



**UNIVERSIDAD SANTO TOMÁS**  
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA  
T U N J A

**Implementación de plataforma web de reservas de espacios deportivos, usando  
desarrollo basado en pruebas TDD.**

Juan Pablo Acevedo Torres

**Trabajo de grado para obtener el título de ingeniero de sistemas**

**DIRECTOR**

Luis Fernando Castellanos Guarín

Ingeniero de sistemas

**UNIVERSIDAD SANTO TOMÁS SECCIONAL TUNJA  
DIVISION DE INGENIERIA Y ARQUITECTURA  
FACULTAD DE INGENIERIA DE SISTEMAS  
2019**

## Resumen

En el ámbito de la gestión de espacios deportivos en la universidad Santo Tomas seccional Tunja, existe el problema de hacer una rápida gestión de reservas y consultas sobre espacios deportivos ya académicos libres. El objetivo de este trabajo es solucionar estos inconvenientes en cuanto a la gestión de espacios y reducir los contratiempos de estos, usando un desarrollo ágil basado en pruebas o TDD, que garantizara tener un software de calidad y con errores mínimos a diferencia de los métodos de desarrollo tradicionales.

**Palabras clave:** desarrollo basado en pruebas, TDD, Espacios deportivos, gestión de espacios, gimnasio.

## Abstract

*In the field of sports space management at the Santo Tomas Tunja sectional university, there is the problem of making a quick management of reservations and inquiries about free sports and academic spaces. The objective of this work is to solve these inconveniences in terms of space management and reduce the setbacks of these, using an agile development based on tests or TDD, which will guarantee to have a quality software and with minimal errors unlike the methods of Traditional development*

**Keywords:** test based development, TDD, sports spaces, space management, gym.

# CONTENIDO

## Contenido

<b><u>1. INTRODUCCIÓN</u></b> .....	<b>5</b>
<b><u>2. PLANTEAMIENTO DEL PROBLEMA</u></b> .....	<b>6</b>
2.1. ÁRBOL DE PROBLEMAS.....	6
2.2. DEFINICIÓN DEL PROBLEMA.....	7
2.3. IDENTIFICACIÓN ESFUERZOS ANTERIORES.....	7
2.4. JUSTIFICACIÓN.....	7
<b><u>3. OBJETIVOS</u></b> .....	<b>9</b>
3.1. OBJETIVO GENERAL.....	9
3.2. OBJETIVOS ESPECÍFICOS .....	9
<b><u>4. ALCANCE Y LIMITACIONES</u></b> .....	<b>10</b>
4.1. ALCANCE. ....	10
4.2. LIMITACIONES .....	10
<b><u>5. MARCO CONCEPTUAL</u></b> .....	<b>12</b>
5.1. METODOLOGÍAS.....	13
<b><u>6. METODOLOGIA</u></b> .....	<b>16</b>
6.1. EXPLORACIÓN PREVIA: .....	16
6.2. ANÁLISIS:.....	16
6.3. DISEÑO DE LA PROPUESTA: .....	16
6.4. DISEÑO DE ARQUITECTURA DE LA APLICACIÓN: .....	16
6.5. IMPLEMENTACIÓN .....	16
<b><u>7. LÍNEA DE INVESTIGACIÓN</u></b> .....	<b>17</b>
<b><u>8. DESARROLLO DEL PROYECTO</u></b> .....	<b>18</b>
8.1. EXPLORACIÓN PREVIA: .....	18

<b>8.2. ANÁLISIS:</b> .....	<b>20</b>
<b>8.3. REQUISITOS :</b> .....	<b>22</b>
8.3.1. VISTA ADMINISTRADOR.....	22
8.3.2. VISTA DE INSTRUCTOR.....	23
8.3.3. VISTA DE USUARIO FINAL .....	23
<b><u>9. ARQUITECTURA Y DISEÑO DE LA PLATAFORMA</u></b> .....	<b>24</b>
<b>9.1. BASE DE DATOS</b> .....	<b>24</b>
<b>9.2. DICCIONARIO DE DATOS</b> .....	<b>25</b>
<b>9.3. DIAGRAMAS DE SECUENCIA.</b> .....	<b>30</b>
<b><u>10. ARQUITECTURA DE LA PLATAFORMA</u></b> .....	<b>31</b>
<b>10.1. DIAGRAMA DE CASOS DE USO DEL SISTEMA</b> .....	<b>31</b>
<b>10.2. DIAGRAMA DE COMPONENTES DEL SISTEMA</b> .....	<b>31</b>
<b>10.3. DISEÑO DE LA PLATAFORMA</b> .....	<b>32</b>
10.3.1. VISTA INDEX:.....	32
10.3.2. VISTA LOGIN: .....	32
10.3.3. VISTA ADMINISTRADOR:.....	33
10.3.4. VISTA USUARIOS: .....	33
VISTA USUARIO .....	34
10.3.5. VISTA RESERVA .....	35
10.3.6. FASE DE ITERACIONES.....	35
10.3.7. FASE DE PRODUCCIÓN.....	35
10.3.8. FASE DE MANTENIMIENTO.....	36
<b><u>11. RECURSOS</u></b> .....	<b>37</b>
<b>11.1. PRESUPUESTO</b> .....	<b>37</b>
<b>11.2. RECURSO HUMANO</b> .....	<b>37</b>
<b>11.3. SOFTWARE EMPLEADO</b> .....	<b>37</b>
<b><u>12. BIBLIOGRAFIA</u></b> .....	<b>40</b>

## **1. Introducción**

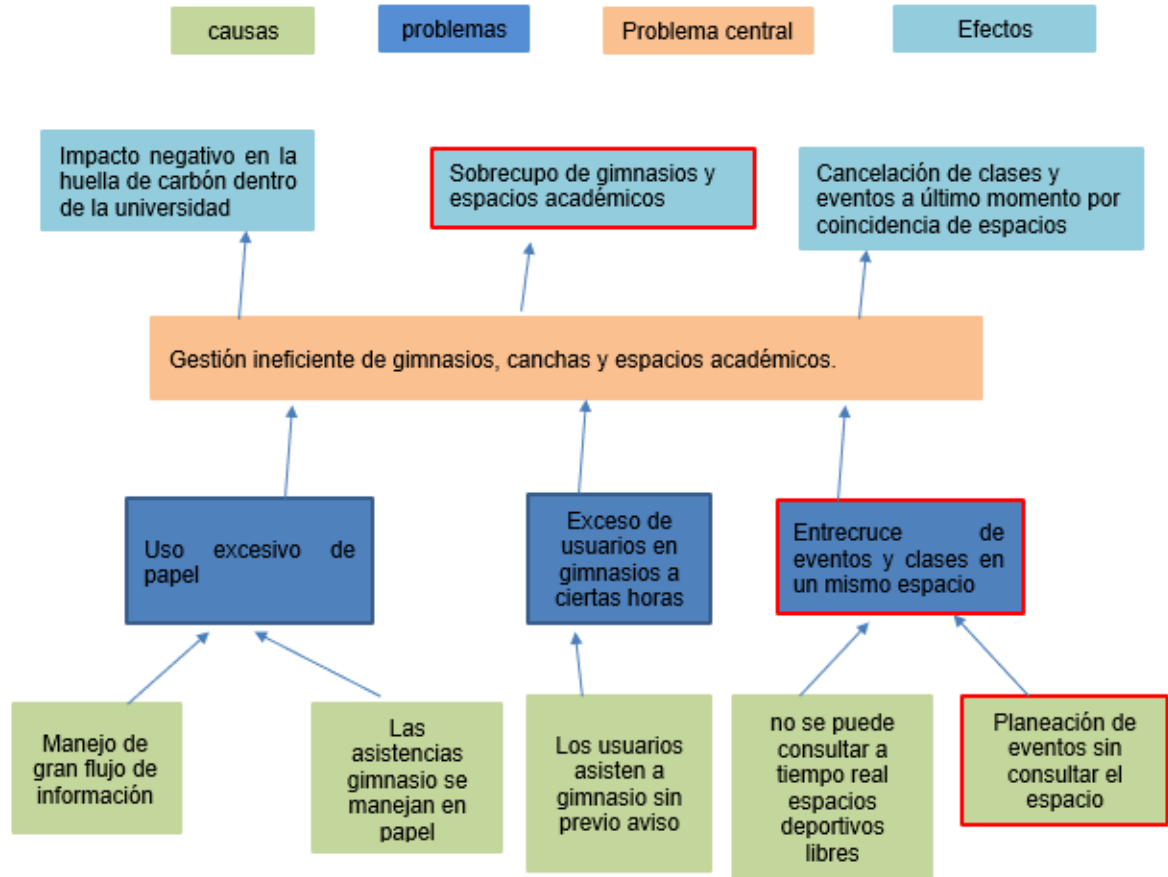
En el presente anteproyecto, se propone llevar a cabo el desarrollo e implementación de una plataforma web que permita administrar la asignación de espacios deportivos y el control de asistencias a estos en la universidad Santo Tomas seccional Tunja.

Con la implementación de este sistema en la universidad, se busca mejorar el acceso a los gimnasios y demás espacios deportivos de la universidad para los usuarios de la universidad (estudiantes, profesores, administrativos y egresados), quienes a través de una cuenta de usuario en la plataforma web, podrá hacer consulta de clases libres disponibles e inscribirse a estas ya sea mediante la oficina de bienestar o mediante la plataforma si el evento no tiene restricción de inscripción.

Es importante resaltar que la practica deportiva es importante para el desarrollo integral de los miembros de la universidad, enfocado principalmente en el uso de los gimnasios de la universidad, por lo que es importante un control de estos para que no halla sobre cupos y todos puedan hacer uso de estos espacios.

## 2. PLANTEAMIENTO DEL PROBLEMA

### 2.1. Árbol de problemas



## **2.2. Definición del Problema.**

Durante el transcurso de 2019 la información de cada clase de gimnasio, se encuentra almacenada en planillas Excel y listas de papel, el sistema que se maneja actualmente en el departamento de bienestar para el manejo de los gimnasios y otros espacios, hace que en se gaste aproximadamente una resma de papel al mes en impresiones de listas y documentos para solicitar la reserva de estos

Cuando se plantea generar un evento como un campeonato de microfútbol, toma aproximadamente una semana hábil para que el departamento de bienestar pueda asignar una o varias canchas que estén disponibles, debido a que se debe hacer una consulta previa a las listas de clases que se están dictando para evitar un entrecruzamiento de horarios.

El origen de este planteamiento radica en la gran cantidad de información que genera el control de lista de una sola clase y evento, que con el tiempo se va incrementando en forma continua, lo que de alguna manera dificulta el manejo eficiente y oportuno de esta información para la toma de decisiones. Esto conlleva a un aumento relacional de tiempo en lo que respecta al proceso de elaboración de los distintos informes de asistencias y uso de espacios entre otros encomendados.

## **2.3. Identificación Esfuerzos Anteriores**

Con anterioridad a esto se intentó llevar estos registros en plantillas de Excel, y en trabajos de tesis anteriores se plantearon sistemas de reservas, pero con enfoques diferentes, como apartar una cita médica.

## **2.4. Justificación**

El Proyecto de Software viene a ofrecer una alternativa de solución a la permanente necesidad por parte del Departamento de bienestar, de mantener la información de espacios

de forma centralizada, lo que permitirá que se pueda mantener un control en tiempo real de las clases y eventos que se puedan realizar en estos espacios, y que a la vez ésta pueda ser entregada de manera rápida y oportuna, a los usuarios que la soliciten.

Se hace necesario realizar este proyecto para el departamento de bienestar universitario de la universidad Santo Tomas Seccional Tunja con el fin de agilizar la gestión de espacios de gimnasio y espacios académicos para la realización de clases y eventos. Considerando que la creación de un sistema centralizado de manejo de espacios permitirá agilizar los procesos relacionados con la gestión y control de estos en un menor tiempo, menor gasto de recursos y evitando problemas como entrecruzamiento de espacios.

puede ser muy provechoso para el mejoramiento de tiempos en cuanto a la resolución de reservas y generaría simplicidad para el departamento de bienestar universitario como, por ejemplo; al momento de requerir alguna información sobre un espacio para la realización de una clase de Yoga para 14 participantes, el sistema permitirá mostrar que espacios cumplen con el espacio suficiente y el horario disponible.

una vez el sistema web demuestre su funcionalidad en los gimnasios y espacios deportivos, este podrá escalarse a una mayor cobertura, como laboratorios y auditorios entre otros, dejando una fácil adaptabilidad a necesidades futuras.

### 3. OBJETIVOS

#### 3.1. Objetivo General

Mejorar la gestión de espacios deportivos y académicos para la universidad Santo Tomas Tunja mediante la construcción e implementación de una plataforma web para la administración de clases, eventos deportivos y clases dentro de los espacios de esta, usando desarrollo ágil basado en pruebas TDD.

#### 3.2. Objetivos específicos

- Analizar los procesos de registro y seguimiento de la información de los diferentes eventos y clases dentro de la universidad Santo Tomas seccional Tunja.
- Aplicar el “**Método para optimizar el tiempo de la fase desarrollo en proyectos Web con calidad**” ...
- Determinar los requerimientos funcionales y no funcionales.
- Analizar los datos obtenidos y a partir de estos implementar una solución optima
- Diseñar el modelo de la base de datos.
- Diseñar la interfaz gráfica.
- Implementar el modelo de la base de datos y la interfaz gráfica.

## **4. ALCANCE Y LIMITACIONES**

### **4.1. Alcance.**

El presente trabajo comprende el desarrollo de una plataforma web para asignación de espacios deportivos como gimnasios y espacios físicos para el departamento de bienestar universitario de la Universidad Santo Tomas seccional Tunja. La plataforma incluirá un sistema de registro y control de usuarios, actividades o eventos, periodos académicos, y gestión de reservas de gimnasio y espacios deportivos, como también un sistema de interfaz gráfica de calendario para ver los horarios asignados a cada espacio cada espacio. Incluye una base de datos en la cual se almacenará toda la información, permitiendo a los usuarios y administradores adquirir más confianza sobre la veracidad de los datos almacenados y la reducción en tiempos de respuesta de gestión de clases y eventos, permitiendo un seguimiento más detallado a el flujo de información de estos espacios.

### **4.2. Limitaciones**

Este proyecto seguirá las normas establecidas en la universidad y leyes vigentes, si en algún momento estas son modificadas durante o después de la realización del proyecto no serán contempladas tales modificaciones.

Teniendo en cuenta el tiempo límite para el desarrollo no se tendrán la implementación de nuevos requisitos, para esta versión del proyecto, cualquier requisito adicional se tomarán como parte de los requerimientos de una siguiente versión.

Al finalizar la etapa de análisis de requerimientos y diseño con la aprobación total de las especificaciones del sistema, solo se tendrán en cuenta cambios menores como la interfaz, sin contemplar cambios a la estructura de base de datos y backend que conlleven un mayor gasto de recursos para entregar en el lapso de tiempo estipulado inicialmente en el calendario de trabajo.

Los nuevos requisitos que involucren cambios en el sistema o nuevas funcionalidades, serán contemplados para las siguientes versiones.

## 5. MARCO CONCEPTUAL

**Aws:** Amazon Web Services es una colección de servicios de computación en la nube pública que en conjunto forman una plataforma de computación en la nube, ofrecidas a través de Internet por Amazon.com. Es usado en aplicaciones populares como Dropbox, Foursquare, HootSuite.

**PostgreSQL:** PostgreSQL es un sistema de gestión de bases de datos relacional orientado a objetos y de código abierto, publicado bajo la licencia PostgreSQL, similar a la BSD o la MI.

**Php:** acrónimo recursivo en inglés de PHP: Hypertext Preprocessor, es un lenguaje de programación de propósito general de código del lado del servidor originalmente diseñado para el preprocesado de texto plano en UTF-8.

**Nginx:** es un servidor web/proxy inverso ligero de alto rendimiento y un proxy para protocolos de correo electrónico. Es software libre y de código abierto, licenciado bajo la Licencia BSD simplificada; también existe una versión comercial distribuida bajo el nombre de Nginx Plus.

**Materialize:** es un marco CSS moderno y receptivo basado en Material Design by Google. es un lenguaje de diseño que combina los principios clásicos del diseño exitoso junto con la innovación y la tecnología. El objetivo de Google es desarrollar un sistema de diseño que permita una experiencia de usuario unificada en todos sus productos en cualquier plataforma.

**Bootstrap:** es un kit de herramientas de código abierto para desarrollar con HTML, CSS y JS. Realice rápidamente prototipos de sus ideas o cree su aplicación completa con nuestras variables y mixins Sass, sistema de cuadrícula sensible, componentes precompilados extensos y complementos potentes creados en jQuery.

**jQuery:** es una biblioteca multiplataforma de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el

árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web.

**Fullcalendar:** es un plugin de jQuery escrito por Adam Shaw que presenta datos en forma de calendario o agenda. Puede cargar los eventos directo por AJAX

**Docker:** es un proyecto de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software, proporcionando una capa adicional de abstracción y automatización de virtualización de aplicaciones en múltiples sistemas operativos.

**Laravel:** Laravel es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP 5 y PHP 7. Su filosofía es desarrollar código PHP de forma elegante y simple, evitando el "código espagueti". Fue creado en 2011 y tiene una gran influencia de frameworks como Ruby on Rails, Sinatra y ASP.NET MVC

## 5.1. Metodologías

- **eXtreme Programming :**

(programación extrema) también llamado XP, es una metodología que tiene su origen en 1996, de la mano de Kent Beck, Ward Cunningham y Ron Jeffries. A diferencia de Scrum, XP propone solo un conjunto de prácticas técnicas, que, aplicadas de manera simultánea, pretenden enfatizar los efectos positivos de en un proyecto de desarrollo de Software.

- **Scrum**

Scrum es una metodología ágil para la gestión de proyectos relacionados con la construcción (desarrollo) de Software basado en un sistema de sprints, Un Sprint tiene dos características fundamentales: a) una duración fija entre 1 a 4 semanas ; b) cada Sprint se ejecuta de forma consecutiva (sin tiempo muerto entre un sprint y otro). El objetivo principal es transformar un conjunto de ítems requeridos por el cliente en un incremento de funcionalidad 100% operativa para el software.

- **Desarrollo Dirigido por Tests (TDD)**

Carlos Blé Jurado se refiere este como una técnica técnica de diseño e implementación de software incluida dentro de la metodología XP. TDD es una

técnica para diseñar software que se centra en tres pilares fundamentales:

- La implementación de las funciones justas que el cliente necesita y no más.
- La minimización del número de defectos que llegan al software en fase de producción.
- La producción de software modular, altamente reutilizable y preparado para el cambio.

En esta metodología Pasamos de pensar en implementar tareas, a pensar en ejemplos los mas certero y reales posibles que eliminen la ambigüedad creada por el lenguaje natural. Hasta ahora estábamos acostumbrados a que las tareas, o los casos de uso, eran las unidades de trabajo más pequeñas sobre las que ponerse a desarrollar código. Con TDD intentamos traducir el caso de uso o tarea en X ejemplos, hasta que el número de ejemplos sea suficiente como para describir la tarea sin lugar a malinterpretaciones de ningún tipo, es decir empezamos a realizar ejemplos cada vez más complejos hasta que no exista la posibilidad de error.

- **Prototipado**

El modelo de prototipos permite que todo el sistema, o algunos de sus partes, se construyan rápidamente para comprender con facilidad y aclarar ciertos aspectos en los que se aseguren que el desarrollador, el usuario, el cliente estén de acuerdo en lo que se necesita así como también la solución que se propone para dicha necesidad y de esta forma minimizar el riesgo y la incertidumbre en el desarrollo, este modelo se encarga del desarrollo de diseños para que estos sean analizados y prescindir de ellos a medida que se adhieran nuevas especificaciones, es ideal para medir el alcance del producto, pero no se asegura su uso real.

Este modelo principalmente se lo aplica cuando un cliente define un conjunto de objetivos generales para el software a desarrollarse sin delimitar detalladamente los requisitos de entrada procesamiento y salida, es decir cuando el responsable no está seguro de la eficacia de un algoritmo, de la adaptabilidad del sistema o de la forma en que interactúa el hombre y la máquina. Este modelo se encarga principalmente de ayudar al ingeniero de sistemas y al

cliente a entender de mejor manera cuál será el resultado de la construcción cuando los requisitos estén satisfechos.

El paradigma de construcción de prototipos tiene tres pasos:

- Escuchar al cliente. Recolección de requisitos. Se encuentran y definen los objetivos globales, se identifican los requisitos conocidos y las áreas donde es obligatorio más definición.
- Construir y revisar la maqueta (prototipo).
- El cliente prueba la maqueta (prototipo) y lo utiliza para refinar los requisitos del software.

Este modelo es útil cuando:

El cliente no identifica los requisitos detallados.

El responsable del desarrollo no está seguro de la eficiencia de un algoritmo, sistema operativo o de la interface hombre-máquina.

(Peeleger, 2002)

## **6. METODOLOGIA**

En la propuesta del presente trabajo de grado se planteó trabajar de la siguiente forma:

### **6.1. Exploración previa:**

se recopilará información a partir de entrevistas y documentación proporcionada por bienestar universitario, y consulta de aplicaciones ya existentes relacionadas con los requisitos que vayan surgiendo para la aplicación.

### **6.2. Análisis:**

Una vez recopilada la información, se hará una lista de requerimientos y se definirán los módulos necesarios para el funcionamiento del sistema.

### **6.3. Diseño de la propuesta:**

A partir del análisis previo y los requisitos de la plataforma, se pasará a definir los módulos y las vistas de usuarios del sistema.

### **6.4. Diseño de arquitectura de la aplicación:**

Se usará una arquitectura que se ajuste a las funcionalidades propuestas en el punto anterior, se creará un módulo básico que contenga todas las funciones para certificar que las pruebas cumplan con los requisitos mínimos para ser el estándar a manejar en el proyecto.

### **6.5. Implementación**

Se realizará la codificación de la plataforma haciendo uso de un framework de desarrollo web, y se crearan pruebas por cada módulo, antes de su implementación en frontend, para asegurar la calidad de los métodos.

## **7. LÍNEA DE INVESTIGACIÓN**

Tecnologías Actuales y Desarrollo de Software.

## **8. DESARROLLO DEL PROYECTO**

Durante el transcurso del proyecto, se determinó que la metodología propuesta resulto ser la más adecuada para la implementación de la plataforma.

A continuación, se mostrará lo realizado en cada una de las fases metodológicas.

### **8.1. Exploración previa:**

EL proyecto se centra en dos fases, la primera consta del diseño y desarrollo del módulo de backend para el área de gimnasios de la universidad santo tomas seccional Tunja, y la segunda fase, se centra en la implementación del software de gimnasio y adaptado para los demás espacios deportivos de la universidad.

El proceso de desarrollo del software comenzó inicialmente con la reunión del 02 de febrero de 2019 con los profesores Jimmy Anceno Cubides Cubides, Wilson Hernández Sánchez, Sergio Lopez Betancourth y el coordinador de deportes Harold Rodrigo Ortega Mogollon. En esta reunión analizaron las problemáticas que tiene bienestar universitario con respecto a la administración de los espacios deportivos, compuestas especialmente los gimnasios de la universidad, respecto a las problemáticas propuestas por el grupo de bienestar universitario y como solución se propuso desarrollar un software que ayude a solucionar estos inconvenientes e implementar una plataforma de reservas dinámicas del gimnasio.

Los profesores de bienestar entregaron sus puntos de vista respecto a los usos de los gimnasios con las problemáticas las cuales se centran en los manejos de los espacios y adecuar los horarios para el mejor uso del espacio de gimnasio ya que se excede la capacidad para poder trabajar con mayor comodidad y tener buenos usos de las maquinas, todo esto prestado por bienestar universitario.

Los requerimientos del grupo de bienestar son:

- uso de reservas para acceder al gimnasio, para mejor uso de los franjas deportivas (trx,halterofilia.etc) planteadas y ofertadas por bienestar.
- realizar reservas de espacios deportivos por medio de la plataforma
- se limitará el cupo de cada gimnasio a un máximo de 40 personas (centro históricamente) y 25 personas (av. universitaria) respecto a la capacidad del gimnasio.
- los usuarios que incurran en inasistencia (3) repetidamente injustificadas, conllevarán a aplicar algún tipo de sanción, ejemplo suspensión de reservas por un tiempo.
- se tiene en cuenta que el rol de usuario en el aplicativo se trabajará en los 4 tipos de perfiles usuarios como lo son (estudiantes, administrativos, docentes, egresados)
- aparte del rol de usuario existirá un rol administrador el cual pueda generar los reportes, según el periodo o lapso de tiempo que se necesite, también este rol de administrador podrá ofertar los espacios deportivos en los horarios que se proponen.
- la plataforma tendrá la capacidad de mostrar el horario, el espacio ofertado por bienestar, al igual si hay cupos para ese espacio o no hay cupos.
- en caso de no existir cupos en la sede seleccionada, el aplicativo arrojará la disponibilidad de haber cupos en la otra sede.
- es necesario contar con el control de asistencias ya que para los profesores del gimnasio entregan un reporte de asistentes mensualmente y reemplazar el uso de los formatos de asistencia reduciendo el uso de papel.
- se identificará al usuario por medio del documento de identidad.
- se podrá tener en cuenta posibles nuevos módulos para próximas actualizaciones del aplicativo

Posteriormente a esta reunión se realizó otra con el profesor de deportes Jimmy Anceno Cubides Cubides el día 28 de febrero de 2019 donde se hizo la siguiente recopilación de información:

Los horarios de gimnasio cubren de 7:30 am a 10:00 am y de 1:00 pm a 9:00 pm y los días sábados de 7:00 am a 1:00 pm, donde el profesor aclara que hay clases que no se dictan todo el semestre si no en algunos días específicos, por lo cual, es importante que ese pueda inscribir actividades en cualquier momento, también se especifica que ellos necesitan saber cuántos estudiantes hay inscritos por clase.

Una vez terminada esta reunión y con documentación anexa que envió el departamento de bienestar, se dio por concluida la etapa de recopilación de información.

## **8.2. Análisis:**

A partir de la información recopilada en la etapa anterior se procede a hacer una lista de funcionalidades que tendrá el sistema.

- Inscribir usuarios y asignarles un rol dentro del sistema.
- Controlar que usuarios pueden inscribirse a un evento deportivo, cultural o a una clase en específico.
- Monitorear que no existan sobrecupos en gimnasios.
- Inscribir actividades deportivas y culturales generales.
- Inscribir periodos académicos incluyendo vacacionales
- Inscribir espacios deportivos en cada sede.
- Inscribir clases asignándole un espacio.
- Reservas de espacios registrados para una clase o evento deportivo.
- Consultar las clases asignadas a un espacio mediante el uso de un calendario.
- Inscripción a eventos libres, solo con estar registrado y activo en el sistema.
- Generar una lista de los usuarios o inscritos de cada clase.
- Generar un horario de las clases con espacios y fechas ya asignadas.
- Control de sanciones a usuarios.

Teniendo en cuenta el análisis del análisis de la información y la lista de requerimientos se determinó que la plataforma web contara con tres tipos de usuarios Administrador, Instructor o Profesor y usuario final.

A su vez los usuarios pueden tener funciones diferentes en el sistema, las cuales se asignarán mediante roles.

- Módulo de usuarios.
- Módulo de actividades.
- Módulo de sedes.
- Módulo de espacios deportivos
- Módulo de clases
- Módulo de reservas
- Módulo de informes

vista instructor

- Módulo de clases.

vista usuario final

- Módulo de clases

ANEXO A. Documento de Requerimientos de software

### **8.3. Requisitos :**

Dentro de este contexto una vez realizadas las operaciones de recolección y análisis de requerimientos se hace el diseño de los módulos que formara la plataforma.

Vista administrador:

- Módulo de usuarios.
- Módulo de actividades.
- Módulo de sedes.
- Módulo de espacios deportivos
- Modulo de clases
- Módulo de reservas
- Módulo de informes

vista instructor

- Módulo de clases.

vista usuario final

- Módulo de clases

#### **8.3.1. Vista Administrador**

Esta sección de sistema es el pilar de la plataforma ya que es el encargado del control absoluto sobre la gestión de usuarios, espacios deportivos, clases, reservas y periodos académicos, dividido en varios módulos.

##### **1. Módulo de usuarios:**

Este módulo es el encardo de toda la gestión correspondiente a la información de cada usuario.

##### **2. Módulo de actividades:**

Este modulo administra todo los eventos deportivos y culturales, impartidas por el departamento de bienestar universitario y otros.

##### **3. Módulo de sedes:**

Este módulo hace toda la gestión de sedes de la universidad.

#### **4. Módulo de espacios deportivos:**

Este módulo hace toda la gestión de espacios deportivos como gimnasios y canchas a cada sede de la universidad Santo Tomas seccional Tunja.

#### **5. Módulo de periodos académicos:**

Este módulo hace toda la gestión de periodos académicos y hace la validación de que periodo este activo.

#### **6. Módulo de clases:**

Este módulo permite toda la gestión correspondiente a la gestión de eventos deportivos y culturales, asignándole un instructor a una actividad registrada, y la asignación de usuarios a esta.

#### **7. Módulo de reservas:**

Este módulo gestiona toda la función principal del sistema, reserva de espacios deportivos para asignarlos a clases ya inscritas en un horario académico valido.

### **8.3.2. Vista de instructor**

La vista de instructor contará con un único módulo ya que solo necesita hacer consultas y gestión de asistencias.

#### **1. Módulo de clases:**

Este módulo permite ver la información de las clases asignadas (usuarios, horarios), inscritas y hacer control de asistencia de alumnos o usuarios. También podrá ver y registrarse a clases que sean públicas o no tengan restricciones para su inscripción.

### **8.3.3. Vista de usuario final**

La vista de instructor contará con un único módulo ya que solo necesita hacer consultas de horario.

#### **1. Módulo de clases:**

Este módulo permite ver la información de las clases inscritas. También podrá ver y registrarse a clases que sean públicas o no tengan restricciones para su inscripción.

## 9. ARQUITECTURA Y DISEÑO DE LA PLATAFORMA

El propósito de la siguiente base de datos es poder contar en todo momento con el respaldo de una secuencia de pasos y flujo de datos que correspondan con lo estipulado en la etapa de diseño.

### 9.1. Base de datos

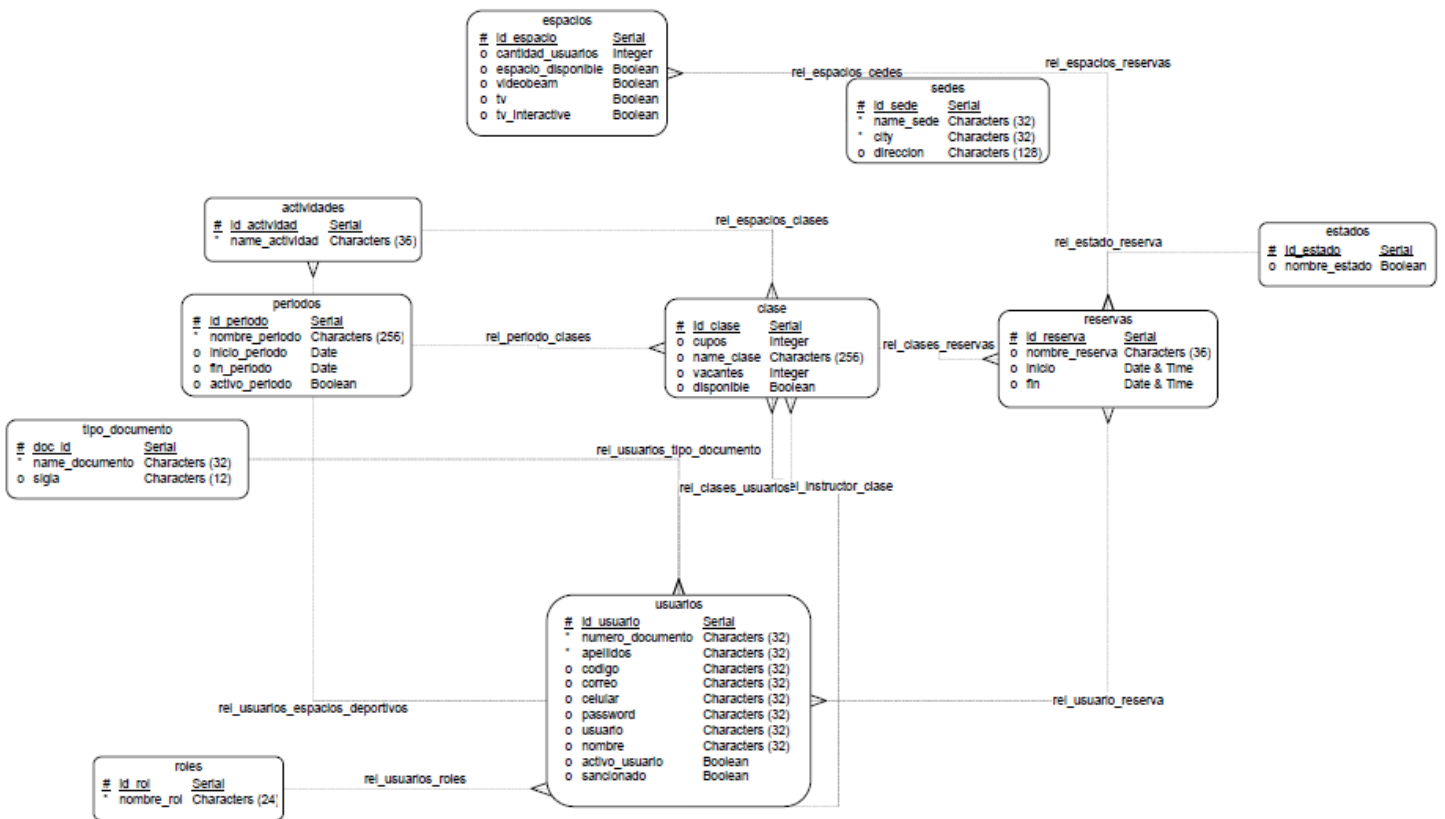


Ilustración 1: modelo conceptual

## 9.2. Diccionario de datos

Tabla 1-rol						
Campo	Tipo	Longitud	Llave primaria	Llave foránea	obligatorio	descripción
Id	Serial		x		x	Llave primaria de la tupla
name	Text	24			x	Nombre de la tupla "único"

Tabla 2 - Tipo de documento						
Campo	Tipo	Longitud	Llave primaria	Llave foránea	obligatorio	descripción
Id	Serial		x		x	Llave primaria de la tupla
Name	Text	24			x	Tipo de documento válido en Colombia "único"
sigla	Text	6			x	Sigla de identificación del documento válido en Colombia de la tupla "único"

Tabla 3 - Periodos						
Campo	Tipo	Longitud	Llave primaria	Llave foránea	obligatorio	descripción
Id	Serial		x		x	Llave primaria de la tupla
Name	Text	24			x	Nombre de la tupla "único"
Inicio	Date					Fecha de inicio del periodo académico
Fin	Date					Fecha fin del periodo académico
Activo	Boolean					Identificación de periodo académico activo

Tabla 4 – Sedes						
Campo	Tipo	Longitud	Llave primaria	Llave foránea	obligatorio	descripción

Id	Serial		x		x	Llave primaria de la tupla
name	Text	24			x	Nombre del estado
city	Text				x	Ciudad de la sede
address	Text					Dirección de la sede

<b>Tabla 5 – Espacios</b>						
<b>Campo</b>	<b>Tipo</b>	<b>Longitud</b>	<b>Llave primaria</b>	<b>Llave foránea</b>	<b>obligatorio</b>	<b>descripción</b>
Id	Serial		x		x	Llave primaria de la tupla
Name	Text	24			x	Nombre de la tupla “único”
Code	Text					Código del espacio
Id_sede	Intger			x		Sede del espacio
Cupos	Integer				x	Capacidad de usuarios
available	Boolean					Identificación si el espacio está disponible
Tv	Boolean					Identifica si el espacio cuenta con tv
Video-beam	Boolean					Identifica si el espacio cuenta con Videobeam
Tv-interactive	Boolean					Identifica si el espacio cuenta con tv-interactivo

<b>Tabla 6 - Actividades</b>						
<b>Campo</b>	<b>Tipo</b>	<b>Longitud</b>	<b>Llave primaria</b>	<b>Llave foránea</b>	<b>obligatorio</b>	<b>descripción</b>
Id	Serial		x		x	Llave primaria de la tupla

Name	Text	24			x	Nombre de la tupla "único"
encargado	Integer				x	Indica que departamento está encargado de la actividad o evento

<b>Tabla 7 – Estados</b>						
<b>Campo</b>	<b>Tipo</b>	<b>Longitud</b>	<b>Llave primaria</b>	<b>Llave foránea</b>	<b>obligatorio</b>	<b>descripción</b>
Id	Serial		x		x	Llave primaria de la tupla
Name	Text	24			x	Nombre del estado

<b>Tabla 8 – usuario</b>						
<b>Campo</b>	<b>Tipo</b>	<b>Longitud</b>	<b>Llave primaria</b>	<b>Llave foránea</b>	<b>obligatorio</b>	<b>descripción</b>
Id	Serial		x		x	Llave primaria de la tupla
Name	Text	24			X	Nombre del estado
lastname	Text	36				Apellido del usuario
Email	email	36				Correo institucional
Code	Text	12				Código USTA o número de documento
Mobile	Text	16				Número de celular
password	Text	24				Contraseña de acceso al sistema, encriptación bcrypt
Active	Boolean					Indica si esta activo o inactivo en el sistema
sanctioned	Boolean					Indica si el usuario está sancionado
asistencia	Boolean					Valor para control de asistencia
Id_rol	Integer			x	x	Rol en el sistema, por defecto es estudiante

identification	Text			x	x	Numero o código de identificación
Id_doc	Integer				x	Tipo de documento del usuario, por defecto cedula de ciudadanía “CC”

<b>Tabla 9 – clases</b>						
<b>Campo</b>	<b>Tipo</b>	<b>Longitud</b>	<b>Llave primaria</b>	<b>Llave foránea</b>	<b>obligatorio</b>	<b>descripción</b>
Id	Serial		x		x	Llave primaria de la tupla
Name	Text	24			x	Nombre del estado
Cupos	Integer				x	Cupos de la clase
vacantes	Integer					Cupos de la clase disponibles” Consulta”
Available	Boolean					Indica si la clase está disponible o es de uso libre
Id_actividad	Integer				x	Actividad asociada a la clase
Id_periodo	Integer			x	x	Periodo académico de la clase
Id_instructor	Integer			x	x	Profesor o instructor de la clase

<b>Tabla 10 – reservas</b>						
<b>Campo</b>	<b>Tipo</b>	<b>Longitud</b>	<b>Llave primaria</b>	<b>Llave foránea</b>	<b>obligatorio</b>	<b>descripción</b>
Id	Serial		x		x	Llave primaria de la tupla
Name	Text	24				Nombre de la reserva, por defecto nombre de la clase asociada
Id_clase	Integer			x	x	Clase asociada a la reserva
Id_espacio	Integer			x	x	Espacio asociado a la reserva
Id_estado	Integer			x	x	Estado de la reserva, por defecto 1 “activo”

Inicio	Date				x	Dia y hora inicial de la reserva
Fin	Date				x	Dia y hora final de la reserva

### 9.3. Diagramas de secuencia.

Crear una reserva para una clase.

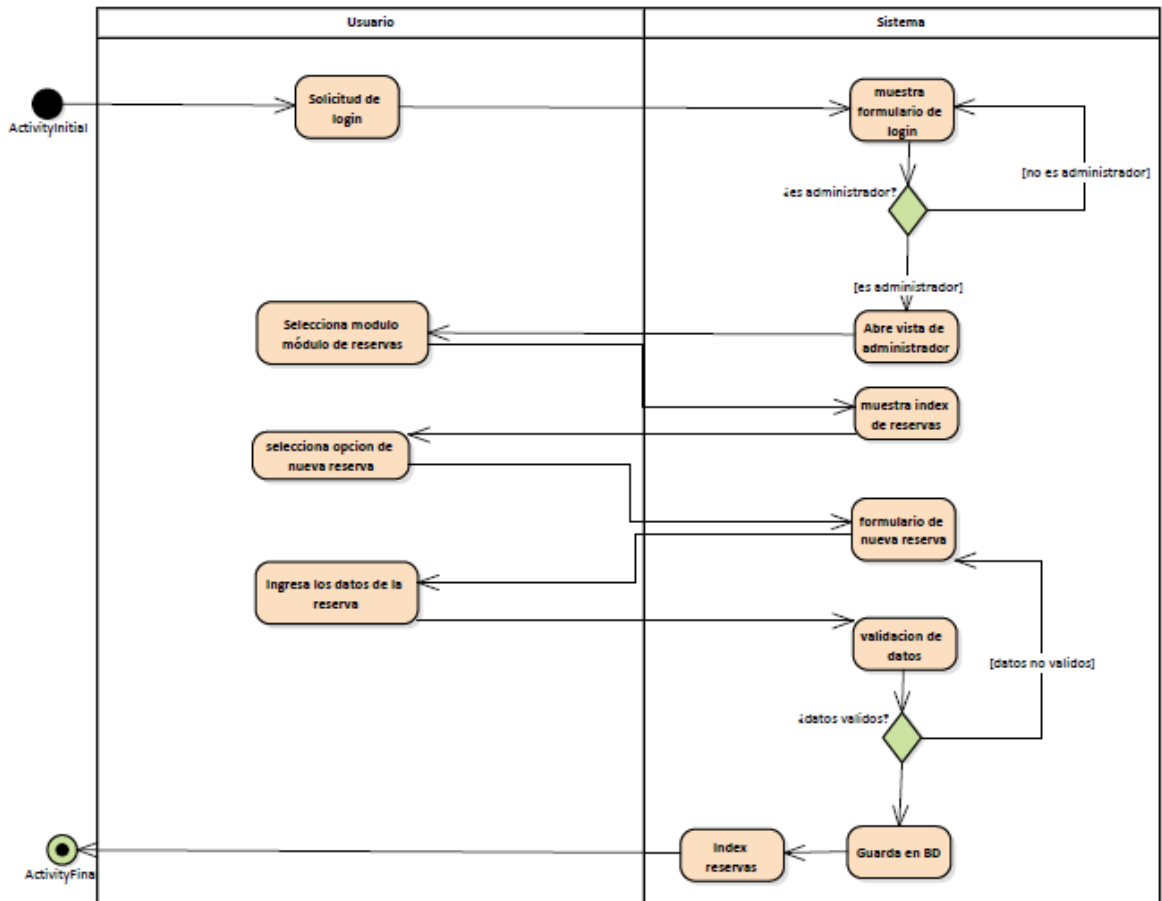
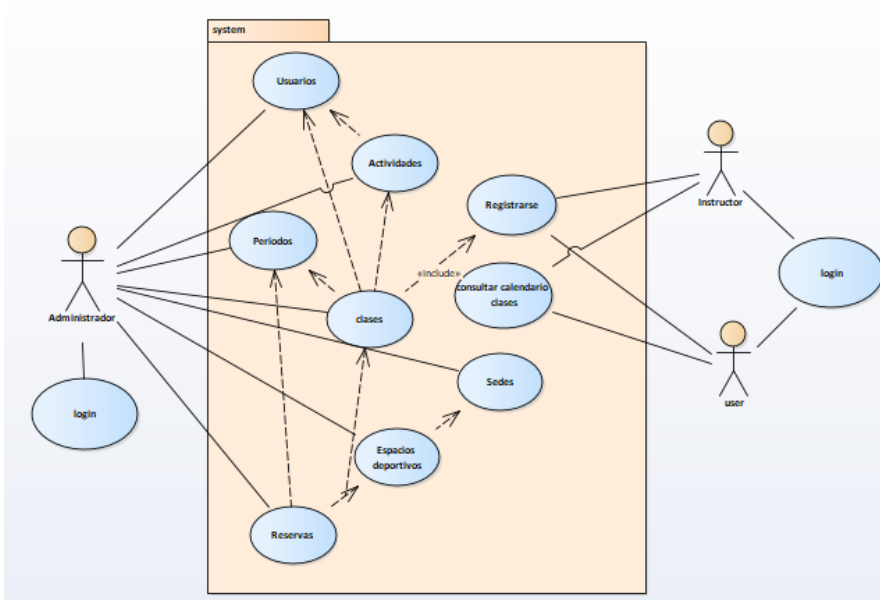


Ilustración 2 Diagrama de procesos de reservas  
Diagrama procesos

## 10. ARQUITECTURA DE LA PLATAFORMA

### 10.1. Diagrama de casos de uso del sistema

Ilustración 3 Diagrama de casos de uso



### 10.2. Diagrama de componentes del sistema.

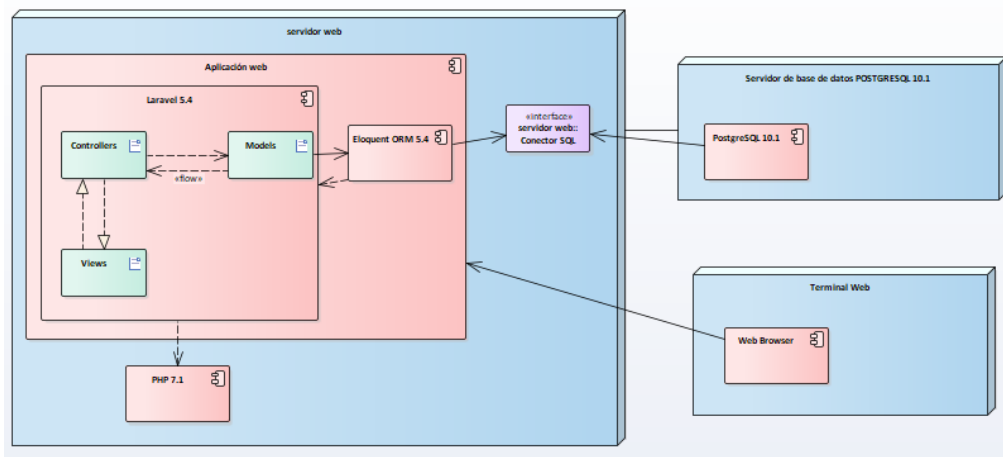


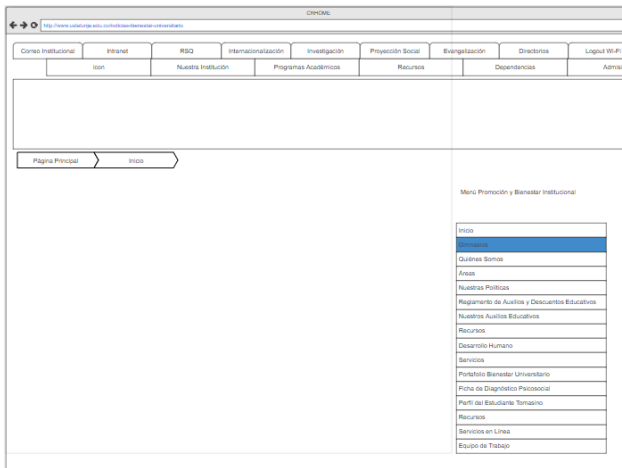
Ilustración 4: Diagrama de componentes

### 10.3. Diseño de la plataforma.

Teniendo en cuenta los requerimientos de la plataforma antes mencionados, se procedió hacer un diseño previo de la plataforma

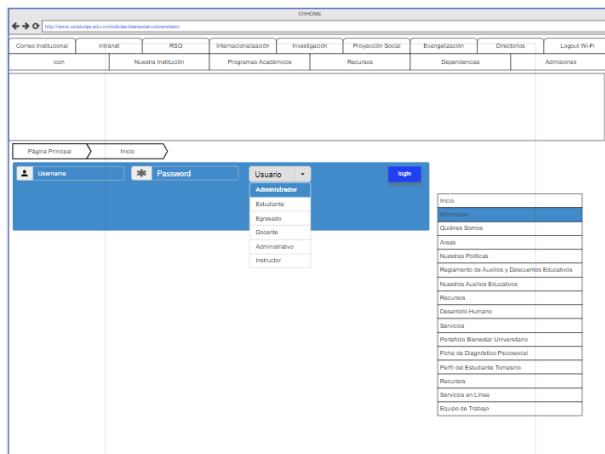
#### 10.3.1. Vista Index:

la pantalla de inicio, se muestra todo el contenido de la universidad en la parte inferior izquierda se encuentra el menú del sistema, donde se encontrará una columna para nuestro aplicativo en la opción gimnasios



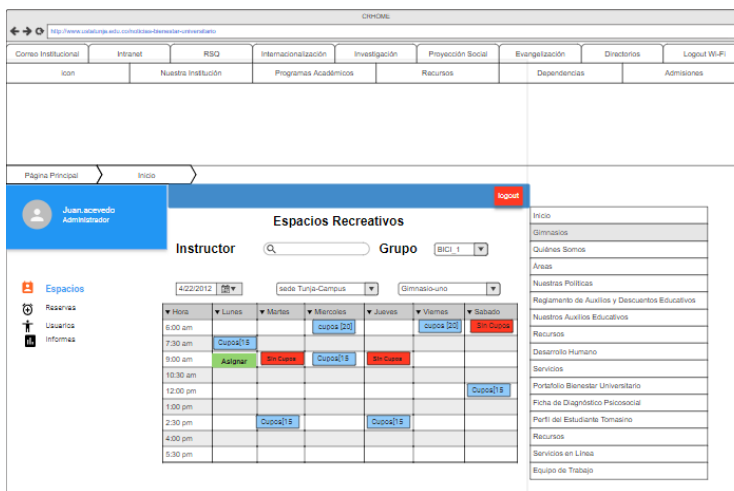
#### 10.3.2. Vista login:

la pantalla de login es la pantalla que nos da el introducción a las de más vistas donde se muestra un campo para digitar el usuario (correo electrónico institucional) y otro para la contraseña( contraseña campus virtual), como también se muestra una lista desplegable con los tipos de usuarios para el sistema



### 10.3.3. Vista administrador:

en esta vista el administrador interactúa con la aplicación agregando los espacios, sacando reportes, como la visualización de las reservas como de usuarios



### 10.3.4. Vista Usuarios:

el usuario interactúa al visualizar los distintos horarios, como los espacios deportivos que ofertan en bienestar, los cupos disponibles, Si se hace click en logout, en estudiante o administrador lo envía a la página de login.

CHORDL

Inicio

**Juan Lucero**  
Administrador

### Lista general de usuarios

Administración de Sistemas | Estudiantes

Nombre	Usuario	Código	Sección	Última Sesión
Usuario	usuario1.ejemplo	100001	<input type="checkbox"/>	15/02/2019
Usuario	usuario1.ejemplo	100001	<input type="checkbox"/>	15/02/2019
Usuario	usuario1.ejemplo	100001	<input type="checkbox"/>	15/02/2019
Usuario	usuario1.ejemplo	100001	<input type="checkbox"/>	15/02/2019
Usuario	usuario1.ejemplo	100001	<input type="checkbox"/>	15/02/2019
Usuario	usuario1.ejemplo	100001	<input type="checkbox"/>	15/02/2019

Inicio

- Gimnasios
- Quiénes Somos
- Áreas
- Nuestras Políticas
- Reglamento de Asuntos y Descuentos Educativos
- Nuestros Asuntos Educativos
- Recursos
- Desarrollo Humano
- Servicios
- Portafolio Bienestar Universitario
- Forma de Diagnóstico Psicosocial
- Perfil del Estudiante Tommaso
- Recursos
- Servicios en Línea
- Estado de Trabajo

**Vista usuario:** el usuario visualiza los cupos disponibles en cada clase.

CHORDL

Inicio

**Juan Ulloa**  
Estudiante

4/22/2012

Español Recreativo

Sección Turista Campus

Dimas-Luna

Hora	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
6:00 am	Cupos: 9	Cupos: 8	10 Cupos			
7:30 am						
9:00 am						
10:30 am						
12:00 pm						
1:00 pm						
2:30 pm						
4:00 pm						
5:30 pm						

Inicio

- Gimnasios
- Quiénes Somos
- Áreas
- Nuestras Políticas
- Reglamento de Asuntos y Descuentos Educativos
- Nuestros Asuntos Educativos
- Recursos
- Desarrollo Humano
- Servicios
- Portafolio Bienestar Universitario
- Forma de Diagnóstico Psicosocial
- Perfil del Estudiante Tommaso
- Recursos

### 10.3.5. Vista reserva

El estudiante genera su reserva seleccionando el horario y la fecha

The image displays two side-by-side screenshots of a web application interface for making reservations. Both screenshots show a user profile at the top left, a search bar, and a dropdown menu for selecting a course. Below these are two input fields for 'Bici-1' and 'Entrenamiento libre', followed by two input fields for '6:00 am' and '4/22/2012'. At the bottom of each form are two buttons: 'Cancelar' (Cancel) and 'Inscribir' (Reserve).

### 10.3.6. Fase de iteraciones

Durante la fase de iteraciones se estableció la prioridad de los módulos de cada usuario, así como la jerarquía de cada uno de los submódulos de la plataforma de reservas, así como toda la recopilación previa de información de clases de semestres anteriores.

### 10.3.7. Fase de producción

En esta fase se hizo la implementación de la plataforma web usando un desktop de la facultad de ingeniería de sistemas para la fase de pruebas iniciales. Durante esta implantación se llevó a cabo las pruebas funcionales del sistema, usando datos reales de semestres anteriores, para la búsqueda de errores antes de una fase de producción final en el servidor principal de la universidad.

Las pruebas se realizaron en los siguientes hosts:

- Celular Huawei YII con sistema operativo Android 5.1
- Laptop ASUS x541U con sistema operativo Windows 10 de 64 bits
- Laptop HP Pavilion x360
- Laptop DELL Inspiron 1545 con sistema operativo Windows 7 de 32 bits

### **10.3.8. Fase de mantenimiento**

Durante esta fase se realizó la corrección de los errores encontrados durante la fase de producción, permitiendo así que la aplicación sea estable y funcional.

## 11. Recursos

### 11.1. presupuestó

Concepto	Valor Unidad	Cantidad	Valor total
Servidor local de pruebas	\$0 pesos	1	\$0
Internet	\$35000/mes	9	\$315000
Curso de Laravel online	\$180000	1	\$180000
Laptop Asus X541U	\$1750000	1	\$1750000
Desktop	\$850000	1	\$850000
Transportes	\$3300	120	\$396000
Papelería	\$35	40	\$1400

Tabla 1:Presupuesto

### 11.2. Recurso Humano

Nombre	rol	Horas empleadas
Luis Fernando Castellanos Guarín	Director Proyecto	320
Iván Fernando Fonseca Barinas	Tutor de tesis	96
Juan Pablo Acevedo Torres	Desarrollador	720

Tabla 2 Recurso Humano

### 11.3. Software empleado

Software	versión	valor
Sublime Text	3.1	Software libre
PostgreSQL	9.6	Versión libre
Libre office	Office 365	

Laravel	5.4	Software libre
XAMPP	7.1.13	Software libre

Tabla 3 :Software utilizado

## CONCLUSIONES

La realización de los modelos funcionales de Los módulos, así como el levantamiento de requerimientos permitió garantizar que éstas funcionan acordes a lo solicitado.

La base de datos fue diseñada con base en el modelo conceptual con el fin de ser fácilmente adaptable a mejoras futuras.

La plataforma web fue creada teniendo en cuenta los diferentes roles (administrador y empresa) según su nivel de permisos en el sistema, con el fin de mantener una jerarquía en el manejo de la información de este sistema y así mantener la información actualizada.

El backend se hizo usando el ORM eloquent y usando el estándar de rutas GET Y POST, así como la exportación de datos en formato JSON, lo que permite una mayor compatibilidad con otros frameworks., escalabilidad y flexibilidad y además se requiere un menor uso del procesador.

Las pruebas funcionales de la aplicación se llevaron acabó de todo el desarrollo, usando el modelo de pruebas TDD para así poder garantizar el correcto funcionamiento de éstas, antes de llegar a la fase de producción.

## 12. BIBLIOGRAFIA

- [1] Bahit, E. (2012). *Scrum y eXtreme Programming para Programadores*. Buenos Aires, Argentina.
- [2]. Brian Terlson. (30 de junio de 2019). *ECMA international*. Obtenido de <https://www.ecma-international.org/publications/files/ECMA-ST/ECMA-262.pdf>
- [3]. Jurado, C. B. (2010). *Diseño Ágil con TDD*. Lulu.com.
- [4]. Manuaba, I. B. (2019). Combination of Test-Driven Development and Behavior-Driven Development for Improving Backend Testing Performance. *Procedia Computer Science*, 79-86.
- [5]. Peleeeger, S. L. (2002). *Ingeniería de software*. PRENTICE HALL ARGENTINA.
- [6]. Pressman, R. (2010). *Ingeniería del Software. Un enfoque práctico*. USA: McGraw-Hill.
- [7]. Iván Camilo Ocampo Garzón (2016). *diseño y desarrollo de un prototipo de aplicación móvil para estimar paquetes turísticos de deportes extremos en el departamento de cundinamarca*. Universidad Distrital Francisco José De Caldas.
- [8].  
Juan Manuel Vela Gomez Jhonnatan, Camilo Ortiz Soriano (2008). Desarrollo De Un Aplicativo Para El Registro Y Seguimiento De La Información De Los Procesos Judiciales En El Consorcio Aj Abogados. *Universidad De San Buenaventura Facultad De Tecnología*.