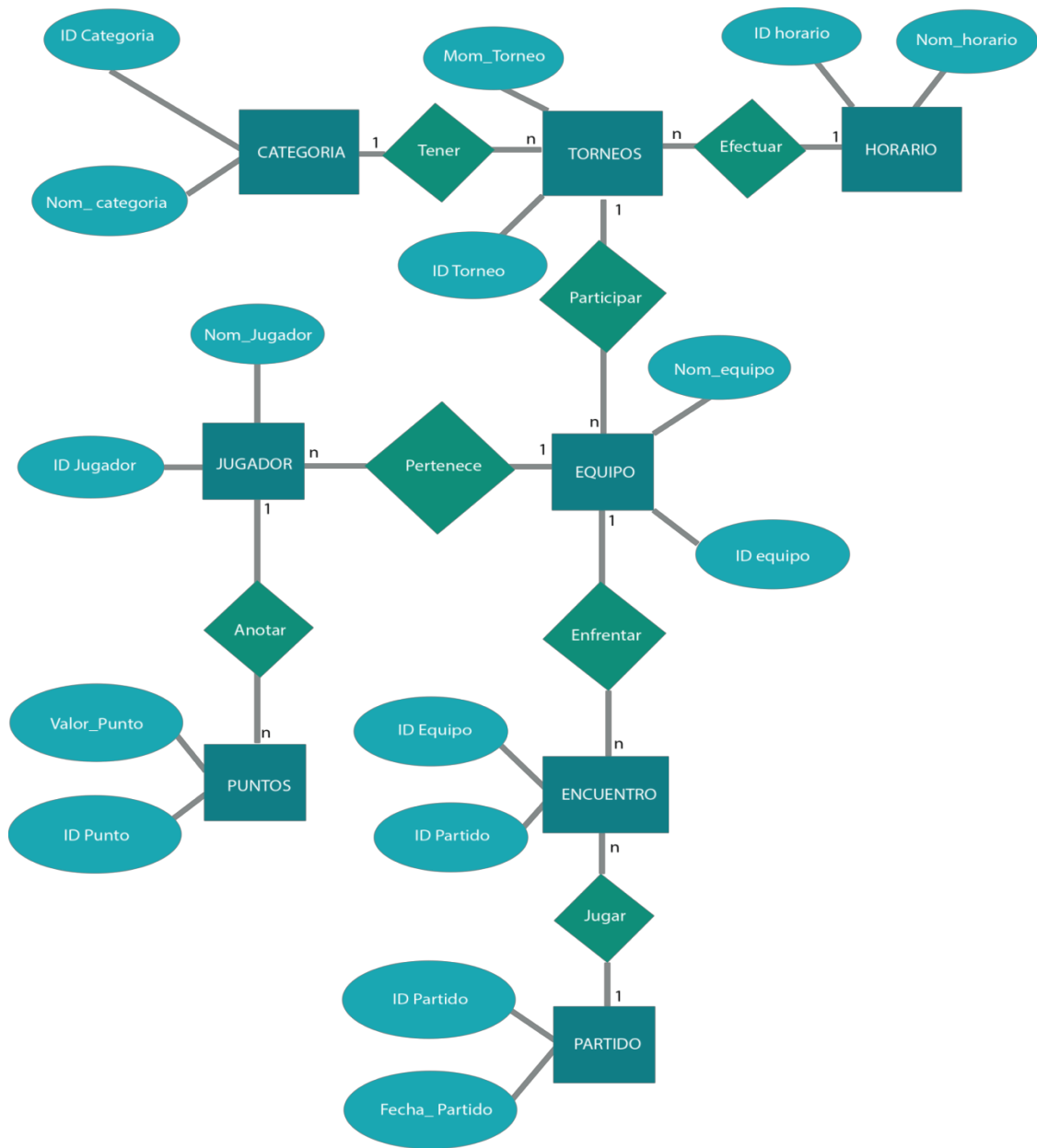


Diagrama 7 Modelo entidad relacion

3.3.2 Diagramas Dinámicos

Con estos diagramas se representa el comportamiento que debe tomar el sistema cuando los usuarios acceden e interactúan con todas las opciones que se les pueden brindar.

Diagrama Casos de uso:

Son los diagramas que se encargan de describir el comportamiento del sistema al afrontar un requerimiento funcional e interacción con los roles del sistema.

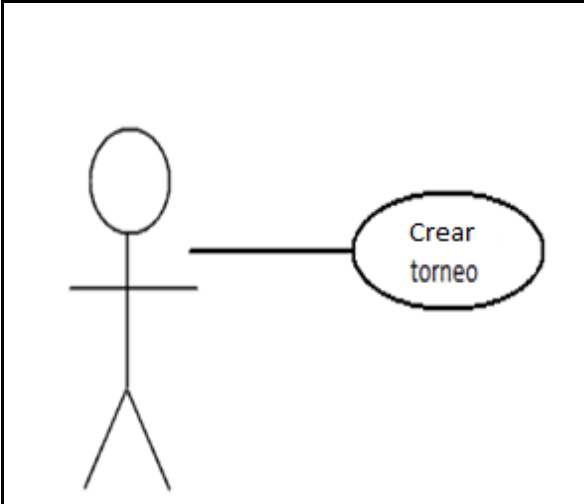
Diagrama 1: caso de uso crear torneo		
	id	1
	actor	administrador
	descripción	El administrador ingresa al sistema para crear un nuevo torneo de cualquier categoría
	Pre-condición	Tener permiso de administrador
	flujo normal	El administrador crea un torneo
	Pos-condición	El torneo se crea y los demás usuarios pueden verlo
	flujo alternativo	
	descripción	El torneo ya está creado
	Pos-condición	Se crea como un nuevo torneo

Diagrama 8 Caso de uso Crear Torneo

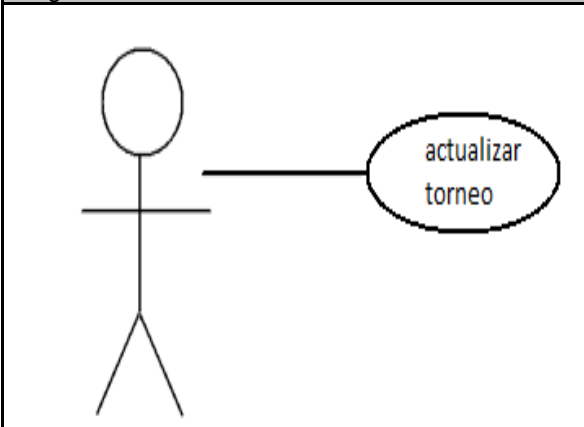
Diagrama 2: caso de uso actualizar torneo		
	id	2
	actor	administrador
	descripción	El administrador actualiza un torneo
	Pre-condición	Tener permiso de administrador
	flujo normal	El administrador actualiza el torneo
	Pos-condición	El torneo queda actualizado
	flujo alternativo	
	descripción	El torneo ya está en marcha
	Pos-condición	mensaje de error

Diagrama 9 caso de uso actualizar torneo

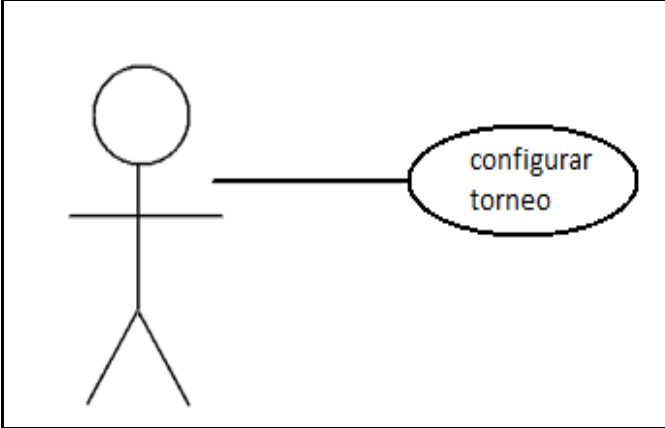
Diagrama 3: caso de uso configurar torneo		
	id	3
	actor	administrador
	descripción	El administrador crea o actualiza un torneo
	Pre-condición	Permiso de administrador
	flujo normal	El administrador configura un torneo
	Pos-condición	El torneo queda actualizado
	flujo alternativo	
	descripción	El torneo ya está en marcha
	Pos-condición	mensaje de error

Diagrama 10 caso de uso configurar torneo

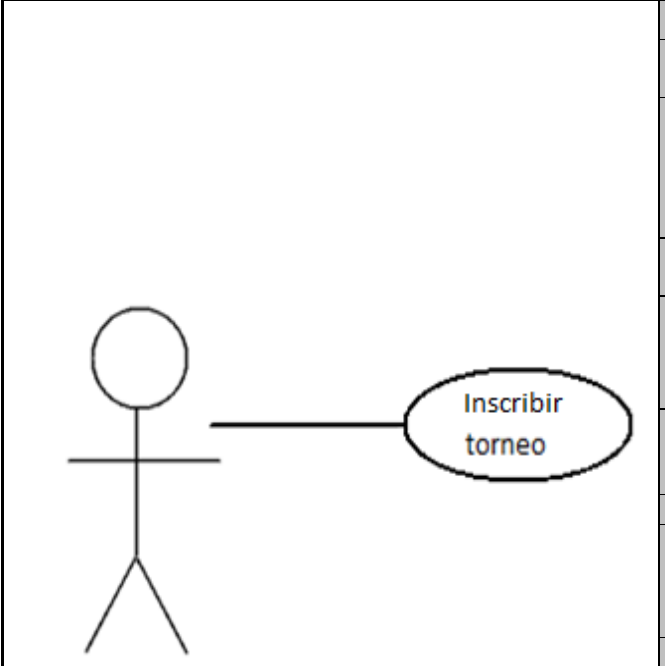
Diagrama 4: caso de uso Inscribir torneo		
	id	4
	actor	Administrador y/o Usuario
	descripción	El administrador o usuario puede ingresar al sistema para inscribir un equipo en el torneo que desea
	Pre-condición	Estar logeado en el sistema r
	flujo normal	El usuario ingresa al sistema selecciona el torneo que desee e inscribe su equipo.
	Pos-condición	El queipo queda registrado en el torneo que se selecciono
	flujo alternativo	
	descripción	Ya no hay cupos para el torneo o la fecha de inicio del torneo ya ha pasado
	Pos-condición	mensaje de error

Diagrama 11 Caso de uso Inscribir Torneo

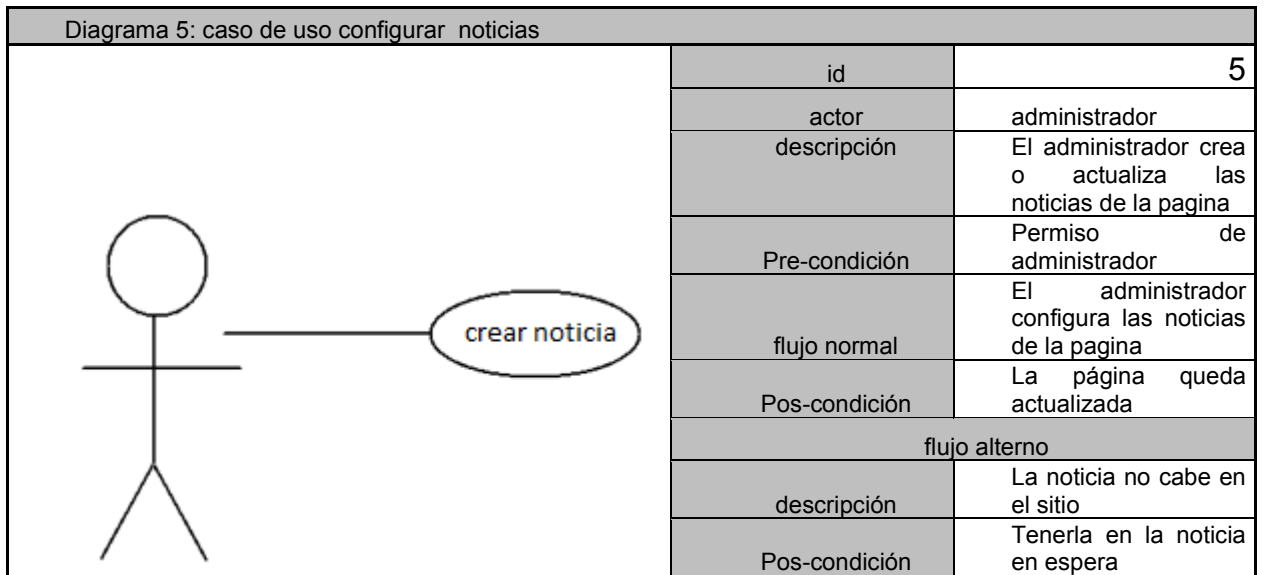


Diagrama 12 caso de uso configurar noticias.

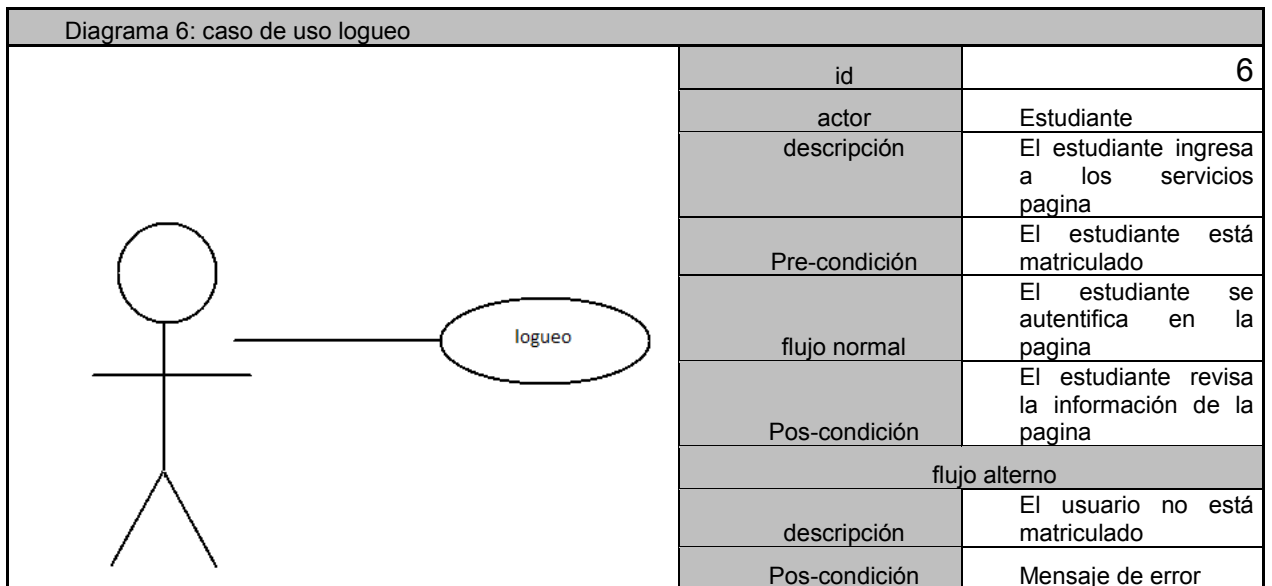


Diagrama 13 caso de uso logueo.

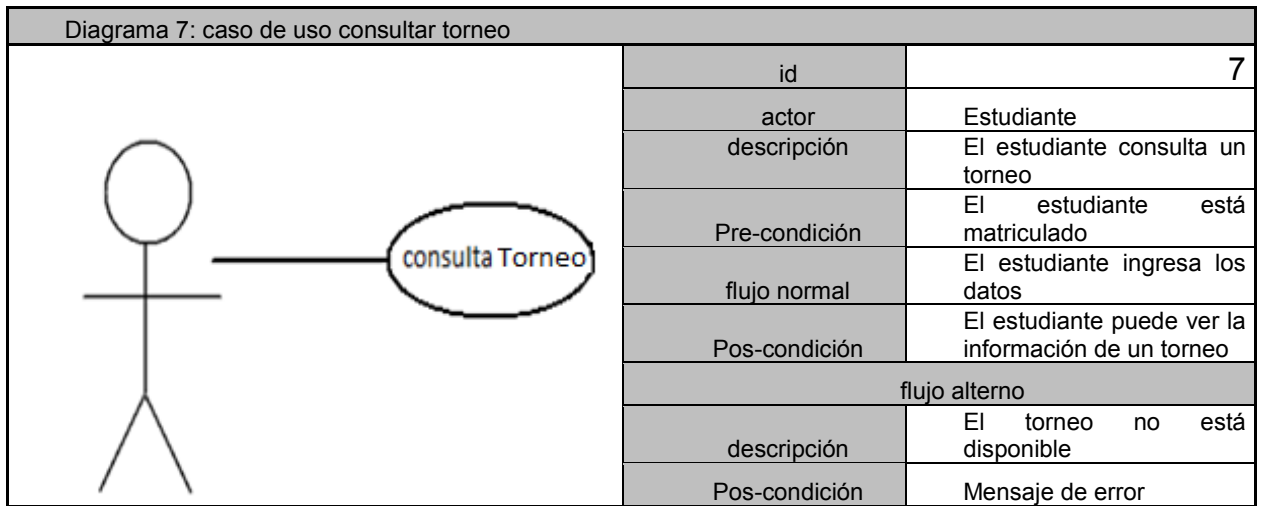


Diagrama 14 caso de uso consultar torneo.

Diagramas de secuencia

Los diagramas de secuencia son utilizados para modelar la iteración entre los objetos de un sistema según UML.

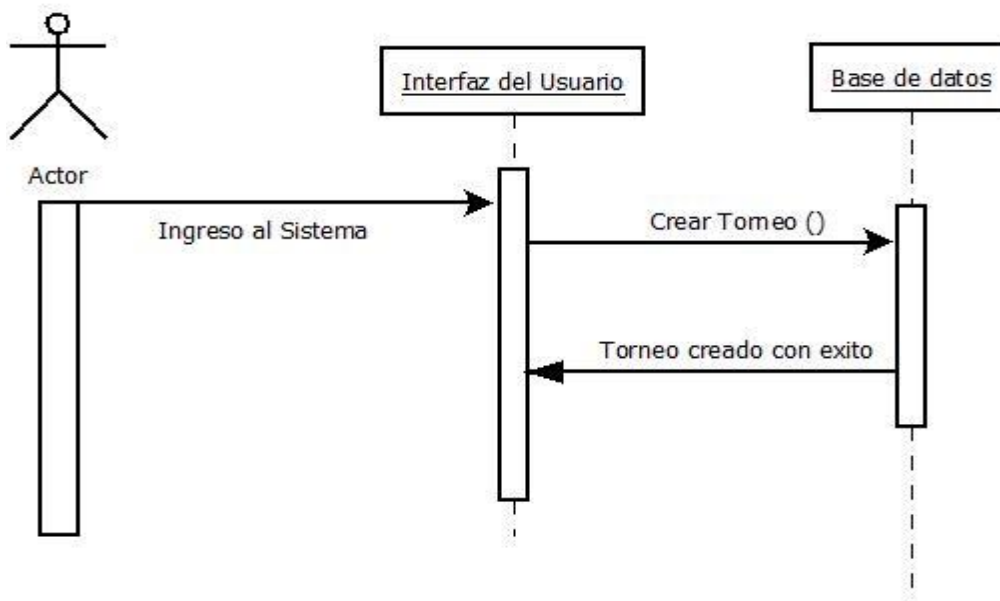


Diagrama 15 Secuencia crear Torneo

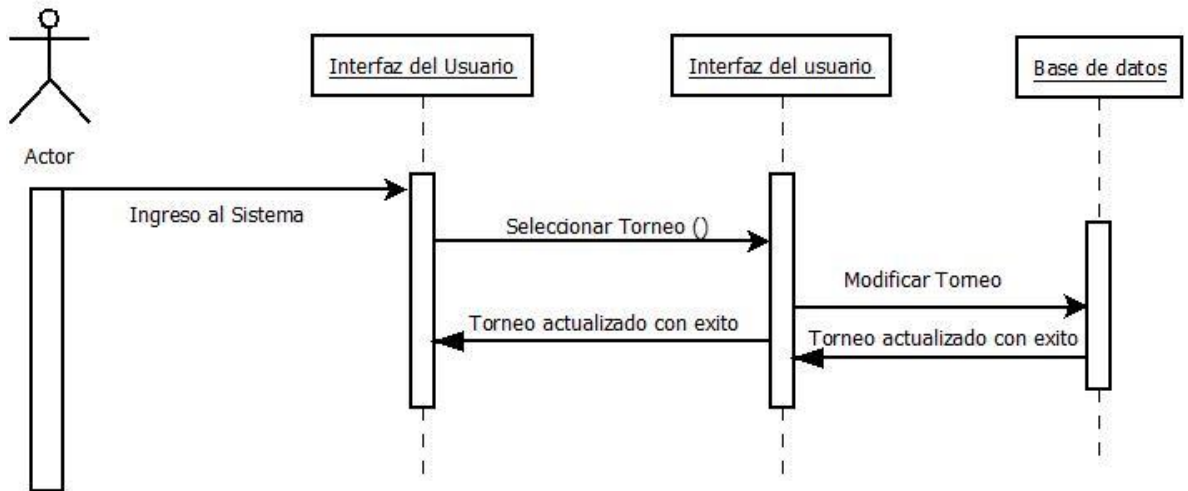


Diagrama 16 Secuencia Modificar Torneo

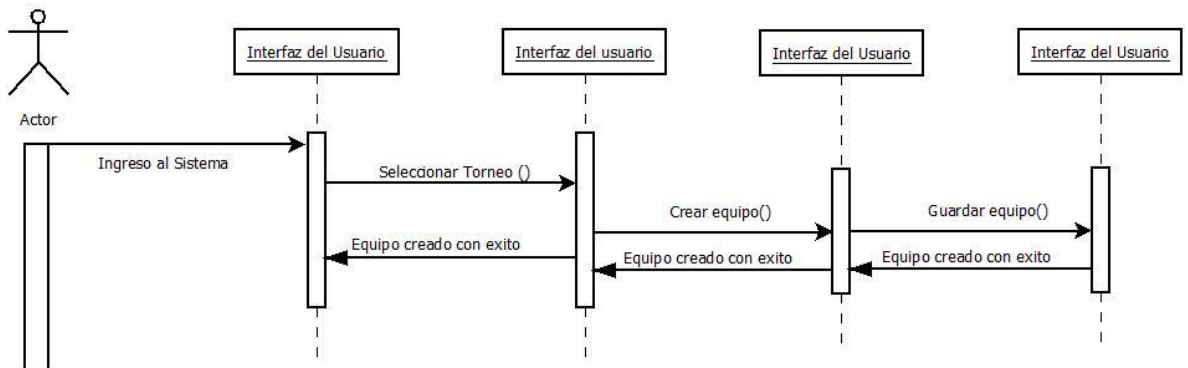


Diagrama 17 Secuencia Crear Equipo

4 Glosario

Para que se pueda entender mejor este documento se han seleccionado los siguientes términos teniendo en cuenta su complejidad.

- **Página web:** documento creado en formato HTML (Hypertext Markup Language) que es parte de un grupo de documentos hipertexto o recursos disponibles en el World Wide Web. Una serie de páginas web componen lo que se llama un sitio web.
- **Software:** conjunto de programas, documentos, procedimientos, y rutinas asociadas con la operación de un sistema de cómputo. Distinguiéndose de los componentes físicos llamados hardware. Comúnmente a los programas de computación se les llama software; el software asegura que el programa o sistema cumpla por completo con sus objetivos, opera con eficiencia, esta adecuadamente documentado, y suficientemente sencillo de operar.
- **Sistema:** En un sistema informático se utilizan computadoras para almacenar, procesar y/o acceder a información.
- **Patrón de diseño:** base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces.
- **Interfaz:** parte de un programa que permite el flujo de información entre un usuario y la aplicación, o entre la aplicación y otros programas o periféricos. Esa parte de un programa está constituida por un conjunto de comandos y métodos que permiten estas intercomunicaciones.
- **Base de datos:** es una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una empresa o negocio en particular.

- **Servidor:** es un tipo de software que realiza ciertas tareas en nombre de los usuarios. El término servidor ahora también se utiliza para referirse al ordenador físico en el cual funciona ese software, una máquina cuyo propósito es proveer datos de modo que otras máquinas puedan utilizar esos datos.
- **PHP:** Lenguaje de programación usado generalmente en la creación de contenidos para sitios web. Es un lenguaje interpretado especialmente usado para crear contenido dinámico web y aplicaciones para servidores, aunque también es posible crear aplicaciones gráficas utilizando la biblioteca GTK+.
- **MySQL:** MySQL es un sistema de gestión de bases de datos (SGBD) multiusuario, multiplataforma y de código abierto. MySQL pertenece a la compañía sueca MySQL AB, a la que le pertenece casi todos los derechos del código fuente. La compañía desarrolla y mantiene el sistema, vendiendo soporte y servicios, como también las licencias para usar MySQL.
- **PostgreSQL:** es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales.
- **LDAP:** (Protocolo compacto de acceso a directorios) es un protocolo estándar que permite administrar directorios, esto es, acceder a bases de información de usuarios de una red mediante protocolos TCP/IP. Las bases de información generalmente están relacionadas con los usuarios, pero, algunas veces, se utilizan con otros propósitos, como el de administrar el hardware de una compañía.
- **TCP/IP:** son las siglas de Protocolo de Control de Transmisión/Protocolo de Internet (en inglés Transmission Control Protocol/Internet Protocol), un sistema de protocolos que hacen posibles servicios Telnet, FTP, E-mail, y otros entre ordenadores que no pertenecen a la misma red.

5. Conclusiones

Gracias a la metodología de desarrollo SCRUM se pudo realizar un análisis, diseño y desarrollo ordenado del sistema de información orientado a la web VU para torneos de la Corporación Universitaria Minuto de Dios para consolidar información de estos eventos.

El sistema provee una mayor seguridad en el tránsito de la información e incentiva a más estudiantes a participar de los torneos. Tras una interfaz muy amigable con el usuario, se ve el continuo interés de los estudiantes por visitar este sitio.

Se logra resolver el problema planteado, al construir un sistema de información constante, que sirve para inscripción de torneos, además de esto provee toda información relacionada con estos, como lo son horarios, fechas resultados, anuncios, noticias importantes, etc .

BIBLIOGRAFIA

- UML El lenguaje unificado de Modelado GrandyBooch, JimRumbaugh e Ivar Jacobson; <http://elvex.ugr.es/decsai/java/pdf/3E-UML.pdf> Consultada en la fecha 20/10/11. (PUBLICACION ELECTRONICA)
- Lenguaje Unificado de Modelado; Fecha Publicación 10/11/2011 http://es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado#Cr.C3.ADticas_a_UML Consultada en la fecha 23/10/11. (PUBLICACION ELECTRONICA)
- Pergamino virtual.; Fecha Publicación 10/09/2011 http://www.pergaminovirtual.com.ar/definicion/Pagina_web.html en la fecha 09/07/12. (PUBLICACION ELECTRONICA)
- El siglo de torreon; Fecha Publicación 02/12/2002 <http://www.elsiglodetorreon.com.mx/noticia/14864.html> en la fecha 09/07/12. (PUBLICACION ELECTRONICA)
- Elegsa; Fecha Publicación 12/03/2012 <http://www.alegsa.com.ar/Dic/sistema%20informatico.php> en la fecha 09/07/12. (PUBLICACION ELECTRONICA)
- Elegsa; Fecha Publicación 09/12/2011 <http://www.alegsa.com.ar/Dic/interfaz.php> en la fecha 09/07/12. (PUBLICACION ELECTRONICA)
- Elegsa; Fecha Publicación 02/02/2012 <http://www.alegsa.com.ar/Dic/php.php> en la fecha 09/07/12. (PUBLICACION ELECTRONICA)
- Elegsa; Fecha Publicación 02/02/2012 <http://www.alegsa.com.ar/Dic/mysql.php> en la fecha 09/07/12. (PUBLICACION ELECTRONICA)
- Wikipedia; Fecha Publicación 02/06/2012 http://es.wikipedia.org/wiki/Patr%C3%B3n_de_dise%C3%B1o en la fecha 09/07/12. (PUBLICACION ELECTRONICA)
- Maestros del web; Fecha Publicación 02/04/2012 <http://www.maestrosdelweb.com/principiantes/%C2%BFque-son-las-bases-de-datos/> en la fecha 09/07/12. (PUBLICACION ELECTRONICA)
- Maestros del web; Fecha Publicación 22/05/2012 <http://www.masadelante.com/faqs/servidor/> en la fecha 09/07/12. (PUBLICACION ELECTRONICA)

Manuales



Manual Técnico

Miguel Angel Camargo Campos y Fabian Andres Benavides

Sistema VU para la administración de torneos Internos de la corporación
Universitaria Minuto de Dios para la sede principal.

Corporación Universitaria Minuto de Dios Bogotá D.C. – Colombia

Manual Técnico

Instalación y configuración del Sistema VU,
para la administración de torneos internos
de la corporación Universitaria Minuto de Dios
para la sede principal.

Corporación Universitaria Minuto de Dios

Bogotá D.C. – Colombia

Tabla de contenido

pag

1. INTRODUCCION..... 1
1.1 Definiciones Previas 1
2. Modo de Implementación. 3

1. INTRODUCCION

VU es un sistema de información orientado a la web, que busca mejorar el proceso de administración de torneos internos en la sede Principal de la Corporación Universitaria Minuto de Dios (Uniminuto) , como lo es el proceso de inscripción, el manejo de las estadísticas y la comunicación de toda la información relacionada con estos.

El presente manual de usuario, se encarga de especificar el comportamiento de los componentes del proyecto, el modelo de desarrollo, el lenguaje utilizado el comportamiento del sistema y las clases que lo componen.

Se especifica la interacción del software con el usuario y entre componentes, requerimientos del sistema a nivel hardware/software, además de la especificación de los términos utilizados, con el objetivo de guiar y mostrar todo lo trabajado en este software a las personas interesadas.

1.1 Definiciones Previas

Componente: aquellos recursos desarrollados para un fin concreto y que puede formar solo o junto con otros, un entorno funcional requerido por cualquier proceso predefinido.

Módulo: son unidades que pueden ser compiladas por separado y los hace reusables y permite que múltiples programadores trabajen en diferentes módulos en forma simultánea, produciendo ahorro en los tiempos de desarrollo.

MVC: es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

Localhost: nombre reservado de un equipo router o dispositivo.

PHP: Lenguaje de programación usado generalmente en la creación de contenidos para sitios web. Es un lenguaje interpretado especialmente usado para crear contenido dinámico web y aplicaciones para servidores, aunque también es posible crear aplicaciones gráficas utilizando la biblioteca GTK+.

PostgreSQL: es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos

comerciales.

LDAP (*Protocolo compacto de acceso a directorios*) es un protocolo estándar que permite administrar directorios, esto es, acceder a bases de información de usuarios de una red mediante protocolos TCP/IP.

Las bases de información generalmente están relacionadas con los usuarios, pero, algunas veces, se utilizan con otros propósitos, como el de administrar el hardware de una compañía.

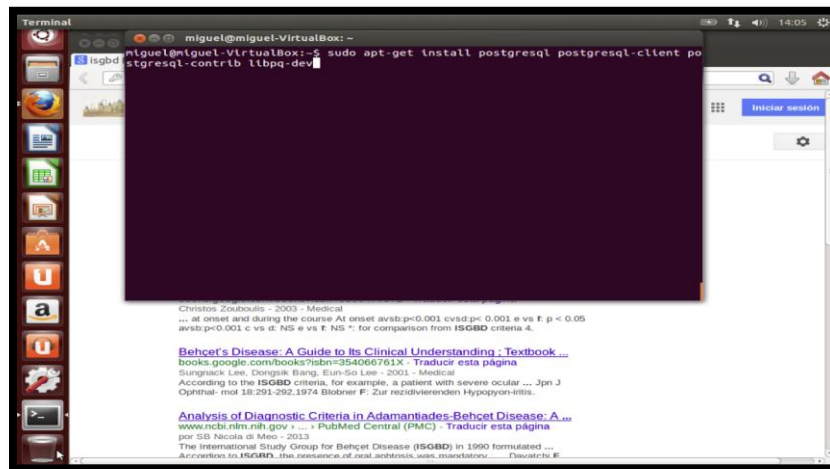
TCP/IP son las siglas de Protocolo de Control de Transmisión/Protocolo de Internet (en inglés *Transmission Control Protocol/Internet Protocol*), un sistema de protocolos que hacen posibles servicios Telnet, FTP, E-mail, y otros entre ordenadores que no pertenecen a la misma red.

VU esta desarrollado con el lenguaje de programación PHP, el cual es orientado a la web, con conexión a una base de datos de PostgreSQL y una conexión al LDAP de Uniminuto, basado en el patrón de diseño MVC, la arquitectura estructural de VU se basa en un único componente que alberga los diferentes módulos del sistema.

2. Modo de Implementación.

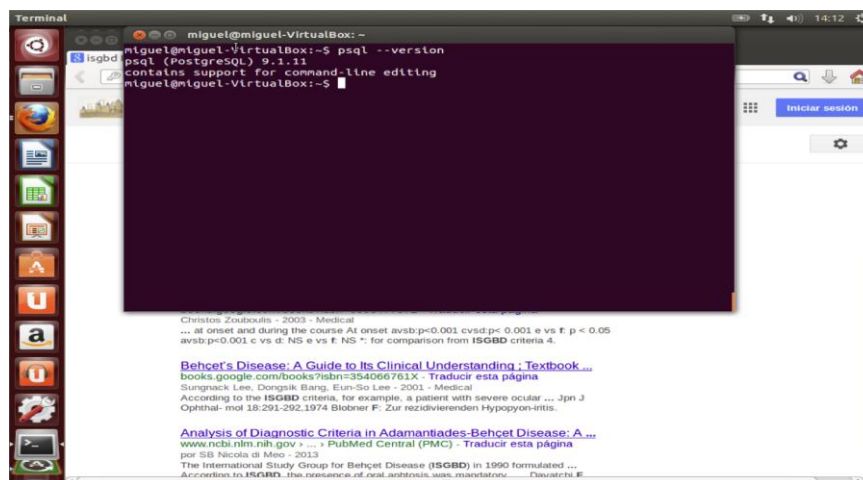
Actualmente lo primero que se debe tener es un servidor con sistema operativo Linux, para la realización de este manual se monto en un servidor Ubuntu 13, En primera instancia debemos configurar el servidor de base de datos, que como se menciona es PostgreSQL.

1. Abrimos una consola y digitamos el comando “**sudo apt-get install postgresql postgresql-client postgresql-contrib libpq-dev**”:

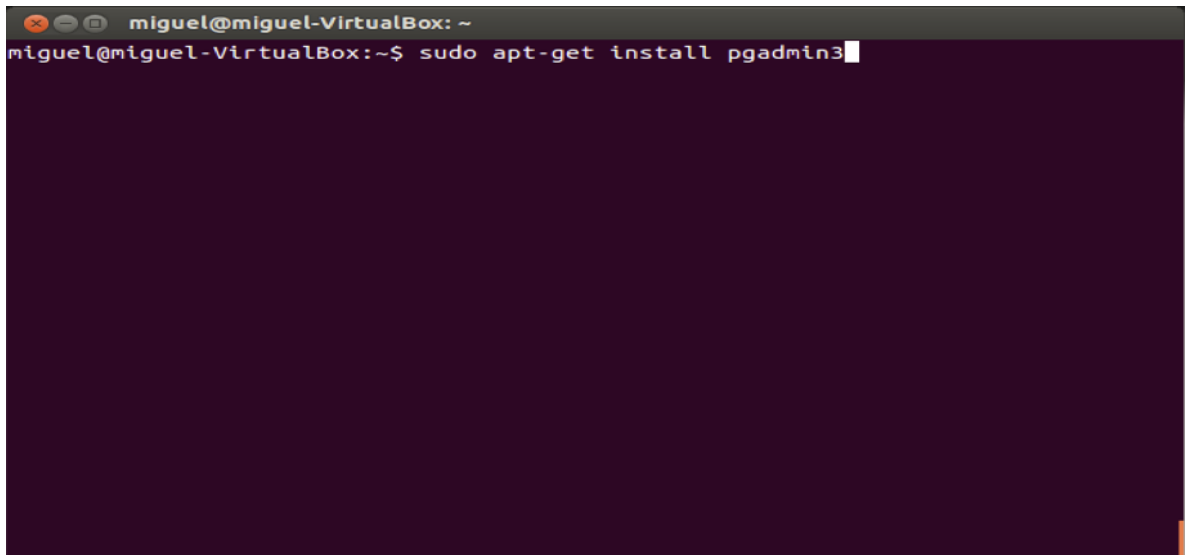


Esto instala el cliente y servidor de la base de datos y algunos scripts de utilidad. El paquete de **libpq-dev** nos servirá para poder compilar la gema **Ruby** para **PostgreSQL** más adelante.

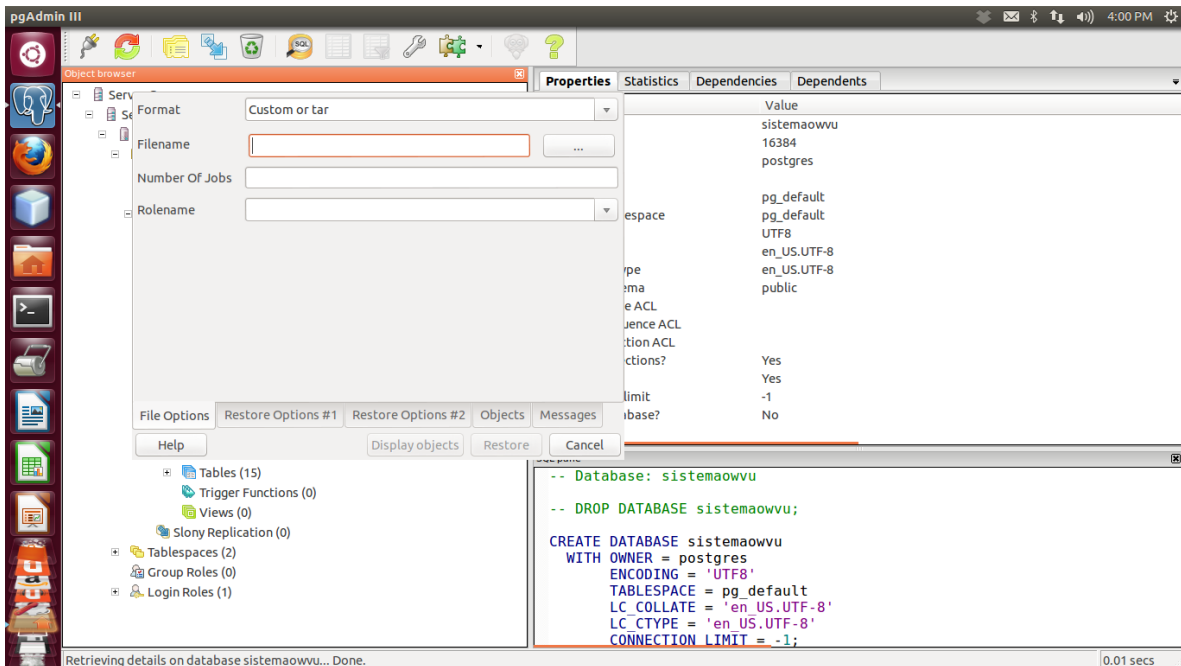
Confirma que la instalación terminó adecuadamente tecleando: “**psql --version**”



2. Para mejor manejo del motor instalamos PGAdmin, para ello tecleamos en una terminal “sudo apt-get install pgadmin3”:



3. Importamos el esquema de la base de datos, para hacer esto abrimos el pgadmin y desde la conexión local, seleccionamos la opción de nueva base de datos, le damos click derecho sobre la nueva base, y seleccionamos la propiedad “restore”, cargamos el archivo del esquema de la base de datos, lo seleccionamos y queda cargado.



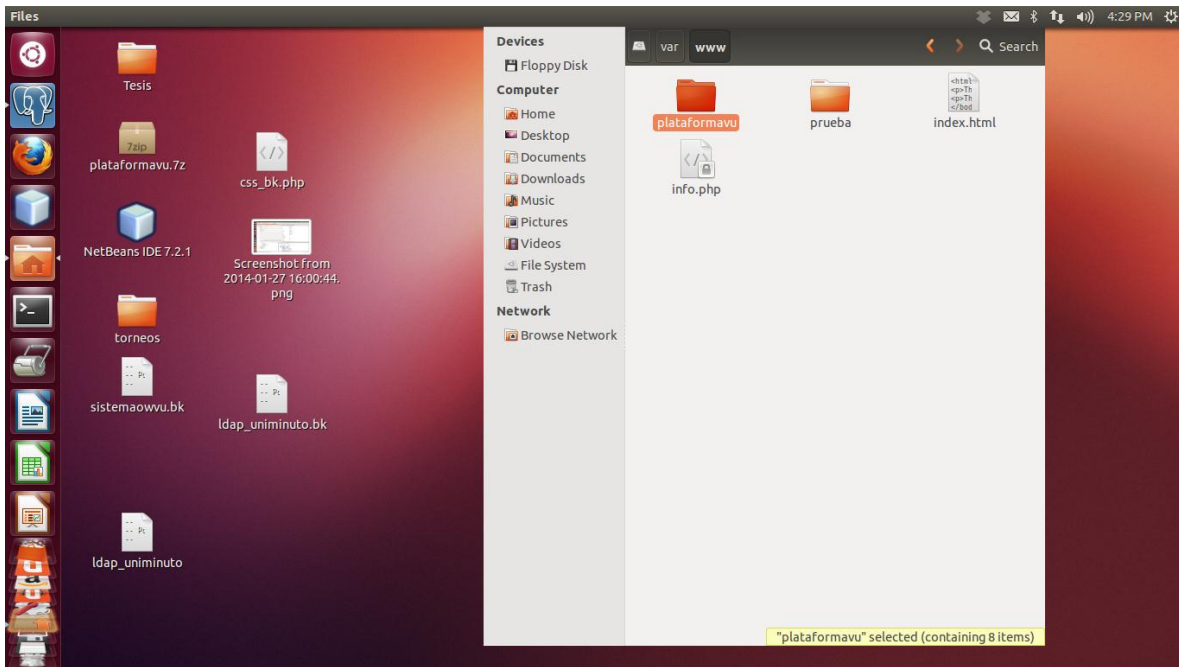
4. Procedemos a instalar el servidor web que en este caso será apache, para instalarlo corremos por terminal el comando “sudo apt-get install apache2”.

```
miguel@miguel-VirtualBox: ~
miguel@miguel-VirtualBox:~$ sudo apt-get apache2
[sudo] password for miguel:
Sorry, try again.
[sudo] password for miguel:
E: Operación inválida: apache2
miguel@miguel-VirtualBox:~$ sudo apt-get install apache2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  apache2-mpm-worker apache2-utils apache2.2-bin apache2.2-common libapr1
  libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap
Paquetes sugeridos:
  apache2-doc apache2-suexec apache2-suexec-custom
Se instalarán los siguientes paquetes NUEVOS:
  apache2 apache2-mpm-worker apache2-utils apache2.2-bin apache2.2-common
  libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap
0 actualizados, 9 se instalarán, 0 para eliminar y 326 no actualizados.
Necesito descargar 3.504 kB de archivos.
Se utilizarán 9.251 kB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]? 
```

5. Instalaremos en el servidor PHP, para realizar esto debemos correr por terminal, el siguiente comando “sudo apt-get install php5 libapache2-mod-php5”.

```
miguel@miguel-VirtualBox: ~
miguel@miguel-VirtualBox:~$ sudo apt-get install php5 libapache2-mod-php5
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes extras:
  apache2-mpm-prefork php5-cli php5-common
Paquetes sugeridos:
  php-pear
Los siguientes paquetes se ELIMINARÁN:
  apache2-mpm-worker
Se instalarán los siguientes paquetes NUEVOS:
  apache2-mpm-prefork libapache2-mod-php5 php5 php5-cli php5-common
0 actualizados, 5 se instalarán, 1 para eliminar y 326 no actualizados.
Necesito descargar 5.798 kB de archivos.
Se utilizarán 17,8 MB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]? 
```

6. Por ultimo copiamos el sistema VU a la carpeta del servidor web para ser desplegado, la dirección de este directorio es `/var/www/`



7. Desde el navegador ingresamos a la URL : `http://localhost/plataformavu`

Manuales



Manual Usuarios

Miguel Angel Camargo Campos y Fabian
Andres Benavides

Sistema VU para la administración de torneos Internos de la corporación
Universitaria Minuto de Dios para la sede principal.

Corporación Universitaria Minuto de Dios Bogotá D.C. – Colombia

Manual Usuarios

Manejo del Sistema VU,
para la administración de torneos internos
de la corporación Universitaria Minuto de Dios
para la sede principal.

Corporación Universitaria Minuto de Dios

Bogotá D.C. – Colombia

Tabla de Contenido

1. INTRODUCCION..... 2

2.Manual del administrador..... 3

3.Manual del usuario 5

1. INTRODUCCION

Este manual tiene como objetivo ser una guía en el uso de los módulos del sistema VU. En su contenido se podrá observar los aspectos más esenciales para poder utilizar de una manera muy fácil dicho sistema.

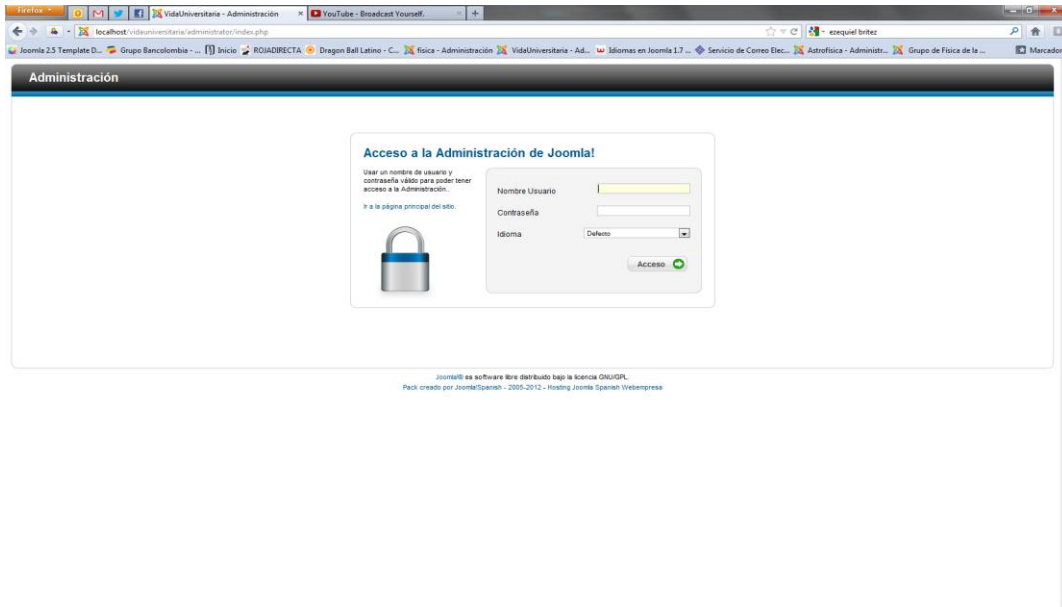
En este manual se encontrara dividido por la sección del usuario y por la sección del administrador.

A continuación se realizará una explicación más profunda del funcionamiento del prototipo, observando la interacción de sus componentes claves.

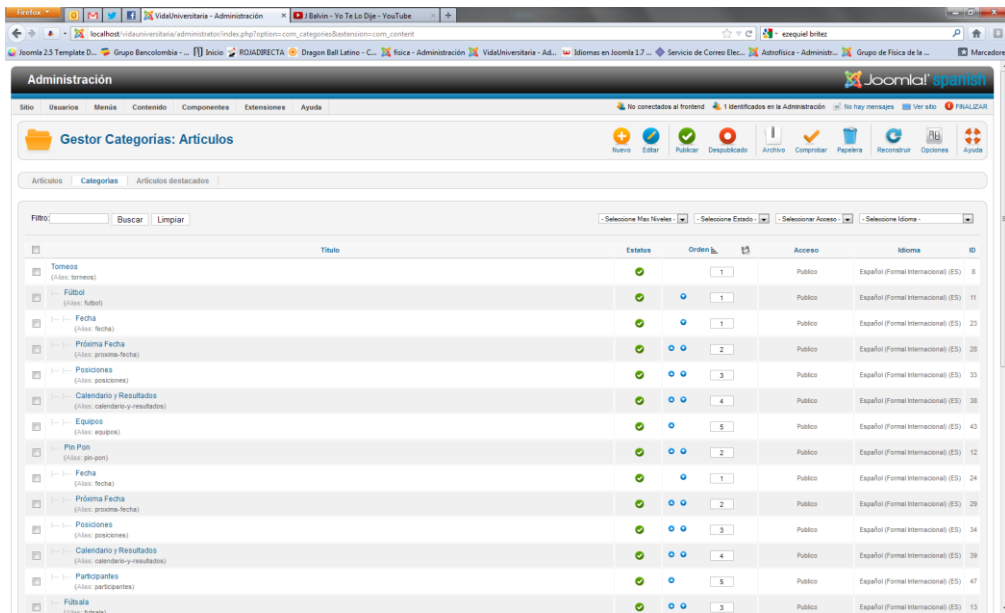
2. Manual del administrador

El administrador cuenta con la página web desde donde puede administrar los torneos, cursos y noticias que los estudiantes ven en la plataforma.

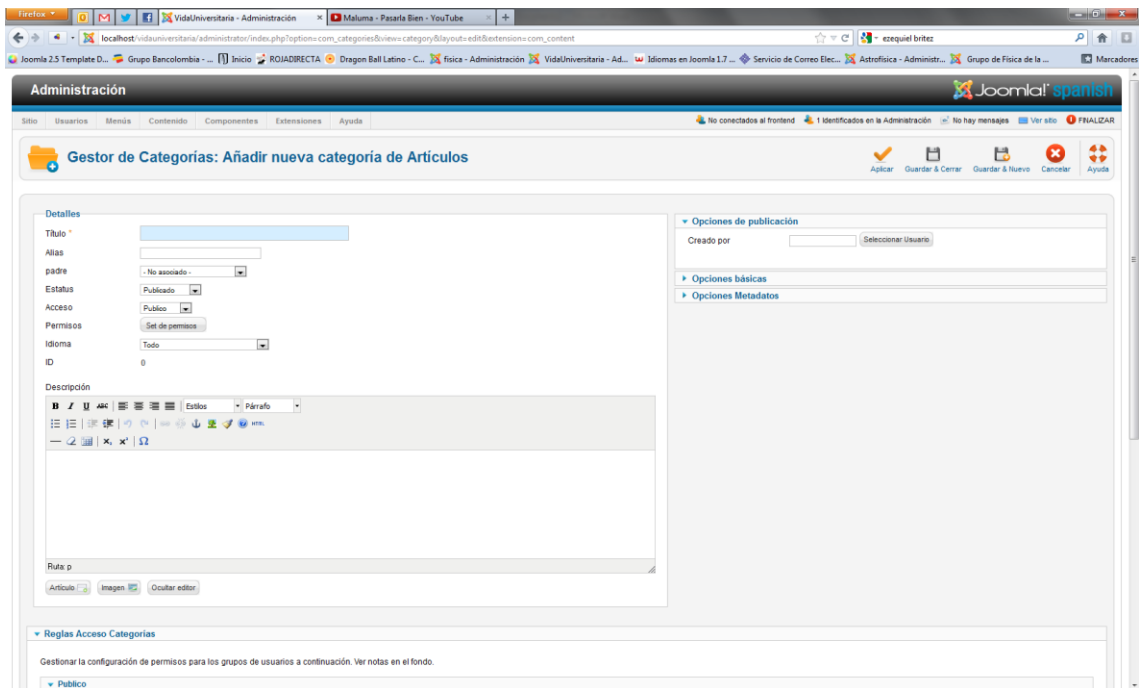
Lo primero que debe hacer el administrador del sitio es autenticarse como administrador con su usuario y contraseña correspondiente.



Al ingresar a la página como administrador, se pueden ver las secciones que se muestra en la página, puede seleccionar la que necesite para configurar lo que sea necesario y así ofrecer un mejor servicio a los estudiantes



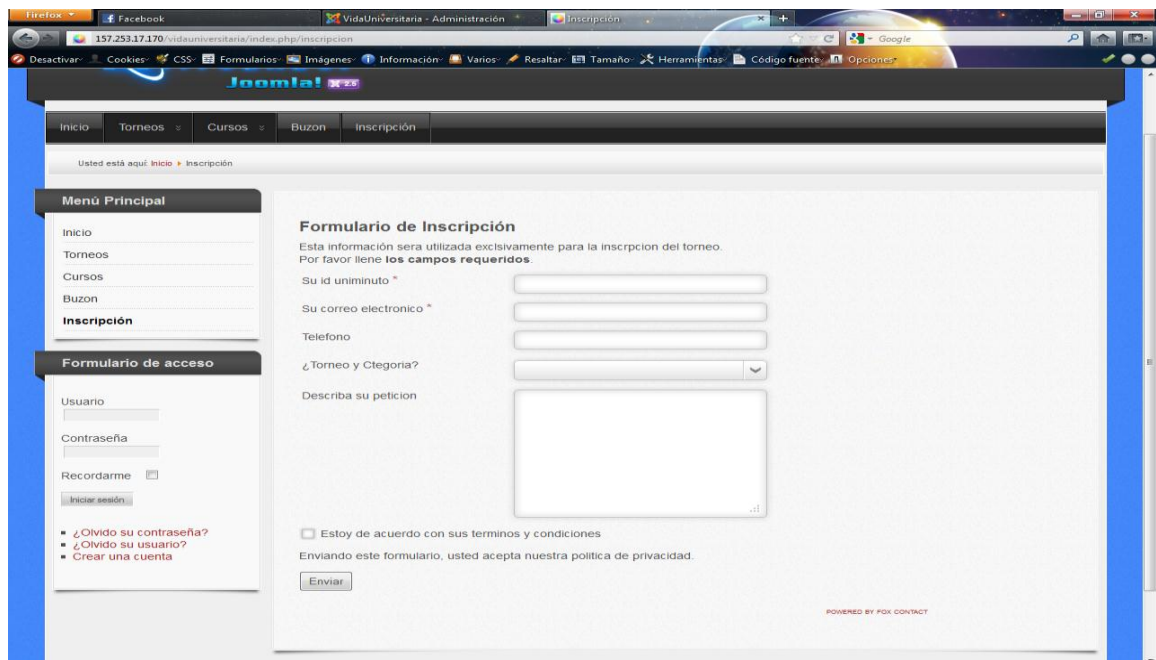
Para agregar una categoría ya sea para crear un torneo un curso el administrador cuenta con un menú donde puede configurar la sección ya sea el nombre o si pertenece a otra sección de la pagina, también puede aplicar los cambios antes de guardar, guardar y terminar el proceso, guardar y crear una nueva sección o cancelar el proceso que está haciendo



3. Manual del usuario

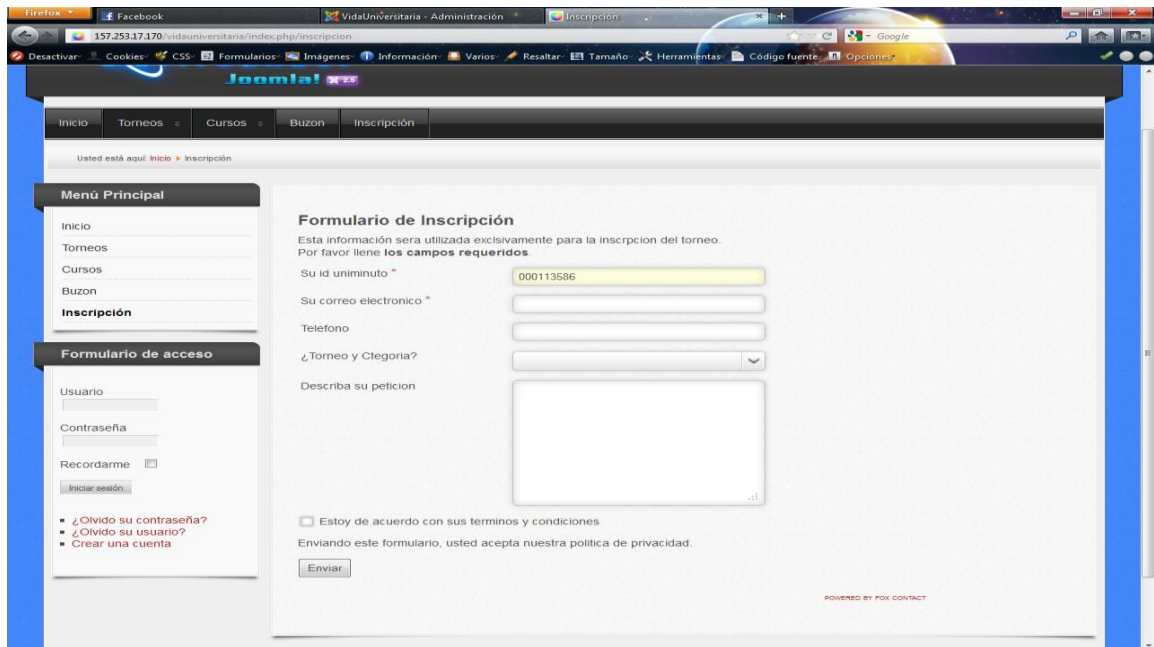
Para inscribir un equipo o persona a un torneo, primer paso es ingresar a la plataforma VU y entrar al ítem del menú “Inscripción”.

Si es un equipo el capitán del equipo se encarga de la inscripción de todos los jugadores.

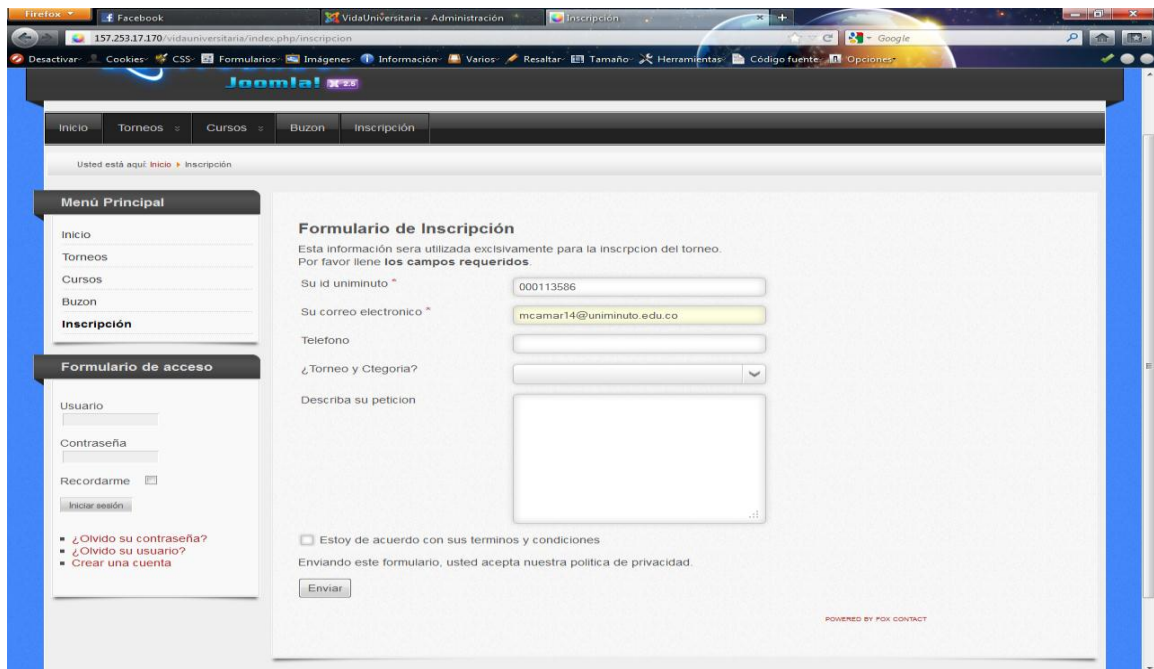


The screenshot shows a Joomla! website interface. At the top, there is a navigation menu with items: Inicio, Torneos, Cursos, Buzon, and Inscripción. Below the menu, there is a breadcrumb trail: "Usted está aquí: Inicio > Inscripción". On the left side, there is a "Menú Principal" with a list of items: Inicio, Torneos, Cursos, Buzon, and Inscripción (highlighted). Below the menu, there is a "Formulario de acceso" with fields for "Usuario" and "Contraseña", a "Recordarme" checkbox, and an "Iniciar sesión" button. There are also links for "¿Olvido su contraseña?", "¿Olvido su usuario?", and "Crear una cuenta". The main content area is titled "Formulario de Inscripción" and contains the following text: "Esta información sera utilizada exclusivamente para la inscripcion del torneo. Por favor llene los campos requeridos." The form includes fields for "Su id uniminuto *", "Su correo electronico *", "Telefono", and a dropdown menu for "¿Torneo y Ctegoria?". There is also a large text area for "Describa su peticion". At the bottom of the form, there is a checkbox for "Estoy de acuerdo con sus terminos y condiciones" and a note: "Enviando este formulario, usted acepta nuestra politica de privacidad." An "Enviar" button is located at the bottom left of the form. The footer of the page says "POWERED BY FOX CONTACT".

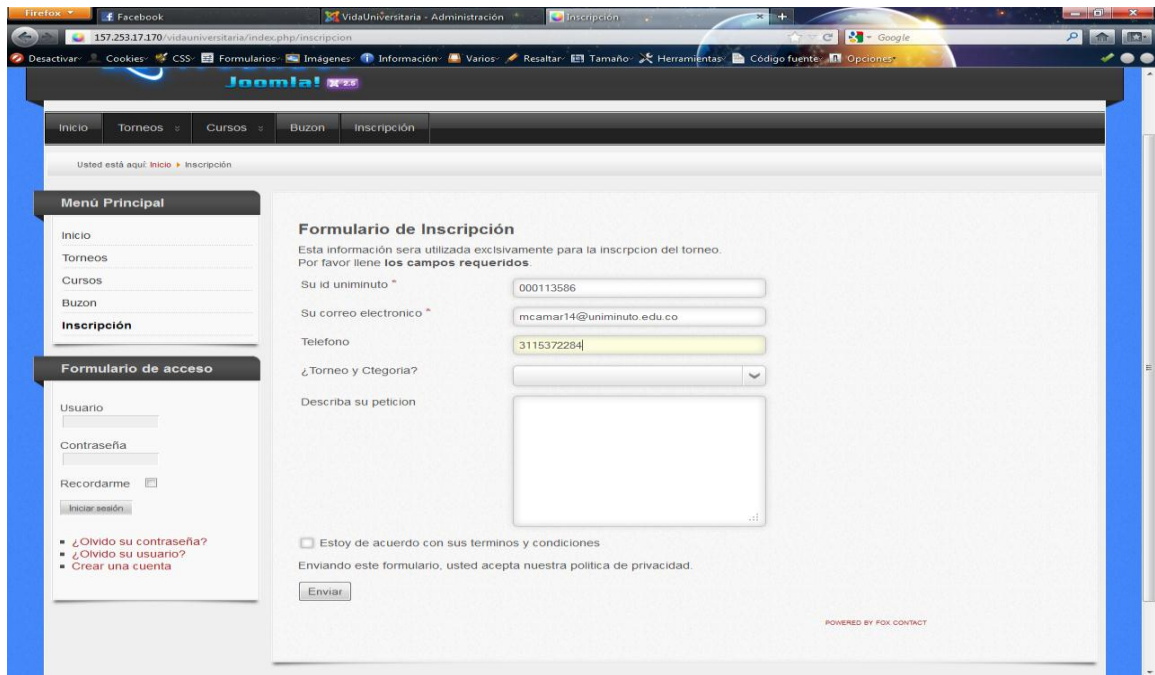
Lo primero que debe hacer el usuario es digitar su ID



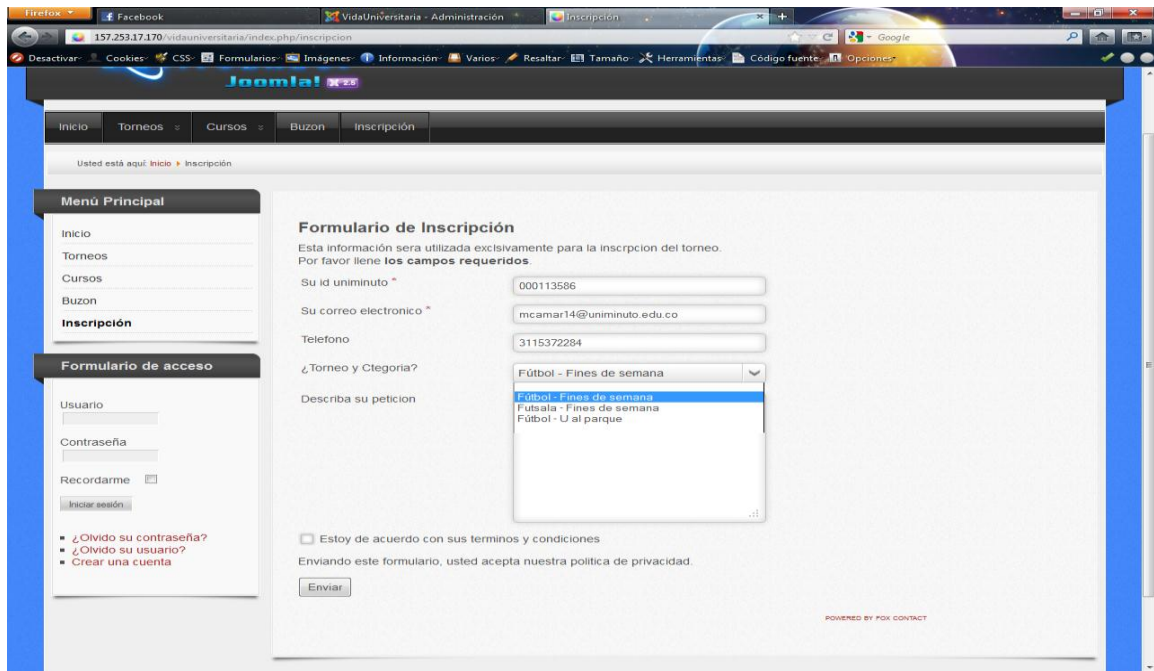
Después de que el usuario ingresa su ID se pide su correo electrónico



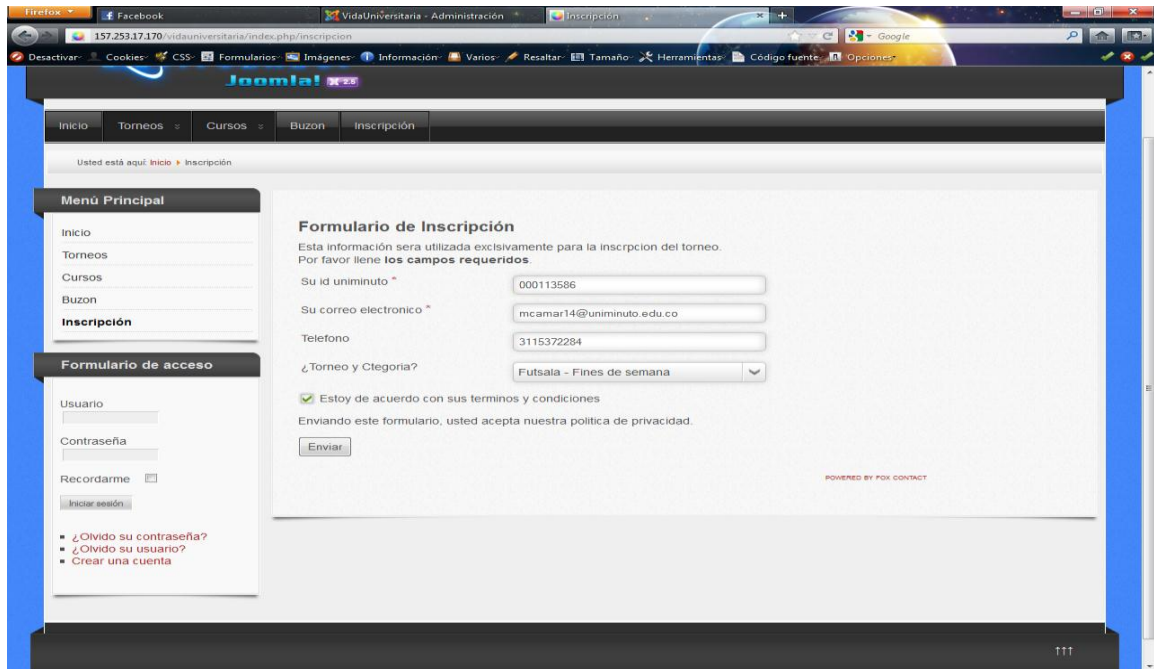
También es necesario que el usuario digite un número de teléfono donde se le pueda contactar



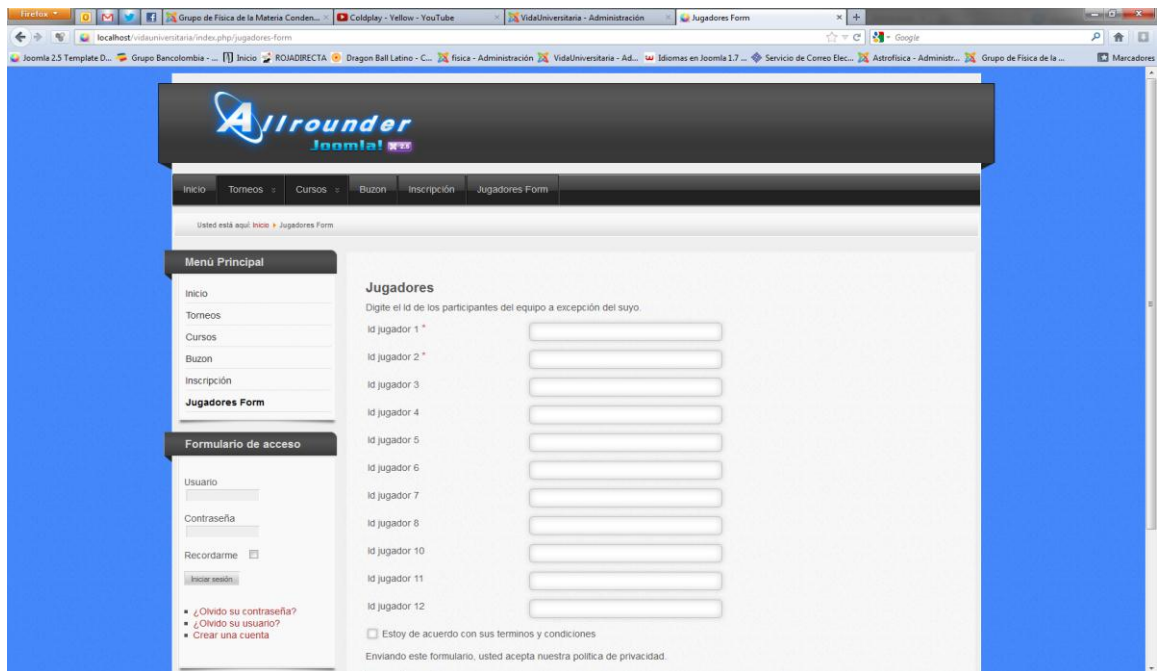
El usuario debe seleccionar el torneo y la categoría en la que desea inscribir su equipo



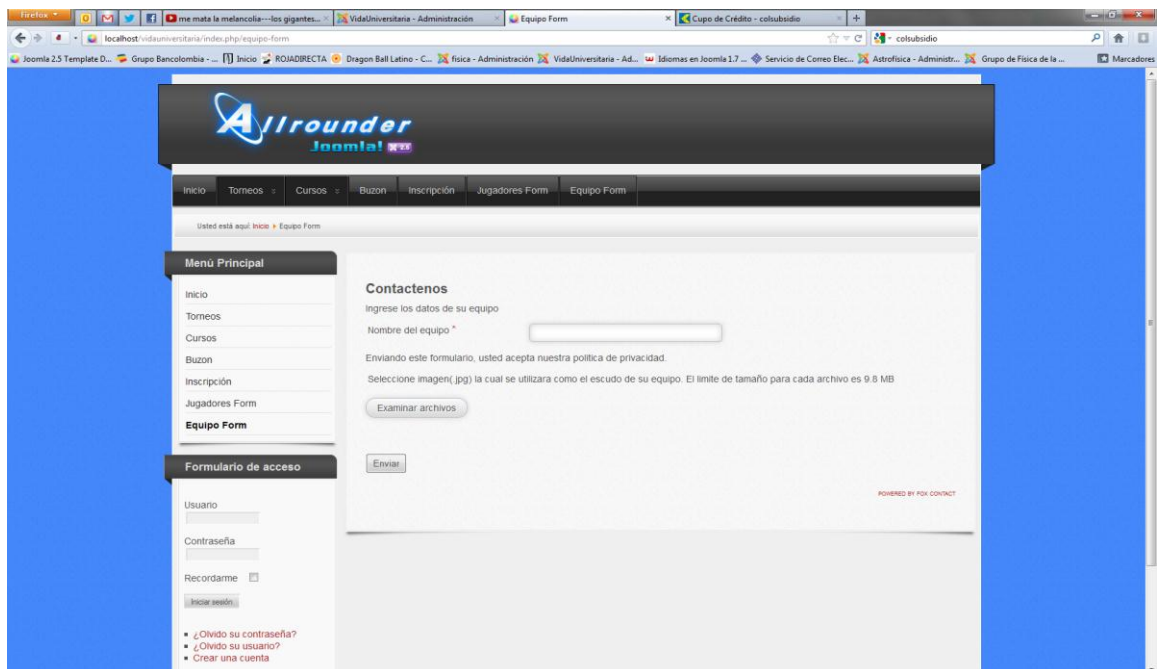
Por último el usuario acepta los términos y condiciones del torneo



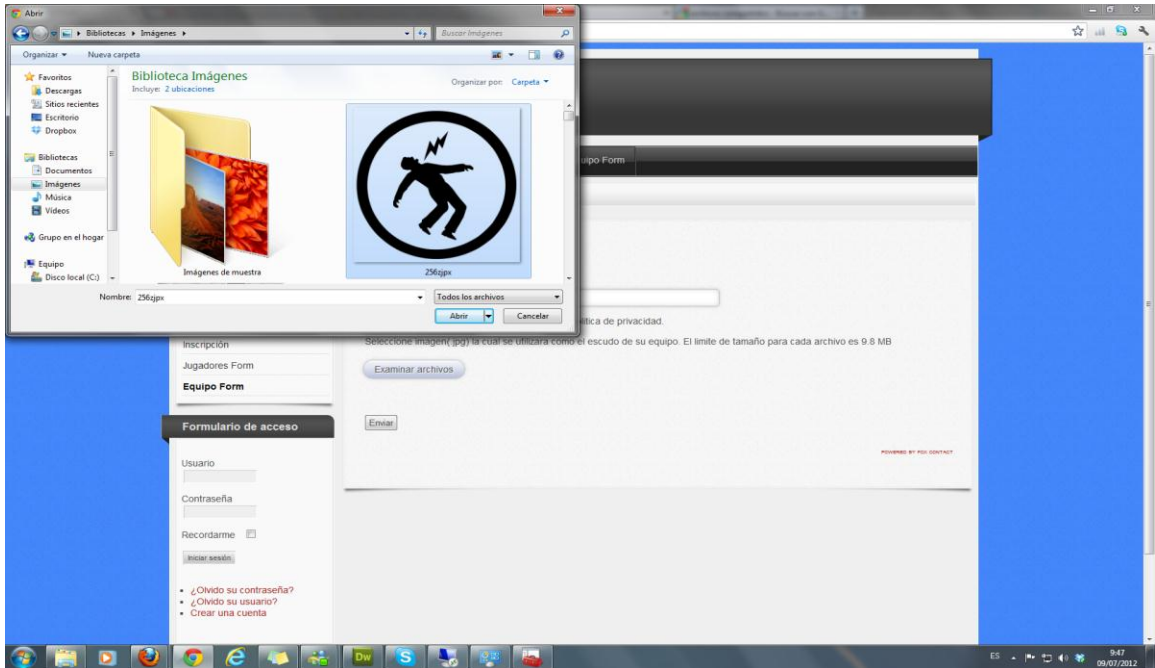
Después que el estudiante ingresa sus datos son solicitados todos los id de los miembros del equipo que esta inscribiendo excepto el de él.



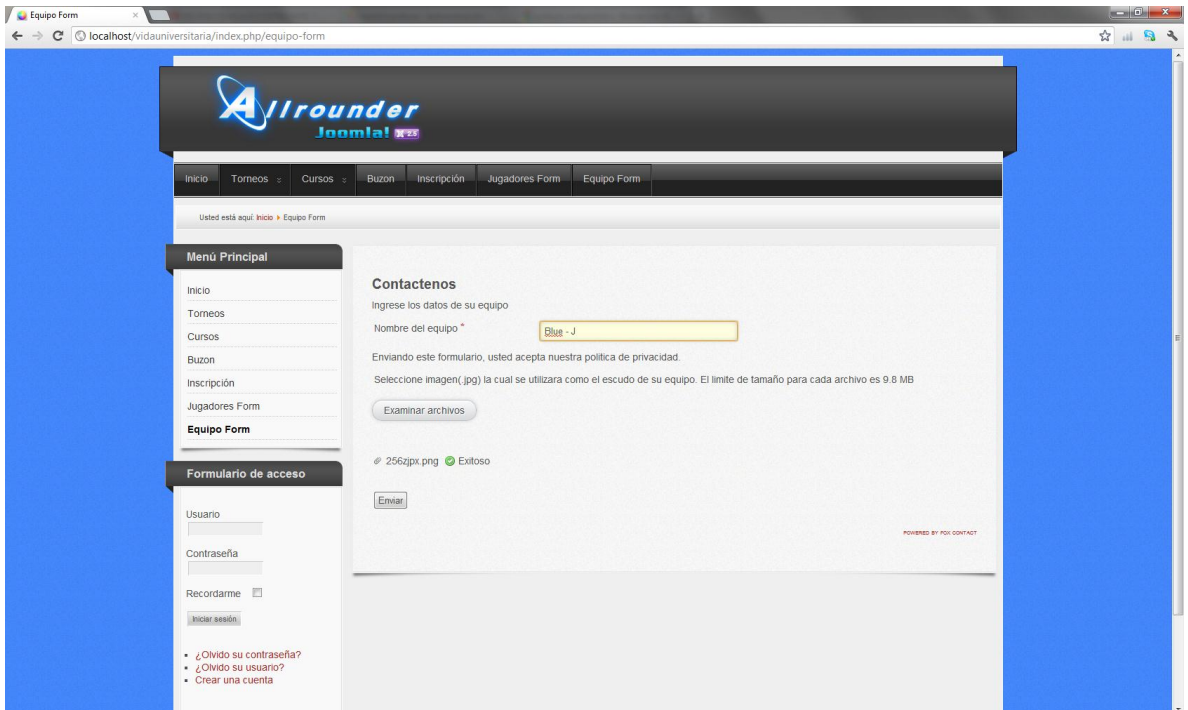
Después de inscribir los jugadores el estudiante selecciona el nombre del equipo y una imagen para utilizarla como escudo del equipo



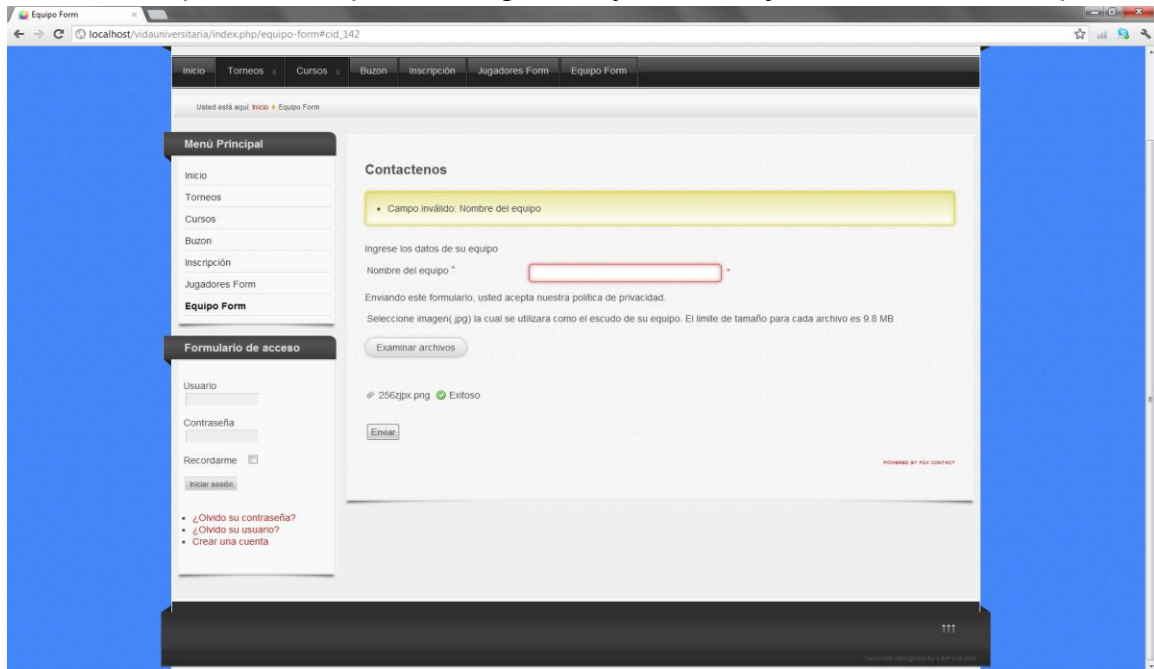
El estudiante debe seleccionar una imagen cuyo formato sea .JPG para que sea el escudo de su equipo.



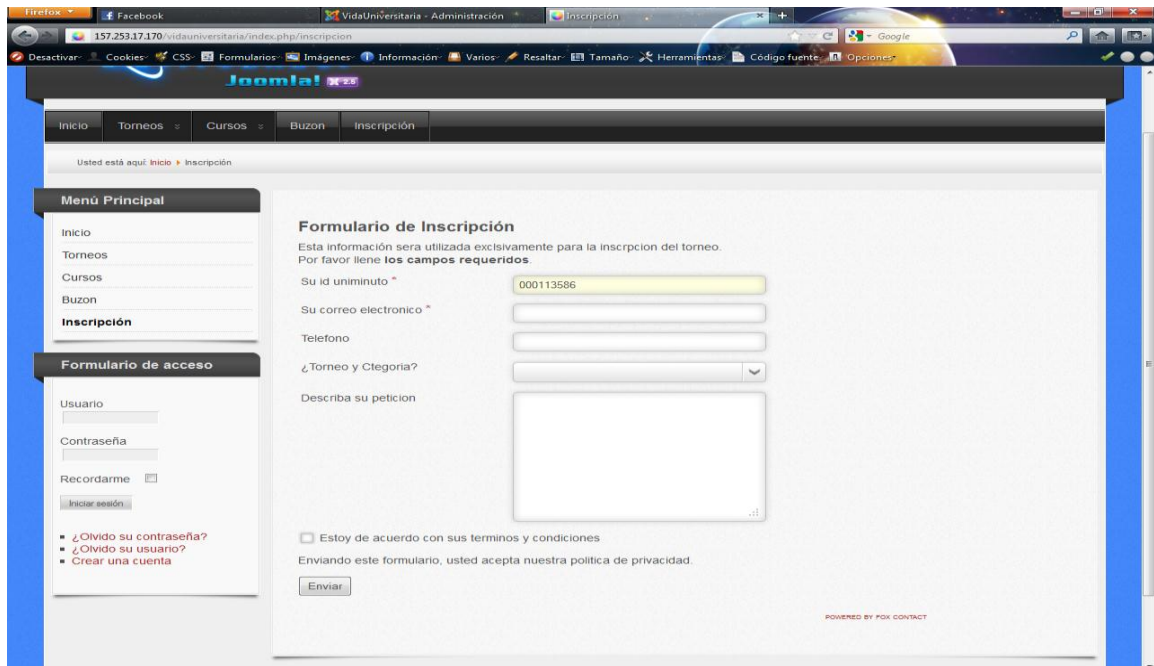
Por último el estudiante envía todos sus datos para el registro del equipo.



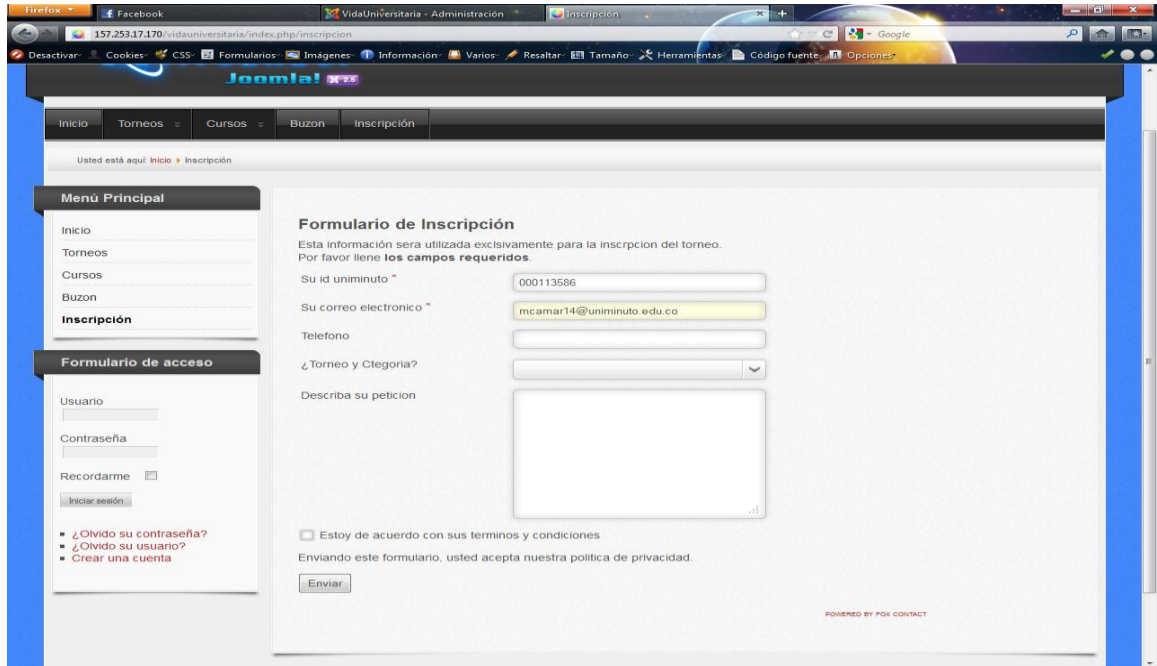
Si no ingresa la información requerida en el formulario aparecerá un mensaje que le advierte que un campo es obligatorio y no lo deja continuar con el proceso.



En la inscripción en los torneos individuales lo que debe hacer el estudiante es diligenciar sus datos empezando por ID



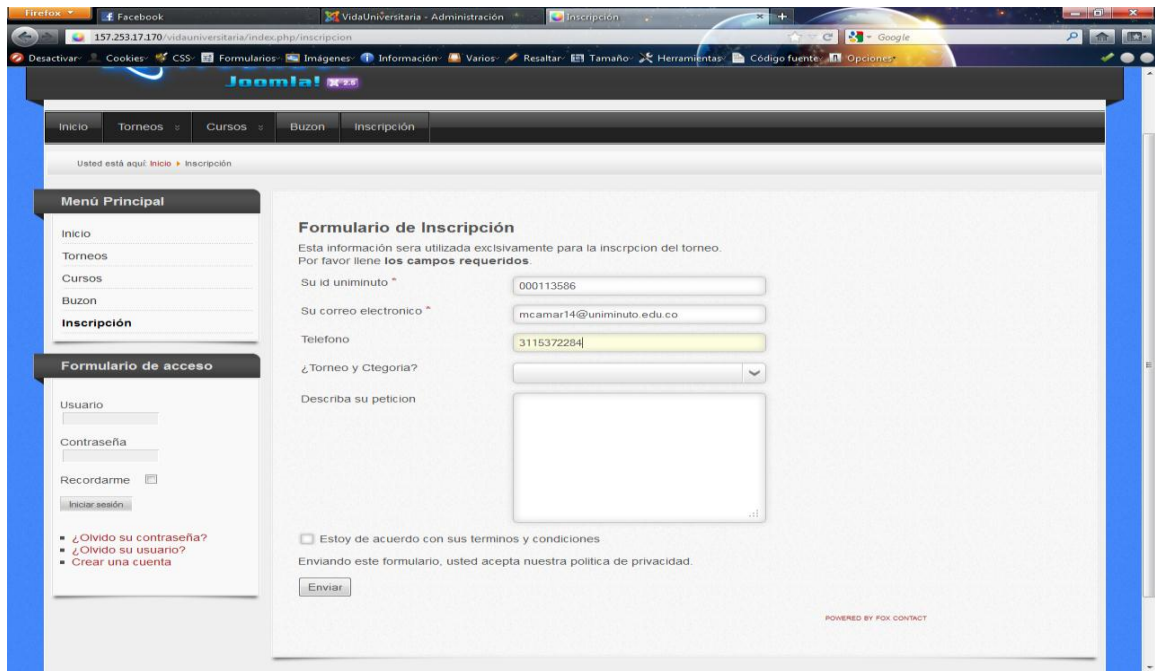
Después de esto el estudiante facilita el correo electrónico donde puede ser contactado



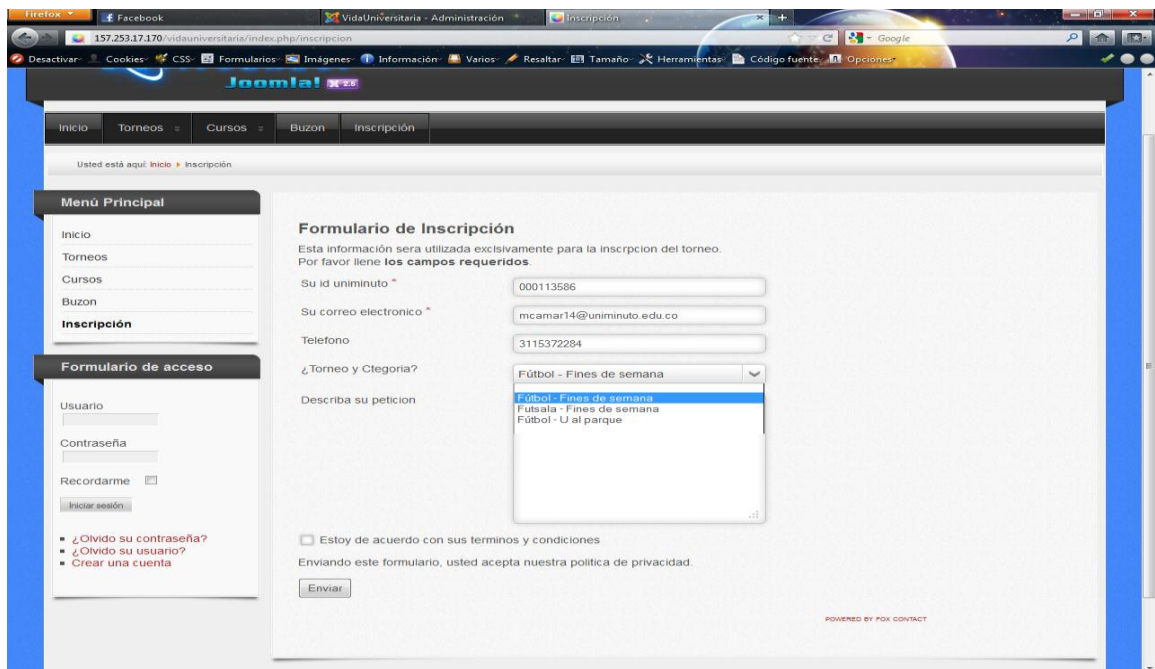
The image shows a screenshot of a Joomla! website's registration page. The browser's address bar shows the URL `157.253.17.170/vidauniversitaria/index.php/inscripcion`. The Joomla! logo is visible at the top. The navigation menu includes 'Inicio', 'Torneos', 'Cursos', 'Buzon', and 'Inscripción'. The main content area is titled 'Formulario de Inscripción' and contains the following fields and options:

- Menú Principal:** A list of site navigation items including 'Inicio', 'Torneos', 'Cursos', 'Buzon', and 'Inscripción'.
- Formulario de acceso:** A section for user login with fields for 'Usuario' and 'Contraseña', a 'Recordarme' checkbox, and an 'Iniciar sesión' button. It also includes links for '¿Olvido su contraseña?', '¿Olvido su usuario?', and 'Crear una cuenta'.
- Formulario de Inscripción:** A registration form with the following elements:
 - Introductory text: 'Esta información sera utilizada exclusivamente para la inscripcion del torneo. Por favor llene los campos requeridos'.
 - Fields: 'Su id uniminuto' (with value '000113586'), 'Su correo electronico' (with value 'mcamar14@uniminuto.edu.co'), 'Telefono', and '¿Torneo y Ctegoria?' (a dropdown menu).
 - A large text area for 'Describa su peticion'.
 - A checkbox for 'Estoy de acuerdo con sus terminos y condiciones'.
 - A note: 'Enviando este formulario, usted acepta nuestra politica de privacidad'.
 - An 'Enviar' button.

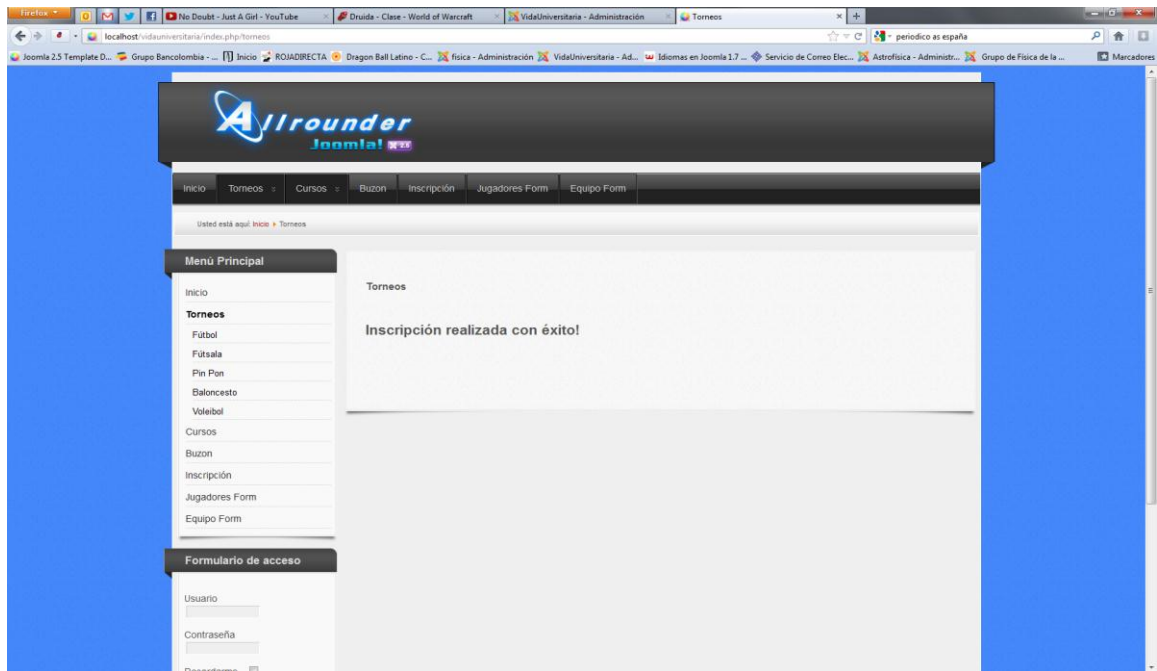
También se solicita un número telefónico para contactar al estudiante



Para finalizar el estudiante selecciona el torneo y la categoría en la que quiere participar



Al finalizar la inscripción saldrá un mensaje que indica que el proceso se realizo con éxito



Modificar usuario

Después de haber registrado el equipo, el estudiante que registro el equipo puede agregar o eliminar más jugadores dependiendo de las reglas y número de participantes que se apliquen para cada torneo. Lo que debe hacer el estudiante es ingresar a la plataforma con su ID y su contraseña para hacer los cambios en su equipo

