

**SISTEMA DE GESTION DE INFORMACION DE MONITORIAS
ACADEMICAS PARA EL PROGRAMA DE INGENIERIA DE SISTEMAS Y
COMPUTACION DE UTP**

**JUAN DAVID HERRERA MARTINEZ
SEBASTIAN ZAPATA COLORADO**

**UNIVERSIDAD TECNOLOGICA DE PEREIRA
FACULTAD DE INGENIERIAS, PROGRAMA DE INGENIERIA
DE SISTEMAS Y COMPUTACION
PEREIRA, RISARALDA
MAYO 2016**

**SISTEMA DE GESTION DE INFORMACION DE MONITORIAS
ACADEMICAS PARA EL PROGRAMA DE INGENIERIA DE SISTEMAS Y
COMPUTACION DE UTP**

**JUAN DAVID HERRERA MARTINEZ
SEBASTIAN ZAPATA COLORADO**

DOCUMENTO FINAL

Director: CARLOS ALBERTO OCAMPO SEPULVEDA

**UNIVERSIDAD TECNOLOGICA DE PEREIRA
FACULTAD DE INGENIERIAS, PROGRAMA DE INGENIERIA
DE SISTEMAS Y COMPUTACION
PEREIRA, RISARALDA**

MAYO 2016

TABLA DE CONTENIDO

	Pagina
GLOSARIO	1
INTRODUCCION	4
PLANTEAMIENTO DEL PROBLEMA	5
ANTECEDENTES	6
JUSTIFICACION DEL PROYECTO	7
OBJETIVO GENERAL	8
OBJETIVOS ESPECIFICOS	8
METODOLOGIA	9
CRONOGRAMA	28
DESARROLLO	29
ANALISIS DE RESULTADOS	43
CONCLUSIONES	51
RECOMENDACIONES	52
BIBLIOGRAFIA	53
LISTA DE ANEXOS	54
ANEXOS	57

GLOSARIO

PROTOTIPO: Los prototipos son una representación limitada de un producto, permite a las partes probarlo en situaciones reales o explorar su uso. A nivel de informática es un producto de software inicial que cumple con los requerimientos mínimamente funcionales para la realización de alguna tarea o proceso en específico.

DISPOSITIVO: Hace referencia a periféricos que son utilizados para capturar algún conjunto de datos de la realidad humana.

LENGUAJE DE PROGRAMACIÓN (informática): Notación utilizada por los programadores para escribir programas. Un lenguaje tiene una sintaxis (las palabras y símbolos utilizadas para escribir códigos de programa), una gramática (las reglas que definen una secuencia de palabras y símbolos significativos y correctos) y semántica. Java es un lenguaje de programación¹.

FRAMEWORK: Es un conjunto de herramientas de software que funciona como un marco o esquema para desarrollar en algún lenguaje de programación. Facilita la codificación de código fuente ya que trae por defecto configuraciones, vinculaciones y plantillas que el programador no necesita volver a codificar.

TABLETA DIGITALIZADORA: Es un dispositivo que permite capturar trazos físicos como lo haría un lápiz y un papel pero llevados digitalmente a un computador. Generalmente estos trazos son capturados de manera de vector o una imagen bidimensional.

LECTOR CODIGO DE BARRAS: dispositivo óptico que captura mediante un rayo láser el código de barras de un elemento y es transmitido a un computador. Estos lectores ayudan a agilizar procesos de identificación de productos.

¹ Glosario de términos de programación. Fundamentos de Programación: Algoritmos, Estructuras de datos. Joyanes Aguilar, Luis.

MÓDULOS: Fase seccional de un software. Comprende una parte del software total que ayuda a una tarea en específico. En general, un módulo en informática hace referencia a una cantidad de prestaciones procesadas que sirven de entrada a otro módulo.

MONITORIA: Proceso en el cual un estudiante con habilidades en cierta asignatura brinda sus conocimientos y experiencias a estudiantes que no lo poseen.

MOCKUP: Es una demostración de evaluación de diseño de algún prototipo de desarrollo para una aproximación del objetivo final del diseño. Se realiza gráficamente un modelo que ayude al cliente a visualizar lo que se va a alcanzar por parte del grupo de desarrollo en un ambiente similar al cual se va a desplegar.

FRONT-END: Hace referencia al desarrollo web realizado para toda la visualización por parte del usuario que va a hacer uso de ella. Todo se enmarca en el diseño adecuado y agradable para facilitar la interacción entre el usuario y el servidor.

BACK-END: Hace referencia al desarrollo que se hace a un aplicativo para interactuar con el servidor de datos ofreciendo valores congruentes que el diseño front-end necesita para su uso adecuado. El desarrollo back-end busca el mejor performance de la aplicación según el lenguaje de programación que se emplee.

MODELO-VISTA-CONTROLADOR: Separación de datos, interfaz de usuario y toda la lógica que se necesita en un desarrollo de software para que se realice una tarea en específico. Es un patrón de diseño de software que ayuda a los desarrolladores a realizar programas poderosos con una trazabilidad entendible en el manejo de variables y de secuencias de ejecución. El modelo maneja la lógica y privilegios del aplicativo con la base de datos. El controlador modifica, responde e invoca todas las peticiones que el modelo ejecute según su interacción con las vistas. La vista muestra toda la información del modelo de salida en un esquema interactivo y entendible para el usuario.

DIRECCION IP: Identifica de manera lógica mediante un número único a un dispositivo de red como un computador, según la jerarquía de la red a la cual pertenezca.

RUBY²: Un lenguaje de programación dinámico y de código abierto enfocado en la simplicidad y productividad. Su elegante sintaxis se siente natural al leerla y fácil al escribirla.

GEMA: Es el término utilizado en el lenguaje de programación Ruby para lo que en programación computacional es denominado Librerías. Estas gemas o RubyGems son librerías que ayudan con funcionalidades específicas que pueden ser embebidas en programas de Ruby para cumplir con funcionalidades específicas en un formato estándar.

² Ruby. Definición en línea: '<https://www.ruby-lang.org/es/>'

INTRODUCCION

Responder efectivamente al ingreso de estudiantes a la Universidad Tecnológica de Pereira, así como asegurar su permanencia y egreso exitoso, es uno de los objetivos de la institución que se debe desplegar para mejorar la experiencia de los estudiantes, haciendo uso de diversos mecanismos para garantizar su formación profesional tal como lo puede ofrecer el programa de acompañamiento tutorial.

Un mecanismo que se utiliza para cumplir el objetivo de la universidad, es la implementación de monitores académicos del programa de ingeniería de sistemas y computación, el cual cuenta con una serie de procesos como lo son la convocatoria y selección de los monitores, la correcta vinculación de estos, la inducción, la programación de sus salones y horarios, su certificación y pago.

En un tiempo en el que la tecnología forma parte crucial en el desarrollo de las instituciones educativas que buscan altos niveles de calidad, implementar herramientas tecnológicas que optimicen los procesos académicos se hace casi indispensable. El desarrollo de un prototipo web que gestione correctamente el mecanismo de monitorías académicas aplicado a los estudiantes de ingeniería de sistemas y computación de la Universidad Tecnológica de Pereira pretende entonces apuntar a la mejora continua de los procesos de la institución.

PLANTEAMIENTO DEL PROBLEMA

El control de los laboratorios del programa de ingeniería y sistemas de la Universidad Tecnológica de Pereira, es un proceso sencillo pero poco práctico actualmente debido a que, el personal capacitado para realizar las monitorias en las distintas salas asignadas, solo tienen papel como comprobante de ingreso de los estudiantes que hacen uso de dichos laboratorios, creando inconsistencias o incoherencias en la documentación de seguimiento, llegando al punto en el cual el contenido diligenciado pueda ser catalogado como inservible por fraude o ilegibilidad. Este mal uso de los recursos para administrar el registro de estudiantes beneficiarios del servicio, hace que la persona encargada de comprobar dicho contenido se tarde demasiado tiempo en dar validez y transcribir a formato digital todos los formularios entregados y sin duda alguna, al utilizar este mecanismo de control se incrementa el indicador de inconsistencia de datos por mala transcripción.

Otro punto a tratar dentro de la gestión de información, es el difícil seguimiento de los monitores en los laboratorios y horarios asignado según sea su contratación, ya que la persona encargada de certificar la asistencia del monitor debe desplazarse a lo largo del campus universitario de la Universidad Tecnológica de Pereira, para dar validez de la normativa.

ANTECEDENTES

Dada las observaciones realizadas al manejo actual de información para el programa de acompañamiento tutorial en el programa de ingeniería de sistemas y computación de la Universidad Tecnológica de Pereira, se establece lo siguiente:

- El proceso de documentación por parte de los monitores de los laboratorios, ha sido realizado con formatos de papel que son diligenciado por los estudiantes que hacen uso de dichos laboratorios, siendo completados todos los campos correspondientes y finalizando con una firma que garantiza la veracidad de su presencia. Una vez entregados los formatos por parte de los monitores al personal encargado de administración de información, deben de ser primero verificados para dar validez haciendo uso de las evidencias y posteriormente, transcritos a una hoja de cálculo en computador para llevar un control.
- El seguimiento de los monitores en los laboratorios, se ha estado realizando mediante el desplazamiento físico a lo largo del campus universitario del personal encargado para comprobar el cumplimiento del requisito normativo. Estos datos deben de ser transcritos a una hoja de cálculo en computador para llevar un control.
- El personal encargado de realizar el control para cada monitor, se basa en las hojas de cálculo que se modificaron a lo largo del mes, dando así según las horas de trabajo realizadas y la asistencia en los laboratorios, el valor que se le va a pagar al monitor por su trabajo.

JUSTIFICACION DEL PROYECTO

La Universidad Tecnológica de Pereira tiene como objetivo garantizar un proceso educativo exitoso para los estudiantes. Como gestión para alcanzar dicho objetivo se propone un plan de monitorias académicas. Actualmente este proceso de monitorias no cuenta con un sistema que permita realizar las tareas necesarias de manera eficiente. Por lo tanto se están generando unas cargas laborales que pueden llegar a entorpecer el proceso de monitorias, implicando así un problema en el camino a alcanzar el objetivo institucional.

Se propone el desarrollo de un prototipo que gestione los procesos ligados al programa de monitorias académicas que trataría este problema haciéndolo más eficiente y fiable en el proceso. Las firmas digitales se convierte en una de las evidencias primordiales a la hora de trabajar en este proyecto ya que, debe de ser anexado a cada registro que se va a ingresar a la base de datos, puesto que la firma y número de cédula de ciudadanía o código institucional, son muestras legales y formales de la presencia de alguna persona en algún establecimiento, se necesita dispositivos que capturen dicha información. Para esto se debe de trabajar con hardware comercial y de fácil adquisición que apunten a sistematizar el proceso actual.

Haciendo una integración metodológica de ingeniería de software para el desarrollo del proyecto, el prototipo prevé reducción de costos de operatividad como logística, papel, polvo láser e impresoras para expedir los formatos actuales. Pero también brindar un mejor servicio con calidad a los estudiantes que hacen uso de los laboratorios, a los monitores que velan por el correcto uso de estos lugares y sin lugar alguna, a las personas encargadas de administrar a todos los monitores contratados.

OBJETIVO GENERAL

Desarrollar un prototipo web, que gestione los procesos de información y seguimiento de los monitores y laboratorios académicos del programa de Ingeniería de Sistemas y Computación, de la UTP.

OBJETIVOS ESPECIFICOS

- Implementar una base de datos que permita llevar un control de los documentos
- Usar metodologías de la ingeniería de software aplicadas al trabajo de grado.
- Ejecutar plan de pruebas al prototipo.
- Generar manual de usuario con las funcionalidades del prototipo.

METODOLOGIA

Dada las condiciones para el despliegue del prototipo y su rápido desarrollo, fue necesario la elección de una metodología ágil que involucrara un fácil manejo de las actividades de programación como también una versatilidad en el cambio de requerimientos por parte de los involucrados del proyecto. Dada esta situación, la metodología que cumple con estas características de adecuación permanente entre programadores y clientes, es la “Programación Extrema³” cuyo fundamento se basa en la simplicidad, implementando diseños que permitan un desarrollo eficiente como también práctico y dinámico para los interesados.

Gracias a la simplicidad de esta metodología, se pueden crear estrategias entre los programadores que permitan un fácil manejo del desarrollo como del mantenimiento del sistema en desarrollo, es decir que al momento que cada uno de los desarrolladores hace un avance significativo de los módulos creados, se pueden realizar pruebas pertinentes a dicha sección para asegurar en mayor medida la compatibilidad con los diferentes módulos que se están desarrollando por otras personas.

Al momento de realizar actualizaciones y pruebas para continuar con los módulos restantes, se debe informar a los demás desarrolladores las funcionalidades, percances, alcances y sugerencias del módulo creado. Esto sin duda crea una asertiva comunicación utilizando diferentes canales y herramientas para que los desarrolladores se encuentren constantemente enterados del progreso del desarrollo total.

La retroalimentación se convierte en un valor constante que se tiene que repetir cada vez que se realice un avance significativo en las funcionalidades que los interesados del prototipo han estipulado, incentivando una clara vinculación directa que apunta a un beneficio común para ambas partes que es el despliegue adecuado del prototipo en desarrollo. Para que dicha retroalimentación sea constructiva y no en su sentido contrario, es necesario el respeto por parte de los desarrolladores en acatar las sugerencias o ideas del cliente como del cliente en escuchar

³ Programación Extrema. Extreme Programming Explained, Kent Beck.

por parte de las personas que realizan los avances en el desarrollo, lo que puede ser posible de alcanzar dadas las funcionalidades que apunten a un acorde funcionamiento del programa computacional. Por esta serie de criterios o valores se ha elegido para el desarrollo del prototipo web la metodología de ingeniería de software “Programación Extrema” adecuándose a las necesidades de los involucrados. Consiguiente a esto se crean cuatro (4) fases que la metodología implanta para el despliegue adecuado de la misma.

1. FASE DE PLANIFICACION DEL PROYECTO

1.1 Historias de Usuario

Estas historias de usuario se convierten en la pieza clave para los desarrolladores del prototipo, ya que muestran las funcionalidades principales exigidas por el administrador del prototipo junto a sus respectivas observaciones para que cumplan a cabalidad con lo esperado.

Historia de Usuario	
Número: 1	Nombre: Gestión de Solicitudes Monitores
Prioridad en Negocio: Media	Iteración Asignada: 1
<p>Descripción: Se requiere que el sistema le permita al usuario administrador recibir las solicitudes de los estudiantes que aspiran al cargo de monitor para la creación de las cuentas. Y además solicitudes posee un estado por defecto, estos estados son:</p> <ul style="list-style-type: none"> · Aprobado · Rechazado <p>Una vez la solicitud cambie de estado, lo cual lo define el administrador, el perfil monitor está en condiciones de ingresar al sistema.</p>	
<p>Observaciones: La aprobación de los monitores está regida bajo el criterio del administrador del prototipo.</p>	
<p>Responsable: Sebastian Zapata Colorado</p>	

Historia de Usuario	
Número: 2	Nombre: Gestión de Monitorias
Prioridad en Negocio: Alta	Iteración Asignada: 2
Descripción: El sistema debe permitir crear, eliminar y establecer periodos de habilitación de las monitorias según las asignaturas determinadas por el programa de ingeniería de sistemas y computación.	
Observaciones: Solo los monitores habilitados están en condiciones de utilizar esta funcionalidad. .	
Responsable: Juan David Herrera Martínez	

Historia de Usuario	
Número: 3	Nombre: Formatos de Seguimiento a Estudiantes y Monitores
Prioridad en Negocio: Media	Iteración Asignada: 3
Descripción: Se requiere que los formatos digitales cumplan con el diseño actual utilizado en versión física para proporcionar consistencias con el modelo vigente. Este modelo digital debe permitir visualizar dichos formatos de los distintos monitores habilitados.	
Observaciones: Formato 1: VICERRECTORÍA ACADÉMICA - CONTROL DE ASISTENCIA Formato 2: VICERRECTORÍA ACADÉMICA - CONTROL DE ASISTENCIA A SALAS DE CÓMPUTO	
Responsable: Sebastian Zapata Colorado	

Historia de Usuario	
Número: 4	Nombre: Seguimiento de Monitores
Prioridad en Negocio: Media	Iteración Asignada: 5
Descripción: Se requiere que el sistema le permita al usuario administrador visualizar los monitores activos junto a sus actividades realizadas en la jornada de trabajo, como también su ubicación en las distintas salas de cómputo en tiempo real.	
Observaciones: Solo serán visibles los monitores que estén cumpliendo su labor en las diferentes salas de cómputo.	
Responsable: Juan David Herrera Martínez	

Historia de Usuario	
Número: 5	Nombre: Perfiles de Usuario
Prioridad en Negocio: Media	Iteración Asignada: 5
Descripción: Se requiere que el sistema permita que los usuarios puedan iniciar sesión sea bajo el perfil Administrador o Monitor, como también su cambio de contraseña o recuperación de la misma dada las condiciones que se generen.	
Observaciones: La recuperación de contraseña se hará mediante un envío de instrucciones al correo electrónico del usuario registrado.	
Responsable: Juan David Herrera Martínez	

1.2 Release Planning

Dada las historias de usuario realizadas previamente se estipulan una serie de tiempos para cumplir con las funcionalidades que se esperan del prototipo, para esto es necesario de la experiencia de los desarrolladores como también de la disposición por parte del administrador del prototipo en encontrar una ventana de tolerancia dados los diferentes panoramas que se podrían generar al momento del inicio de codificación de código fuente y que implicarían una serie de retrasos o demoras en la entrega de dicha funcionalidad en una versión del prototipo.

Los tiempos que se han estipulado se pueden apreciar en la sección de **Cronograma** en este documento para evitar redundancia de información y facilitando la fácil comprensión de los procesos elaborados.

Para crear una buena experiencia en el manejo del prototipo por parte del administrador como también de los Monitores, se realizarán desarrollos en las vistas que faciliten el aspecto y sobretodo que convierta a prototipo en una herramienta de fácil manejo.

- **Historia de Usuario 1**

Para permitir el registro de los monitores al prototipo y un despliegue total de la información en las diferentes secciones de consultas que se pueden crear para ofrecer información completa y detallada para el perfil Administrador, es necesario que esta

historia de usuario se maneje con mucha prudencia de la mano con el diseño, desarrollo y creación de la base de datos, garantizando una trazabilidad de información con ayuda de filtros que imposibiliten el acceso y modificación dependiendo de los dos roles establecidos en el prototipo.

El tiempo límite para asegurar estas funcionalidades depende de una (1) semana con una prioridad Alta para los desarrolladores, ya que cuenta como una estructura para el correcto manejo de información. El proceso de evaluación para alcanzar el objetivo de esta historia de usuario consiste en permitir el aislamiento de información entre los dos roles del prototipo junto al cambio de estado que debe realizar el perfil Administrador al perfil Monitor dadas las condiciones o criterio del primero.

- **Historia de Usuario 2**

La distribución de las diferentes vistas que se generan en el prototipo para los dos perfiles o roles, juega un papel importante para esta historia de usuario, ya que es necesario evitar el exceso de contenido en cada una de ellas permitiendo así un fácil e intuitivo uso. Es necesario utilizar Mockups⁴ para facilitar al administrador del prototipo una aproximación real de lo que llegaría a ser una vista del prototipo y así visualizar la correcta disposición para proseguir a transcribir código fuente buscando una realidad acertada conservando el criterio y observaciones del administrador, de la mano con la respuesta de los programadores dados los alcances para crear las funcionalidades descritas en la historia de usuario.

Para esto es necesario trabajar una (1) semana con una prioridad Media para los desarrolladores en el planteamiento de las distribuciones gráficas, esto abarca también la información mostrada en cada una de ellas. La aprobación de los Mockups por parte del administrador del prototipo jugará el papel definitorio en la aceptación y ejecución en desarrollo de código fuente.

⁴ Mockup. Realizados con la herramienta Balsamiq Mockups para escritorio, Versión 2.2.24

- **Historia de Usuario 3**

Existen dos formatos de manera física con los cuales se trabaja actualmente en el programa de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira, se pueden visualizar en el Anexo **A.2** y **A.3**, en las páginas 58 y 59 respectivamente. Dichos formatos se deben de aproximar de manera digital en las vistas para que sea de fácil apreciación o seguimiento tanto para el perfil Administrador como para el perfil Monitor, facilitando una familiarización en los procesos que se realizan actualmente pero definitivamente mostrando mejoras significativas al alcance del prototipo.

Para este proceso se requiere de una (1) semana con una prioridad Baja para los desarrolladores en la búsqueda de la aproximación visual haciendo uso de distribución adecuadas según las vistas generadas previamente.

- **Historia de Usuario 4**

En la búsqueda de generar un control del perfil administrador al perfil monitor del prototipo, es necesario hacer uso de direcciones IP de los computadores usados por los monitores para crear un seguimiento con su actividad en la fecha de consulta. Cabe resaltar que es necesario mostrar solo los que se encuentran realizando actividades para no saturar de información la vista que muestre el reporte de los monitores actuales trabajando.

Para realizar esta funcionalidad es necesario usar una (1) semana con prioridad Media para los programadores ya que es necesario interactuar con las sesiones activas que se encuentren utilizando el prototipo. La evaluación de este requerimiento consta con la visualización en el perfil Administrador de todos los monitores que se encuentren haciendo uso del prototipo y que se encuentren trabajando.

- **Historia de Usuario 5**

Según los dos perfiles de usuario que se tienen en el prototipo, es necesario la correcta separación de funcionalidades con la ayuda de un inicio de sesión tradicional con correo electrónico y una contraseña que realice la bifurcación deseada. También es importante realizar una restauración de contraseña si algún usuario del prototipo la ha olvidado, como también se debe colocar un enlace con la opción de cambio de contraseña a una cuenta.

Para cumplir con estos requerimientos, es necesario usar dos (2) semanas con prioridad Alta para los programadores ya que esta separación de roles como de restauración o cambio de contraseña, desempeñan un papel fundamental en la privacidad y veracidad de la información. La evaluación del éxito de estos procesos consta en un inicio de sesión apropiado según el rol de cada usuario que utilice el prototipo como también de su restauración y modificación de su respectiva contraseña.

1.3 Programación en Pareja

La programación en pareja es una cuestión de desarrollo óptima y acertada en esta metodología de ingeniería de software ya que mientras una persona se encuentra transcribiendo código fuente tratando de usar su capacidad de ingenio y destreza haciendo énfasis en la calidad y funcionalidad, la otra persona se encuentra al tanto de que tan buena práctica se está haciendo con el método o desarrollo, junto a las variantes y consecuencias de hacer uso de esas prácticas en la transcripción de código. Esto va de la mano de las habilidades y experiencia de cada uno de los desarrolladores para este prototipo ya que si alguno tiene ventaja en front-end o en back-end, es necesario acatar su conocimiento para realizar mejores prácticas que promuevan un desarrollo más robusto e inclusive más limpio, que se verá reflejado en el despliegue del prototipo.

1.4 Reuniones Diarias

Las reuniones constantes de los desarrolladores para hacer hincapié en todos los avances, problemas, sugerencias y soluciones son de vital importancia para tratar temas que puedan colocar en riesgo las fases de desarrollo del prototipo ya que, sin la realización de estas reuniones la motivación junto al seguimiento de todo el proceso se puede tornar comprometida por la desinformación constante implicando una serie de redundancias o falta de acciones para desarrollar algo en específico. Esto abre un catálogo de medios de comunicación para los desarrolladores buscando el mejor rendimiento en pro de la maximización de tiempo de desarrollo sin olvidar estas reuniones.

2. FASE DE DISEÑO

2.1 Diseños Simples

Para promover un diseño simple hay que utilizar diferentes conceptos que ayuden a cumplir este objetivo como lo es el patrón de arquitectura de software Modelo-Vista-Controlador (MVC) para un entorno web, creando una interacción limitada entre el servidor y el usuario final donde este último se ve beneficiado por el contenido adquirido de manera eficiente. Todo esto también se ve enlazado con la cantidad de datos que se desee mostrar en cada una de las vistas prevaleciendo solo lo fundamental con un manejo fácil e intuitivo, lo que es generalmente buscado por todos los usuarios. Cada salto de página o mensaje mostrado debe estar completamente ligado a una serie de eventos que muestren una beneficencia para el usuario disminuyendo la cantidad de pasos para afrontar a la solución de alguna consulta.

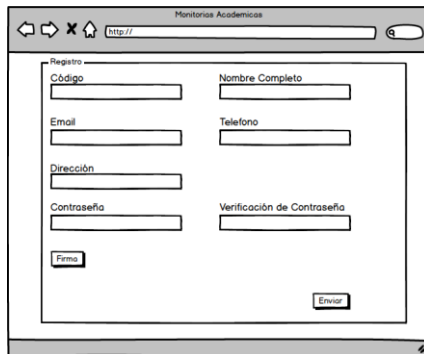
En un entorno web es necesario jugar con muchos aspectos dada la cantidad de canales de acceso a estos contenidos, contando con su conectividad que puede ir desde cable de cobre pasando por medios inalámbricos hasta satelitales. El contenido debe de ser poco pero robusto en su utilización para crear fuerte impacto y perdurar en los mercados existentes.

Teniendo en cuenta lo mencionado previamente, es crucial realizar diseños simples con ayuda de herramientas que tanto programadores como los usuarios de los programas, puedan interactuar y encontrar soluciones eficientes. Para este prototipo web, se maneja una serie de Mockups que facilitan el entendimiento de las vistas planeadas según las historias de usuario elaboradas previamente.

- **Historia de Usuario 1**

Registro de Monitores

Es crucial solicitar al monitor únicamente los campos necesarios para la identificación y vinculación en el prototipo web, para tener un filtro de información coherente con lo solicitado por el administrador del prototipo.



The image shows a browser window titled "Monitores Académicos" with a URL bar containing "http://". The main content area is a registration form with the following fields and controls:

- Registro** (Section Header)
- Código** (Text input field)
- Nombre Completo** (Text input field)
- Email** (Text input field)
- Telefono** (Text input field)
- Dirección** (Text input field)
- Contraseña** (Text input field)
- Verificación de Contraseña** (Text input field)
- Enviar** (Submit button)
- Finca** (Small button or checkbox)

Aprobación de Monitores Por Parte del Administrador

Según los datos diligenciados por el Monitor que desea registrarse, es necesario que el administrador del prototipo apruebe o rechace la solicitud bajo su criterio de una manera fácil.



- **Historia de Usuario 2**

Lista de Monitores Por Parte del Perfil Administrador

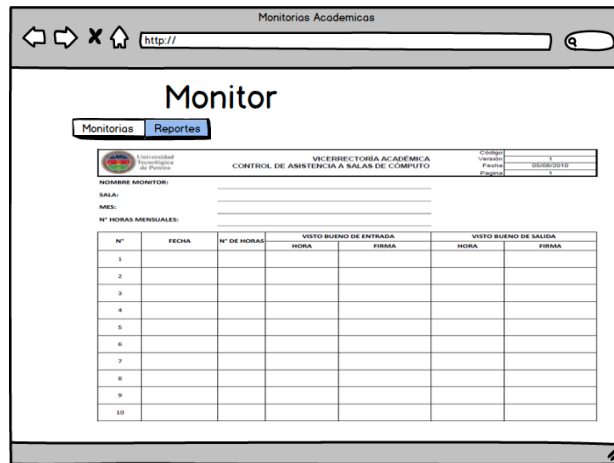
El perfil Administrador puede ver a los monitores con sus respectivas monitorias, como también debe crear, eliminar y establecer el periodo en el cual el monitor va a tener activa la monitoria para trabajar.



- **Historia de Usuario 3**

Formatos Perfil Monitor

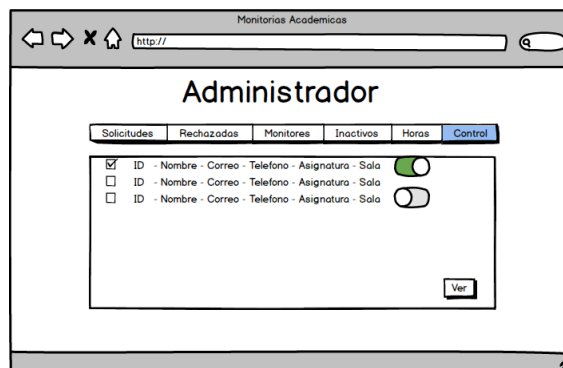
Se debe visualizar un formato digital parecido al utilizado físicamente para el control de asistencia y de monitores y estudiantes.



- **Historia de Usuario 4**

Control de Monitores

El perfil Administrador debe poder visualizar todos los monitores que se encuentren con una monitoria iniciada y trabajando en ella de manera sencilla e intuitiva.

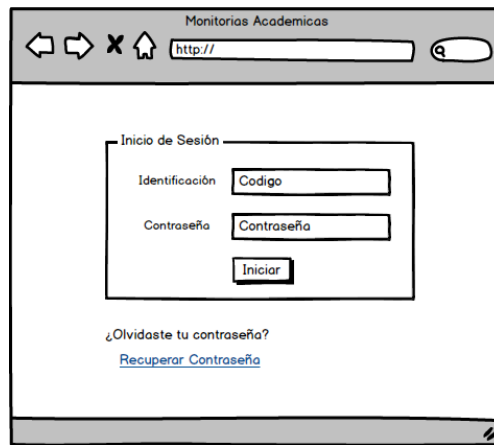


- **Historia de Usuario 5**

Iniciar Sesión y Restaurar Contraseña

El éxito de esta funcionalidad consiste en la simplicidad y que tan intuitiva puede llegar a ser inclusive para una persona que no conozca por completo la funcionalidad del prototipo, permitiendo el inicio de sesión e inclusive recuperar la contraseña si la ha

olvidado. Mostrar información relacionada en esta vista evita mostrar enlaces redundantes en barras de navegación para tratar un tema en particular.



Cambiar Contraseña

Tanto el perfil Monitor como el perfil Administrador del prototipo deben tener la funcionalidad de cambiar su contraseña con la ayuda de un link en la parte superior derecha.



2.2 Riesgos

Todos los riesgos que se pueden generar a la hora de hacer uso de esta metodología de programación extrema en conjunto del patrón de arquitectura de software MVC⁵, es la cantidad de inquietudes en el manejo de desarrollo del prototipo con el lenguaje de programación y

⁵ MVC, Modelo-Vista-Controlador

framework utilizado para crear cada una de las vistas y funcionalidades, ya que al generar una programación tan ágil y versátil, los problemas se pueden generar a la vuelta de unas pocas líneas de código imposibilitando el avance secuencial del desarrollo. Para esto sin discusión alguna, hay que trabajar arduamente en el manejo de los errores tratando de solucionar ágilmente todas las inquietudes con expertos en el tema, manuales de usuario o en foros en línea que puedan dar solución oportuna, de no ser así es muy probable que el cronograma se vea comprometido desfasando en semanas la entrega final.

Otro riesgo que se puede generar es el cambio de requerimientos constante o que tergiversen la funcionalidad del prototipo por parte del cliente o administrador, ya que podría crear un cambio que produzca una reacción en cadena en los módulos ya desarrollados que deben adecuarse para cumplir con las expectativas de los requerimientos actualizados.

Un factor de riesgo que no se debe descartar es la salud o ausencia de los desarrolladores del prototipo ya que al trabajar en pareja, la ausencia de uno podría comprometer en primera instancia la velocidad de desarrollo e intrínsecamente la calidad del mismo.

2.3 Refactorizar

La necesidad de reutilizar código libre debe consistir en la base para el éxito de la programación extrema, ya que al crear código fuente de propósito específico por desarrolladores toma demasiado tiempo debido a que se debe garantizar la homologación de panoramas a los cuales se pueda enfrentar el código desarrollado. Para contrarrestar el impacto en tiempo es necesario hacer uso de código libre y adecuarlo a las funcionalidades del prototipo, esto crea un margen potencial que se verá reflejado en dedicación puntual al prototipo y en la mejora continua a percances que se den por manejo de código fuente externo. No usar la refactorización implicaría un aumento considerable en el tiempo de entrega final del prototipo web para cumplir con los requerimientos establecidos, pero incumpliendo con las fechas establecidas previamente.

Con ayuda de un framework que posibilite la inserción de código fuente externo para agilizar el desarrollo del prototipo con respecto a calidad contra tiempo, se puede incrementar el performance del mismo implicando mejoras en procesos y resultados. Por ejemplo, se debe emplear una búsqueda de librerías de código que faciliten el Inicio de Sesión con su respectivo manejo de sesiones activas, librería para cargar imágenes o archivos multimedia, manejo de aspecto visual del prototipo y finalmente que facilite la información de datos de navegación para obtener direcciones IP para el control de estudiantes.

3. FASE DE CODIFICACION

Trabajar en una actividad productiva en el manejo del código fuente desarrollado es necesario utilizar diferentes estrategias que apunten a crear un seguimiento a las versiones que se vayan produciendo en las diferentes fases de desarrollo. Esto implica una armonía para los desarrolladores al momento trabajar con confianza y sin riesgo de tomar demasiado tiempo en buscar errores que se hayan cometido y se deben revertir para volver a una versión estable.

Para desplegar esta fase de codificación se va a utilizar una plataforma operativa basada en Windows 7 o superior, donde también se utilizarían herramientas para esta plataforma con licenciamiento libre.

Existen programas que cumplen la funcionalidad de crear versiones en una carpeta de desarrollo como lo es **TortoiseSVN Versión 1.9.3**, el cual es utilizado para el almacenamiento de los archivos de desarrollo en un servidor privado. Con esta herramienta se garantiza que los desarrolladores puedan desempeñar exploraciones en las mejoras del prototipo web y estipular criterios al momento de guardar cambios respetando las pruebas respectivas. Estas pruebas consisten en la obligación por parte de cada desarrollador en garantizar la funcionalidad del trabajo realizado tomando en cuenta las expectativas con la que se inició el desarrollo y los resultados obtenidos una vez implementado.

Al momento de ver la integridad y consistencia entre las entradas y salidas de datos, valores, consultas o vistas, se crea una versión estable con la cual cada desarrollador puede al momento de obtener datos del repositorio, estos cumplirán con las funcionalidades homologadas hasta dicha versión.

El framework utilizado para este desarrollo del prototipo web es **Rails⁶ Versión 4.2.6** con uso del lenguaje de programación **Ruby⁷ Versión 2.2.4** y el manejo de estilos HTML, CSS y JavaScript de **Bootstrap⁸ Versión 3.3.6**

La elección de este conjunto de herramientas fue basada en la demanda de funcionalidades que se deben cumplir para los requerimientos del prototipo de manera fácil, ágil y con estabilidad, ya que al momento de desarrollar dichas funcionalidades se puede encontrar documentación confiable y de fácil acceso vía web para atacar temas en específico.

Cabe resaltar que en Ruby existen Gemas⁹ por defecto para el correcto uso del framework facilitando tareas para los desarrolladores. Dichas gemas son código fuente libres que se encuentran listas para ser utilizadas en el prototipo web, como lo es para el inicio de sesión, manejo de sesiones y contraseñas de usuario con la gema “**Devise¹⁰**”.

La gema “**Carrierwave¹¹**” permite la carga de archivos para aplicaciones web en Ruby on Rails. Esta gema facilita la carga de imágenes que representarán las firmas de los estudiantes y monitores que hacen uso del prototipo web.

⁶ Rails. Descripción en línea en ‘<http://rubyonrails.org/>’

⁷ Ruby. Descripción en línea en ‘<https://www.ruby-lang.org/es/>’

⁸ Bootstrap. Descripción en línea en ‘<http://getbootstrap.com/>’

⁹ Gemas. Documentación de Ruby on Rails. Versión Web. ‘<https://rubygems.org/>’

¹⁰ Devise. Información y contenido en línea: ‘<https://github.com/plataformatec/devise>’

¹¹ Carrierwave. Información y contenido en línea: ‘<https://github.com/carrierwaveuploader/carrierwave>’

4. FASE DE PRUEBAS

La fase de pruebas consiste en conjunto de procesos que evalúan tanto el rendimiento como la funcionalidad del prototipo, teniendo en cuenta los requerimientos que se han establecido, las cuales se dividen en:

4.1 Test Unitario

Las pruebas que se realizarán se dividirán en los módulos o historias de usuario que se han planteado junto con el cliente y consiste en evaluar la funcionalidad de cada módulo introduciendo los datos requeridos y analizando los resultados.

- **Historia de Usuario 1**

1. Realizar solicitudes para crear una cuenta de monitor desde la página de inicio del prototipo.
2. Verificar desde el perfil administrador que la solicitud por parte del usuario haya sido efectuada y comprobar su estado inicial.
3. Verificar que desde el perfil Administrador se pueda aprobar o rechazar las solicitudes recibidas por parte de los usuarios y verificar que estas sean ubicadas en las secciones correspondientes.

- **Historia de Usuario 2**

1. Desde el perfil Administrador, en la sección de monitores, se asignan monitorías a los estudiantes habilitados estableciendo diferentes asignaturas y horarios, posteriormente se verificará desde los perfiles activos de monitores su correcta creación.

2. Desde el perfil Administrador en la sección de monitores se eliminarán diferentes monitorias al azar y desde los perfiles de monitores activos se verificará que las monitorias seleccionadas hayan sido correctamente eliminadas.

- **Historia de Usuario 3**

1. Se llenarán los formularios requeridos desde el perfil monitor, verificando que cumplen con los campos requeridos de los formatos que se utilizan actualmente de manera física en la Universidad Tecnológica de Pereira. Ver Anexo **A.2** y **A.3**, en las páginas 58 y 59 respectivamente.
2. Se buscarán los formularios diligenciados y se verificará la consistencia de la información suministrada.

- **Historia de Usuario 4**

1. Iniciar sesión desde varios perfiles de monitor y simular una jornada de trabajo.
2. Verificar desde el perfil administrador que los monitores que se encuentren en jornada de trabajo están activos, como también que se pueda visualizar el salón en el que se encuentran y con su respectiva monitoria asignada.

- **Historia de Usuario 5**

1. Realizar el inicio de sesión desde diferentes perfiles como monitor y desde el administrador.

2. Realizar el cambio de contraseña de los diferentes perfiles y verificar de nuevo el inicio de sesión con la nueva contraseña.
3. Recuperar la contraseña del perfil administrador y el perfil monitor.

4.2 Test de Integración

Estas pruebas consistirán en evaluar módulos que están relacionados y verificar que se acoplen correctamente.

- **Historias de Usuario 1 y 5**

1. Esto consiste en realizar diferentes solicitudes de registro de monitor desde la página de inicio e iniciar sesión desde el perfil administrador para aprobar al azar algunas de estas solicitudes. Enseguida verificar que las solicitudes aprobadas permitan el inicio de sesión de los usuarios que las realizaron y que dichas solicitudes rechazadas no permitan que se inicie la sesión.
2. Desde los perfiles de usuario aprobados realizar el cambio de contraseña y verificar.

- **Historias de Usuario 2, 3 y 4**

1. Desde el perfil Administrador se crearán diferentes monitorias para los monitores asignados, en una fecha de habilitación al azar entre la fecha actual y una fecha lejana, se verificará en cada perfil monitor la efectiva creación y el estado de la monitoria, habilitándola si dicha monitoria se creó bajo la fecha actual o inhabilitándola si la fecha es lejana.
2. Desde los perfiles de monitor se darán inicio a jornadas de trabajo verificando que se pueda iniciar correctamente una monitoria, como también que se puedan diligenciar

correctamente los formatos de asistencia de estudiantes y monitores siempre y cuando esté habilitada la monitoria.

3. Desde el perfil administrador se verificará que los monitores en jornada de trabajo puedan ser visualizados.

4.3 Test de Estrés

Las pruebas consisten en someter el prototipo a un estado crítico y consistirá en enviar la mayor cantidad de datos posibles buscando un posible fallo del sistema, además de datos incoherentes que permitan vulnerabilidades al prototipo. Para esto es necesario:

- Usar software que genere una cantidad masiva de peticiones buscan bloquear el sistema.
- Observar el comportamiento del prototipo web en el servidor durante la prueba de estrés.

4.4 Test Funcional

Las pruebas consisten en realizar una evaluación de todos los módulos del prototipo, y verificar que todos los requerimientos son cumplidos a cabalidad.

- Se simulará una jornada normal de trabajo de monitores.

4.5 Test de Aceptación

El test de aceptación consiste en lograr cumplir las expectativas del usuario, por lo que el prototipo se someterá a las pruebas del usuario administrador y los usuarios monitor.

CRONOGRAMA

ACTIVIDAD	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12
1. Análisis, creación e implementación de Base de Datos.	X											
2. Planteamiento de la distribución gráfica de las interfaces.	X											
3. Diseño y desarrollo de formularios web de los distintos formatos.		X										
4. Creación de interfaz de interacción del usuario con el prototipo.		X	X	X	X	X	X					
5. Desarrollo de sesiones de usuario.				X	X							
6. Ingreso de firma digital al sistema como verificador de contenido.					X	X	X					
7. Fase de Pruebas I								X				
8. Realización manual de usuario								X				
9. Capacitación de personal									X			
10. Equipar al personal con el prototipo web									X			
11. Fase de Pruebas II										X		
12. Despliegue del prototipo.											X	

DESARROLLO

Este desarrollo constó de una serie de pasos que ayudaron a cumplir los requerimientos que el cliente o administrador del prototipo planteó en todas las historias de usuario realizadas en el inicio de este proyecto. Cabe resaltar también que se realizaron una serie de mejoras de estos requerimientos para mejorar la experiencia en todo el entorno del prototipo como también se realizó un manual de usuario para brindar una capacitación a los usuarios dependiendo su perfil, bien sea Monitor o Administrador. Esto sin duda crea un entorno agradable de trabajo para ambos perfiles brindando las funcionalidades que satisfagan sus necesidades básicas en el manejo del prototipo web.

Para mayor claridad con respecto a la visualización de las vistas y funcionalidades del prototipo web para esta sección del documento evitando la redundancia de información e imágenes, pueden consultar el **Manual De Usuario**.

Como se mencionó previamente, los pasos que se trataron para el desarrollo del prototipo web son:

1. HISTORIA DE USUARIO

Como materia fundamental en este proceso, se tuvo en cuenta cada uno de los aspectos resaltados por el cliente o administrador del prototipo, ya que a medida que las funcionalidades atacaran las necesidades existentes en la Universidad Tecnológica de Pereira, en el programa de Ingeniería de Sistemas y Computación en el tema de monitorias académicas, se mostrarían una mejora considerable en los procesos mejorando la congruencia de datos de los trabajadores involucrados en dichas monitorias. Para esto se tuvo en cuenta también el criterio de los programadores según las valoraciones pertinentes para afrontar el desarrollo del prototipo web, ya que si los requerimientos iban más allá de sus capacidades, estos demostrarían una disminución de tiempo

de productividad en la codificación alargando el tiempo del cronograma para cumplir con las actividades estipuladas.

1.1 Reuniones con el Administrador

Se realizaron reuniones con el cliente o administrador del prototipo para mejorar el conocimiento de los desarrolladores del prototipo web para entender los procesos con los cuales las monitorias de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira trabajan, implicando una armonía de conceptos y funcionalidades. En dichas reuniones se informó por parte del administrador, la importancia de establecer un control a los usuarios bajo el perfil de Monitor, con el cual solo el perfil administrador pudiera consultar su actividad para evitar el desplazamiento físico a los diferentes laboratorios de cómputo o salas disponibles donde se estaban realizando las monitorias. También la importancia de conservar los formatos de trabajo los cuales se pueden apreciar en los anexos **A.2** y **A.3**, en las páginas 58 y 59 respectivamente.

Dichas reuniones con el administrador del prototipo se generaban cada vez que los desarrolladores encontraban dudas para la codificación de código fuente de algún módulo en particular y se realizaban en la secretaria del programa de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira, las cuales no superaban los 15 minutos de duración por sesión de consulta.

1.2 Análisis de Requerimientos

El análisis de requerimientos de cada una de las historias de usuario prevaleció para los desarrollos del prototipo, asegurar un inicio de sesión adecuado para crear una separación de información entre el perfil Administrador y Monitor.

La siguiente tabla resume los niveles de prioridad que los desarrolladores del prototipo web establecieron para las diferentes historias de usuario.

Historias de Usuario		
Número	Nombre	Prioridad
1	Gestión de Solicitudes Monitores	Media
2	Gestión de Monitorias	Alta
3	Formatos de Seguimiento a Estudiantes y Monitores	Baja
4	Seguimiento de Monitores	Media
5	Perfiles de Usuario	Alta

Estos niveles de prioridad van ligados al nivel de complejidad de desarrollo ya que, intentar de disminuir la cantidad de errores y dar funcionalidad a los módulos con resultados consistentes en condiciones normales, es intentar de solucionar las posibles entradas de datos erróneas que podrían hacer entrega de datos incoherentes o vulnerar alguna especie de filtro establecido. Por eso, la historia de usuario dos (2) y cinco (5) contiene tanto nivel de importancia.

Primero que todo la historia de usuario número dos (2) es el corazón o esencia de este proyecto, ya que la permitir la correcta gestión de las monitorias por parte del administrador del prototipo da pie para crear los reportes o consultas con los cuales el administrador puede tomar decisiones dadas las intensidades de cargas de trabajo que le da a cada monitor. Una buena práctica en el desarrollo de este módulo puede significar el éxito de la consistencia de los datos que se dispondrán pero en caso contrario puede significar un desastre en materia de solución al problema. También, la distribución de datos mostrados en la vista que contiene las funcionalidades de “Crear” y “Eliminar” monitorias fue importante para tomar el nivel de complejidad de este módulo ya que, sin una buena distribución dada una gran cantidad de monitores habilitados, se podría tornar incómodo y fatigador.

Para la historia de usuario cinco (5) su nivel de prioridad se consideró Alta por la línea sutil que debe tratar para imposibilitar o admitir mostrar un contenido de datos según los perfiles que existan en el prototipo. Realizar todo este proceso desde cero se puede tornar demorado por lo

que fue necesario hacer uso de una gema que ayuda a cumplir esta funcionalidad como lo hace “**Devise**” para gestionar toda la actividad de sesiones activas y contraseñas, la cual tuvo que ser acomodada a las prestaciones de este prototipo web con ayuda de su respectiva documentación en línea.

La historia de usuario uno (1) con su prioridad Media tuvo un impacto moderado en el desarrollo ya que al momento de aceptar o rechazar la solicitud de algún monitor por parte del perfil administrador, solo era necesario cambiar un estado en la tabla de la base de datos para poder iniciar sesión y aplicar un filtro para efectuarlo. Esta funcionalidad dio pie para agregar la posibilidad de que el perfil administrador pudiera también “**Inhabilitar**” algún monitor que estuviese aprobado, que incumpliera con algún criterio definido como también la posibilidad de “**Habilitar**” al monitor si era necesario, revirtiendo la funcionalidad anterior. Esto sin duda sirve para tener consultas resumidas y en orden para una fácil adquisición.

Según la historia de usuario cuatro (4) su prioridad se basa en la actividad de sesión en la cual cada monitor se encuentra en el momento que el perfil administrador desea consultar. Esto significa que al momento que una monitoria esté iniciada por un monitor, el administrador puede visualizar en qué monitoria se encuentra y bajo qué dirección ip se encuentra trabajando. Este proceso no significa mayor complicación para los desarrolladores y puede producir para el administrador del prototipo una ayuda fundamental para tomar decisiones.

Para la historia de usuario tres (3) la prioridad Baja se fundamenta en la distribución visual que se realiza con ayuda de código HTML y CSS para traducir de forma similar a los formatos físicos.

1.3 Cronograma

El Cronograma se realizó de manera posterior al análisis de las historias de usuario ya que según toda la información que se describió anteriormente, las fases debían ser congruentes en las funcionalidades y de manera incremental aportar en la integridad de las funciones del prototipo

web. Cabe resaltar que dicho cronograma se realizó basado en la experiencia y pericia de los desarrolladores visualizando el alcance en materia de codificación de código fuente.

El cronograma se puede apreciar en la Sección de **Cronograma** en este documento para evitar redundancia de contenido.

Se describirán cada una de las actividades de manera simplificada:

1. Hacer uso de un modelo relacional para establecer las tablas necesarias con sus columnas respectivas. Con la ayuda de esta base de datos, se da punto de partida con el desarrollo del prototipo web.
2. Este análisis se realizó con ayuda de mockups para facilitar la interacción entre el cliente y los desarrolladores para su posterior aprobación o rechazo. La simplicidad prevalece en los diseños y colores del prototipo para evitar fatiga visual.
3. El diseño es basado a un estilo de CSS definido por Bootstrap para cada uno de los formularios que se realizaron en el prototipo web.
4. Programación por parte de los desarrolladores en el framework y lenguaje de programación seleccionado buscando la aproximación del cumplimiento de requerimientos establecidos.
5. Desarrollar el inicio de sesión, cambio y recuperación de contraseña para los dos perfiles de usuarios del prototipo web como lo son el perfil Administrador y perfil Monitor.
6. Desarrollar la opción de carga de archivos multimedia con extensión .JPG y .PNG para la firma digital utilizada en los formularios requeridos en el prototipo web.
7. Fase de pruebas con respecto al servidor y los desarrolladores permitiendo simular una operación de trabajo normal del prototipo desde el servidor.

8. Realizar el manual de usuario del prototipo web con ayuda de imágenes de las funcionalidades según los perfiles de usuario.
9. Capacitar al administrador del prototipo y a un monitor en las funcionalidades del prototipo web alojado en internet.
10. Registrar algunos monitores para realizar pruebas al prototipo web.
11. Fase de pruebas simulando una monitoria normal en una sala de cómputo del programa de ingeniería de sistemas y computación de la universidad tecnológica de Pereira con un monitor del programa.
12. Inicio de operaciones a partir de la fecha establecida para hacer uso del prototipo web.

2. HERRAMIENTAS

Las herramientas son sin lugar a duda los elementos más importantes que se requieren al momento de empezar a desarrollar un aplicativo web. Actualmente en el mercado se puede encontrar una amplia gama de herramientas para escoger que permiten maximizar los recursos de un proyecto, así que teniendo en cuenta que el desarrollo del prototipo web se trabajó bajo la metodología de programación extrema se hizo necesario buscar herramientas que se ajustaran a las características de agilidad, de fácil usabilidad y de comunicación, además de tener en cuenta algunos asuntos legales como lo serían los temas de licenciamiento de software.

2.1 Selección de Herramientas

Una de las herramientas más importantes que se seleccionó fue el **framework** de trabajo. La selección de éste se hizo con la intención de buscar un entorno de desarrollo en el que se pudiesen abordar de manera más eficaz los procesos que implican los requerimientos que fueron

solicitados y desarrollados por parte tanto del cliente como de los desarrolladores del prototipo. Atendiendo a este modelo de selección se optó por utilizar un entorno de desarrollo web conocido como **Ruby on Rails**. Este permitió desarrollar el proyecto bajo un modelo de desarrollo ágil implementado una arquitectura de modelo vista controlador que envuelve una cantidad considerable de ventajas, entre ellas la de separar la lógica del diseño haciendo que el trabajo pudiese ser realizado más eficientemente por los dos desarrolladores.

Asegurar la constante comunicación y el buen trabajo en equipo fue indispensable para el desarrollo, así que atendiendo a estas necesidades se buscó una herramienta que permitiese realizar un desarrollo de manera colaborativa, lo que llevó al equipo de desarrollo a utilizar una herramienta llamada **TortoiseSVN** que gestiona el control de versiones, que sin duda es una labor invaluable que se hace al momento de realizar cambios en la programación o configuración del prototipo ya que constantemente se actualizaba la configuración y programación del prototipo, además de que está maneja licencias públicas generales lo que permite que su uso sea gratuito.

Empezar a desarrollar un prototipo web en con una nueva herramienta de trabajo no es una labor sencilla, por lo que el equipo de desarrolladores contó durante el desarrollo del proyecto con la asesoría de un experto, una persona con gran conocimiento en el manejo del Framework (**Ruby on Rails**) y gran experiencia en el desarrollo plataforma web. Con esta asesoría fue posible que los desarrolladores del prototipo implementarán las buenas prácticas de programación web requeridas por este **framework**.

Para el prototipo los desarrolladores optaron por buscar un servidor web que ofreciera una infraestructura segura, que generará confidencialidad y disponibilidad de la información. Siguiendo estos lineamientos se optó por adquirir un servidor en **DigitalOcean**¹². Es un servidor en la nube que permite crear copias de seguridad de manera periódica y automáticamente o de manera muy sencilla cada vez que se desee, además de implementar un filtro para todo el tráfico de red ofrece una característica que es la alta disponibilidad, que consiste en el periodo de disponibilidad de servicio lo que genera gran confianza al momento de utilizar el prototipo.

¹² DigitalOcean. Simple Cloud Computing, Built for Developers. En línea: '<https://www.digitalocean.com/>'

3. CODIFICACIÓN

En la etapa de codificación se implementó la metodología de programación en parejas. Lo que se buscó fue incrementar la eficiencia de desarrollo sin dejar de lado la calidad de código que se utilizó. La codificación siempre estuvo ligada a las historias de usuario desarrolladas, buscando cumplir con los requerimientos definidos entre el administrador del prototipo y los desarrolladores además se hizo un análisis general de las mismas para desarrollar un diseño relacional un la base de datos manteniendo la consistencia de la información. Durante el proceso de codificación se utilizó el framework **RubyOnRails**, que permitió llevar un proceso organizado bajo los estándares de programación, trabajando bajo una arquitectura modelo vista controlador, el framework permitió utilizar herramientas de desarrollo denominadas gemas, que implementan código que es de uso general, como la creación de formularios, inicios de sesión y registros de usuario, permitiendo desarrollar de manera muy rápida gran parte de las historias de usuario.

3.1 ¿Cómo Se Programó?

Bajo un método de programación en parejas, los desarrolladores trabajaron bajo constante comunicación. En la mayor parte de desarrollo se trabajó en un mismo espacio lo que permitió que el desarrollo de las ideas se produjera fácilmente, evitando caer en redundancias de código o errores lógicos. Además se utilizó un sistema control de versiones que controló los cambios de código que se iban presentando, esto les permite trabajar el mismo código de manera separada, ya que cuando un desarrollador realiza un cambio en el prototipo el otro puede simplemente actualizar su código, con los cambios generados, además el sistema evita el cambio de código que cualquier desarrollador sin su previo consentimiento.

3.2 Disposición de Desarrollo en Cuanto a Historia de Usuario

Los tiempos de programación están ligados directamente a las historias de usuario, ya que éstas tienen definidas unos plazo de entrega establecidos con el administrador del prototipo, por lo que

lo desarrolladores concentraban sus esfuerzos en los módulos con las prioridades más altas en razón a los tiempos de entrega.

En cuanto al orden se definió que empezar por la historia de usuario 5 que especifica que se deben crear los perfiles de usuario Administrador y Monitor que harán uso del prototipo, es lo ideal ya que crea las bases necesarias para dar desarrollo a los demás módulos del prototipo y su prioridad está definida como alta.

Siguiendo el orden de prioridades se trabajó en la historia de usuario 2 que describe la gestión de las monitorias, permitiendo crear, administrar y realizar monitorias, cumpliendo así con el núcleo funcional del prototipo.

La historia de usuario que gestiona las solicitudes de monitores, fue la siguiente en ser desarrollada, con una prioridad media, se generaron unos formularios de registro permitiendo cumplir con los requerimientos de la historia de usuario.

La siguiente historia de usuario en ser desarrollada fue la historia de usuario 4 que describe el seguimiento de las monitorias, con una prioridad media permitiendo al administrador visualizar en tiempo real las sesiones activas de monitores.

Por último se realizó la historia de usuario que describe cómo deben ser creados los formatos monitores, esto se hizo creando formularios que atendieran a los formatos físicos entregados. Ver **A.1** y **A.2**

3.3 Estrategias

Para el desarrollo del prototipo se usaron como base las historias de usuario en orden de prioridades y funcionalidades, permitiendo llevar una línea lógica de desarrollo. Haciendo uso del framework **RubyOnRails** que trabaja bajo una arquitectura modelo-vista-controlador, utilizando la estrategia de trabajar bajo dos modelos de desarrollo **Front-end** y **Back-end**.

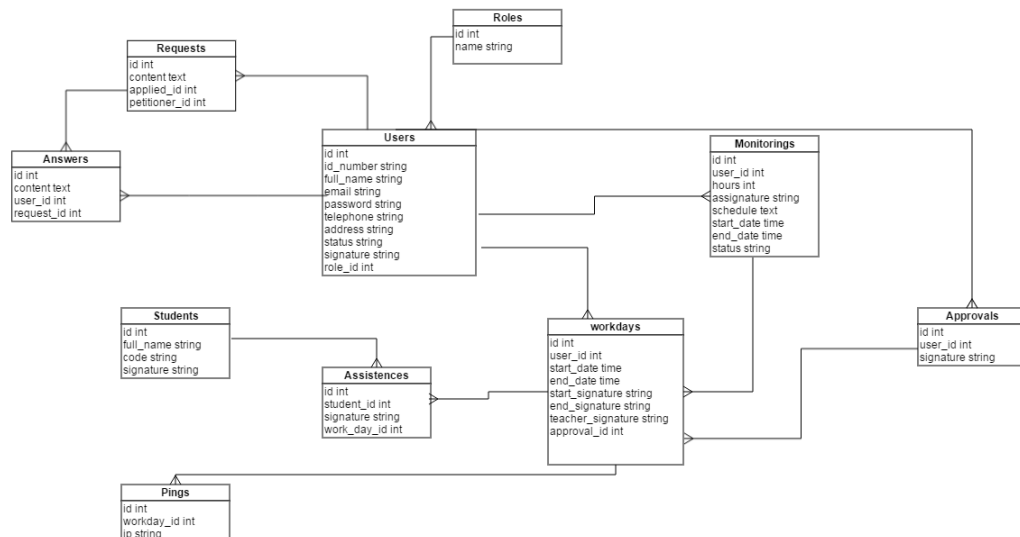
Ambos desarrolladores asumieron los roles para trabajar, **front-end** quien se ocupó de maquetar la estructura de código generando las vistas del prototipo, de codificar los diseños de hojas de estilo para crear un ambiente de trabajo óptimo y de agregar funcionalidades **Javascript** para generar una mejor interacción con el usuario, además de implementar respuestas con **JQuery** evitando recargar de manera improductiva el prototipo. Mientras el desarrollador **Back-end** se ocupó de manejar los procesos que ocurren por debajo cómo manejar el acceso a las bases de datos, esto para crear consultas de información como lo son datos de monitores, formularios de estudiantes, información de perfiles y sesiones del prototipo.

Durante el desarrollo de las historias de usuario cuando se termina de codificar cada módulo se realizan cierta cantidad de pruebas para verificar el cumplimiento de los requerimientos de cada una, lo que permitió un desarrollo más estable y una optimización de código en diferentes módulos.

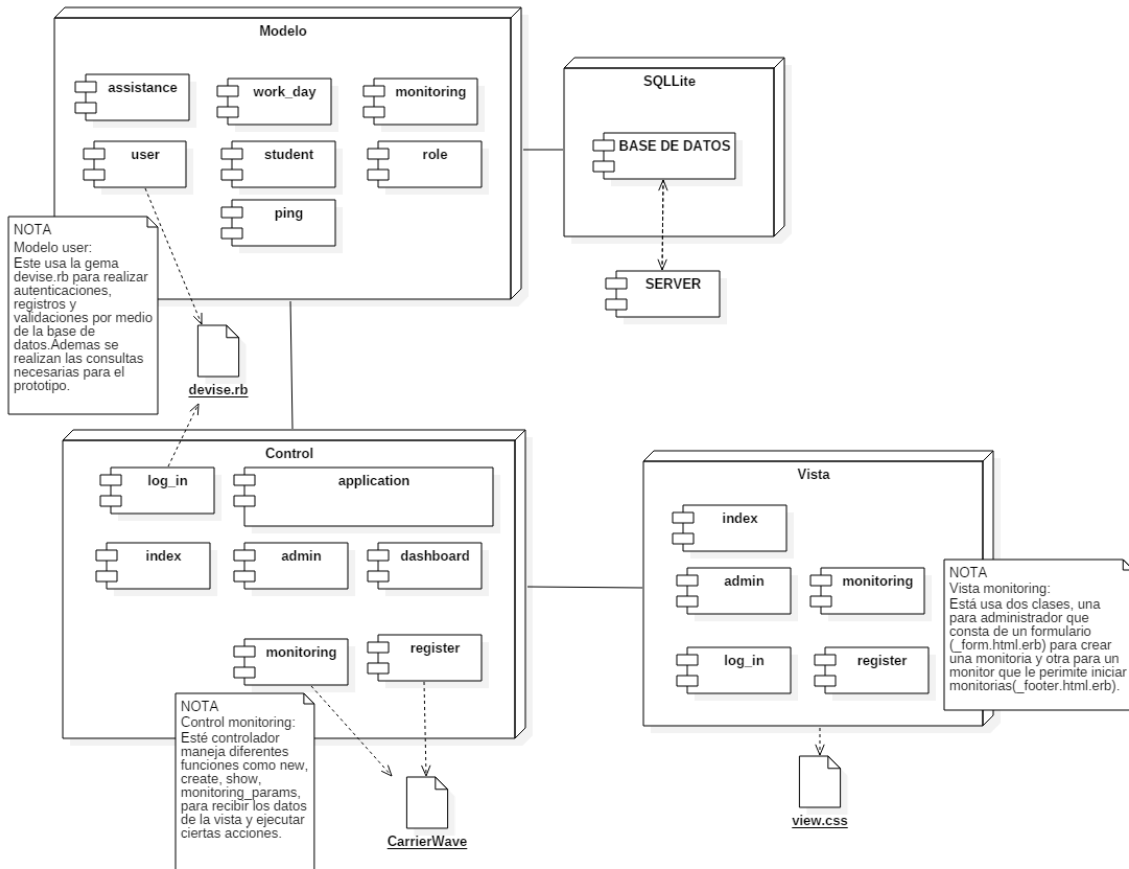
4. DIAGRAMAS

Estos diagramas ayudan a entender los componentes del prototipo web de una manera más sencilla y rápida sin tener que explorar directamente el código fuente, evitando largas horas de comprensión de contenido que puede entorpecer el conocimiento del mismo.

4.1 Base de Datos



4.2 Diagrama de Componentes



- **Modelo User**

Los modelos creados en **Ruby on Rails** trabajan bajo un concepto conocido como **Mapeo Objeto-Relacional**¹³ (ORM) heredado de la clase **ActiveRecord**¹⁴. Esto permite trabajar las bases de datos relacionales como objetos lo que facilita de manera considerable ciertas tareas sobre la base de datos como las búsquedas, la creación de nuevos datos o la actualización de los mismos.

El modelo **User** hace uso de una gema conocida como **Devise.rb**¹⁵ que ha sido creada con el propósito de facilitar ciertas funcionalidades que son usuales al momento de desarrollar una plataforma web. Dentro de estas funcionalidades encontramos autenticación mediante

¹³ Mapeo Objeto-Relacional. Definición en línea: 'http://guides.rubyonrails.org/active_record_basics.html'

¹⁴ Active Record. Definición en línea: 'http://guides.rubyonrails.org/active_record_basics.html'

¹⁵ Devise.rb. Definición y contenido en línea: '<https://github.com/plataformatec/devise>'

la base de datos, registro de datos y validaciones con el propósito de trabajar módulos como Inicio de Sesión o Registro de Usuarios. En este modelo se definen una cantidad de relaciones entre los demás modelos del prototipo que permiten establecer coherencia entre los mismos además de interactuar de manera correcta con la base de datos.

Dentro del modelo **User** se crearon las consultas necesarias a la base de datos que son requeridas por el prototipo para cumplir con los requerimientos establecidos además de ciertas funciones que permiten agregar datos a la base de datos, eliminarlos o actualizarlos según sea necesario. Es preciso indicar que todos los modelos del prototipo trabajan de manera similar cambiando solo las relaciones y las consultas requeridas.

- **Control Monitoring**

El controlador de Monitoring maneja una cantidad de funciones que son necesarias para trabajar correctamente los datos según sea solicitado por la vista de Monitorings.

Dentro de las funciones o métodos se pueden observar algunas como new, create, update y destroy, las cuales permiten crear o destruir una nueva monitoria en caso de ser solicitado bajo una vista de administrador, o de iniciar una monitoria se está es solicitada desde una vista monitor. Igualmente cuenta con una función que le permite recibir de manera filtrada ciertos parámetros enviados desde la vista lo que proporciona un nivel de seguridad mayor.

Obedeciendo a la función solicitada el controlador se encarga utilizar el modelo de Monitoring para realizar las consultas o cambios pertinentes a la base de datos y de la misma forma trabajan los demás controladores del prototipo.

- **Vista Monitoring**

La vista monitoring está segmentada en diferentes partes que tienen un propósito específico.

Monitoring-Vista de Usuario Administrador

La vista que está creada para trabajar bajo el usuario administrador permite a éste visualizar un formulario que le ayudará a enviar información como nombre, fechas de inicio y fechas de fin para la creación de una monitoria, además de enviar implícitamente los datos del monitor a quien se le crearán.

Monitoring-Vista de Usuario Monitor

Esta vista está creada para trabajar bajo un usuario monitor y lo que permite es visualizar los formularios necesarios para iniciar una monitoria activa enviando datos como firmas iniciales y firmas finales. Éstas son necesarias para cumplir con los requerimientos del prototipo.

Todas las vistas del prototipo usan hojas de estilo (**CSS¹⁶**) para manejar una presentación estándar adecuada para el prototipo.

5. REUNIONES DE AVANCE

Se realizaron una par de reuniones con el administrador del prototipo mostrando el avance significativo del prototipo web ya que sin duda alguna son necesarias las opiniones de la persona que va a hacer uso de él. Estas reuniones consistían en mostrar las funcionalidades y su interacción con el entorno del prototipo guardando el criterio de simplicidad para que todo el ambiente fuera consistentemente e intuitivo para el usuario. También fue indispensable escuchar las recomendaciones u observaciones que el administrador suministraba a los desarrolladores del prototipo para dar una aproximación a la solución que éste estaba esperando.

¹⁶ CSS. Definición, ejemplos y tutorial en línea: '<http://www.w3schools.com/css/>'

6. MANUAL DE USUARIO

El manual de usuario se realizó con el fin de brindar un soporte informativo a los usuarios que vayan a hacer uso del prototipo web para mejorar su experiencia. Dicho manual de usuario trata los dos perfiles existentes en el prototipo web como el Administrador, así como también el perfil Monitor, mostrando sus funcionalidades y lo que se puede apreciar en cada una de las vistas al interactuar con el prototipo haciendo uso de capturas de pantallas para facilitar la comprensión de las acciones que se están llevando a cabo.

Cabe resaltar que el manual de usuario únicamente muestra las funcionalidades disponibles y algunas recomendaciones de algunas actividades en específico del prototipo web pero no trata algún tipo de error. Para verificar que el manual de usuario cumpla con la finalidad de realizar un acompañamiento en el aprendizaje del prototipo web, se hicieron pruebas con un grupo de cinco personas naturales que hicieron lectura ayudando a encontrar errores en la codificación y explicaciones de las funcionalidades del prototipo web.

El manual de usuario se puede visualizar en el Anexo **A.32** en la página 82 de este documento.

ANALISIS DE RESULTADOS

Según lo establecido en la sección de **Metodología** de este documento específicamente en la fase de pruebas, se realizaron dichos procedimientos para crear unos análisis de resultados que se describirán según la clasificación previa.

1. TEST UNITARIO

1.1 Historia de Usuario 1

- Realizar solicitudes para crear una cuenta de monitor desde la página de inicio del prototipo.

La prueba resultó ser un éxito al verificar que el formulario no permite enviar la solicitud si este no es diligenciado correctamente y si se envían los datos en caso de que la información esté completa. Ver imagen en Anexos **A.4** y **A.5**, en la página 60 de este documento.

- Verificar desde el perfil administrador que la solicitud por parte del usuario haya sido efectuada y comprobar su estado inicial.

La prueba resultó ser un éxito al verificar que los datos enviados en el formulario estaban acordes a los datos visualizados por el administrador. Ver imagen en Anexo **A.6** en la página 61 de este documento.

- Verificar que desde el perfil Administrador se pueda aprobar o rechazar las solicitudes recibidas por parte de los usuarios y verificar que estas sean ubicadas en las secciones correspondientes.

La prueba resultó ser un éxito ya que el usuario Administrador permite rechazar o aprobar una solicitud, además ubica la información en la sección que le corresponde. Ver imágenes en Anexos **A.7** y **A.8** en las páginas 61 y 62 respectivamente.

1.2 Historia de Usuario 2

- Desde el perfil Administrador, en la sección de monitores, se asignan monitorias a los estudiantes habilitados estableciendo diferentes asignaturas y horarios, posteriormente se verificará desde los perfiles activos de monitores su correcta creación.

Esta prueba ha sido superada pues el administrador permitió crear diferentes monitorias en horarios al azar, además desde el perfil monitor se pudo constatar que las monitorias fueron creadas. Ver imágenes en Anexos **A.9**, **A.10**, **A.11** y **A.12** desde las páginas 62 a 64 de este documento.

- Desde el perfil Administrador en la sección de monitores se eliminarán diferentes monitorias al azar y desde los perfiles de monitores activos se verificará que las monitorias seleccionadas hayan sido correctamente eliminadas.

La prueba resultó positiva al eliminar una monitoria al azar y verificar que está no estuviera disponible para el monitor al que se le eliminó. Ver imágenes en Anexos **A.13** y **A.14** en las páginas 64 y 65 de este documento.

1.3 Historia de Usuario 3

- Se llenarán los formularios requeridos desde el perfil monitor, verificando que cumplen con los campos requeridos de los formatos que se utilizan actualmente de manera física en la Universidad Tecnológica de Pereira. Ver Anexo **A.2** y **A.3** en las páginas 58 y 59 de este documento.

Esta prueba resultó en éxito, ya que se puede observar que la plataforma permite diligenciar los datos requeridos por los formatos utilizados actualmente. Ver imagen en Anexo **A.15** en la página 65 de este documento.

- Se buscarán los formularios diligenciados y se verificará la consistencia de la información suministrada.

Esta prueba resultó ser un éxito ya que los datos diligenciados son consistentes con los datos consultados. Ver imagen en Anexo **A.16** en la página 66 de este documento.

1.4 Historia de Usuario 4

- Verificar desde el perfil administrador que los monitores que se encuentren en jornada de trabajo están activos, como también que se pueda visualizar el salón en el que se encuentran y con su respectiva monitoria asignada.

La prueba para esta historia de usuario fue exitosa ya que desde el perfil de administrador es posible visualizar a los monitores que están actualmente en una monitoria. Ver imagen en Anexo **A.17** en la página 66 de este documento.

1.5 Historia de Usuario 5

- Realizar el inicio de sesión desde diferentes perfiles como monitor y desde el administrador.
- Realizar el cambio de contraseña de los diferentes perfiles y verificar de nuevo el inicio de sesión con la nueva contraseña.
- Recuperar la contraseña del perfil administrador y el perfil monitor.

Para realizar esta primera corroboramos la contraseña actual (Ver imagen en Anexo **A.18** en la página 67 de este documento) para el usuario administrador. En la sección de cambio de contraseña se ingresa la contraseña anterior y se reemplazó por una más segura (Ver imagen en Anexo **A.19** en la página 67 de este documento).

2. TEST DE INTEGRACION

Estas pruebas consistirán en evaluar módulos que están relacionados y verificar que se acoplen correctamente.

2.1 Historia de Usuario 1 y 5

- Esto consiste en realizar diferentes solicitudes de registro de monitor desde la página de inicio e iniciar sesión desde el perfil administrador para aprobar al azar algunas de estas solicitudes. Enseguida verificar que las solicitudes aprobadas permitan el inicio de sesión de los usuarios que las realizaron y que dichas solicitudes rechazadas no permitan que se inicie la sesión. Verificar el cambio de contraseña.

Para realizar estas pruebas se corroboró que la información con la que trabaja el prototipo fuera consistente en cada uno de los módulos. Al realizar la prueba de integridad para estas historias de usuario se encontró que los datos se movían a través del prototipo de manera correcta. Ver imágenes en Anexos **A.12** pagina 64, **A.8** pagina 62, **A.5** pagina 60 y **A.6** pagina 61 de este documento.

2.2 Historias de Usuario 2, 3 y 4

- Desde el perfil Administrador se crearán diferentes monitorias para los monitores asignados, en una fecha de habilitación al azar entre la fecha actual y una fecha lejana, se verificará en cada perfil monitor la efectiva creación y el estado de la monitoria, habilitándola si dicha monitoria se creó bajo la fecha actual o inhabilitándola si la fecha es lejana.
- Desde los perfiles de monitor se darán inicio a jornadas de trabajo verificando que se pueda iniciar correctamente una monitoria, como también que se puedan diligenciar correctamente los formatos de asistencia de estudiantes y monitores siempre y cuando esté habilitada la monitoria.
- Desde el perfil administrador se verificará que los monitores en jornada de trabajo puedan ser visualizados.

Las pruebas de integración para estas historias de usuario fueron aprobadas, ya que los datos que se manejaban entre los módulos eran correctos, tanto información de inicios de sesión como los datos de los formularios y los monitores. Ver imágenes en Anexos **A.8** pagina 62, **A.9** pagina 62, **A.10** página 63, **A.11** página 63, **A.12** página 64, **A.15** página 65 y **A.17** página 66 de este documento.

3. TEST DE ESTRÉS

Las pruebas fueron desarrolladas con un computador de uso personal con las siguientes características:

- **Procesador:** Intel® Core™ i7-3630QM CPU @ 2.40GHz x 8
- **Memoria:** 7,7 GiB
- **Sistema Operativo:** elementary OS 0.3 Freya (64-bit) Basado en Ubuntu 14.04

Para mayor información de las especificaciones del computador de pruebas, ver imagen en Anexo **A.20** página 68 de este documento.

Dichas pruebas se realizaron al índice del prototipo web por el puerto 80, enviando una serie de solicitudes al servidor donde se encontraba alojado (**DigitalOcean**¹⁷). La cantidad y tiempo de estas peticiones fueron diferentes, lo cual permitió dividir esta fase de pruebas en dos secciones.

La intensidad de uso de recursos por parte del servidor donde se encontraba el prototipo web al momento de recibir la lluvia de peticiones, como el uso de Disco, Banda Ancha y CPU, se puede apreciar en los Anexos **A.24 Y A.25** en las páginas 71 y 72 de este documento.

3.1 ApacheBench

Se realizó pruebas con **ApacheBench**¹⁸ versión 2.3 para plataforma Linux, una serie de 10000 peticiones tomando un total de 569.465 segundos para realizar dichas peticiones. Según las pruebas muestra un flujo correcto para una demanda a grandes proporciones para conseguir acceso al aplicativo sin colapsar los servicios.

¹⁷ DigitalOcean. . Simple Cloud Computing, Built for Developers. En línea: '<https://www.digitalocean.com/>'

¹⁸ ApacheBench. Copyright 1996 Adam Twiss, Zeus Technology Ltd, '<http://www.zeustech.net>' , Licenced to The Apache Software Foundation, '<http://www.apache.org/>'

Los resultados de estas pruebas concretamente y al detalle, se pueden visualizar en imágenes en los Anexos **A.21** y **A.22**, página 69 y 70 de este documento.

Haciendo una comparación de pruebas de stress con la herramienta ApacheBech a dos sitios web (Ver anexo **A.26** y **A.27**, de las páginas 73 al 75 de este documento) se puede concluir que los resultados son satisfactorios, ya que la concurrencia de estos dos anexos mencionados son modestos a comparación del realizado al prototipo web a una tasa mucho mayor a las realizadas a los anexos A.26 y A.27.

3.2 Siege

Esta prueba **Siege**¹⁹ versión 3.0.5-1 para plataforma Linux, se realizó por 59.95 segundos al prototipo web procesando 1111 transacciones, con ninguna transacción fallida. También se pudo apreciar el tiempo más corto de transacción el cual fue de 0.46 segundos y la más larga que fue de 6.26 segundo. Esto muestra una respuesta oportuna a un posible ingreso masivo al prototipo web, implicando que el servicio no colapse y que el tiempo de respuesta sea aceptable según las condiciones del servidor web.

Los resultados de estas pruebas se pueden visualizar concretamente y al detalle como imagen en el Anexo **A.23** página 71 de este documento.

Se hace hincapié a la importancia sobre las transacciones exitosas ya que son primordiales para el uso del prototipo web y haciendo comparaciones con pruebas a dos páginas web, que se pueden apreciar en los Anexos **A.28** y **A.29**, páginas 76 y 77 de este documento., que no todos los sitios poseen esta cualidad tan importante y que puede poner en riesgo la integridad de los recursos en línea. También cabe resaltar que el rango de transacciones más demoradas se encuentra en un promedio esperado dada la cantidad de solicitudes que se realizaron en un lapso corto de tiempo con respecto al prototipo web desplegado en este proyecto de grado, mostrando una respuesta aceptable al acceso de información.

¹⁹ Siege. In Ubuntu Trusty. Descripción en línea, '<https://launchpad.net/ubuntu/trusty/amd64/siege/3.0.5-1>'

4. TEST FUNCIONAL

Inicialmente se realizó una prueba con cinco personas naturales en tres días al del prototipo web, haciendo uso de funcionalidades del perfil Monitor como Registro, Iniciar Monitoria, Inscribir Estudiante a Monitoria, Finalizar Monitoria y Cambiar Contraseña para visualizar errores que se pudieron eludir por parte de los desarrolladores del prototipo web.

Posteriormente el test constó de una visita a las instalaciones de la Universidad Tecnológica de Pereira, en las cuales se pretende hacer uso del prototipo web en un ambiente similar de trabajo de monitorias, con acompañamiento del Administrador del prototipo web, un monitor del programa de Ingeniera de Sistemas y Computación de la Universidad Tecnológica de Pereira, el director del trabajo de grado y los dos desarrolladores de este proyecto de grado.

Toda la descripción de ésta actividad se encuentra descrita en el acta firmada por los asistentes del proceso. Se puede apreciar en el Anexo **A.30** página 78 de este documento.

5. TEST DE ACEPTACION

Este test se realizó con el administrador del prototipo web validando la aceptación de los requerimientos tratados y desarrollados para este proyecto de grado, tratando las historias de usuario y ante todo la interacción del prototipo desplegado acomodándose a las necesidades que el Administrador expresaba en cada una de dichas historias de usuario.

La evidencia de este proceso se puede verificar en el acta firmada por los desarrolladores, director de proyecto de grado y administrador del prototipo en el Anexo **A.31** página 80 de este documento.

CONCLUSIONES

Con el desarrollo del prototipo web se intentó alcanzar el pilar de la seguridad de la información que consta de ciertos aspectos importantes como los son la integridad, confiabilidad y la disponibilidad de la información, que incrementaron en cierta medida un factor de calidad en los procesos y tareas de la Universidad Tecnológica de Pereira.

Trabajar un proyecto bajo una estructura de procesos formales como una metodología de ingeniería de software, no solo permite disminuir tiempos de desarrollo sino también los errores que podrían presentarse durante todo el proceso, además de aumentar los índices de éxito permitiendo cumplir con todos los objetivos en el tiempo establecido.

Basándonos en la filosofía de desarrollo, la cual consta de no inventar lo que ya está inventado, se realizó un trabajo de manera más eficiente y eficaz acorde con los estándares de trabajo a nivel mundial con mecanismos como la refactorización de código fuente libre disponible en internet acoplándolos a las necesidades del proyecto maximizando recursos y funcionalidades existentes.

Aprovechar la disponibilidad y la disposición de los estudiantes de la Universidad Tecnológica de Pereira para realizar proyectos de la institución es beneficioso para ambos debido a que los estudiantes además de demostrar los conocimientos que se han adquirido, afianzarán sus habilidades y ganarán experiencia y la institución obtendría proyectos que aportaran al mejoramiento de la calidad en sus procesos.

RECOMENDACIONES

Como todo desarrollo computacional que se realiza, siempre van a existir mejoras que pueden maximizar la funcionalidad y agregar contenidos que incrementen la experiencia entre el usuario final y el programa realizado. Este es el caso del prototipo web de este proyecto de grado, se pueden crear nuevas funcionalidades que aporten y optimicen procesos existentes. Por ejemplo:

1. Se puede optar por mejorar la actualización del contenido en las diferentes vistas, sincronizándose con el servidor para no hacer uso de refrescar o actualizar la página para cargar variaciones de contenido.
2. Trabajar en las firmas digitales de tal manera que no sea necesario la carga de archivos de contenido multimedia como imágenes, sino que sea una vinculación completa del prototipo web con la captura de trazos realizados con una tableta digitalizadora.
3. Incluir a profesores en la autenticación y veracidad de las diferentes monitorias. Cabe resaltar que esto puede ser poderoso siempre y cuando sea evaluado por el programa de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira, cambiando el sistema de monitorias académicas existente.
4. Filtrar información o reportes por fechas o semestres facilitando la búsqueda de información.

BIBLIOGRAFIA

L. Joyanes, "Fundamentos de Programación: Algoritmos, Estructuras de Datos", Editorial McGraw-Hill, 2010.

BECK, Kent. *Extreme programming explained: embrace change*. addison-wesley professional, 2000.

HIBBS, Curt. Ruby on Rails. *OnLamp. com*. O'Reilly Media, 2005, vol. 13.

BERUBE, David. *Practical Ruby Gems*. Apress, 2007.

FLANAGAN, David. *JavaScript: the definitive guide*. "O'Reilly Media, Inc.", 2006.

MCFARLAND, David Sawyer. *JavaScript & jQuery: the missing manual*. "O'Reilly Media, Inc.", 2011.

DUCKETT, Jon. *Accessible XHTML and CSS Web Sites: Problem-Design-Solution*. John Wiley & Sons, 2005.

Ruby, El Mejor Amigo De Un Desarrollador, disponible en línea en español:

<<https://www.ruby-lang.org/es/>>

Ruby Gems, Gemas para la comunidad Ruby. Disponible en línea en español:

<<https://rubygems.org/>>

Bootstrap. Disponible en línea:

< <http://getbootstrap.com/>>

LISTA DE ANEXOS

	Pagina
A.1 ACUERDO CONSEJO ACADEMICO	57
A.2 CONTROL ASISTENCIA ESTUDIANTES	58
A.3 CONTROL ASISTENCIA MONITORES	59
A.4 PRUEBA H1_1	60
A.5 PRUEBA H1_2	60
A.6 PRUEBA H1_3	61
A.7 PRUEBA H1_4	61
A.8 PRUEBA H1_5	62
A.9 PRUEBA H2_1	62
A.10 PRUEBA H2_2	63
A.11 PRUEBA H2_3	63
A.12 PRUEBA H2_4	64
A.13 PRUEBA H2_5	64
A.14 PRUEBA H2_6	65

A.15 PRUEBA H3_1	65
A.16 PRUEBA H3_2	66
A.17 PRUEBA H4_1	66
A.18 PRUEBA H5_2	67
A.19 PRUEBA H5_3	67
A.20 SISTEMA OPERATIVO DE PC DE PRUEBAS	68
A.21 APACHE_BENCH 01	69
A.22 APACHE_BENCH 02	70
A.23 SIEGE	71
A.24 DISK	71
A.25 BANDWIDTH-CPU	72
A.26 APACHE_BENCH APLICACIÓN 1	73
A.27 APACHE_BENCH APLICACIÓN 2	74
A.28 SIEGE APLICACIÓN 1	76
A.29 SIEGE APLICACIÓN 2	77
A.30 ACTA DE REUNION No. 1	78

A.31 ACTA DE REUNION No. 2 80

A.32 MANUAL DE USUARIO SGMUTP 82

ANEXOS

A.1 ACUERDO CONSEJO ACADÉMICO

Nombre del Autor:

Consejo Académico de la Universidad Tecnológica de Pereira.

Título del documento:

Acuerdo consejo académico número 11

Fecha de publicación:

17 de junio de 2015

Resumen

“POR MEDIO DEL CUAL SE ESTABLECE Y REGLAMENTA EL SISTEMA INSTITUCIONAL DE ACOMPAÑAMIENTO ACADÉMICO EL CONSEJO ACADÉMICO DE LA UNIVERSIDAD TECNOLÓGICA DE PEREIRA, en uso de sus atribuciones legales y estatutarias, y CONSIDERANDO


Que es deber y prioridad de la Universidad Tecnológica de Pereira garantizar el ingreso equitativo de estudiantes, su permanencia y egreso exitoso.

Que el objetivo institucional 'Cobertura con Calidad de la Oferta Educativa" busca garantizar un proceso de formación exitoso de los estudiantes.

Que se han realizado diversos diagnósticos que apuntan a identificar las asignaturas de mayor nivel de pérdida y las condiciones que constituyen riesgo académico de deserción.

Que se hace necesario fortalecer los mecanismos de acompañamiento académico a la población estudiantil”.

A.2 CONTROL ASISTENCIA ESTUDIANTES

 Universidad Tecnológica de Pereira	VICERRECTORÍA ACADÉMICA CONTROL DE ASISTENCIA	Código	
		Versión	1
		Fecha	05/08/2010
		Página	1

NOMBRE MONITOR:


ASIGNATURA:

FECHA:

N° HORAS:

N°	ESTUDIANTE	CÓDIGO	FIRMA
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			

A.3 CONTROL ASISTENCIA MONITORES

 Universidad Tecnológica de Pereira	VICERRECTORÍA ACADÉMICA CONTROL DE ASISTENCIA A SALAS DE CÓMPUTO		Código	
			Versión	1
			Fecha	05/08/2010
			Página	1

NOMBRE MONITOR: _____

SALA: _____

MES: _____

N° HORAS MENSUALES: _____

N°	FECHA	N° DE HORAS	VISTO BUENO DE ENTRADA		VISTO BUENO DE SALIDA	
			HORA	FIRMA	HORA	FIRMA
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						

A.4 PRUEBA H1_1

Registro de Monitor sgmntp

Codigo

Nombre completo Completa este campo
Juan David Herrera Martínez

Email
juand.herrera@hotmail.com

Telefono
333 33 33

Dirección
Universidad Tecnologica de Pereira

Contraseña (8 Caracteres Mínimos)

Confirmación de contraseña

Firma de estudiante
[Seleccionar archivo](#) Selena.png

Solicitar Registro

Ingeniería de Sistemas y Computación - UTP
Juan David Herrera Martínez - Sebastian Zapata Colorado
© 2018 SGMUTP / All rights reserved.

A.5 PRUEBA H1_2

Registro de Monitor sgmntp

Codigo
1088302970

Nombre completo
Juan David Herrera Martínez

Email
juand.herrera@hotmail.com

Telefono
333 33 33

Dirección
Universidad Tecnologica de Pereira

Contraseña (8 Caracteres Mínimos)

Confirmación de contraseña

Firma de estudiante
[Seleccionar archivo](#) Selena.png

Solicitar Registro

Ingeniería de Sistemas y Computación - UTP
Juan David Herrera Martínez - Sebastian Zapata Colorado
© 2018 SGMUTP / All rights reserved.

A.6 PRUEBA H1_3

The screenshot shows a web browser window with the URL 104.131.80.25. The page title is SGMUTP. The user is logged in as Juan David Herrera Martinez. The main navigation menu includes Solicitudes, Solicitudes Rechazadas, Monitores, Monitores Inactivos, Consolidado Horas, and Control. The 'Solicitudes' tab is active, displaying a table with the following data:

ID	Código	Nombre	Fecha Solicitud	Correo Electrónico	Teléfono	Dirección	
13	1088302970	Juan David Herrera Martinez	05/06/2016	juand.herrera@hotmail.com	333 33 33	Universidad Tecnologica de Pereira	Aprobar Rechazar

The footer contains the text: Ingeniería de Sistemas y Computacion - UTP, Juan David Herrera Martinez - Sebastian Zapata Colorado, and © 2016 SGMUTP / All rights reserved.

A.7 PRUEBA H1_4

The screenshot shows the same SGMUTP web application. The user is still logged in as Juan David Herrera Martinez. The 'Solicitudes Rechazadas' tab is active, displaying a table with the following data:

ID	Código	Nombre	Fecha Solicitud	Correo Electrónico	Teléfono	Dirección	
13	1088302970	Juan David Herrera Martinez	05/06/2016	juand.herrera@hotmail.com	333 33 33	Universidad Tecnologica de Pereira	Reestablecer

The footer contains the text: Ingeniería de Sistemas y Computacion - UTP, Juan David Herrera Martinez - Sebastian Zapata Colorado, and © 2016 SGMUTP / All rights reserved.

A.8 PRUEBA H1_5

The screenshot shows the SGMUTP web application interface. The browser address bar displays the URL `104.131.80.25`. The page header includes the SGMUTP logo, navigation links for 'Manual' and 'Enlaces de Interes', and the user name 'Juan David Herrera Martinez'. The main content area features a tabbed interface with 'Monitores' selected. Below the tabs is a table with the following data:

Código	Nombre	Correo Electrónico	Teléfono	Monitorias	Monitoria	Monitor
1088302970	Juan David Herrera Martinez	juand.herrera@hotmail.com	333 33 33		Crear	Inhabilitar

The footer of the page contains the text: 'Ingeniería de Sistemas y Computacion - UTP', 'Juan David Herrera Martinez - Sebastian Zapata Colorado', and '© 2016 SGMUTP / All rights reserved.'

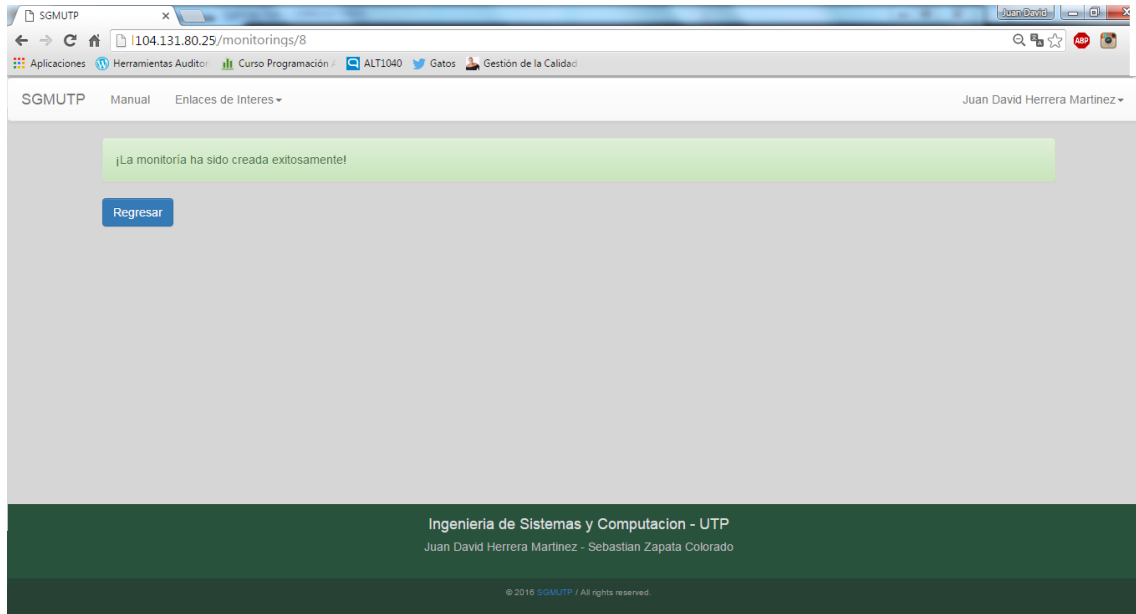
A.9 PRUEBA H2_1

The screenshot shows the SGMUTP web application interface with a modal form titled 'Crear Monitoria SGMUTP' open. The form contains the following fields and elements:

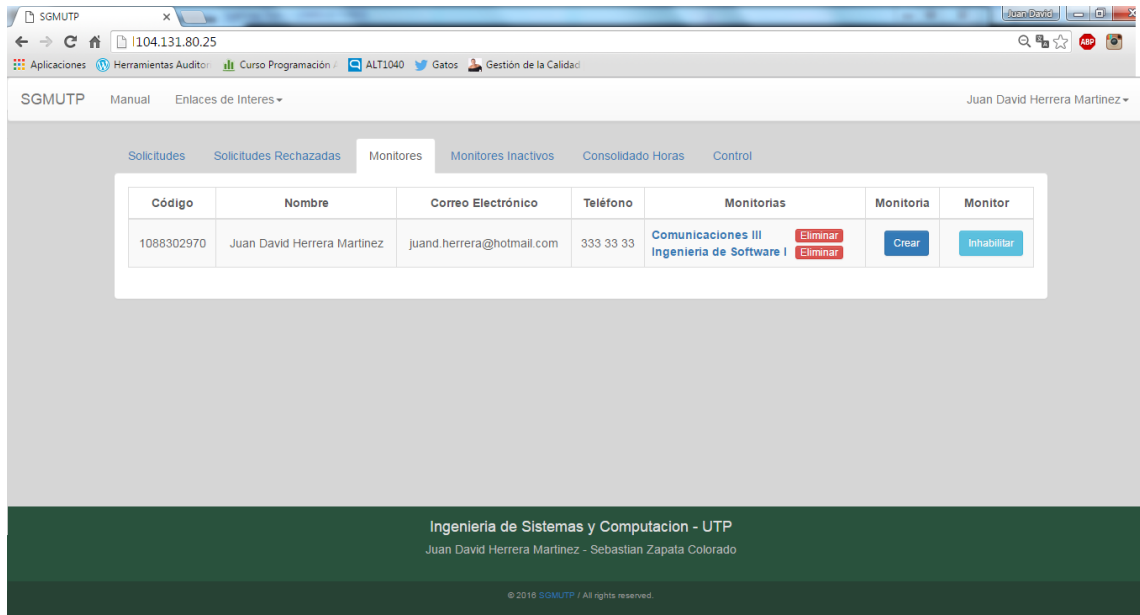
- Nombre de Monitoria:** A text input field containing 'Comunicaciones III'.
- Fecha de inicio:** A date input field containing '05/05/2016'.
- Fecha de finalización:** A date picker showing 'mayo de 2016' with a calendar view. The calendar highlights the date '13'.

The footer of the page contains the text: 'Ingeniería de Sistemas y Computacion - UTP', 'Juan David Herrera Martinez - Sebastian Zapata Colorado', and '© 2016 SGMUTP / All rights reserved.'

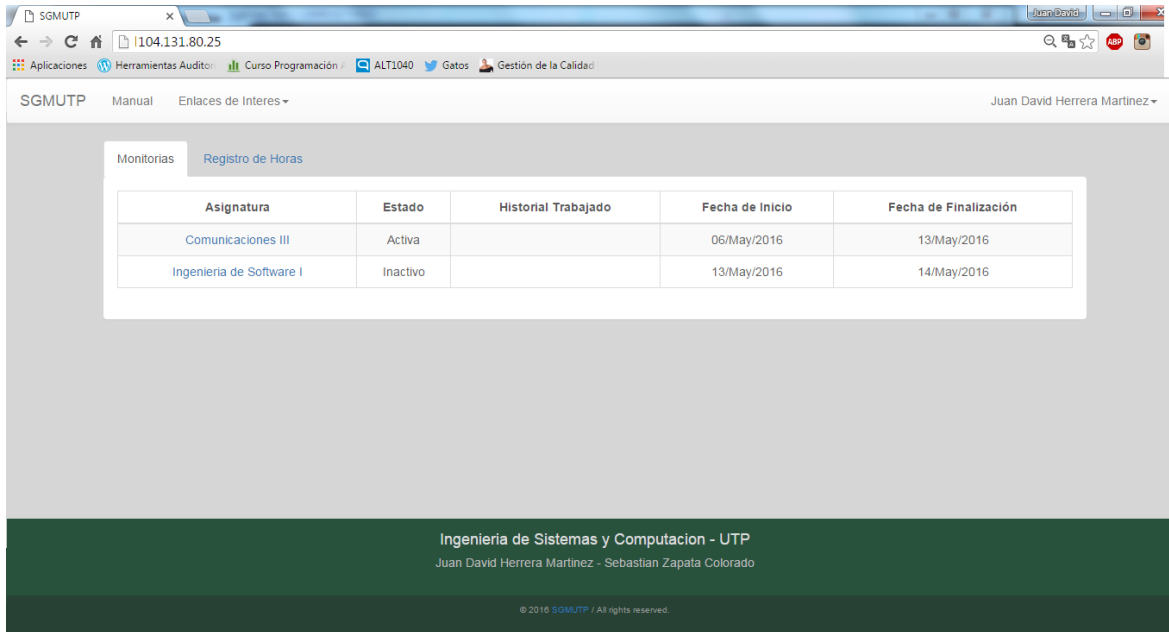
A.10 PRUEBA H2_2



A.11 PRUEBA H2_3



A.12 PRUEBA H2_4



SGMUTP Manual Enlaces de Interés Juan David Herrera Martínez

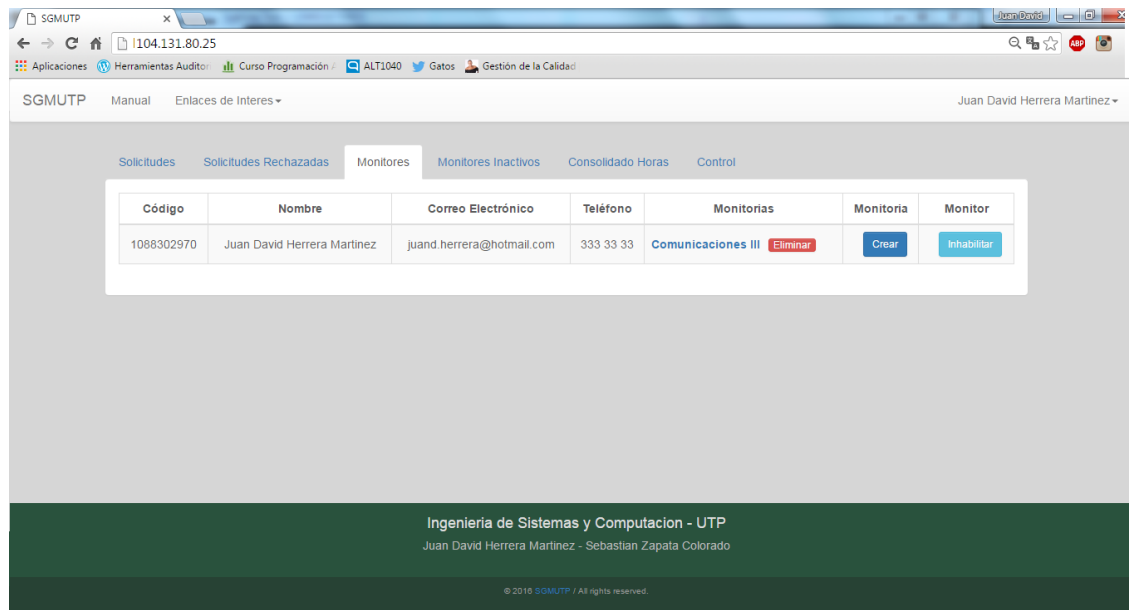
Monitorias Registro de Horas

Asignatura	Estado	Historial Trabajado	Fecha de Inicio	Fecha de Finalización
Comunicaciones III	Activa		06/May/2016	13/May/2016
Ingeniería de Software I	Inactivo		13/May/2016	14/May/2016

Ingeniería de Sistemas y Computacion - UTP
Juan David Herrera Martínez - Sebastian Zapata Colorado

© 2016 SGMUTP / All rights reserved.

A.13 PRUEBA H2_5



SGMUTP Manual Enlaces de Interés Juan David Herrera Martínez

Solicitudes Solicitudes Rechazadas Monitores Monitores Inactivos Consolidado Horas Control

Código	Nombre	Correo Electrónico	Teléfono	Monitorias	Monitoria	Monitor
1088302970	Juan David Herrera Martínez	juand.herrera@hotmail.com	333 33 33	Comunicaciones III Eliminar	Crear	Inhabilitar

Ingeniería de Sistemas y Computacion - UTP
Juan David Herrera Martínez - Sebastian Zapata Colorado

© 2016 SGMUTP / All rights reserved.

A.14 PRUEBA H2_6

SGMUTP Manual Enlaces de Interés Juan David Herrera Martinez

Monitorias Registro de Horas

Asignatura	Estado	Historial Trabajado	Fecha de Inicio	Fecha de Finalización
Comunicaciones III	Activa		06/May/2016	13/May/2016

Ingeniería de Sistemas y Computacion - UTP
Juan David Herrera Martinez - Sebastian Zapata Colorado
© 2016 SGMUTP / All rights reserved.

A.15 PRUEBA H3_1

SGMUTP Manual Enlaces de Interés Juan David Herrera Martinez

Nombre monitor Juan David Herrera Martinez

Fecha 06/05/2016

Asignatura Comunicaciones III

Código	Estudiante	Firma
<input type="text" value="Código"/>		<input type="button" value="Seleccionar archivo"/> Ningún archivo seleccionado

Número	Estudiante	Codigo	Firma
1	sebastian	1088302207	
2	Juan David	1088302970	

Ingeniería de Sistemas y Computacion - UTP
Juan David Herrera Martinez - Sebastian Zapata Colorado
© 2016 SGMUTP / All rights reserved.

A.16 PRUEBA H3_2

The screenshot shows a web browser window with the URL `104.131.80.25/student_assitences?workday_id=2`. The page header includes the SGMUTP logo, navigation links (Manual, Enlaces de Interes), and the user name Juan David Herrera Martinez. The main content area features the logo of Universidad Tecnológica de Pereira and the text "VICERECTORIA ACADEMICA CONTROL DE ASISTENCIA". Below this is a table with the following data:

ID	Código	Estudiante	Firma	Firma Original
2	1088302207	sebastian		
3	1088302970	Juan David		

The footer of the page contains the text "Ingeniería de Sistemas y Computacion - UTP" and "Juan David Herrera Martinez - Sebastian Zapata Colorado", along with a copyright notice "© 2016 SGMUTP / All rights reserved."

A.17 PRUEBA H4_1

The screenshot shows a web browser window with the URL `104.131.80.25`. The page header includes the SGMUTP logo, navigation links (Manual, Enlaces de Interes), and the user name Juan David Herrera Martinez. The main content area features a navigation menu with options: Solicitudes, Solicitudes Rechazadas, Monitores, Monitores Inactivos, Consolidado Horas, and Control. The "Control" option is selected, and the page title is "Control de Monitores". Below this is a table with the following data:

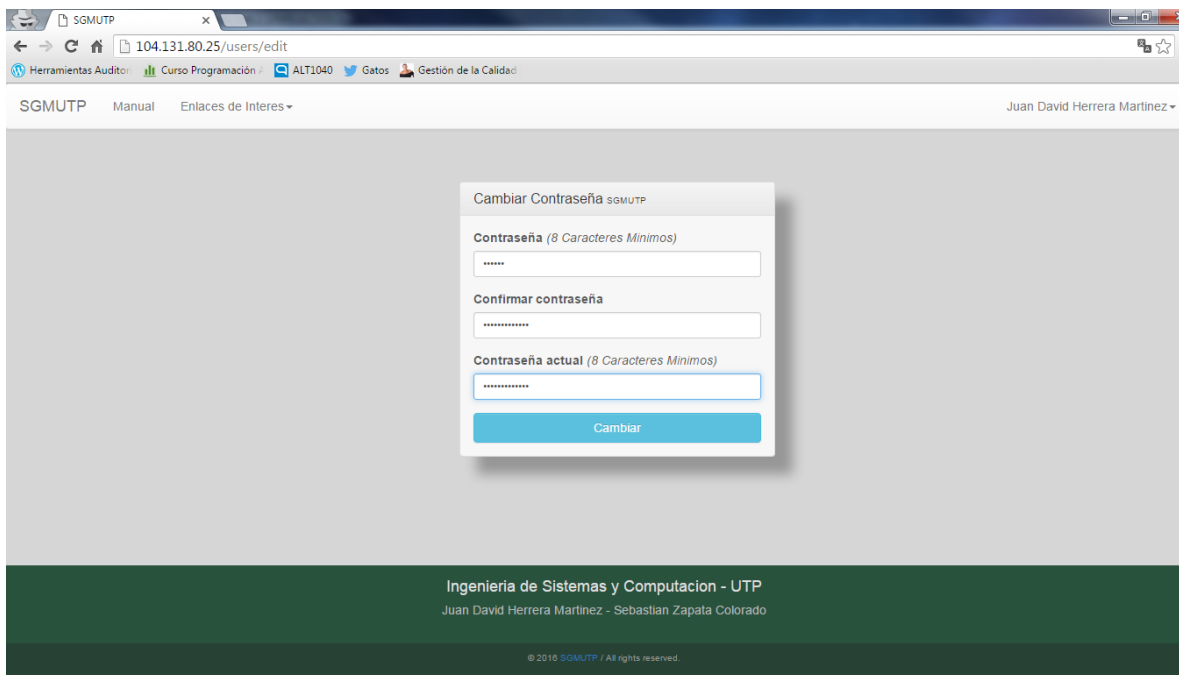
Nombre	Asignatura	Dirección IP	Sala	Estado
Juan David Herrera Martinez	Comunicaciones III	127.0.0.1		Sesión Activa

The footer of the page contains the text "Ingeniería de Sistemas y Computacion - UTP" and "Juan David Herrera Martinez - Sebastian Zapata Colorado", along with a copyright notice "© 2016 SGMUTP / All rights reserved."

A.18 PRUEBA H5_2

```
user.create(email: "juand.herrera@utp.edu.co", password: "abc123", password_confirmation: "abc123", full_name: "Juan David Herrera Martinez", role: admin)
```

A.19 PRUEBA H5_3



The screenshot shows a web browser window with the URL `104.131.80.25/users/edit`. The page title is "SGMUTP" and the user is logged in as "Juan David Herrera Martinez". The main content area displays a modal form titled "Cambiar Contraseña SGMUTP". The form contains three input fields: "Contraseña (8 Caracteres Minimos)", "Confirmar contraseña", and "Contraseña actual (8 Caracteres Minimos)". A blue "Cambiar" button is located at the bottom of the form. The footer of the page includes the text "Ingeniería de Sistemas y Computación - UTP", "Juan David Herrera Martinez - Sebastian Zapata Colorado", and "© 2016 SGMUTP / All rights reserved."

A.20 SISTEMA OPERATIVO DE PC DE PRUEBAS



The screenshot shows the 'About' window of elementary OS 0.3 Freya (64-bit). The window title is 'Acerca de' and it includes a search bar for options. The main content area displays the elementary OS logo, the version 'elementary OS 0.3 Freya (64-bit)', and the base system 'Basado en Ubuntu 14.04'. A link to the 'Sitio web' is provided. Below this, the 'Hardware' section lists the following specifications:

- Procesador: Intel® Core™ i7-3630QM CPU @ 2.40GHz × 8
- Memoria: 7,7 GiB
- Gráficos: Intel Corporation 3rd Gen Core processor Graphics Controller (rev 09)
NVIDIA Corporation GK107M [GeForce GT 740M] (rev a1)
- Almacenamiento: 112,9 GB

At the bottom of the window, there are three buttons: a help button (question mark), 'Sugerir traducciones', 'Informar de un problema', and 'Buscar actualizaciones'.

A.21 APACHE_BENCH 01

```
x
+ x parcial: ab
us3r@us3r-N46VB:~/Documentos/parcial$ ab -n 10000 -c 100 -g grafica.dat
a http://104.131.80.25:80/users/sign_in
This is ApacheBench, Version 2.3 <$Revision: 1528965 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net
/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking 104.131.80.25 (be patient)
Completed 1000 requests
Completed 2000 requests
Completed 3000 requests
Completed 4000 requests
Completed 5000 requests
Completed 6000 requests
Completed 7000 requests
Completed 8000 requests
Completed 9000 requests
Completed 10000 requests
Finished 10000 requests

Server Software:
Server Hostname:      104.131.80.25
Server Port:         80

Document Path:       /users/sign_in
Document Length:     8436 bytes

Concurrency Level:   100
Time taken for tests: 569.465 seconds
Complete requests:   10000
Failed requests:     0
Total transferred:   90930000 bytes
HTML transferred:    84360000 bytes
Requests per second: 17.56 [#/sec] (mean)
Time per request:    5694.647 [ms] (mean)
Time per request:    56.946 [ms] (mean, across all concurrent requests)
Transfer rate:       155.93 [Kbytes/sec] received
```


A.22 APACHE_BENCH 02

```
x
+ x parcial: ab
Finished 10000 requests

Server Software:
Server Hostname: 104.131.80.25
Server Port: 80

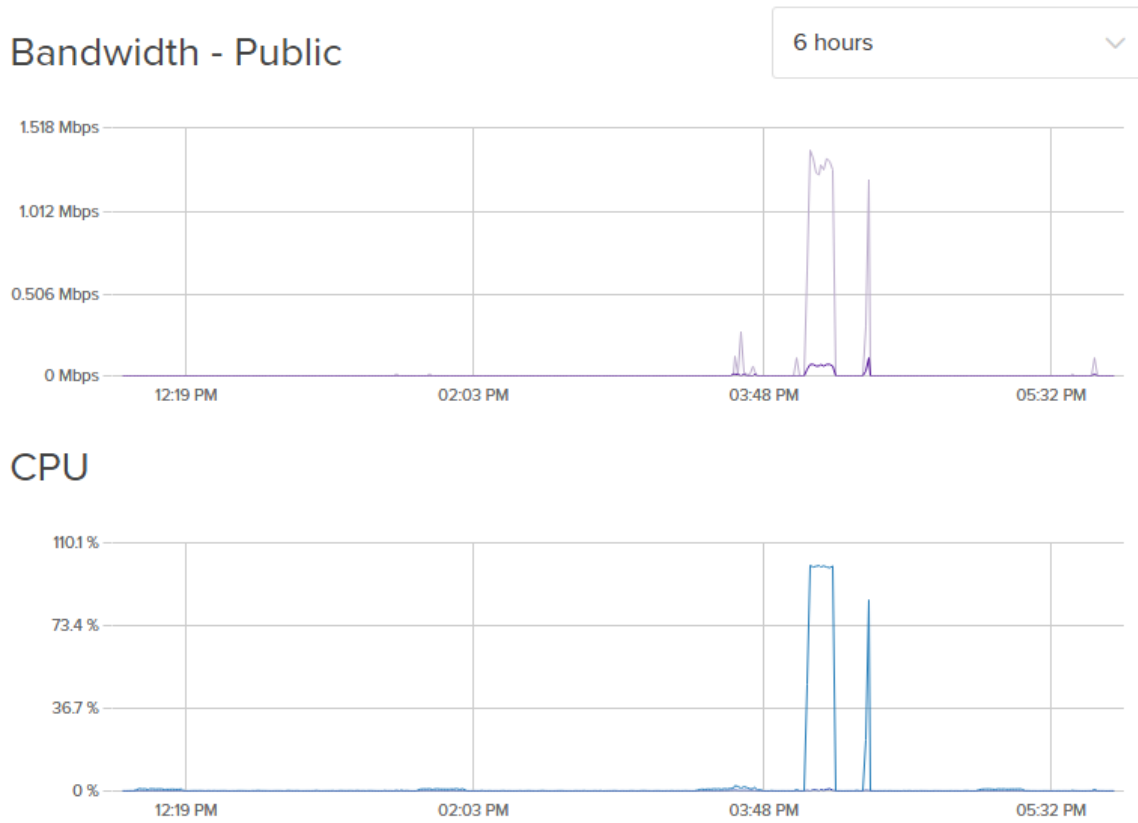
Document Path: /users/sign_in
Document Length: 8436 bytes

Concurrency Level: 100
Time taken for tests: 569.465 seconds
Complete requests: 10000
Failed requests: 0
Total transferred: 90930000 bytes
HTML transferred: 84360000 bytes
Requests per second: 17.56 [#/sec] (mean)
Time per request: 5694.647 [ms] (mean)
Time per request: 56.946 [ms] (mean, across all concurrent requests)
Transfer rate: 155.93 [Kbytes/sec] received

Connection Times (ms)
      min  mean[+/-sd] median  max
Connect:  139  193  62.0   186  1203
Processing: 353 5476 628.2  5411  7527
Waiting:  346 5470 628.1  5404  7524
Total:    517 5669 627.6  5603  7803

Percentage of the requests served within a certain time (ms)
 50%    5603
 66%    5856
 75%    6033
 80%    6183
 90%    6499
 95%    6711
 98%    7015
 99%    7240
100%    7803 (longest request)
```


A.25 BANDWIDTH-CPU



A.26 APACHE_BENCH APLICACIÓN 1

```
Server Software:      Apache
Server Hostname:     nuestra-app.com
Server Port:         80

Document Path:       /
Document Length:     5239 bytes

Concurrency Level:   20
Time taken for tests: 17.604 seconds
Complete requests:   1000
Failed requests:     0
Write errors:        0
Total transferred:   5992584 bytes
HTML transferred:    5248358 bytes
Requests per second: 56.81 [#/sec] (mean)
Time per request:    352.072 [ms] (mean)
Time per request:    17.604 [ms] (mean, across all concurrent requests)
Transfer rate:       332.44 [Kbytes/sec] received

Connection Times (ms)
  min  mean[+/-sd] median  max
Connect:    24   70  62.3   52   403
Processing:  61  257 315.8  153  2953
Waiting:    34  106  94.5   81  1253
Total:      90  328 321.8  221  3039
```

A.27 APACHE_BENCH APLICACIÓN 2

```
This is ApacheBench, Version 2.3 <$Revision: 655654 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking www.genbetadev.com (be patient)....done

Server Software:      nginx/0.7.65
Server Hostname:      www.genbetadev.com
Server Port:          80

Document Path:        /
Document Length:      117491 bytes

Concurrency Level:    10
Time taken for tests: 67.020 seconds
Complete requests:    100
Failed requests:      0
Write errors:         0
Total transferred:    11794800 bytes
HTML transferred:     11749100 bytes
Requests per second:  1.49 [#/sec] (mean)
Time per request:     6702.019 [ms] (mean)
Time per request:     670.202 [ms] (mean, across all concurrent requests)
Transfer rate:        171.86 [Kbytes/sec] received
```

```
Requests per second: 1.49 [# /sec] (mean)
Time per request: 6702.019 [ms] (mean)
Time per request: 670.202 [ms] (mean, across all concurrent requests)
Transfer rate: 171.86 [Kbytes/sec] received
```

Connection Times (ms)

	min	mean[+/-sd]	median	max
Connect:	59	277 377.3	107	1117
Processing:	779	6216 5759.9	4791	37478
Waiting:	110	328 552.0	176	3922
Total:	876	6493 5822.2	5380	38521

Percentage of the requests served within a certain time (ms)

50%	5380
66%	6488
75%	7425
80%	8405
90%	11955
95%	15668
98%	30385
99%	38521
100%	38521 (longest request)

A.28 SIEGE APLICACIÓN 1

```
$ siege -c50 -d10 -t3M http://some.application.com

** SIEGE 2.72
** Preparing 50 concurrent users for battle.
The server is now under siege...[alert] socket: 100081664 select timed
out: Operation timed out

Lifting the server siege...      done.

Transactions:          1398 hits
Availability:          99.93 %
Elapsed time:          179.94 secs
Data transferred:     4.70 MB
Response time:         1.21 secs
Transaction rate:      7.77 trans/sec
Throughput:            0.03 MB/sec
Concurrency:           9.40
Successful transactions: 1398
Failed transactions:   1
Longest transaction:   29.47
Shortest transaction:  0.10
```

A.29 SIEGE APLICACIÓN 2

```
Lifting the server siege... done.  
  
Transactions:                385 hits  
Availability:                100.00 %  
Elapsed time:                76.02 secs  
Data transferred:           0.84 MB  
Response time:              2.40 secs  
Transaction rate:           5.06 trans/sec  
Throughput:                 0.01 MB/sec  
Concurrency:                12.18  
Successful transactions:    385  
Failed transactions:        0  
Longest transaction:        5.73  
Shortest transaction:       0.31
```


A.30 ACTA DE REUNION No. 1



Universidad
Tecnológica
de Pereira

ACTA DE REUNIÓN No. 1
PROCESO: SGMUTP Pruebas

Página 1 de 2
17 de Mayo 2016

Tema: Test Funcional

Lugar: Universidad Tecnológica de Pereira - Edificio 3 Salón 201

Hora de Inicio: 04:30 PM

Hora de Finalización: 05:15 PM

Objetivo de la Reunión:

Realizar el test funcional del prototipo web SGMUTP, simulando una jornada de Monitoria junto a la capacitación para un Monitor habilitado y al Administrador.

Asistentes:

Juan David Herrera Martínez

Sebastian Zapata Colorado

Carlos Alberto Ocampo Sepúlveda

Martha Lucy Estrella Ocampo

Sebastián Vélez Ramírez

Desarrollador

Desarrollador

Director de Proyecto

Secretaria del programa

Monitor

DESARROLLO DE LA REUNIÓN

1. Módulos a Evaluar.

Se evaluaron las historias de usuario establecidas al inicio del proyecto por parte del administrador del prototipo el cual es Martha Lucy Estrella:

1. Gestión de solicitudes Monitores
2. Gestión de Monitorias
3. Formatos de seguimiento a estudiantes y monitores
4. Seguimiento de monitores
5. Perfiles de usuario

2. Prueba a los Módulos.

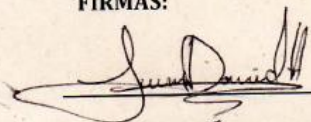
Para evaluar estos puntos es necesario la colaboración de un Monitor habilitado en el programa de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira para hacer su registro en el prototipo web

Simular un ambiente de trabajo del monitor en una monitoria, como también del administrador capacitándolos en las funcionalidades del prototipo web.



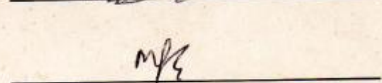
OBSERVACIONES:

FIRMAS:









Juan David Herrera Martinez

Sebastian Zapata Colorado

Carlos Alberto Ocampo Sepilveda

Sebastian Velez Ramirez

Mertha Lucy Estrella

A.31 ACTA DE REUNION No. 2



Universidad
Tecnológica
de Pereira

ACTA DE REUNIÓN No. 2
PROCESO: SGMUTP Pruebas

Página 1 de 2
18 de Mayo 2016

Tema: Test de Aceptación

Lugar: Universidad Tecnológica de Pereira - Edificio 3 Salón 201

Hora de Inicio: 03:00 PM

Hora de Finalización: 03:30 PM

Objetivo de la Reunión:

Aprobación del prototipo web por parte del administrador según los requerimientos acordados.

Asistentes:

Juan David Herrera Martínez

Sebastian Zapata Colorado

Carlos Alberto Ocampo Sepúlveda

Martha Lucy Estrella Ocampo

Desarrollador

Desarrollador

Director de Proyecto

Secretaria del programa

DESARROLLO DE LA REUNIÓN

1. Módulos a Evaluar.

Atendiendo a los requerimientos, se evalúa la funcionalidad integral del prototipo web por parte del administrador buscando el alcance de sus expectativas.

2. Prueba a los Módulos.

Yo Martha Lucy Estrella Ocampo Secretaria del programa de Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira, apruebo el prototipo web "Sistema de Gestión de Información de Monitorías Académicas para el Programa de Ingeniería de Sistemas y Computación de UTP", el día 18 del mes de 05 del año 2016 en la ciudad de Pereira, Risaralda, Colombia.



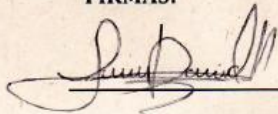
OBSERVACIONES:

Los desarrolladores dejan la base de datos sin contenido de las pruebas como registros temporales de monitores y administradores.

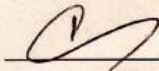
Se crea el Usuario Administrador con cuenta y contraseña definida por Martha Lucy Estrella Ocampo.

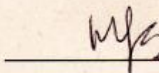
Se establece un Administrador de contingencia por alguna emergencia que acontezca, la cual se hace responsable Martha Lucy Estrella Ocampo.

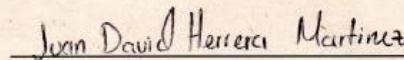
FIRMAS:

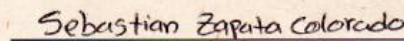


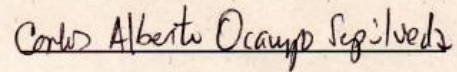


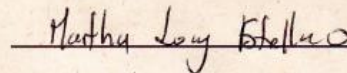












A.32 MANUAL DE USUARIO SGMUTP

MANUAL DE USUARIO

PROTOTIPO WEB

SISTEMA DE GESTION DE INFORMACION DE MONITORIAS ACADEMICAS PARA EL
PROGRAMA DE INGENIRIA DE SISTEMAS Y COMPUTACION DE UTP

JUAN DAVID HERRERA MARTÍNEZ

SEBASTIAN ZAPATA COLORADO

INGENIERIA DE SISTEMAS Y COMPUTACION

UNIVERSIDAD TECNOLOGICA DE PEREIRA

2016

CONTENIDO

	Pagina
1. INTRODUCCION	1
2. PERFIL MONITOR	1
2.1 Acceso Al Prototipo	1
2.2 Registrarse	2
2.3 Iniciar Sesión	3
2.4 Dashboard	3
2.5 Pestaña Monitorias	4
2.5.1 Iniciar Monitorias	5
2.5.2 Ingresar Estudiantes	6
2.5.3 Registrar Estudiantes	8
2.5.4 Finalizar Monitoria	9
2.6 Pestaña Registro De Horas	11
2.6.1 Control de Asistencia de Estudiantes	12
2.7 Cerrar Sesión	13
2.8 Cambiar Contraseña	14
3. PERFIL ADMINISTRADOR	15
3.1 Acceso Al Prototipo	15
3.2 Iniciar Sesión	15
3.2.1 Cerrar Sesión	16
3.2.2 Cambiar Contraseña	17
3.3 Dashboard	17
3.4 Pestaña Solicitudes	18
3.4.1 Rechazar Solicitud	19
3.4.2 Aprobar Solicitud	19

3.5 Pestaña Solicitudes Rechazadas	19
3.5.1 Reestablecer Solicitud	20
3.6 Pestaña Monitores	20
3.6.1 Inhabilitar Monitor	21
3.6.2 Eliminar Monitoria	21
3.6.3 Crear Monitoria	21
3.6.4 Ver Monitorias del Monitor	23
3.6.5 Control de Asistencia de Estudiantes	24
3.7 Pestaña Monitores Inactivos	25
3.7.1 Habilitar Monitor	26
3.8 Pestaña Consolidado de Horas	26
3.8.1 Ver Historial de Horas de Monitores	26
3.9 Pestaña Control	27
3.9.1 Ver Pings de Monitores	28

1. INTRODUCCION

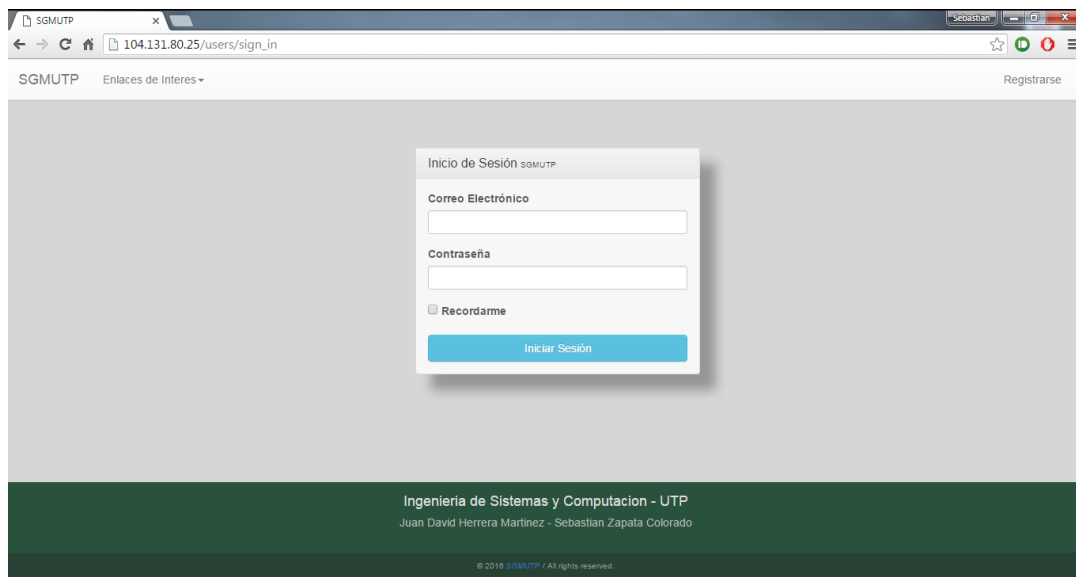
Este manual de usuario se tratará los conceptos y funcionalidades básicas para el manejo del prototipo web dados los dos perfiles (Administrador y Monitor) que se encuentran activos y en capacidad de ejecutar acciones para la interacción de actividades.

2. PERFIL MONITOR

2.1 Acceso al Prototipo

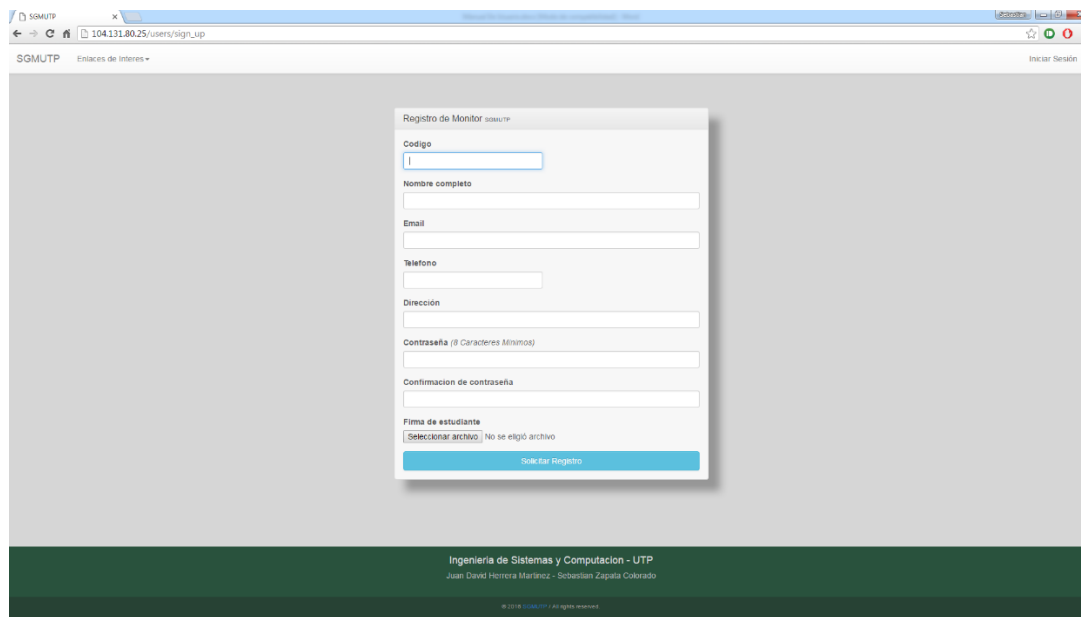
Para hacer uso de todas las funcionalidades del prototipo web, se debe acceder desde cualquier navegador de internet de última generación a la siguiente dirección URL en la barra de navegación: 104.131.80.25

Esta dirección lo guiará a la página inicial o Index donde le mostrará el formulario para iniciar sesión o registrarse si aún no tiene una cuenta.



2.2 Registrarse

Un monitor que desee iniciar sesión es necesario tener una cuenta activada por el Administrador del prototipo. Para esto, es necesario dar click en “**Registrarse**” en la barra de navegación de la página web en la parte superior, en el extremo derecho de la página web se carga una vista donde se despliega un formulario con los datos que son necesarios diligenciar para efectuar un proceso de registro.



The image shows a web browser window displaying the registration form for SGMUTP. The browser's address bar shows the URL "104.131.80.25/users/sign_up". The page title is "SGMUTP" and the navigation bar includes "Inicio Sesion". The registration form, titled "Registro de Monitor SGMUTP", contains the following fields: "Codigo" (with a dropdown menu), "Nombre completo", "Email", "Telefono", "Dirección", "Contraseña (8 Caracteres Mínimos)", "Confirmación de contraseña", and "Firma de estudiante" (with a file selection button labeled "Seleccionar archivo" and the text "No se eligió archivo"). A blue button labeled "Solicitar Registro" is located at the bottom of the form. The footer of the page includes the text "Ingeniería de Sistemas y Computación - UTP", "Juan David Herrera Martínez - Sebastian Zapata Colorado", and "© 2018 SGMUTP. All rights reserved."

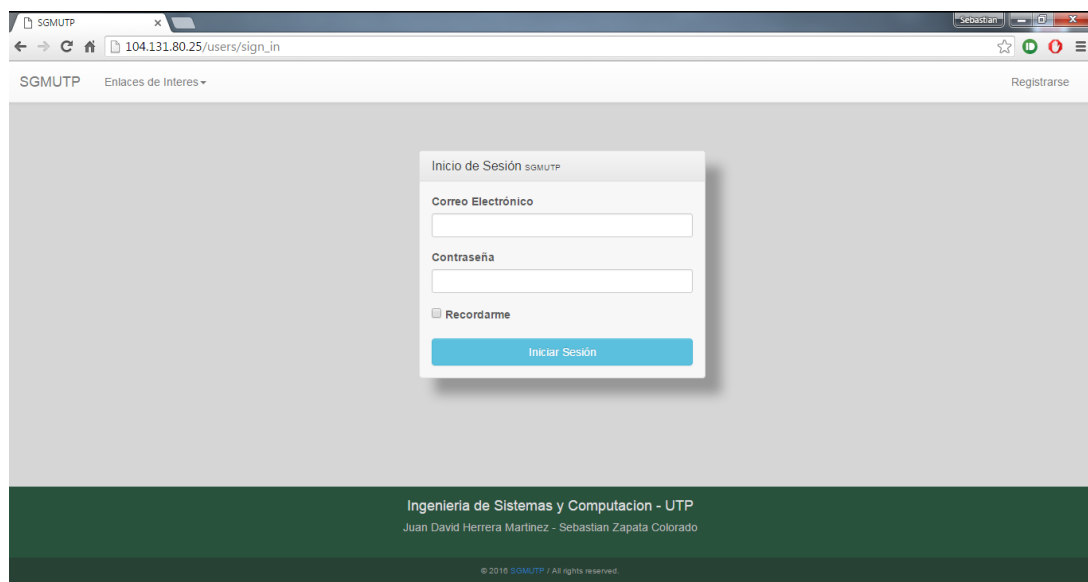
Para que el registro se realice satisfactoriamente, se deben completar los campos de los formularios con la información requerida como también necesario y de vital importancia cargar una firma en formato .JPG o .PNG, de no ser así, **NO** se podrá realizar el debido registro. La firma debe de realizar con una tableta digitalizadora y exportada en cualquiera de los formatos de archivos previamente mencionados. El administrador del prototipo debe de tener una digitalizadora para ayudar a realizar este proceso o en la salas donde se prestarán el servicio de monitorias.

Una vez diligenciado todos los campos y seleccionado el archivo de la firma en las extensiones mencionadas previamente, debe dar click en “**Solicitar Registro**” para finalizar satisfactoriamente el proceso. Después de esto, volverá a la página inicial o Index del prototipo para iniciar sesión.

2.3 Iniciar Sesión

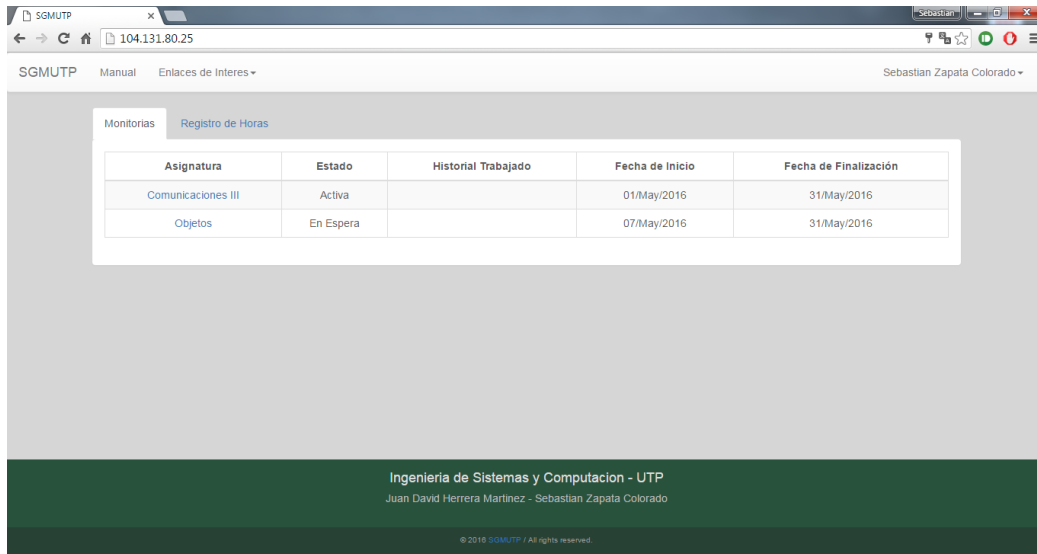
Hasta que el administrador del prototipo no acepte su solicitud de registro, NO podrá iniciar sesión. Comuníquese con el administrador para agilizar el estudio y posterior aprobación o rechazo de su solicitud.

Una vez aceptada la solicitud por parte del administrador, debe de completar los campos del inicio de sesión y dar click en “**Iniciar Sesión**”



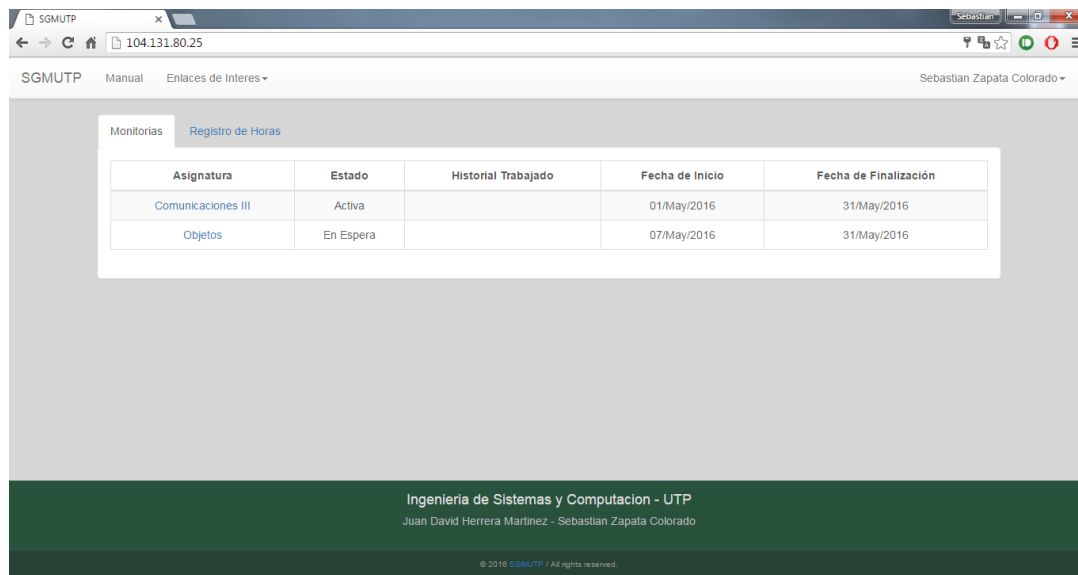
2.4 Dashboard

Una vez iniciada la sesión, se carga una vista donde se encuentran las funcionalidades del Monitor o también denominado “**Dashboard**”. En dicha vista se visualizan dos enlaces llamados: “**Monitorias**” y “**Registro Mensual**”.



2.5 Pestaña Monitorias

En este enlace se puede observar una tabla que contiene las monitorias creadas por el administrador del prototipo y con las que usted como monitor puede empezar a trabajar registrando a los estudiantes que ingresen a las salas de computo.



Detalles de la tabla:

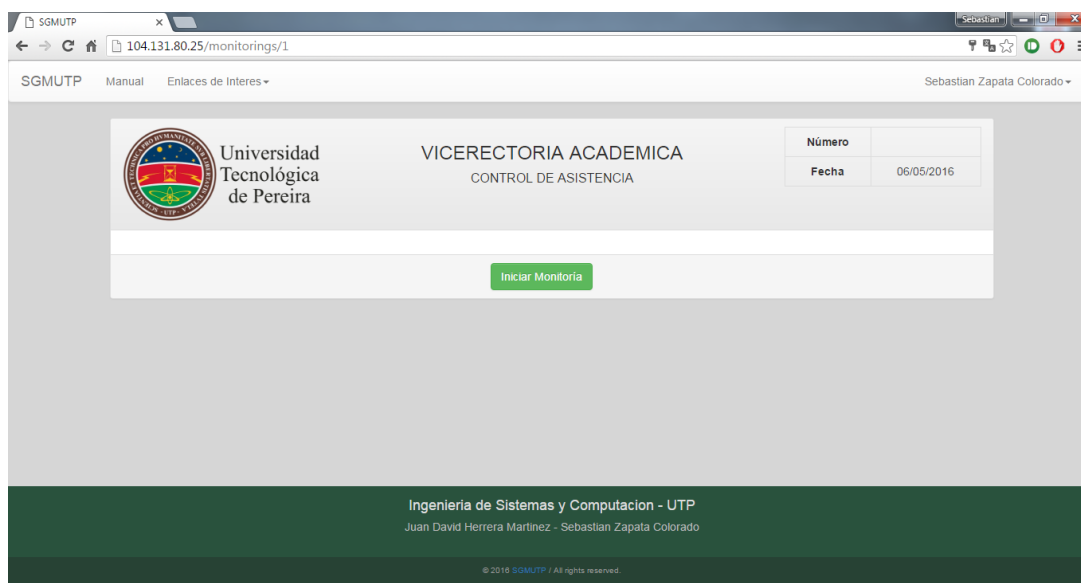
- **Nombre:** De la monitoria a la cual se habilitó.
- **Estado:** Se muestra si está Activa o En Espera.

- **Historial Trabajado:** Esto hace referencia al tiempo total trabajado para esa monitoria desde que fue creada por el administrador. Si este campo se encuentra vacío, significa que usted no ha registrado actividad en esa monitoria.
- **Fecha de Inicio:** Muestra la fecha en la cual usted puede hacer uso de la monitoria para trabajar.
- **Fecha de Finalización:** Muestra la fecha en la que se encontrará activa la monitoria, una vez se cumpla dicha fecha, NO podrá seguir trabajando en esa monitoria.

Al dar click en el nombre de las monitorias que están en la tabla, se puede acceder a ella para empezar a registrar estudiantes que ingresen a las salas de cómputo.

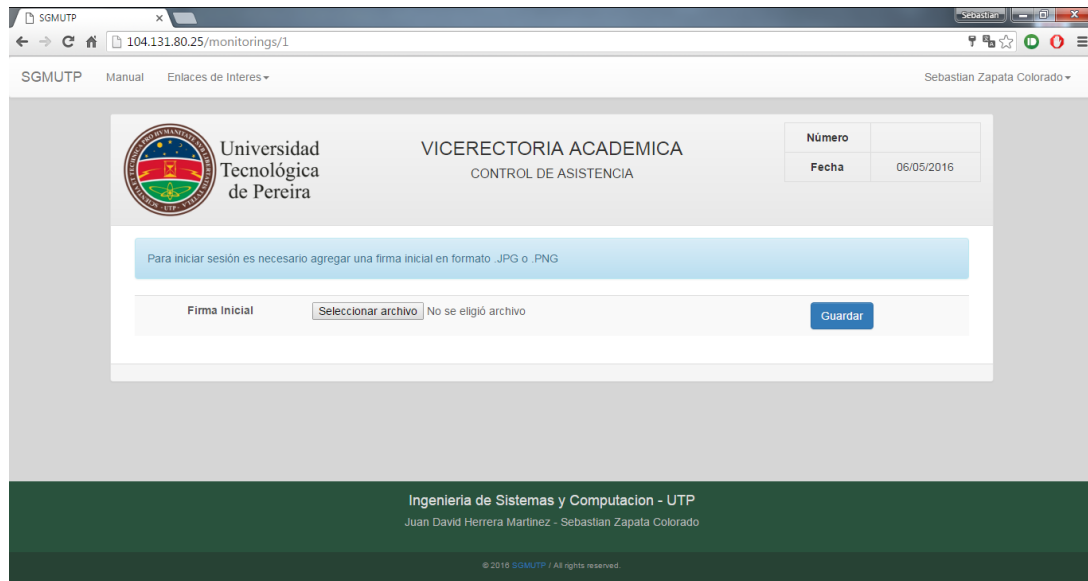
2.5.1 Iniciar Monitorias

Para comenzar a trabajar es necesario dar click en “**Iniciar Monitoria**”. Si usted accedió a una monitoria en una fecha en la cual no es acorde a la establecida por el administrador del prototipo, el botón “**Iniciar Monitoria**” NO aparecerá.



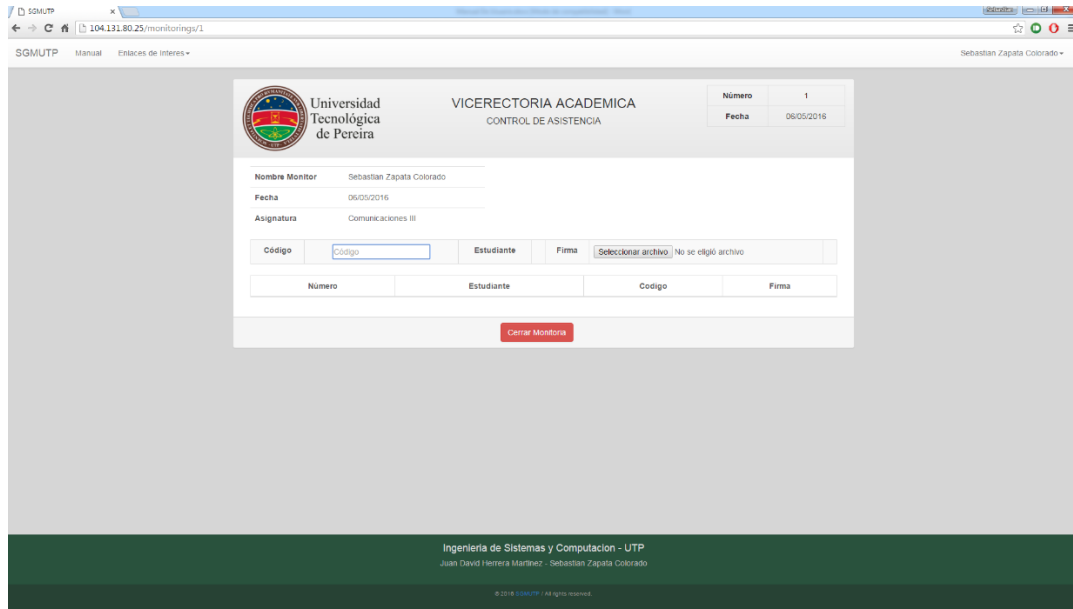
Al dar click saldrá un mensaje de advertencia el cual dice: “**¿Está seguro de que desea iniciar esta monitoria?**”. Si está de acuerdo debe dar click en “**Aceptar**”, en caso contrario en “**Cancelar**”.

Al dar click en “**Aceptar**”, se cargará una nueva vista donde es necesario ingresar la firma del monitor con un archivo desde su computador con extensión .JPG o .PNG con ayuda de una tableta digitalizadora. Si no se carga está opción, no se puede realizar la monitoria. Al seleccionarla, debe dar click en “**Guardar**”.



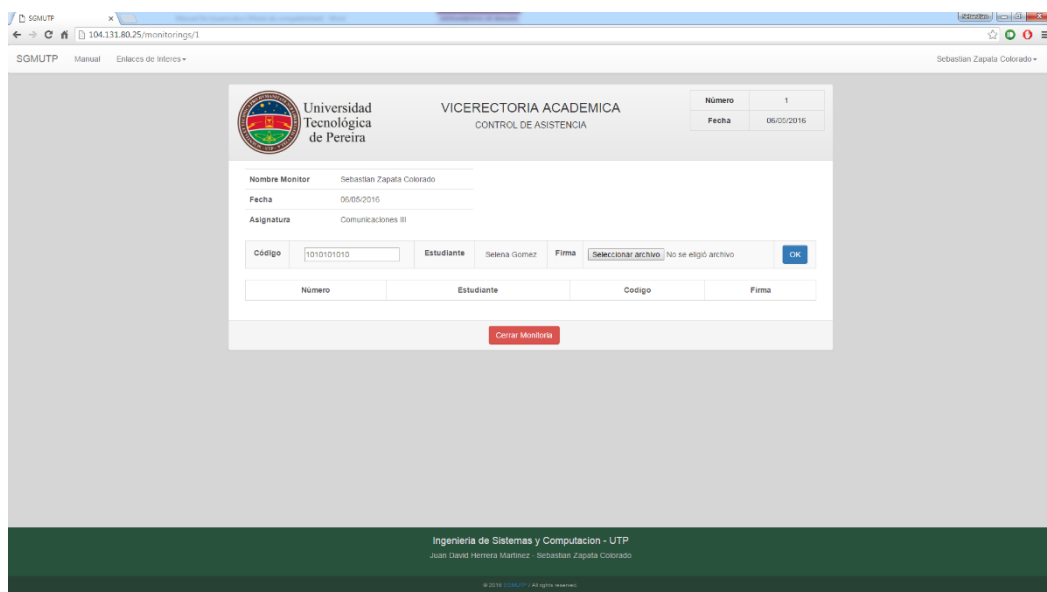
2.5.2 Ingresar Estudiantes

Al iniciar la monitoria satisfactoriamente se debe mostrar una nueva vista donde el monitor puede ingresar el Código y la firma en formato digital de los estudiantes que van a hacer uso de las salas de computo.



Para ingresar el código, se puede hacer mediante el teclado digitando dicho número o con ayuda de un lector óptico de código de barras en el campo “**Código**”, luego debe dar click en cualquier parte de la página para que cargue el estudiante con el nombre completo. Una vez identificado el estudiante aparecerá el botón “**OK**”.

Posteriormente se debe seleccionar el archivo de imagen que contiene la firma del estudiante en extensión .JPG o .PNG para que sea válida. Para finalizar el ingreso es necesario dar click en el botón “**OK**”.



Ya ingresado el estudiante, éste registro se debe cargar automáticamente en la lista con los datos necesarios y desapareciendo el botón “OK”. A partir de este momento se puede hacer ingreso de más estudiantes siguiendo el mismo procedimiento descrito en esta sección.

The screenshot shows a web browser window with the URL 104.131.80.25/monitorings/1. The page header includes the logo of Universidad Tecnológica de Pereira and the text "VICERECTORIA ACADEMICA CONTROL DE ASISTENCIA". The user is identified as Sebastian Zapata Colorado. The main form contains the following fields:

- Nombre Monitor: Sebastian Zapata Colorado
- Fecha: 06/05/2016
- Asignatura: Comunicaciones III

Below these fields is a table for student registration:

Código	Estudiante	Firma
<input type="text" value="Codigo"/>		<input type="button" value="Seleccionar archivo"/> No se eligió archivo

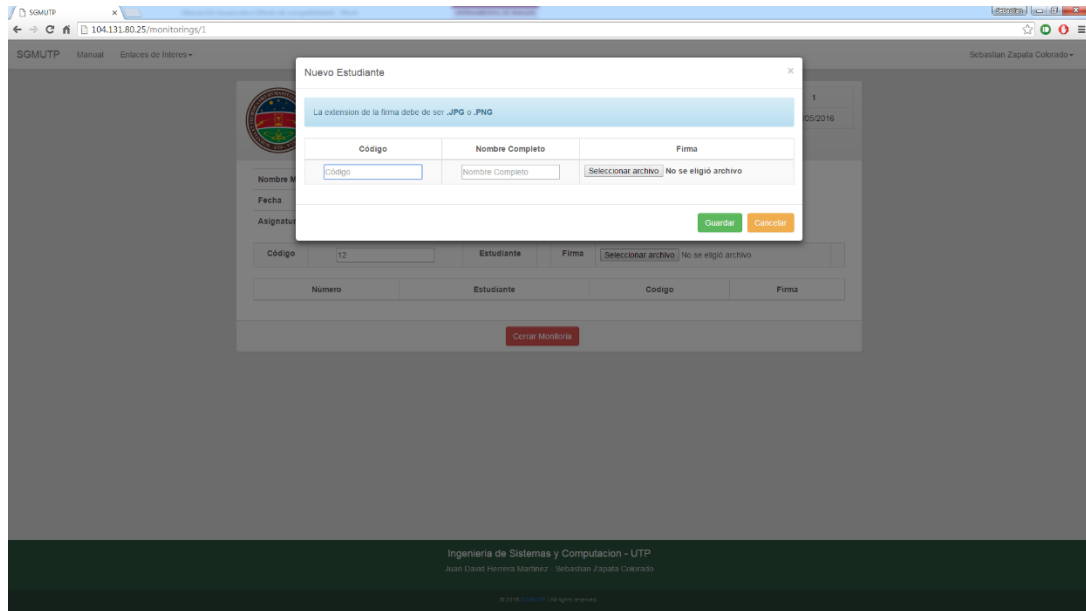
Below this is a summary table:

Número	Estudiante	Código	Firma
1	Selena Gomez	1010101010	

A red button labeled "Cerrar Monitoria" is located below the summary table. The footer of the page reads "Ingeniería de Sistemas y Computación - UTP" and "Juan David Herrera Martínez - Sebastian Zapata Colorado".

2.5.3 Registrar Estudiantes

Si no se encuentra registrado el estudiante al momento de ingresar el código con ayuda del teclado o de un lector óptico de código de barras, se mostrará un mensaje informando “**Estudiante NO encontrado**”. Dando click en “**Aceptar**”, se mostrará una ventana modal permitiendo el registro del estudiante.

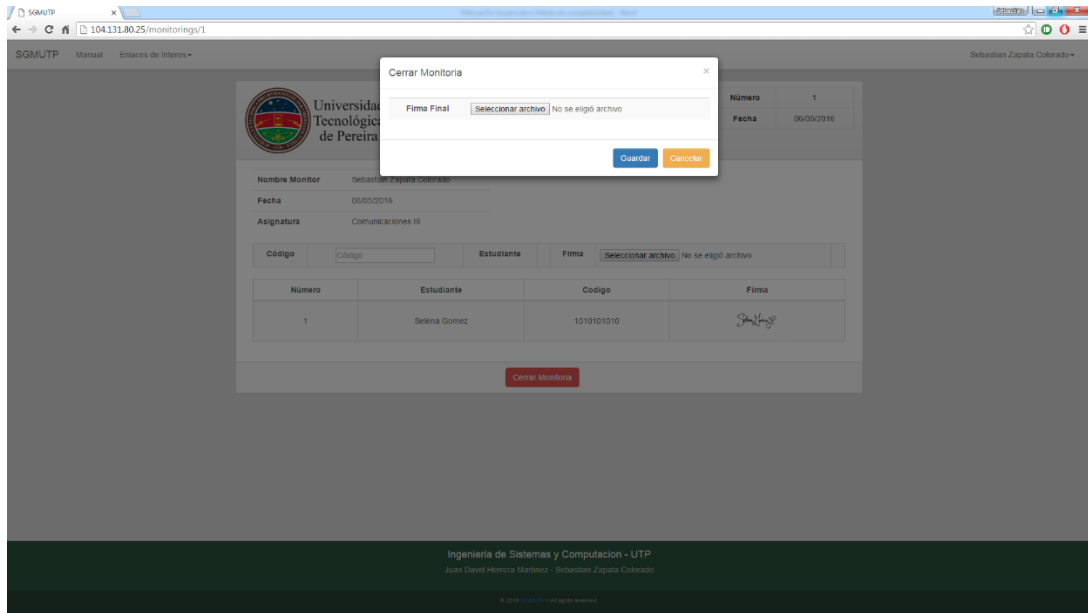


Debe ingresar los datos que se piden en el formulario y seleccionar el archivo de imagen que contiene la firma del estudiante con ayuda de una tableta digitalizadora. Posteriormente se debe dar click en “**Guardar**” y se cerrará la ventana modal automáticamente. A partir de este momento es posible ingresar al nuevo estudiante en la lista.

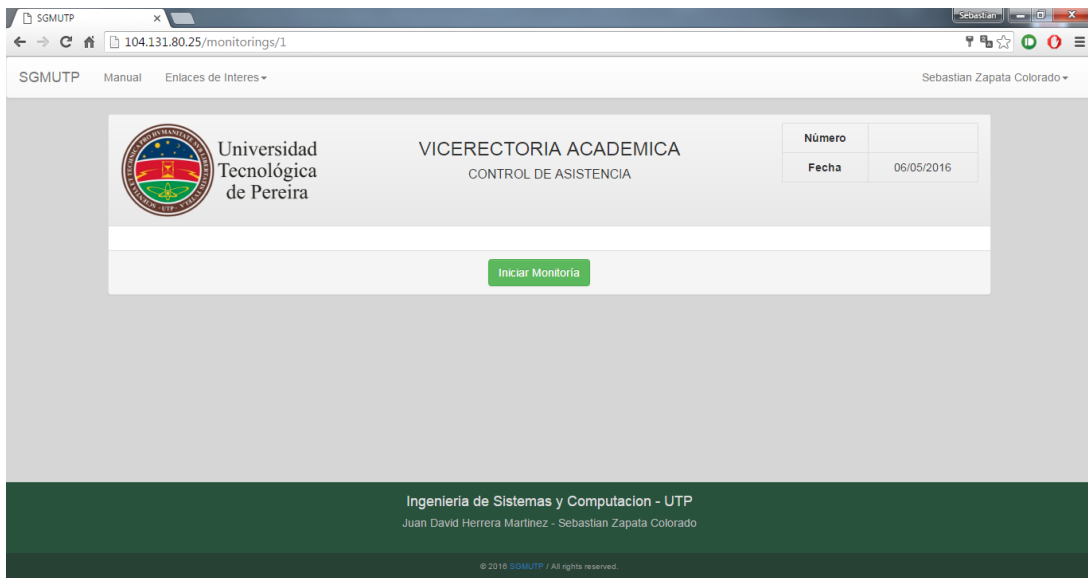
Si opta por la opción “**Cancelar**”, dicho registro **NO** se realizará satisfactoriamente y el estudiante **NO** quedará guardado en la base de datos.

2.5.4 Finalizar Monitoria

Para realizar este proceso, de click al botón rojo en la parte inferior de la página web “**Finalizar Monitoria**”, esto cargará una ventana modal en la cual se debe seleccionar el archivo de imagen con la firma del monitor realizada con ayuda de una tableta digitalizadora. Se debe dar click en “**Guardar**” para efectuar la finalización exitosamente. Si opta por la opción “**Cancelar**”, el cierre de la monitoria **NO** se llevará a cabo y se puede continuar ingresando estudiantes en la lista.

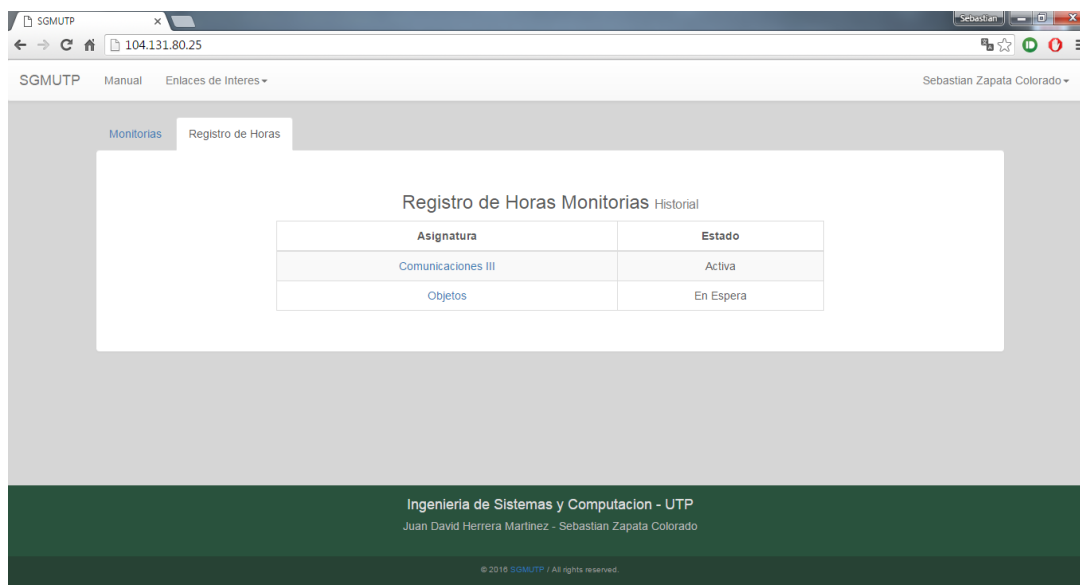


Una vez finalizada la monitoria, se cargará una vista en la cual puede volver a iniciar una nueva monitoria para la misma asignatura de trabajo presionando el botón **“Iniciar Monitoria”** o si quiere ir al **“Dashboard”** es necesario en la barra de navegación del prototipo, dar click en el botón **“SGMUTP”**

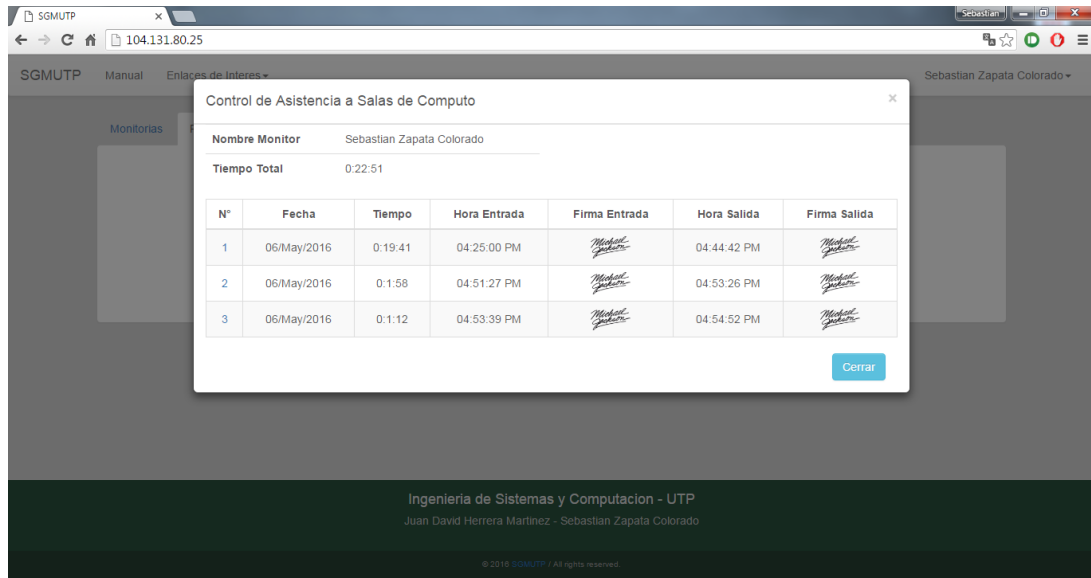


2.6 Pestaña Registro De Horas

En esta pestaña se puede visualizar una tabla con las monitorias creadas por parte del administrador del prototipo y en la cual se puede dar click en la monitoria deseada para ver cuantas horas ha trabajado. El estado se encuentra en “Activa” si ya ha trabajado en la monitoria, en caso contrario aparecerá “En Espera”.



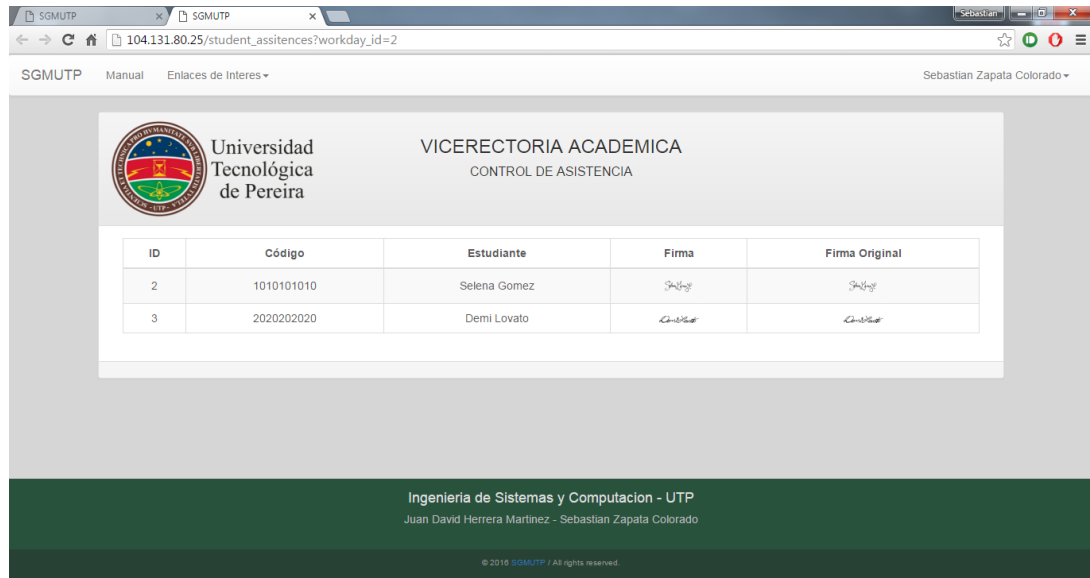
Una vez seleccionada la monitoria, se cargará una ventana modal en la cual se puede visualizar el tiempo total trabajado y una tabla con la descripción de las monitorias trabajadas. En caso en el que se haya seleccionado una monitoria y **NO** se haya mostrado la ventana modal, significa que actualmente **NO** se ha finalizado una monitoria de esa asignatura. Para más detalles sobre como finalizar una monitoria, diríjase a la sección **2.5.4**



2.6.1 Control de Asistencia de Estudiantes

Para visualizar los estudiantes que asistieron a una monitoria, es necesario dar click en la columna “N°” de la monitoria deseada, en la ventana modal que se obtiene en la sección 2.6. Esto creará una nueva pestaña en el navegador de internet con una vista de la lista completa de estudiantes con su respectiva firma en el momento ingresar a la sala de cómputo y la “Firma Original” que fue creada en el momento que se registró el estudiante como se explica en la sección 2.5.3

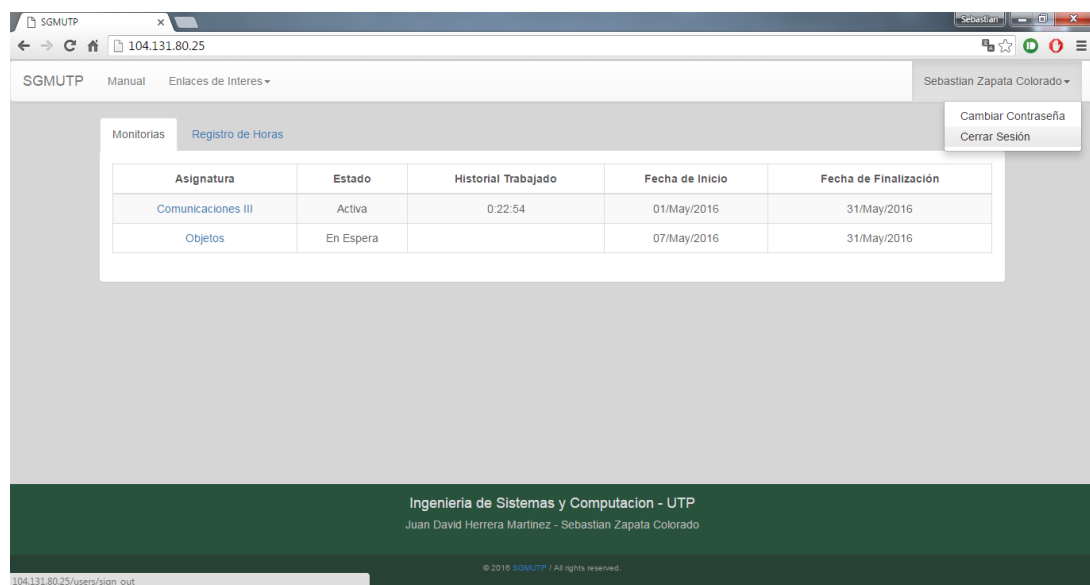
Para salir de esta vista de consulta, simplemente es necesario cerrar la pestaña que se creó en el navegador de internet.



2.7 Cerrar Sesión

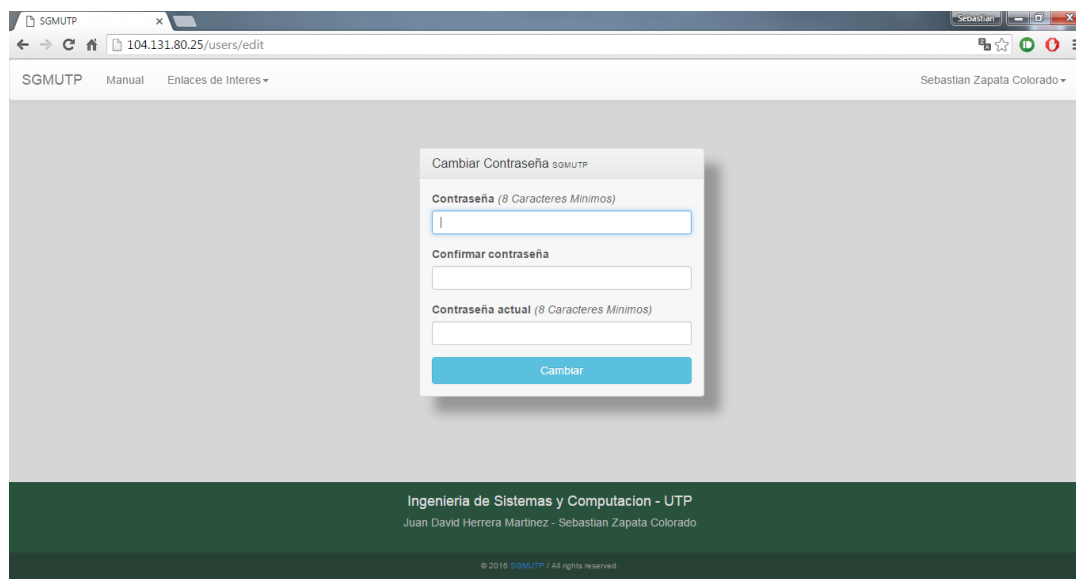
Si desea cerrar sesión, solo es necesario dar click en el nombre de usuario en la barra de navegación del prototipo. Posteriormente se desplegará una lista de opciones en la cual se debe seleccionar “**Cerrar Sesión**” para efectuar la acción.

Al realizar esta acción, se cargará el Index o página inicial del prototipo.



2.8 Cambiar Contraseña

Si desea cambiar la contraseña de su cuenta, es necesario dar click en el **nombre de usuario** en la barra de navegación del prototipo. Posteriormente se desplegará una lista de opciones en la cual se debe seleccionar “**Cambiar Contraseña**” para cargar una nueva vista que lo llevará a un formulario que le ayudará a efectuar la acción.



The screenshot shows a web browser window with the URL 104.131.80.25/users/edit. The page title is 'SGMUTP' and the user name 'Sebastian Zapata Colorado' is visible in the top right. A modal form titled 'Cambiar Contraseña SGMUTP' is centered on the page. The form contains three input fields: 'Contraseña (8 Caracteres Minimos)', 'Confirmar contraseña', and 'Contraseña actual (8 Caracteres Minimos)'. A blue 'Cambiar' button is located at the bottom of the form. The footer of the page includes the text 'Ingeniería de Sistemas y Computación - UTP', 'Juan David Herrera Martínez - Sebastian Zapata Colorado', and '© 2016 SGMUTP / All rights reserved.'

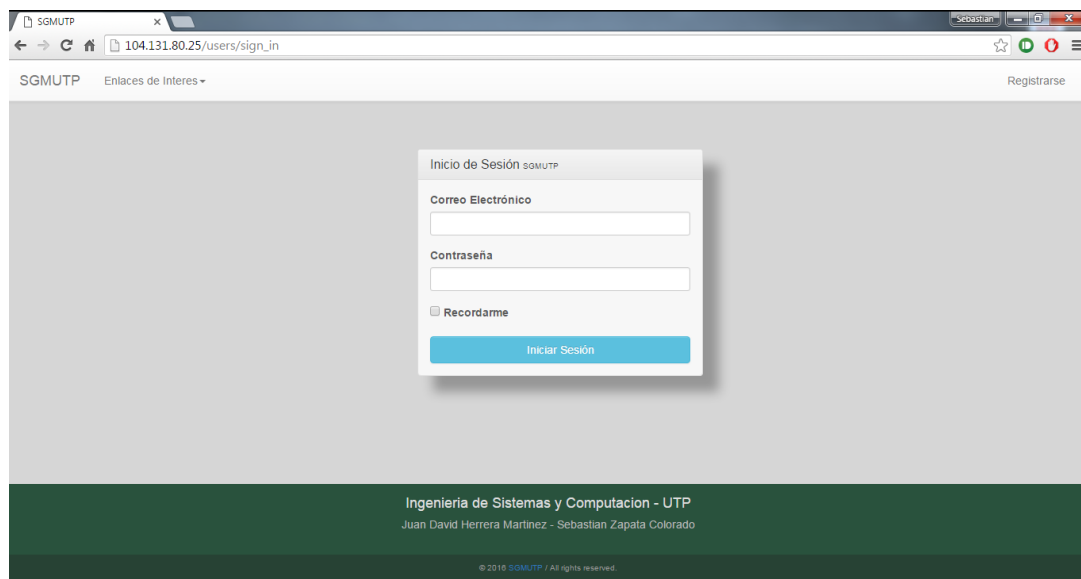
Una vez diligenciado los campos, se debe dar click en el botón “**Cambiar**” para finalizar el proceso. Si tiene inconsistencia en los datos que se solicita diligenciar, mostrará un mensaje de error para que usted corrija los campos en el formulario. En caso que los datos sean consistentes, lo llevará al “**Dashboard**” o página principal del perfil monitor garantizando así el cambio exitoso de contraseña de su cuenta.

3. PERFIL ADMINISTRADOR

3.1 Acceso al Prototipo

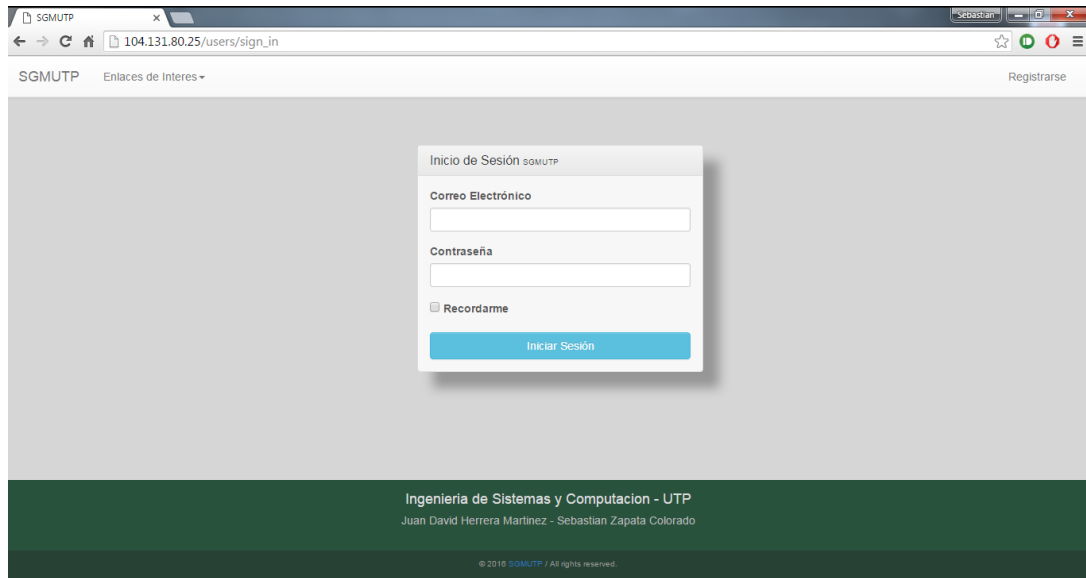
Para hacer uso de todas las funcionalidades del prototipo web, se debe acceder desde cualquier navegador de internet de última generación a la siguiente dirección URL en la barra de navegación: 104.131.80.25

Al cargar la página web, se mostrará el Index o página inicial para poder iniciar sesión.



3.2 Iniciar Sesión

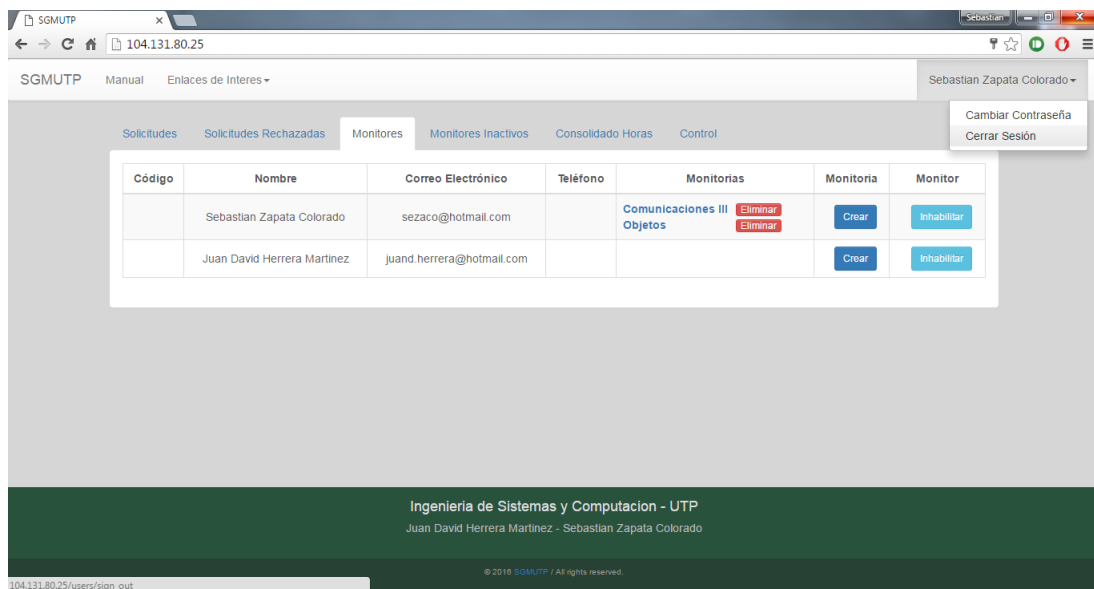
Es necesario llenar los campos solicitados en el Index o página inicial del prototipo, los cuales son: “**Correo Electrónico**” y “**Contraseña**”. Al llenarlos se debe dar click en el botón “**Iniciar Sesión**”.



3.2.1 Cerrar Sesión

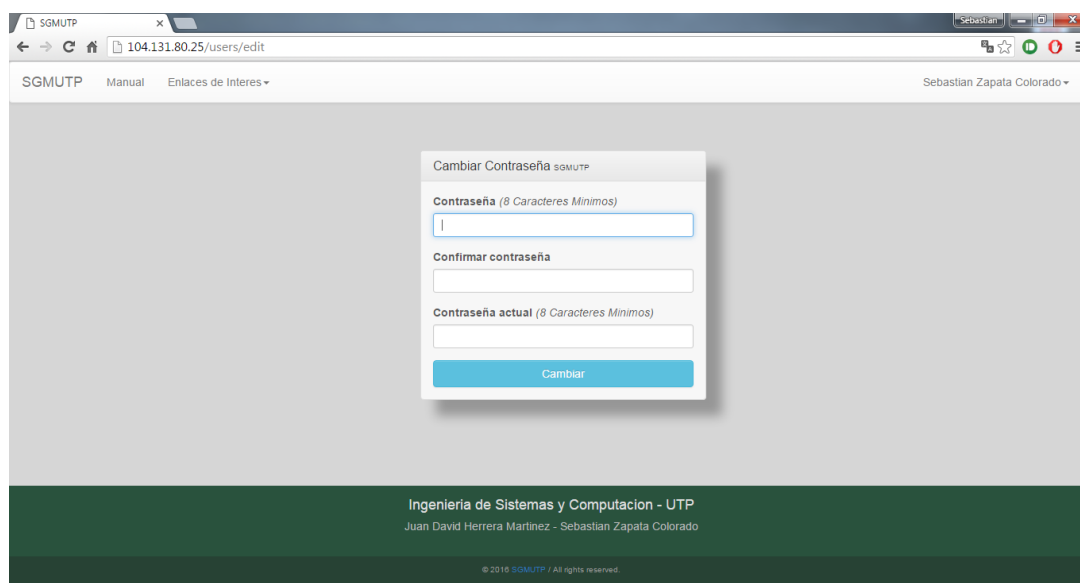
Si desea cerrar sesión, solo es necesario dar click en el nombre de usuario en la barra de navegación del prototipo. Posteriormente se desplegará una lista de opciones en la cual se debe seleccionar “**Cerrar Sesión**” para efectuar la acción.

Al realizar esta acción, se cargará el Index o página inicial del prototipo.



3.2.2 Cambiar Contraseña

Si desea cambiar la contraseña de su cuenta, es necesario dar click en el nombre de usuario en la barra de navegación del prototipo. Posteriormente se desplegará una lista de opciones en la cual se debe seleccionar “**Cambiar Contraseña**” para cargar una nueva vista que lo llevará a un formulario que le ayudará a efectuar la acción.

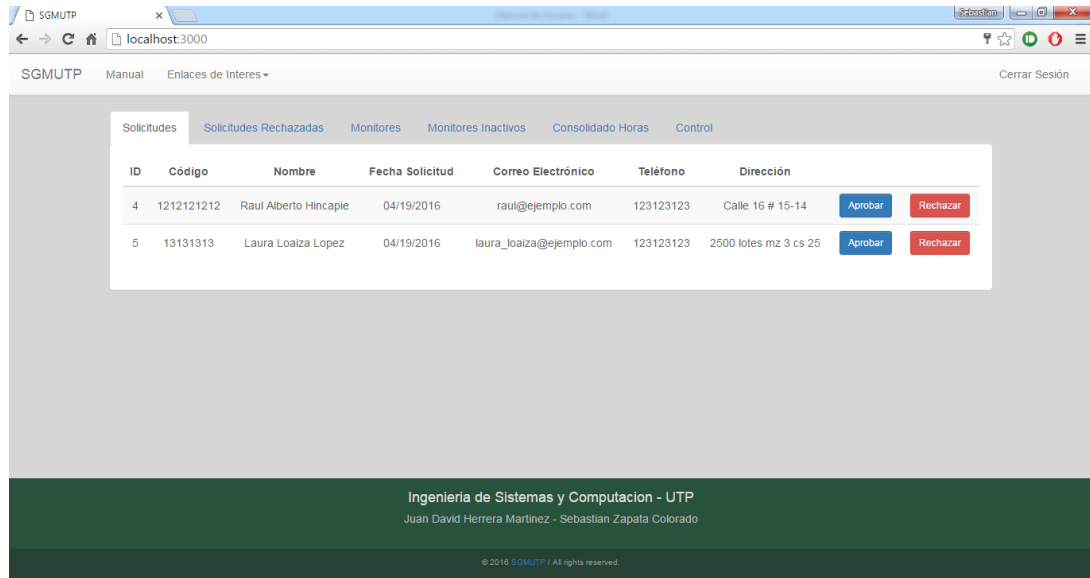


The screenshot shows a web browser window with the URL 104.131.80.25/users/edit. The page title is 'SGMUTP' and the user is identified as 'Sebastian Zapata Colorado'. A modal form titled 'Cambiar Contraseña SGMUTP' is displayed in the center. The form contains three input fields: 'Contraseña (8 Caracteres Minimos)', 'Confirmar contraseña', and 'Contraseña actual (8 Caracteres Minimos)'. A blue 'Cambiar' button is located at the bottom of the form. The footer of the page includes the text 'Ingeniería de Sistemas y Computación - UTP', 'Juan David Herrera Martínez - Sebastian Zapata Colorado', and '© 2016 SGMUTP / All rights reserved.'

Una vez diligenciado los campos, se debe dar click en el botón “**Cambiar**” para finalizar el proceso. Si tiene inconsistencia en los datos que se solicita diligenciar, mostrará un mensaje de error para que usted corrija los campos en el formulario. En caso que los datos sean consistentes, lo llevará al “**Dashboard**” o página principal del perfil administrador garantizando así el cambio exitoso de contraseña de su cuenta.

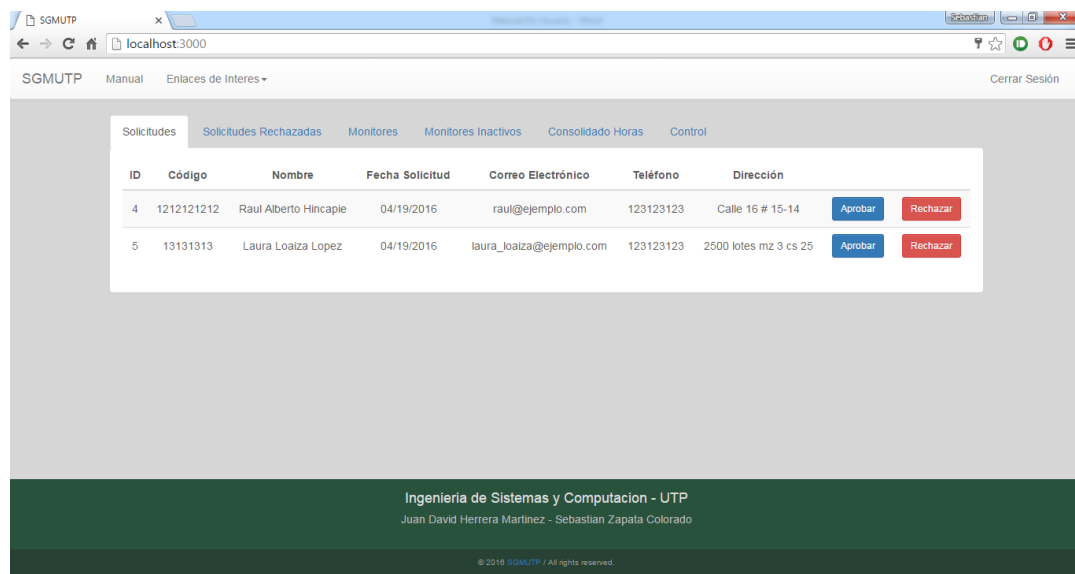
3.3 Dashboard

Una vez iniciada la sesión, se puede acceder a las funcionalidades del perfil Administrador en una vista llamada “**Dashboard**”. En dicha vista se observan seis enlaces llamados: “**Solicitudes**”, “**Solicitudes Rechazadas**”, “**Monitores**”, “**Monitores Inactivos**”, “**Consolidado de Horas**” y “**Control**”.



3.4 Pestaña Solicitudes

En esta pestaña es posible visualizar todas las solicitudes enviadas por los monitores que realizan el proceso de registro en el prototipo. Dichas solicitudes cargaran los datos más relevantes que los monitores diligencian en el formulario de registro, para facilitar al administrador su aprobación o rechazo.



3.4.1 Rechazar Solicitud

En la pestaña “**Solicitudes**” se debe ubicar la solicitud de registro que desea rechazar bajo su criterio. Luego se debe dar click en el botón “**Rechazar**” y posteriormente aparecerá un mensaje de advertencia donde se debe confirmar si quiere efectuar la acción.

La solicitud rechazada aparecerá en la pestaña “**Solicitudes Rechazadas**”.

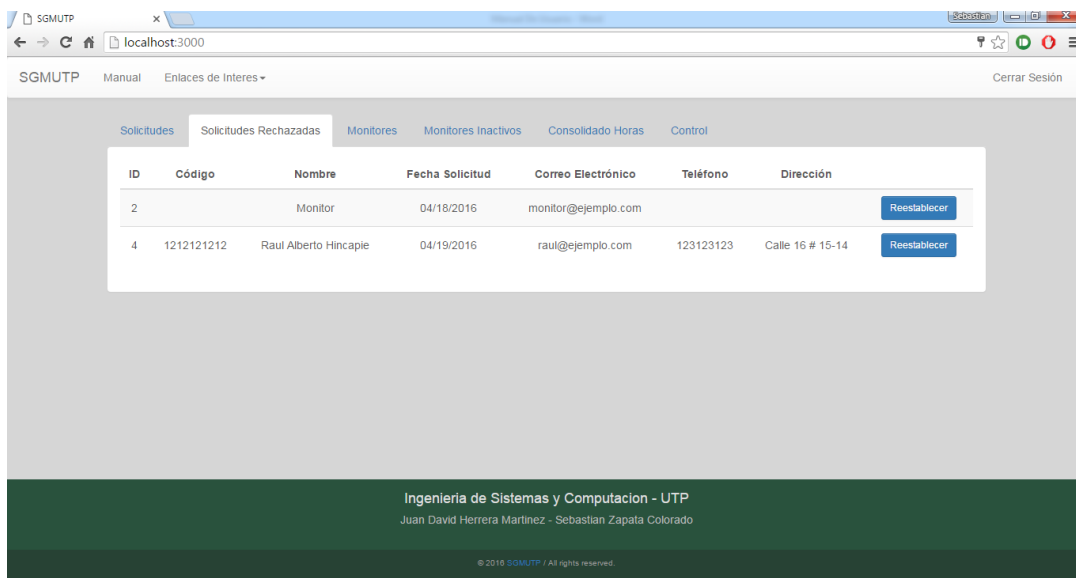
3.4.2 Aprobar Solicitud

En la pestaña “**Solicitudes**” se debe ubicar la solicitud de registro que desea aprobar bajo su criterio. Luego es necesario dar click en el botón “**Aprobar**” para efectuar la acción.

La solicitud aprobada aparecerá en la pestaña “**Monitores**”.

3.5 Pestaña Solicitudes Rechazadas

En esta pestaña se puede visualizar todas las solicitudes que han sido rechazadas bajo su criterio con la información más relevante diligenciada por los monitores.



ID	Código	Nombre	Fecha Solicitud	Correo Electrónico	Teléfono	Dirección	
2		Monitor	04/18/2016	monitor@ejemplo.com			Reestablecer
4	1212121212	Raul Alberto Hincapie	04/19/2016	raul@ejemplo.com	123123123	Calle 16 # 15-14	Reestablecer

Ingeniería de Sistemas y Computación - UTP
Juan David Herrera Martínez - Sebastián Zapata Colorado
© 2016 SGMUTP / All rights reserved.

3.5.1 Reestablecer Solicitud

En la pestaña “**Solicitudes Rechazadas**”, se debe ubicar la solicitud que desea reestablecer y dar click en el botón “**Reestablecer**”. Posteriormente aparecerá un mensaje de advertencia donde se debe confirmar si quiere efectuar la acción.

La solicitud reestablecida aparecerá en la pestaña “**Solicitudes**”.

3.6 Pestaña Monitores

En esta vista se pueden visualizar todos los monitores existentes aprobados con los datos más relevantes para asegurar su identificación. También se pueden visualizar las monitorias que se hayan asignado a cada monitor.

Como funcionalidades de esta pestaña existe:

- **Eliminar:** Monitoria de un monitor en específico.
- **Crear:** Monitoria para un monitor en específico.
- **Inhabilitar:** Monitor bajo el criterio del administrador.

Código	Nombre	Correo Electrónico	Teléfono	Monitorias	Monitoria	Monitor
	Sebastian Zapata Colorado	sezaco@hotmail.com		Comunicaciones III Objetos	Eliminar Eliminar	Crear Inhabilitar
	Juan David Herrera Martinez	juand.herrera@hotmail.com			Crear	Inhabilitar

3.6.1 Inhabilitar Monitor

En la pestaña “**Monitores**”, para inhabilitar un monitor es necesario ubicarlo en la lista. Una vez ubicada la fila en la que se encuentra el monitor se debe dar click en el botón “**Inhabilitar**” para efectuar la acción.

El monitor será ubicado en la pestaña de “**Monitores Inactivos**”.

3.6.2 Eliminar Monitoria

En la pestaña “**Monitores**”, para eliminar una monitoria de un monitor, se debe ubicar la fila de la tabla en la que se encuentra el monitor y en la columna “**Monitorias**”, se podrán observar todas las monitorias activas de ese monitor, cada una de estas monitorias cuenta con un botón “**Eliminar**” que permite eliminarla. Una vez se da click en dicho botón, aparecerá un aviso de confirmación para efectuar la acción.

3.6.3 Crear Monitoria

En la pestaña “**Monitores**”, para crear la monitoria se debe ubicar la fila del monitor al cual se desea asignar una nueva monitoria y dar click en el botón “**Crear**”. Esto lo llevará a una vista donde debe diligenciar los siguientes datos:

Crear Monitoria SGMUTP

Nombre de Monitoria

Fecha de inicio

Fecha de finalización

Ingeniería de Sistemas y Computación - UTP
Juan David Herrera Martínez - Sebastian Zapata Colorado

© 2016 SGMUTP / All rights reserved.

- **Nombre de Monitoria:** Nombre de la asignatura que se le va a realizar monitorias.
- **Fecha de Inicio:** Momento en el cual se encuentra disponible la monitoria para contabilizar el tiempo de trabajo. Dando click en el botón “↓”, se puede mostrar un calendario para ayudar a encontrar la fecha deseada
- **Fecha de Finalización:** Momento límite de disponibilidad de la monitoria para contabilizar el tiempo de trabajo. Dando click en el botón “↓”, se puede mostrar un calendario para ayudar a encontrar la fecha deseada.

SGMUTP Manual Enlaces de Interes Sebastian Zapata Colorado

Crear Monitoria SGMUTP

Nombre de Monitoria

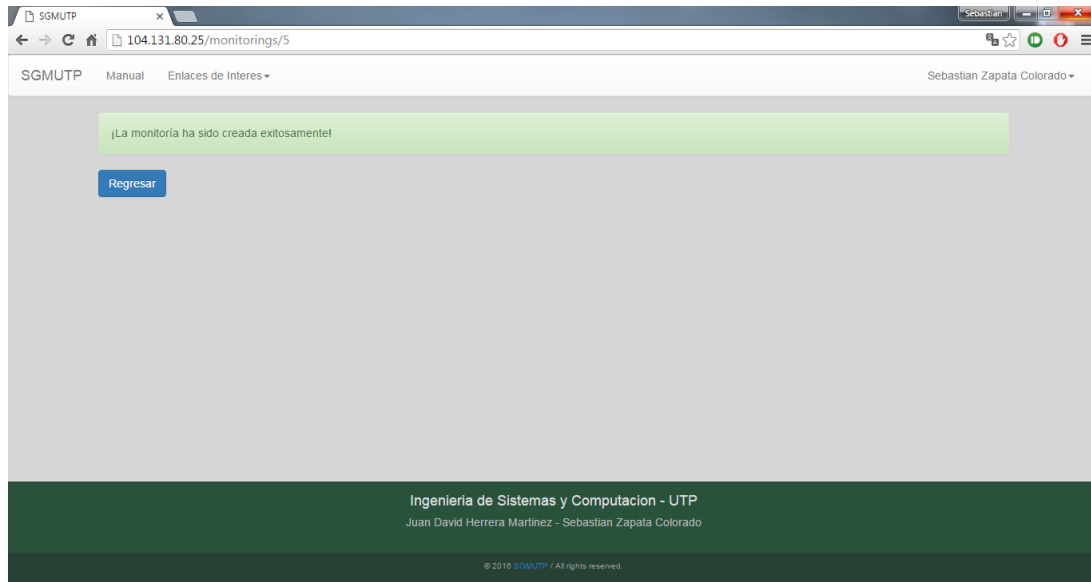
Fecha de inicio

mayo de 2016

dom	lun	mar	mié	jue	vie	sáb
	1	2	3	4	5	6
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Ingeniería de Sistemas y Computación - UTP
 Juan David Herrera Martínez - Sebastian Zapata Colorado
 © 2016 SGMUTP / All rights reserved.

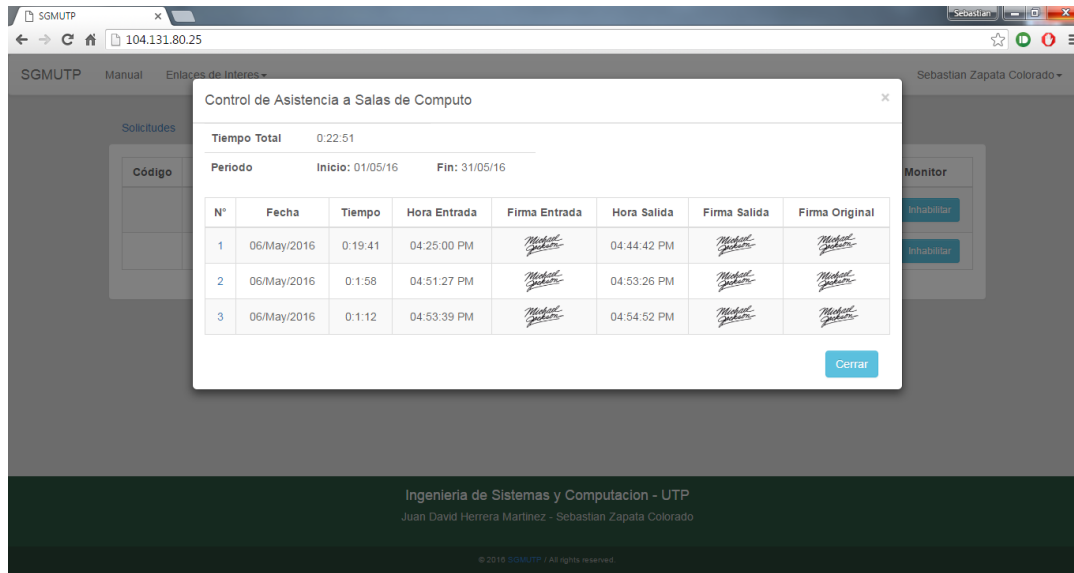
Una vez diligenciado el formulario, se debe dar click en el botón inferior del formulario color azul celeste “**Crear**” para efectuar la acción. Posteriormente aparecerá una vista con el siguiente mensaje “**¡La monitoria ha sido creada exitosamente!**” y un botón de “**Regresar**” para dirigirse a la página inicial de la sesión administrador o “**Dashboard**”.



3.6.4 Ver Monitorias del Monitor

En la pestaña “**Monitores**”, si se desea visualizar las monitorias de cada monitor, es necesario ubicar el monitor en la lista y ubicar la columna “**Monitorias**”. En esta columna seleccione la monitoria que desee explorar y dar click.

Al momento de dar click en una monitoria, se debe mostrar una ventana modal con la información completa del tiempo trabajado por parte del monitor, como también cada una de las monitorias que él ha prestado a esa asignatura. En el informe se muestra también la “**Firma Original**” que corresponde a la firma que el monitor diligenció al mandar la solicitud de registro (Ver sección **2.2**) para verificar si coincide con “**Firma Entrada**” y “**Firma Salida**” de cada monitoria.

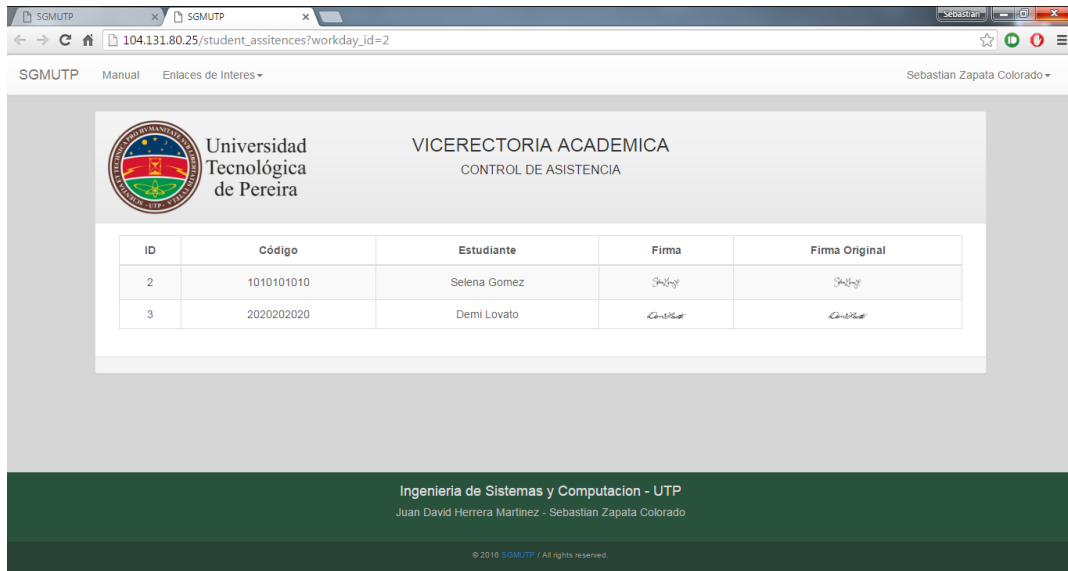


Si al momento de dar click en la monitoria ésta **NO** efectúa ninguna acción, significa que esa monitoria actualmente se encuentra abierta y el monitor no la ha finalizado. Se debe esperar a que el monitor finalice la monitoria para poder ver la información. Para más información sobre como un monitor finaliza una monitoria, ir a la sección **2.5.4**

3.6.5 Control de Asistencia de Estudiantes

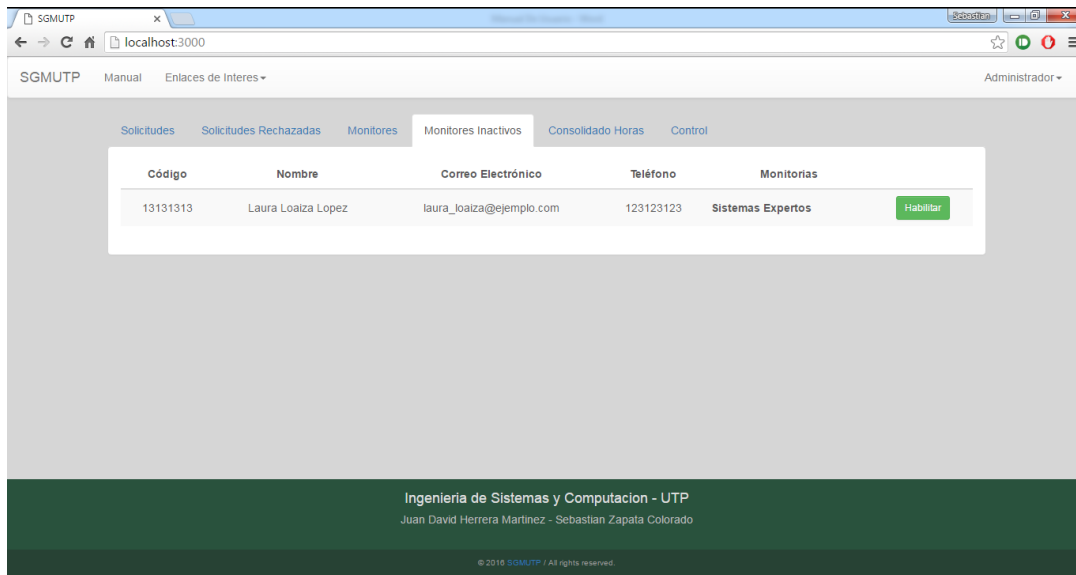
Para ver los estudiantes que asistieron a las monitorias, es necesario visualizar el informe que se obtiene en la sección **3.6.4** y se debe seleccionar en la columna “N°” la monitoria a la cual se desea ver los estudiantes dando click. Esta acción creará una nueva pestaña del navegador de internet con un reporte con todos los estudiantes que estuvieron presentes como también la “**Firma Original**” que se obtuvo en el registro del estudiante (ver sección **2.5.3**) para verificar su veracidad.

Para salir de este reporte basta con cerrar la pestaña que se creó en el navegador de internet.



3.7 Pestaña Monitores Inactivos

En esta vista se pueden visualizar los monitores que han sido inhabilitados por criterio del administrador. Contiene los datos más relevantes de los monitores como también las monitorias que tiene asignadas en el momento. Para poder eliminar las monitorias, el Administrador del prototipo debe habilitar el monitor para proceder a realizar las acciones que se mencionan en la sección 3.6.2



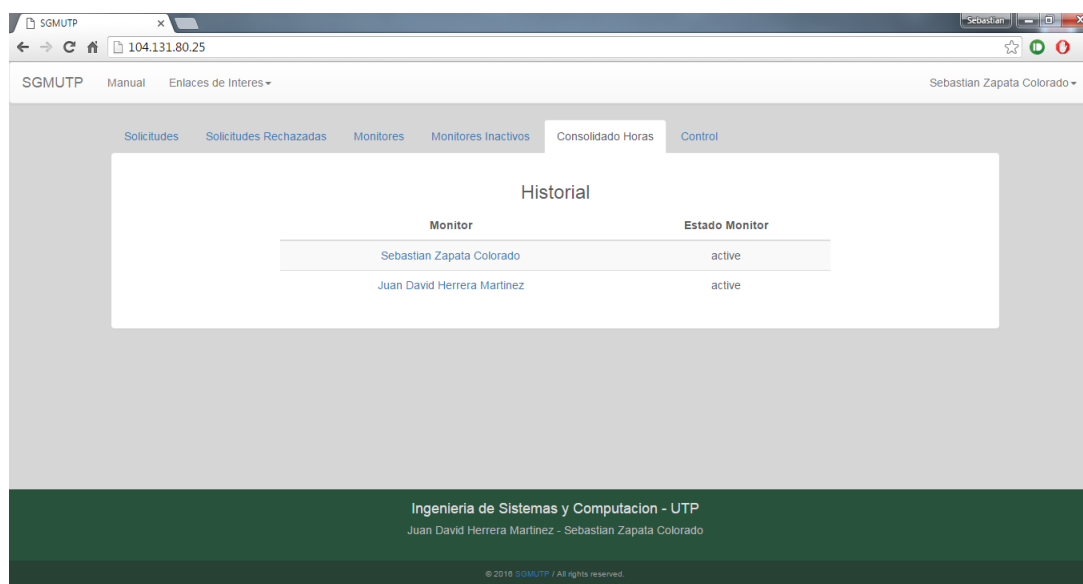
3.7.1 Habilitar Monitor

En la pestaña “**Monitores Inactivos**”, para volver a habilitar un monitor es necesario ubicarlo en la lista y dar click en el botón “**Habilitar**”, posteriormente se mostrará un mensaje de advertencia en el cual se debe confirmar para terminar la acción.

Los monitores habilitados se mostraran en la pestaña “**Monitores**”.

3.8 Pestaña Consolidado de Horas

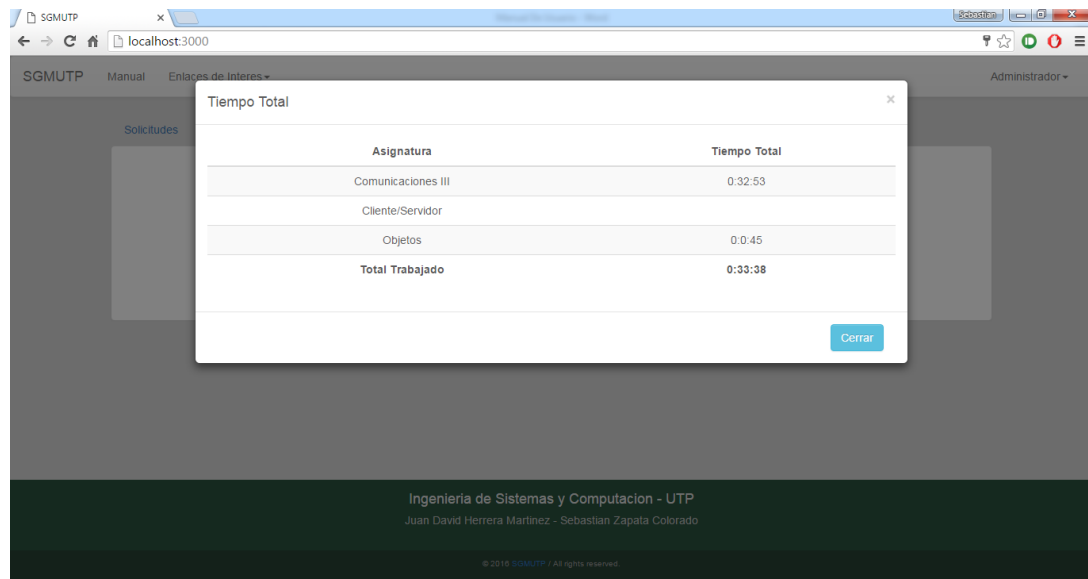
En esta pestaña se pueden ver la lista de los monitores que han realizado por lo menos una monitoria a una asignatura en las fechas de creación establecidas por el administrador del prototipo.



3.8.1 Ver Historial de Horas de Monitores

Una vez ubicado en la pestaña “**Consolidado de Horas**”, en la lista de monitores se debe hacer click en el que se desea explorar la cantidad de horas trabajadas. Posteriormente se muestra una vista modal con la lista de monitorias y el tiempo total trabajado por el monitor para dicha asignatura. Si no se muestra un tiempo en una asignatura, significa que aún el monitor no ha trabajado en ella.

Al final de la lista se muestra el tiempo total trabajado que es el resultado de la suma del tiempo de cada una de las asignaturas.



The screenshot shows a web browser window with the URL localhost:3000. A modal window titled "Tiempo Total" is open, displaying a table with the following data:

Asignatura	Tiempo Total
Comunicaciones III	0:32:53
Cliente/Servidor	
Objetos	0:0:45
Total Trabajado	0:33:38

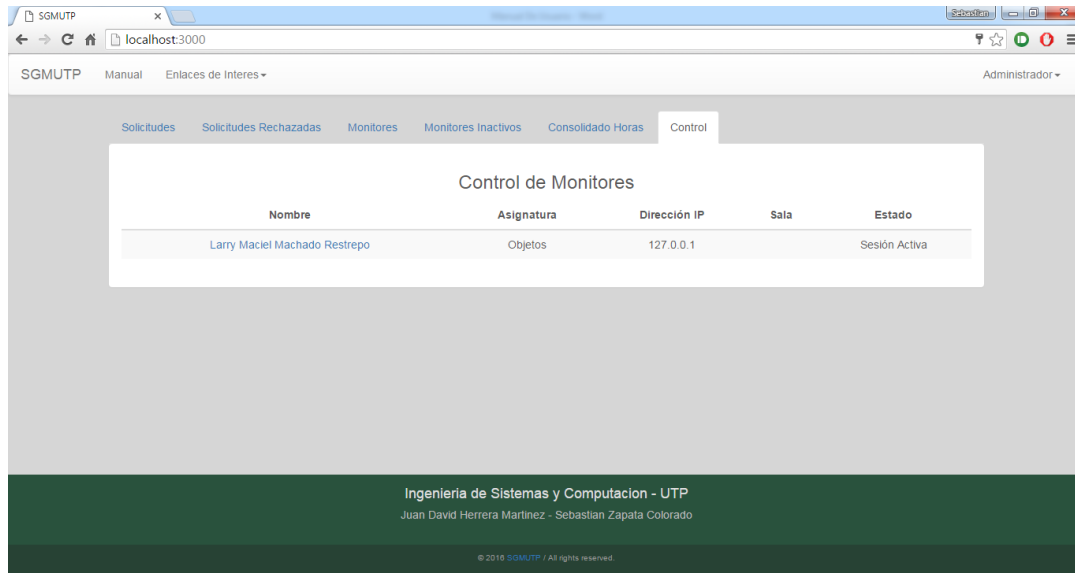
The modal window has a "Cerrar" button at the bottom right. The background of the web application is dimmed, showing a sidebar with "Solicitudes" and a top navigation bar with "SGMUTP", "Manual", and "Enlaces de Interés". The footer of the application reads "Ingeniería de Sistemas y Computación - UTP" and "Juan David Herrera Martínez - Sebastián Zapata Colorado".

Si al momento de dar click en un monitor para ver el reporte descrito anteriormente, **NO** se muestra la ventana modal, significa que el monitor en el momento se encuentra con una monitoria activa y debe finalizar dicha monitoria para poder ver el reporte descrito en esta sección.

3.9 Pestaña Control

En esta pestaña se puede visualizar todos los monitores que se encuentran actualmente trabajando en una monitoria o también es válido decir que esta pestaña muestra a los monitores que han iniciado una monitoria y no han realizado la opción de “**Finalizar Monitoria**”.

Para garantizar la visualización de todos los monitores actuales en tiempo real, se recomienda al administrador refrescar o presionar la tecla “**F5**”. Una vez la lista este actualizada, todos los pings que genere el monitor se cargarán automáticamente **SIN** necesidad de volver a refrescar la página.



3.9.1 Ver Pings de Monitores

Para poder ver los pings es necesario ubicarse en la pestaña “Control” y dar click en el monitor que desea explorar. Posteriormente se visualizará una ventana modal donde se listan los identificadores de las acciones que ha realizado el monitor con la hora y fecha. Estos Pings se generan cada vez que el monitor registra o ingresa un estudiante en su monitoria.

