



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

DESARROLLO DE UN SISTEMA WEB DE REGISTRO DE TUTORÍAS
ACADÉMICAS PARA UNA INSTITUCIÓN DE EDUCACIÓN SUPERIOR

AUTOR

Andrés Alejandro Tapia López

AÑO

2019



FACULTAD DE INGERIERÍA Y CIENCIAS APLICADAS

DESARROLLO DE UN SISTEMA WEB DE REGISTRO DE TUTORÍAS
ACADÉMICAS PARA UNA INSTITUCIÓN DE EDUCACIÓN SUPERIOR.

Trabajo de Titulación presentado en conformidad con los requisitos
establecidos para optar por el título de
Ingeniero en Sistemas de Computación e Informática.

Profesor Guía

MSc. Santiago Ramiro Villarreal Narváez

Autor

Andrés Alejandro Tapia López

Año

2019

DECLARACIÓN DEL PROFESOR GUÍA

"Declaro haber dirigido el trabajo, Desarrollo de un sistema web de registro de tutorías académicas para una institución de educación superior. A través de reuniones periódicas con el estudiante Andrés Alejandro Tapia López, en el semestre 201910, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación".

Santiago Ramiro Villarreal Narváz
Master Science Technologies, Sante a Finalite
Recherche Mention Informatique Specialite Web
Intelligence
C.I.:1713980074

DECLARACIÓN DEL PROFESOR CORRECTOR

"Declaro haber revisado este trabajo, Desarrollo de un sistema web de registro de tutorías académicas para una Institución de Educación Superior, del estudiante Andrés Alejandro Tapia López, en el semestre 201910 dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación".

Anita Elizabeth Yánez Torres
Magister en Gerencia Empresarial
Mención en Gerencia de Proyectos
C.I.: 1802462216

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

“Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.”

Andrés Alejandro Tapia López

C.I.: 1720645223

AGRADECIMIENTOS

Agradezco a mis padres por haberme dado la oportunidad de realizar mis estudios, por los valores que han inculcado en mí, a cada uno de los miembros de mi familia por haberme brindado su apoyo y ayuda incondicional a lo largo de mi carrera estudiantil.

A Dios por haberme dado la fortaleza y Sabiduría para así poder llegar a completar mis objetivos personales y académicos.

DEDICATORIA

El presente trabajo de titulación agradezco a mis padres Cesar Tapia y Zulema López que con su gran sabiduría y consejos han sido grandes pilares para poder formarme como profesional y ser humano.

A mis hermanos, Raquel, Johny, Grace, Anita que me han brindado siempre con humildad su apoyo y consejos a lo largo de mi vida y así poder ir por el camino correcto.

A Dios por estar presente en los momentos difíciles, por guiarme con su fe y sabiduría a lo largo de mi trayectoria como ser humano

RESUMEN

Las tutorías es el mecanismo que con mayor eficiencia se utiliza para la contribuir a mejorar el rendimiento académico de los estudiantes universitarios, este es un espacio que permite resolver dudas y preguntas que en muchas ocasiones no se puede responder en clases debido al poco tiempo que se dispone en la sala de clase.

Las tutorías es una de las herramientas indispensables para alcanzar el éxito académico, por lo que es importante realizar un seguimiento adecuado a este proceso. La tecnología en la actualidad nos da la facilidad de optimizar procesos. En el presente trabajo se propone desarrollar un sistema que permite la gestión adecuada de tutorías de esta manera mejorar la comunicación entre estudiantes y docentes, además de fortalecer el aprendizaje de los estudiantes.

El sistema permite la digitalización y manipulación de los datos de las tutorías, agilizando la manera de trabajar con los recursos universitarios y mejorando la organización estructural académica.

El proyecto comprende tres módulos que se detallan de manera general a continuación:

- Modulo Administrativo
- Modulo Docente:
- Modulo Estudiante:

La arquitectura para el desarrollo del sistema que se uso fue Modelo Vista Controlador (MVC), en cuanto a la metodología a utilizar para el desarrollo del proyecto será SCRUM, la cual facilita tanto la estructuración, como la implementación.

ABSTRACT

The tutorials is the mechanism that is used most efficiently to contribute to improving the academic performance of university students, this is a space that allows to resolve doubts and questions that in many occasions can't be answered in classes due to the short time that available in the classroom.

Tutoring is one of the indispensable tools to achieve academic success, so it is important to follow this process properly. Today's technology gives us the facility to optimize processes. In the present work, it is proposed to develop a system that allows the proper management of tutorials in this way to improve communication between students and teachers, as well as to strengthen student learning.

The system allows the digitalization and manipulation of the data of the tutorials, agility the way to work with the university resources and improving the academic structural organization.

The project comprises three modules that are detailed in a general manner below:

- Administrative Module
- Teaching Module:
- Student Module:

The architecture for the development of the system that was used was Model View Controller (MVC); in terms of the methodology to be used for the development of the project will be SCRUM, which facilitates both the structuring and the implementation.

ÍNDICE

1. Capitulo I. Introduccion	1
1.1. Antecedentes	1
1.2. Alcance	2
1.3. Justificación	4
1.4. Objetivo General.....	5
1.4.1. Objetivos Específicos	5
1.4.2. Metodología a Utilizar.....	6
2. Capitulo I. Marco Teórico	6
2.1. Metodología de diseño Web – Ágil.....	6
2.1.1. SCRUM.....	6
2.2. IDE ICESSCRUM.....	9
2.3. Aplicaciones Web.....	10
2.3.1. Front-End.....	10
2.3.2. Back-End	11
2.4. IDE Visual Studio 2017	11
2.4.1. Modelo MVC 5	11
2.4.2. Razor	12
2.4.3. LINQ	13
2.5. SQL	13
2.5.1. Consultas, Inserciones y Actualizaciones.....	13
2.5.2. Procedimientos Almacenados	13
2.6. Seguridad Informática	14
3. Capitulo II. Descripcion del Proyecto Analisis y Diseño..	14
3.1 . Sprints	15
3.1.1 Sprint 1	15
3.1.1.1 Reconocimiento del problema.....	15
3.1.1.2 Evaluación y síntesis	16

3.1.1.3 Arquitectura	17
3.1.1.3.1 Modelo	17
3.1.1.3.2 Vista	17
3.1.1.3.3 Controlador.....	17
3.1.1.4 Especificación.....	18
3.1.1.5 Product Backlog	19
4. Capitulo III. DesarrolloCodigo y Pruebas	22
4.1 Sprint 2.	22
4.1.1 Carga facultad, carreras, módulos, periodo.	23
4.1.2 Carga automática Excel Docentes	25
4.1.3 Carga automática Excel Estudiantes	26
4.1.4 Carga horario Docentes.....	27
4.1.5 Carga horario Estudiantes	28
4.1.6 Carga automática horario Docente	29
4.1.7 Carga automática horario Estudiante	30
4.2 Sprint 3.	31
4.2.1 Sección de Login para identificación del docente.....	32
4.2.2 Perfil Docente	33
4.2.3 Sección de notificación Docentes.	34
4.3 Sprint 4.	35
4.3.1 Sección de Login para identificación del estudiante.....	36
4.3.2 Perfil Estudiante.....	37
4.3.3 Notificaciones Estudiante.....	38
4.3.4 Búsqueda y petición de tutorías.	39
4.4 Sprint 5.	40
4.4.1 Sección de reportes	40
4.5 Sprint 6.	41
4.5.1 Pruebas	41
5. Conclusiones y Recomendaciones	56
5.1 Conclusiones.....	56
5.2 Recomendaciones	58

REFERENCIAS.....	59
ANEXOS.....	61

ÍNDICE DE FIGURAS

Figura 1. Metodología Scrum	6
Figura 2. Ejemplo de las Historias de Usuarios	9
Figura 3. Ejemplo de Product Backlog.	9
Figura 4. IDE IceScrum	10
Figura 5. Modelo MVC.	12
Figura 6. Arquitectura del desarrollo.	18
Figura 7. Product Backlog	20
Figura 8. Burndown chart del proyecto.....	22
Figura 9. Spring 2	23
Figura 10. Carga facultad carga carreras materias periodo y módulos.....	24
Figura 11. Carga Docentes archivo	25
Figura 12. Carga Estudiantes archivo	26
Figura 13. Carga Horario Docente	27
Figura 14. Carga Horario Estudiante.....	28
Figura 15. Carga Automática Horario Docente	29
Figura 16. Carga Automática Horario Estudiante	30
Figura 17. Sprint 3	31
Figura 18. Login Docente	32
Figura 19. Perfil Docente	33
Figura 20. Notificación Docente	34
Figura 21. Sprint 4	35
Figura 22. Login Docente	36
Figura 23. Perfil Estudiante	37
Figura 24. Horario Estudiante	38
Figura 25. Notificación Estudiante.....	38
Figura 26. Búsqueda y petición de tutoría	39
Figura 27. Sprint 5	40
Figura 28. Tipos de reportes	40

INDICE DE TABLAS

Tabla 1. Burndown chart del proyecto.....	21
Tabla 2. Historia de usuario - HU-TUT016	24
Tabla 3. Historia de usuario - HU-TUT012	25
Tabla 4. Historia de usuario - HU-TUT013	26
Tabla 5. Historia de usuario - HU-TUT002	27
Tabla 6. Historia de usuario - HU-TUT003	28
Tabla 7. Historia de usuario - HU-TUT014	29
Tabla 8. Historia de usuario - HU-TUT015	30
Tabla 9. Historia de usuario - HU-TUT005	32
Tabla 10. Historia de usuario - HU-TUT010	33
Tabla 11. Historia de usuario - HU-TUT007	34
Tabla 12. Historia de usuario - HU-TUT005	36
Tabla 13. Historia de usuario - HU-TUT017	37
Tabla 14. Historia de usuario - HU-TUT008	38
Tabla 15. Historia de usuario - HU-TUT001	39
Tabla 16. Historia de usuario - HU-TUT011	40
Tabla 17. Prueba de aceptación - PA_APT01	41
Tabla 18. Prueba de aceptación - PA_APT02.....	42
Tabla 19. Prueba de aceptación - PA_APT03.....	43
Tabla 20. Prueba de aceptación - PA_APT04.....	44
Tabla 21. Prueba de aceptación - PA_APT05.....	45
Tabla 22. Prueba de aceptación - PA_APT06.....	46
Tabla 23. Prueba de aceptación - PA_APT07.....	47
Tabla 24. Prueba de aceptación - PA_APT08.....	48
Tabla 25. Prueba de aceptación - PA_APT09.....	49
Tabla 26. Prueba de aceptación - PA_APT10.....	50
Tabla 27. Prueba de aceptación - PA_APT11	51
Tabla 28. Prueba de aceptación - PA_APT12.....	52
Tabla 29. Prueba de aceptación - PA_APT13.....	53
Tabla 30. Prueba de aceptación - PA_APT14.....	54
Tabla 31. Prueba de aceptación - PA_APT15.....	54

1. Capítulo I. INTRODUCCION

1.1. Antecedentes

Conforme avanza la educación, así como avanza la tecnología con el paso del tiempo. Las necesidades de innovar sistemas que se adapten a las nuevas formas para un mejor aprendizaje actual en los estudiantes se vuelven vital.

El Internet, así como dispositivos electrónicos, PCs, dispositivos móviles, tablets, entre otros. Se han convertido en una de las herramientas más importantes y fundamentales para la manipulación de información. Por lo cual estos sistemas tecnológicos están siendo usados en diferentes áreas como la educación.

Existen sistemas informáticos, las cuales ayudan a los estudiantes a llevar un cronograma de manera organizada las diferentes actividades y tareas, así como la gestión de calificaciones y cursos de las diferentes áreas de la cual el estudiante es parte.

Basándonos en la explicación anterior y con el debido conocimiento de que la universidad de las Américas ya cuenta con una plataforma virtual para el manejo de los recursos universitarios de cada uno de los estudiantes; la propuesta es desarrollar un sistema web complementario que pueda solventar las diferentes necesidades e inconvenientes que se han presentado como estudiantes en el tema de tutorías y consultas con docentes.

Este sistema permitirá a cada uno de los estudiantes poder interactuar con los docentes de diferentes áreas, con la finalidad de poder contactarse y establecer horas de tutorías. De igual manera realizar consultas de temas relacionadas al aplicativo en la cual el docente se encuentra capacitado. Todo esto desarrollando un sistema web que permita la comunicación del estudiante con el docente, esta comunicación se realizará mediante un formulario de solicitud de tutorías, el cual contenga una lista detallada de los diferentes docentes de la universidad que estén dentro de las horas de trabajo.

1.2. Alcance

El sistema Web cuenta con una interfaz gráfica que se conecta a una base de datos, con la finalidad de mostrar la información académica de los docentes de cada una de las carreras de la facultad actual. Esta información será clasificada en base a los requerimientos necesarios y de interés general para los estudiantes de las mismas.

Este sistema estará aplicado solamente para los docentes que trabajen a tiempo completo en las carreras de la facultad, debido a que la institución obliga a cumplir un cronograma de tutorías constante hacia los estudiantes.

Los estudiantes podrán conocer dicha información en base a sus necesidades. Digamos que Frank es un estudiante de Ing. en Sistemas y desea conectarse con un docente especialista en base de datos. No conoce el nombre del docente o que docentes pueden ayudarlo con su requerimiento. El sistema permitirá buscar docentes mediante los nombres del docente, especialidad, área entre otras. De esta manera Frank podrá llenar la solicitud de tutoría al docente elegido con el tema, fecha, modulo y mensaje, el cual será notificado en la sección del docente y a su vez enviando un correo al mismo de manera automática con la información de Frank.

Además, el sistema permitirá conocer los horarios en el cual cada uno de los docentes puede realizar tutorías. Cada docente debe cumplir un mínimo de horas establecido por la universidad. Además, el docente cuenta con horas de escritorio, en las cuales el docente está consciente que el estudiante está en todo su derecho de solicitar una tutoría en dichas horas. Los horarios en el cual el docente esté fuera de las horas de clases, así como en las horas asignadas obligatoriamente para tutorías por parte de la institución estarán a la vista de los estudiantes en la plataforma de tutorías.

El estudiante tendrá la facilidad de conocer si uno o varios docentes pueden brindar tutorías sobre el mismo tema, de manera que podrá elegir cual es el docente adecuado que cumpla las expectativas al recibir dicha tutoría. La información del docente estará filtrada desde su área de trabajo hasta las materias que el docente brinda en la Facultad.

Una vez elegido el horario de tutoría disponible entre el estudiante y el docente, el estudiante realizara una solicitud a dicho docente para establecer una confirmación del mismo. Al confirmar el docente esta solicitud, el sistema registrará el nuevo horario de tutorías tanto en el cronograma del docente, como el del estudiante. El sistema notificará con un correo tanto al estudiante como al docente una hora antes de la tutoría como recordatorio de la misma.

El docente como el estudiante llevarán un registro de asistencia de las horas de tutorías, con el fin de que la institución pueda establecer el motivo o circunstancias del cumplimiento o incumpliendo de dichas de las tutorías, y aplicar las soluciones más optimas de la manera adecuada tanto con el estudiante como el docente. El sistema proporcionará un reporte periódico, el cual muestre el estado de cada una de las tutorías las cuales han sido solicitadas por los estudiantes, verificando el cumplimiento o incumplimiento de dicha tutoría. Además, el docente puede notificar la inasistencia por parte del estudiante, quedando un registro para dar seguimiento a uno de los mismos.

En el caso que el docente no pueda asistir a la fecha y solicitada por el estudiante, el docente deberá asignar una nueva fecha y hora para la tutoría solicitada.

Si la universidad permite utilizar dicha información el sistema se conectará a los servidores de la misma. Caso contrario el sistema tendrá una sección de registro tanto para estudiantes como docentes de la universidad. Y estas cuentas serán aprobadas una vez que se constate que la información es válida.

El administrador (Universidad) realizará la carga inicial de las carreras de cada una de las facultades de la UDLA mediante el sistema establecido para él. Para este caso se asignará la facultad FICA con todas las carreras que se encuentran en la misma.

El docente realizará la carga personal de los horarios en los cuales este apto para brindar tutorías, así como la modificación o actualización de dicho cronograma.

Además, el sistema se acoplará con responsive a cada uno de los dispositivos en los cuales el estudiante o docente deseen interactuar, la cual contenga una lista detallada de cada uno de los docentes de la institución, con toda su información académica y distribuida por facultades y carreras.

Finalmente, el sistema contará con una sección de reportes las cuales pueden ser revisadas por el administrador, estos reportes contarán con datos estadísticos mediante gráficos de barras para observar el rendimiento actual de cada docente, así como el rendimiento de cada una de las carreras de la facultad, este rendimiento será por: número de matrícula (primera segunda o tercera), por carrera, por materia por matrícula de estudiante y por docente

1.3. Justificación

Es un sistema muy útil para la institución, ya que en muchas ocasiones los estudiantes tienen que acudir a secretaría académica o a su tutor de carrera para conocer información de uno de los docentes que pueda ayudarle a solventar sus dudas. También hay ocasiones en que el estudiante acude a la oficina del docente, pero el mismo no se encuentra en esta y tiene que esperarlo o buscarlo sin la seguridad que pueda comunicarse con el mismo de inmediato.

Este sistema optimiza tiempo, solventa dudas, y establece horarios de tutorías de manera personalizada, directa e inmediata con cada uno de los docentes por parte el estudiante.

La institución está en beneficio de brindar una mejor calidad de educación, mejorando los servicios institucionales a los estudiantes como docentes de la misma. A su vez es una nueva manera de innovar tecnología en educación y estar a la vanguardia de las dificultades presentadas desde el ámbito educacional. Volviéndose un de los pioneros en solventar necesidades educativas mediante soluciones tecnológicas a nivel nacional.

1.4. Objetivo General

Implementar un sistema que permita la gestión, administración y reserva de tutorías universitarias entre alumno – docente para la Facultad de Ingenierías y Ciencias Aplicadas (FICA), mediante las herramientas de desarrollo Web, con la finalidad de mejorar el sistema académico, proporcionado una nueva manera de innovar los servicios institucionales.

1.4.1. Objetivos Específicos

Análisis de requerimientos necesarios para diseñar, estructura e implementar el sistema de gestión de tutorías universitarias.

Diseñar y estructurar una interfaz Web amigable tanto para el docente como el estudiante, la cual contenga todas las herramientas necesarias y concretas tanto de búsqueda como reserva para el uso académico de los estudiantes.

Implementar un diseño de base de datos el cual permita optimizar los recursos a ser utilizados agilizando el tiempo de respuesta de la información, así como el rendimiento y eficacia del sistema.

Desarrollar una aplicación Web, que organice la información de manera óptima, segura y eficaz, integrando los dos objetivos anteriores las cuales serán ejecutas con herramientas de programación ASP.NET MVC en Visual Studio.

1.4.2. Metodología a Utilizar

El presente proyecto de titulación se utilizará el método SCRUM de la metodología Ágil, debido a la necesidad que se ve generada mediante un análisis de ciertos estudiantes de la institución, por la cual se deduce una problemática en la cual se puede plantear una hipótesis con justificación y estructurar los elementos necesarios para la implementación de una plataforma virtual que soluciones el problema.

2. CAPITULO I. MARCO TEÓRICO

2.1. Metodología de diseño Web – Ágil

2.1.1. SCRUM

Scrum es un modelo de gestión que nos permite desarrollar de una manera interactiva con el usuario, agilitando el proceso de implementación, y a su vez minimizando los diferentes riesgos futuros. (Menzinsky & Gertrudis, 2016)



Figura 1. Metodología Scrum
Adaptado de (Gallego, 2018)

Existen tres roles fundamentales para el equipo de desarrollo:

- ScrumMaster: Es el encargado de dirigir y agilizar las operaciones asignadas al equipo de desarrollo. (Alaimo, 2013)
- Product Owner: Es quien representa la parte del negocio, es decir se encarga de que el equipo de desarrollo maneje los requerimientos del usuario, llegando a cumplir y satisfacer sus necesidades obteniendo el producto adecuado. (Alaimo, 2013)
- Equipo de Desarrollo: Es el encargado del conocimiento técnico necesario para el desarrollo en implementación del proyecto. (Alaimo, 2013)

En el progreso de desarrollo de los proyectos en Scrum se utiliza diferentes etapas de interacción denominadas Sprints.

Un Sprint tiene una duración establecida, generalmente no más de un mes. En este periodo de tiempo se planifica y desarrolla una etapa del proyecto, la cual es construida por los desarrolladores (Product Owner) y presentada a final del periodo cumpliendo todos los objetivos de desarrollo e integración de la etapa, constatando el cumplimiento planificado al inicio del Sprint. (Alaimo, 2013)

Una vez finalizado el Sprints, se planifican nuevos Sprints para continuar con el avance del proyecto, teniendo en cuenta las observaciones y reajustes del Sprint anterior en la nueva etapa.

Las historias de usuarios:

Son todas las descripciones concordes a las funcionalidades que va a tener el sistema a ser implementado. Estas historias forman parte de la interacción entre el cliente y el equipo sujeto a la implementación del proyecto, e irán evolucionando según avanza el mismo, Figura n° 2.

Las Historias de usuarios están compuestas en tres fases denominadas las "3C":

- Card: Es una breve conversación llevada por escrito que servirá como recordatorio.

- Conversation: Es una conversación en la cual se confirma el entendimiento de los requerimientos y se concreta el objetivo.
- Confirmation: Son test funcionales para determinar los detalles más relevantes y fijar límites. (Gallegos, 2018)

Entre los componentes de metodología Scrum encontramos los siguientes:

- Product Backlog: Es un listado de características del producto en base a las historias que serán analizadas y que se desea construir, el cual es supervisado por el Product Owner, Figura N°3. (Alaimo, 2013)
- Sprint Backlog: Es el conjunto de PBIs (Product Backlog Items), seleccionados para trabajar en un sprint conjuntamente con las actividades asignadas al equipo de desarrollo, con la final de incrementar potencialmente el desarrollo. (Alaimo, 2013)
- Incremento funcional potencial entregable: Un sprint es un incremento funcional potencial entregable, ya que es el avance periódico que tiene el proyecto. Hay que tomar en cuenta que tiene que cumplir los requisitos de planificación asignados en cada periodo para continuar con el avance del proyecto. (Alaimo, 2013)
- Dinámica (Flujo de Trabajo): Es la dinámica con la cual se implementa el proyecto mediante organización para el desarrollo por etapas (sprints), optimizando y modificando ciertas inconsistencias o actualizaciones en el transcurso del desarrollo. (Alaimo, 2013)
- Scrum Diario: Permite la interacción continua del personal y el compromiso de las actividades planificadas periódicamente. (Alaimo, 2013)
- Revisión de Sprints: Al finalizar cada Sprint, se realiza una reunión con la finalidad de constatar el cumplimiento del mismo. (Alaimo, 2013)
- Retrospectiva: El equipo de trabajo se reúne para hacer una reflexión sobre el trabajo realizado, tomando en cuenta las fortalezas y dificultades para retroalimentarse y mejorar el trabajo continuo en la siguiente planificación. (Alaimo, 2013).

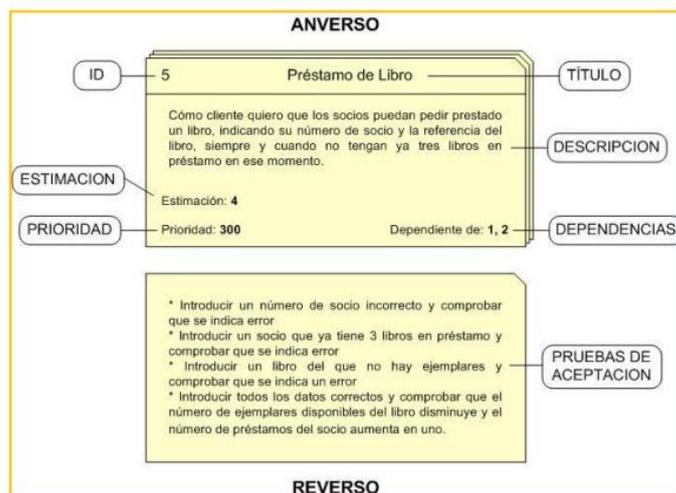


Figura 2. Ejemplo de las Historias de Usuarios
Tomado de (Gallegos, 2018)

ID	PRIORIDAD	DESCRIPCIÓN	ESTIMACIÓN
ESTU-03	MUY ALTA	SECCION DE BUSQUEDA Y PETICIÓN DE TUTORIA	20 Días
DOCE-02	MUY ALTA	SECCION DE CARGA HORARIA	20 Días
ADMIN-01 SUP- ADMIN-01	ALTA	SECCION DE LOGUEO PARA IDENTIFICACION	3 Días
ESTU-01	ALTA	SECCION DE LOGUEO PARA IDENTIFICACION	3 Días
DOCE-01	ALTA	SECCION DE LOGUEO PARA IDENTIFICACION	3 Días
DOCE-03	MEDIA	SECCION DE NOTIFICACIONES	3 Días
ESTU-04	MEDIA	SECCION DE NOTIFICACIONES	3 Días
ESTU-02	MEDIA	PERFIL	3 Días
DOCE-04	MEDIA	PERFIL	3 Días
ADMIN-02 SUP- ADMIN-02	MEDIA	SECCION DE REPORTES	3 Días

Figura 3. Ejemplo de Product Backlog.

2.2. IDE ICESSCRUM

ICESSCRUM es una herramienta de planificación, la cual nos permite organizar las diferentes actividades y tareas del proyecto para la implementación de

software, ayuda a materializar su visión, la comparte con sus equipos y la mantenga actualizada a medida que crece el conocimiento del negocio.

Estas actividades tienen como eje el modelo SCRUM, lo cual podemos organizar por lapsos de tiempo, así como etapas de avances en el proceso de la implementación.

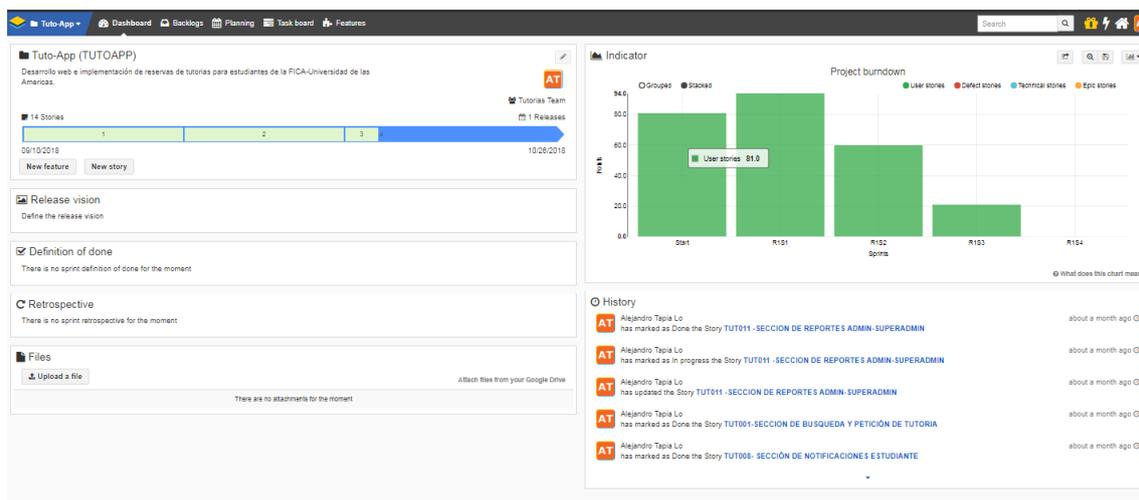


Figura 4. IDE IceScrum

2.3. Aplicaciones Web

2.3.1. Front-End

En desarrollo de software el Front-End es toda la parte de diseño visual, es decir toda la parte interactiva que va a tener los usuarios finales con la aplicación.

En plataformas web se utiliza como lenguaje base o de hipertexto HTML.

Se puede aplicar estilos visuales mediante CSS, y funciones de respuesta inmediata mediante JavaScript o librerías de la misma.

Entre los componentes principales que se va a utilizar para la elaboración del diseño y funcionamiento tenemos:

- HTML, lenguaje base o de hipertexto compatible con los navegadores web.

- CSS, hojas de estilo para un diseño más estético de las páginas.
- JavaScript, lenguaje de programación de lado del cliente para la manipulación de los datos, así como la interactividad de la página.

Estos tres complementos principalmente son fundamentales para un desarrollador de front-end, así el mismo pueda estructurar todo el diseño y desarrollo de la parte del cliente en plataformas web.

2.3.2. Back-End

Se determina back-end toda la parte de software que es “invisible” para el usuario. En esta parte se aplica diferentes lenguajes de programación adaptadas a las necesidades y facilidades de los desarrolladores.

En esta sección se maneja todos los procesos de respuesta a solicitudes hechas por el front-end, así como las diferentes conexiones e interacciones que la aplicación tiene con la base de datos.

2.4. IDE Visual Studio 2017

2.4.1. Modelo MVC 5

Modelo MVC o modelo, vista, controlador, tiene como objetivo independizar cada una de las solicitudes hechas por parte del usuario a la aplicación, con la finalidad de optimizar el tiempo de respuesta y ejecutar solo las sentencias de código solicitadas. (Freeman, 2013)

- Vista: Es la sección de interacción con el usuario, es decir el front-end. Este se ejecuta mediante código HTML, haciendo diferentes solicitudes al controlador.
- Controlador: Obtiene solicitudes de la vista y ejecuta las tareas solicitadas y devuelve una respuesta a la vista (datos).

- Modelo: Es el objeto o clase que depende del controlador para mostrarse en la vista.

El modelo MVC es uno de los modelos con mayor utilidad para aplicaciones de tipo web utilizados. En la figura N° 5 se puede observar un diagrama de flujo de datos basados en modelo MVC.

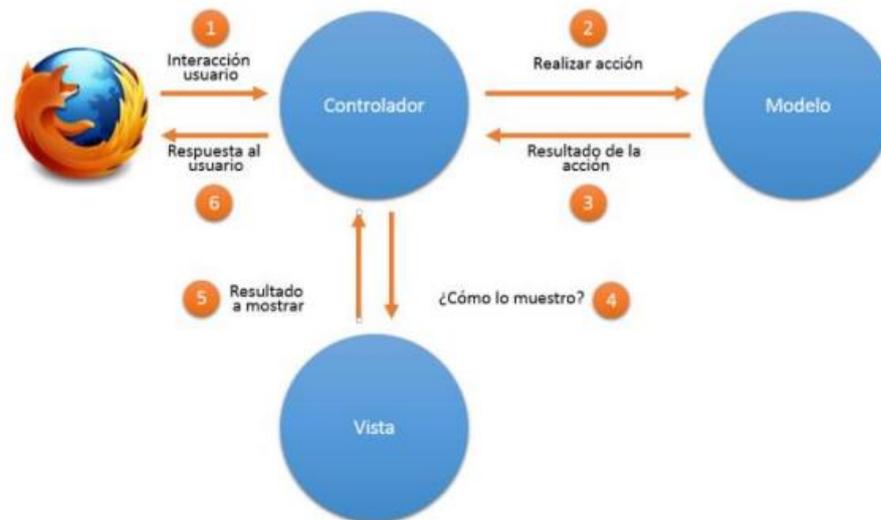


Figura 5. Modelo MVC.
Tomado de (Botella, 2018)

2.4.2. Razor

Es un lenguaje de tipo front-end, el cual combina lenguaje HTML y lenguaje C#, facilitando el uso para desarrolladores de back-end que no tienen mucho conocimiento en front-end. Además, Razor permite trabajar de una manera más ágil al momento de manipular objetos o componentes dinámicos.

El motor de razor compila el código y lo transforma a hipertexto aparentando que todo el front-end fue desarrollado en lenguaje html.

2.4.3. LINQ

Es un lenguaje de manipulaciones datos. Permite obtener, crear y ejecutar las consultas de base de datos mediante la interacción de objetos. LinQ fue diseñado exclusivamente para C#, ya que no había un lenguaje de interpretación que permitiera trabajar directamente con base de datos. (Microsoft, 2015)

2.5. SQL

2.5.1. Consultas, Inserciones y Actualizaciones

Dentro de la gestión de base de datos existen operaciones básicas que nos permiten la interacción y manipulación de los datos, estas son:

- **SELECT:** Realiza consultas de los datos en una tabla.
- **UPDATE:** Actualiza la información de las tablas.
- **DELETE:** Borra Información de las tablas.
- **INSERT:** Inserta nueva información en las tablas. (Oppel & Sheldon, 2010)

2.5.2. Procedimientos Almacenados

Un procedimiento almacenado es una sentencia de código que mantiene una secuencia de procesos ejecutados en un orden, aceptando y devolviendo valores dependiendo del proceso.

Los procedimientos almacenados nos permiten crear acciones específicas para devolver un valor, evitando la redundancia de consultas. Se lo puede comparar con un método en programación orientada a objetos el cual es reutilizado según la necesidad. (Oppel & Sheldon, 2010)

2.6. Seguridad Informática

Al momento de Desarrollar el Sistema de tutorías web, tendremos en cuenta los siguientes aspectos de seguridad con la finalidad de evitar robos o pérdidas de información.

Hay que tener en cuenta estas cuatro áreas fundamentales para evitar que futuras personas o amenazas de software se infiltren en nuestro sistema. Estas áreas son:

- Confidencialidad: Sólo los usuarios autorizados pueden acceder a nuestros recursos, datos e información. (Universidad Internacional de Valencia, 2018)
- Integridad: Sólo los usuarios autorizados deben ser capaces de modificar los datos cuando sea necesario. (Universidad Internacional de Valencia, 2018)
- Disponibilidad: Los datos deben estar disponibles para los usuarios cuando sea necesario. (Universidad Internacional de Valencia, 2018)
- Autenticación: Estás realmente comunicándote con los que piensas que te estás comunicando. (Universidad Internacional de Valencia, 2018)

Cabe destacar que por más seguro que sea el sistema, el recurso humano es un punto por el cual pueden vulnerar nuestros sistemas. Por lo cual es necesario capacitar a los usuarios que van a manipular el mismo.

3. CAPITULO II. DESCRIPCION DEL PROYECTO ANALISIS Y DISEÑO

En este capítulo se detalla cada uno de los requerimientos del sistema, así como los roles y secciones que tiene la aplicación web, con el fin de aplicar la metodología scrum para evaluar el nivel de dificultad de cada una de las secciones del sistema. De este modo se asignará un tiempo en base a la puntuación obtenida y se le otorgará un nivel de prioridad para la realización.

3.1 . Sprints

Para poder tener un correcto manejo del proyecto con metodología se determinará las diferentes historias de usuario las cuales nos servirá para manejar de forma rápida y correcta el desarrollo.

Para lo cual se procede a describir en orden los siguientes Sprint.

3.1.1 Sprint 1

En el primer sprint corresponde al reconocimiento del problema y la situación inicial, evaluación y síntesis, modelo de solución, especificaciones del desarrollo del Product Backlog para posteriormente dividir en los siguientes sprints.

3.1.1.1 Reconocimiento del problema

Muestra los roles del sistema, así como la funcionalidad de las secciones, con el fin de organizar las diferentes partes del sistema para su desarrollo. Permite organizar el tiempo y prioridad para mejorar la efectividad, así el equipo de desarrollo trabaja con una mejor manera y organización.

PROYECTO: **SISTEMA DE** COD: TUT01 (SCRUM 0)

GESTIÓN DE TUTORÍAS

ACADÉMICAS

Reconocimiento del Problema SEPTIEMBRE -01

1. AMBITO DEL PROYECTO

OBJETO

El presente sistema se enmarca dentro del plan de Desarrollo de Software, con la finalidad de automatizar el mecanismo de registro, control y gestión de tutorías de FICA, siendo capaz de simplificar las actividades llevados hasta el momento por los docentes y los estudiantes, y a su vez formando parte del sistema universitario de la UDLA sin interrumpir su funcionamiento.

Desde el punto de vista estratégico de la Universidad, Este sistema es requerido para mejorar el sistema de enseñanza académica, fortaleciendo la confianza, así como el desempeño del estudiante.

Desde el punto de vista informático, el sistema debe permitir:

- Realizar coexistir con las plataformas universitarias y formar parte de ellas.
- Integrar los esfuerzos necesarios para que el equipo de trabajo cumpla con las expectativas y solicitudes que el sistema de tutorías requiera.
- Cumplir los requerimientos necesarios para que el sistema funcione de manera correcta y cumpla con las necesidades tanto de los docentes como de los estudiantes en beneficio con la universidad.

3.1.1.2 Evaluación y síntesis

Establecemos un análisis de las diferentes secciones del sistema, las cuales son requeridas en base a las necesidades y problemáticas presentadas.

Se denota una calificación de baja, medio o alta con la finalidad de evaluar el nivel de exigencia que se presenta al realizar el desarrollo del sistema.

Esta calificación estará presente en cada uno de los Sprint que se encuentra en la sección de historias de usuario que empiezan en el capítulo 2.

3.1.1.3 Arquitectura

En base al análisis y evaluación de los requerimientos se genera un modelo de solución, el cual presenta la funcionalidad del sistema de tutorías, mediante un diagrama de procesos en las cuales están seccionadas las diferentes áreas.

En el desarrollo de mi proyecto como antes se mencionó se utilizará una arquitectura modelo, vista, controlador (MVC) ya que las ventajas que tiene este framework nos ayuda a separar la lógica del producto y la interfaz del usuario, que permitirán tener un mejor análisis y desempeño en la aplicación web, Figura N° 6.

3.1.1.3.1 Modelo

Contiene una representación de los datos que maneja el sistema. En el modelo es en donde trabajaremos con los diferentes métodos para manejar la información del proyecto.

3.1.1.3.2 Vista

Es la interfaz gráfica de los diferentes módulos que contaremos en la aplicación. En el desarrollo web se utilizó HTML CSS Y JAVASCRIPT, para desplegar los datos al usuario. En la aplicación se usó html para que nuestra web sea responsive, lo cual nos permitirá navegar desde cualquier dispositivo de forma amigable por la interfaz.

3.1.1.3.3 Controlador

Nos ayuda a actuar como intermediario entre los modelos y vistas, gestionando el flujo de información entre ellos para manejar la lógica del negocio.

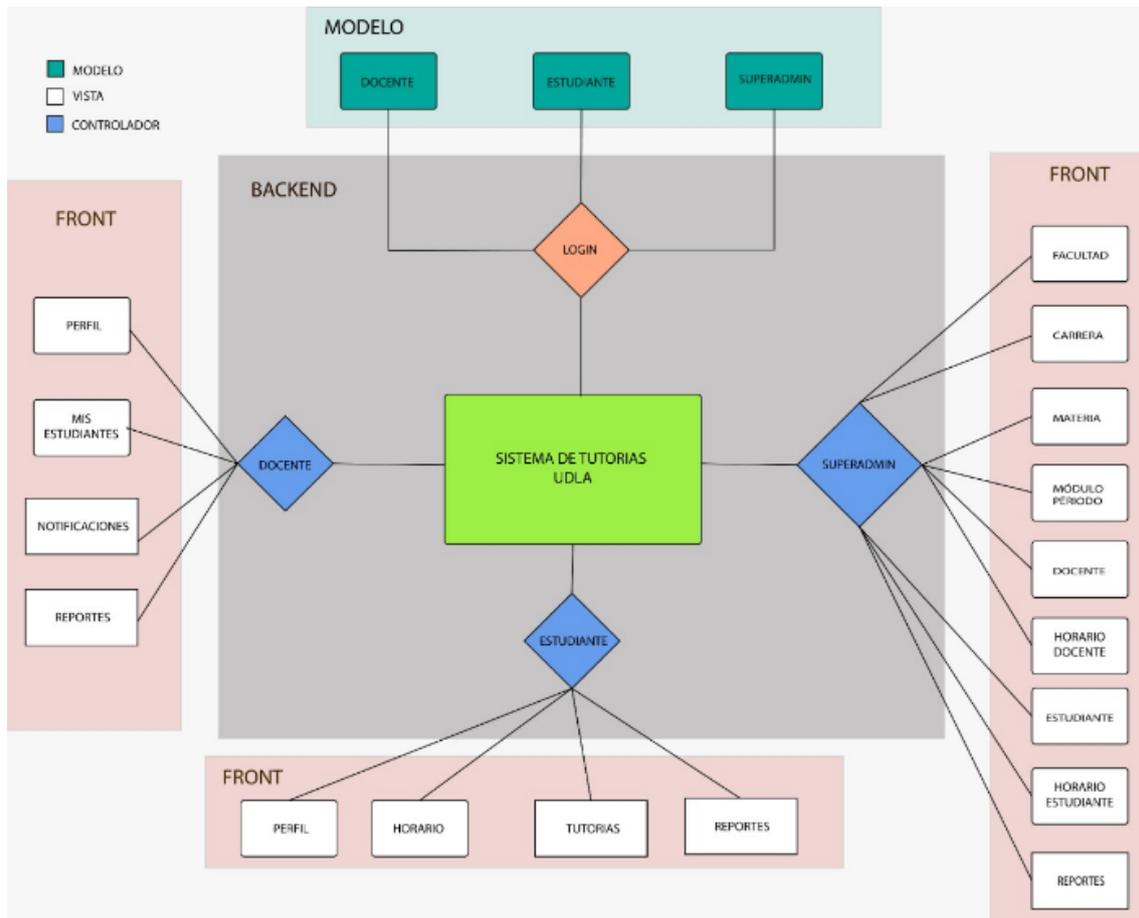


Figura 6. Arquitectura del desarrollo.

3.1.1.4 Especificación

En base al modelo de la solución generado anteriormente se especifican las diferentes secciones, dividido en dos grupos:

BACKEND

El backend es el modelo de funcionamiento del sistema que abarca toda la gestión de los procesos. Controla las diferentes secciones en base a los requerimientos de las diferentes entidades (ESTUDIANTE, DOCENTE, ADMINISTRADOR).

Secciones

- Login(Proceso)
- Manipulación de eventos de las entidades (Acciones de los usuarios en el sistema).
- Código de programación (ASP.NET Y SQL SERVER).
- Seguridades.
- Optimización (Buenas prácticas de desarrollo)

FRONTEND

El frontend es la capa visual en la cual maneja la experiencia de usuario de las diferentes entidades, facilitando de uso del sistema.

Secciones:

- Login(Vista)
- Interacción de componentes (botones, formularios, enlaces).
- Lenguaje de etiquetado (HTML).
- Lenguaje de estilos (CSS).
- Lenguaje de Interactividad (JavaScript).
- Optimización y Rendimiento (Simple Page Aplicación SPA).
- Vistas Parciales y Componentes.

3.1.1.5 Product Backlog

Esta sección muestra la lista de trabajo pendiente separado en sprints. Los cuales fueron realizados concorde al cronograma de actividad de la metodología scrum.

Rank	Description	Cost
★ 2	TUT002 -SECCIÓN DE CARGA HORARIA DOCENTE	21 \$
★ 1	TUT001-SECCION DE BUSQUEDA Y PETICIÓN DE TUTORIA	21 \$
★ 3	TUT004 -SECCIÓN DE LOGUEO PARA IDENTIFICACIÓN SUPER ADMIN-ADMIN	5 \$
★ 4	TUT005 -SECCION DE LOGUEO PARA IDENTIFICACION DOCENTE	5 \$
★ 5	TUT006 -SECCION DE LOGUEO PARA IDENTIFICACION ESTUDIANTE	5 \$
★ 6	TUT007 -SECCION DE NOTIFICACIONES DOCENTE	8 \$
★ 7	TUT008- SECCIÓN DE NOTIFICACIONES ESTUDIANTE	8 \$
★ 8	TUT009 -PERFIL DOCENTE	21 \$
★ 9	TUT003 -SECCIÓN DE CARGA HORARIA ESTUDIANTE	5 \$
★ 10	TUT010 -PERFIL DOCENTE	21 \$
★ 11	TUT011 -SECCION DE REPORTES ADMIN-SUPERADMIN	21 \$
★ 12	TUT012 -CARGA AUTOMATICA EXCEL DOCENTES	5 \$
★ 13	TUT013 -CARGA AUTOMATICA EXCEL ESTUDIANTES	5 \$
★ 14	TUT014 - CARGA AUTOMATICA HORARIO DOCENTE EXCEL	5 \$
★ 15	TUT015 - CARGA AUTOMATICA HORARIO ESTUDIANTE EXCEL	5 \$
★ 16	TUT016 -CARGA-FACULTAD-CARRERA-MATERIAS-PERIODO-MODULOS	8 \$

Figura 7. Product Backlog

Burndown Chart

Seccionaremos el trabajo por tiempo, todas las actividades del backlog serán representadas en días, dando un total de días en realizar las actividades especificadas del backlog.

Con este ejercicio se mide el avance en el tiempo en base el cronograma programado, dándonos un tiempo de estimación, así como el progreso actual.

Tabla 1.
Burndown chart del proyecto

Tareas / Horas	Iniciales	utilizadas Día 5	utilizadas Día 4	utilizadas Día 3	utilizadas Día 2	utilizadas Día 1	TOTAL
HU - TUT001	21	6	1	4	7	4	22
HU - TUT002	21	6	4	3	4	4	21
HU - TUT003	5	2	2	1		1	6
HU - TUT004	5		1	1	1	2	5
HU - TUT005	5	1		1	2	1	5
HU - TUT006	8	2	1	2	1	2	8
HU - TUT007	8	2	1	2		2	7
HU - TUT008	21	6	4		4	4	18
HU - TUT009	5	1		1	2	1	5
HU - TUT010	21	4	4	2	2	4	16
HU - TUT011	21	4	4	4	4	4	20
HU - TUT012	5	1		1	3	1	6
HU - TUT013	5		2		1	1	4
HU - TUT014	5	2			1	1	4
HU - TUT015	5	2	0	1	1	1	5

HU - TUT015	8	1	1	2	2	2	8
Horas Actuales	169	122	98	75	42	9	346
Horas Estimadas	169	128.8	96.6	64.4	32.2	0	322

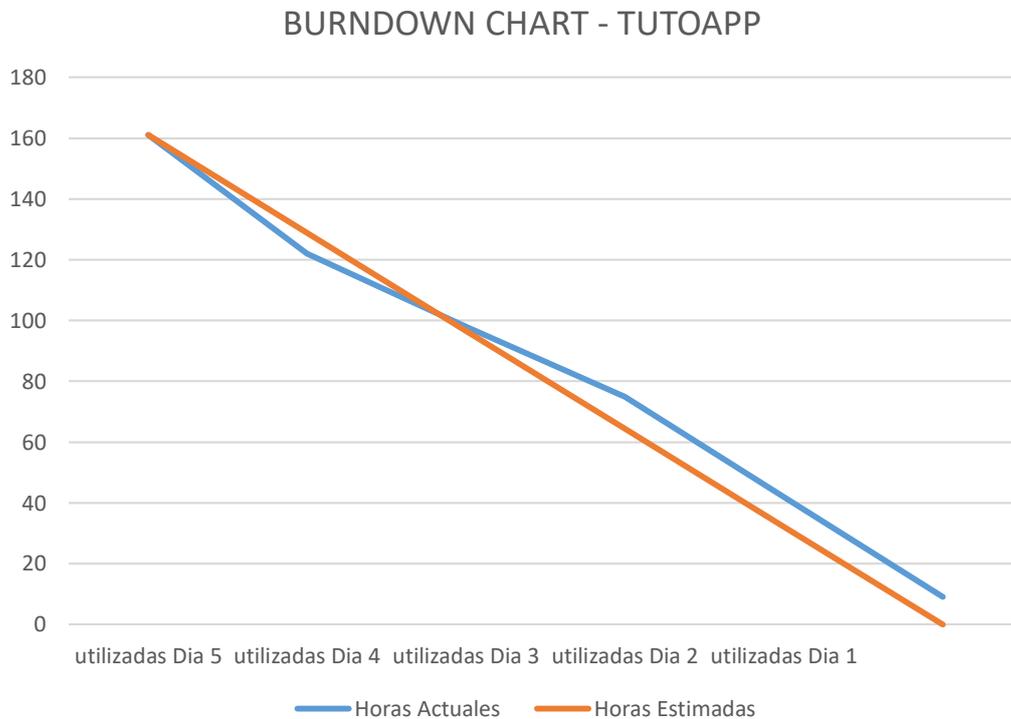


Figura 8. Burndown chart del proyecto

4. CAPITULO III. DESARROLLO CODIGO Y PRUEBAS

Describe el proceso de análisis de datos en el cual se plantea una estructuración de diseño que solucionará tanto el Diseño visual, como la estructura operacional del proyecto.

4.1 Sprint 2.

En este Spring empezaremos con una de las cosas más importantes de la aplicación el desarrollo de los módulos de las diferentes cargas de datos tales

como: el ingreso de facultad carrera, módulos, periodo adicional carga horarias en el docente carga horaria del estudiante de forma manual y automática se agrega también.

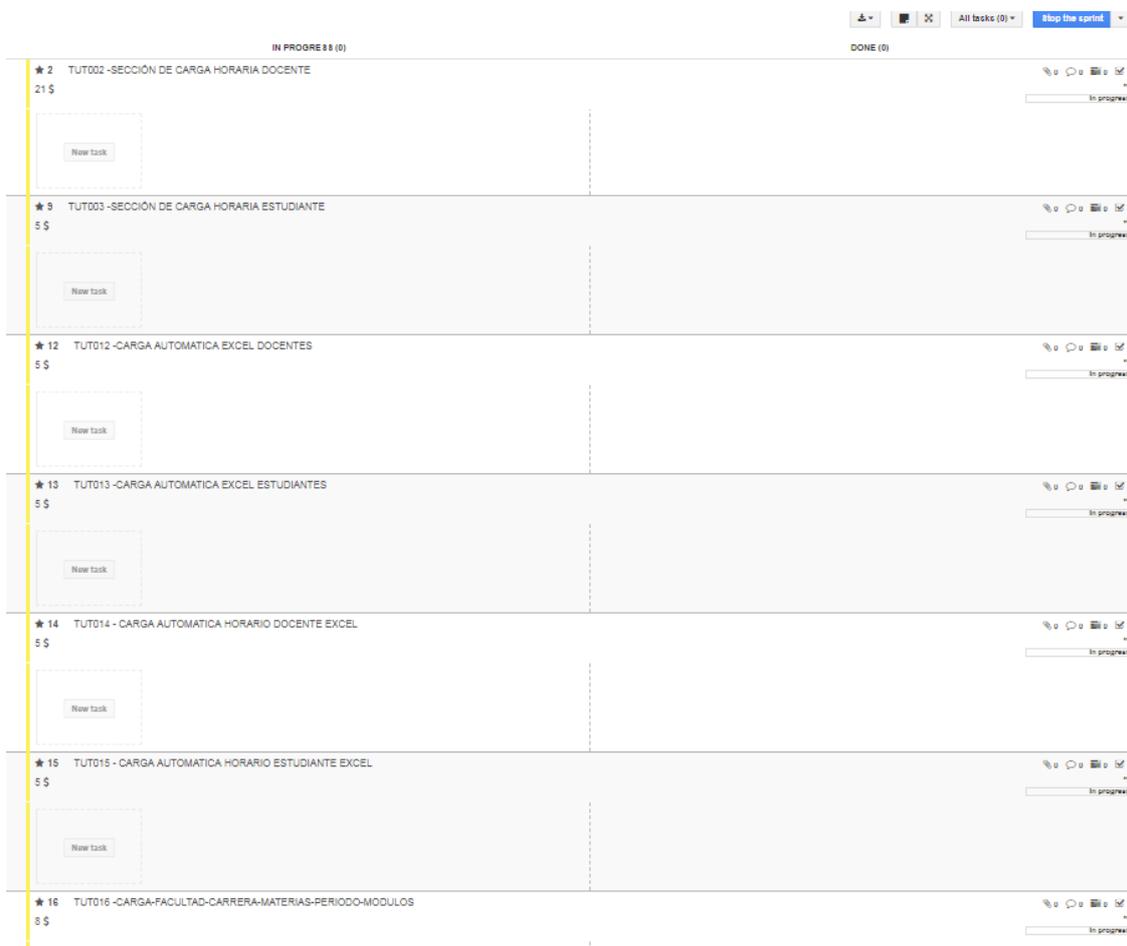


Figura 9. Spring 2

4.1.1 Carga facultad, carreras, módulos, periodo.

Se realiza el desarrollo de la carga de información descrita la cual se realiza manualmente dentro de cada módulo correspondiente.

4.1.1.1 Historia de usuario - HU-TUT016

Tabla 2.

Historia de usuario - HU-TUT016

HISTORIA DE USUARIO	
ID: HU-TUT016	Usuario: Admin
Nombre Historia: TUT016 -CARGA-FACULTAD-CARRERA-MATERIAS-PERIODO-MODULOS	
Prioridad: Media	Obtenido por: Andrés Tapia
	Modulo Asignado: Facultad Carrera Materias Módulos Periodo
OBJETO: Admin	
<p>En la carga de información del perfil admin el usuario cargar toda la información correspondiente a facultad carreras materias periodo y módulos los mismo que son administrables por el usuario e ingresados de forma manual.</p>	



Figura 10. Carga facultad carga carreras materias periodo y módulos

4.1.2 Carga automática Excel Docentes

Se procede con el desarrollo para que el perfil admin realice la carga de manera automática a través de un Excel de la información de Docentes.

4.1.2.1 Historia de usuario - HU-TUT012

Tabla 3.

Historia de usuario - HU-TUT012

HISTORIA DE USUARIO	
ID: HU-TUT012	Usuario: Docente
Nombre Historia: TUT012 -CARGA AUTOMATICA EXCEL DOCENTES	
Prioridad: Media	Obtenido por: Andrés Tapia
	Modulo Asignado: Admin
OBJETO: Docente	
<p>En la carga de información del perfil admin el usuario cuenta con la funcionalidad de cargar los datos correspondientes a cada docente de manera automática seleccionando un archivo que ingresara los datos en nuestras tablas.</p>	

DOCENTE

Ingresar, actualiza o elimina las Docentes actuales de la Universidad

Primer Nombre	Segundo Nombre	Apellido Paterno	Apellido Materno	Cedula
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Email	Celular	<input type="button" value="Ingresar"/>		
<input type="text"/>	<input type="text"/>			

Cargar Docentes desde Excel

No se eligió archivo

Figura 11. Carga Docentes archivo

4.1.3 Carga automática Excel Estudiantes

Se procede con el desarrollo para que el perfil admin realice la carga de manera automática a través de un Excel de la información de Estudiantes.

4.1.3.1 Historia de usuario - HU-TUT013

Tabla 4.

Historia de usuario - HU-TUT013

HISTORIA DE USUARIO	
ID: HU-TUT013	Usuario: Estudiante
Nombre Historia: TUT013 -CARGA AUTOMATICA EXCEL ESTUDIANTES	
Prioridad: Media	Obtenido por: Andrés Tapia
	Modulo Asignado: Admin
OBJETO: Docente	
<p>En la carga de información del perfil admin el usuario cuenta con la funcionalidad de cargar los datos correspondientes a cada estudiante de manera automática seleccionando un archivo que ingresara los datos en nuestras tablas.</p>	

Estudiante

Facultad	Carrera	Matricula	Primer Nombre	Segundo Nombre
FICA ▾	INGENIERIA DE SOFTWARE ▾	<input type="text"/>	<input type="text"/>	<input type="text"/>
Apellido Paterno	Apellido Materno	Email	Celular	<input type="button" value="Ingresar"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	

Cargar Estudiantes desde Excel

<input type="button" value="Seleccionar archivo"/>	<input type="button" value="No se eligió archivo"/>	<input type="button" value="Cargar Lista"/>
--	---	---

Figura 12. Carga Estudiantes archivo

4.1.4 Carga horario Docentes

Se desarrollará el módulo de la carga manual de horario docentes en el cual se asigna diferentes campos para poder ingresar la información.

4.1.4.1 Historia de usuario - HU-TUT002

Tabla 5.

Historia de usuario - HU-TUT002

HISTORIA DE USUARIO	
ID: HU-TUT002	Usuario: Docente
Nombre Historia: TUT002 -SECCIÓN DE CARGA HORARIA DOCENTE	
Prioridad: Media	Obtenido por: Andrés Tapia
	Modulo Asignado: Admin
OBJETO: Docente	
<p>Esta sección permite realizar las cargas horarias del docente mediante el usuario admin, el cual se ingresa a la plataforma y el sistema organiza la información en las subsecciones de: horas de clase campos como:</p> <ul style="list-style-type: none"> • Facultad • Carrera • Materias • Módulos • Periodos • Aula • Paralelo 	

Horario de Clases del Docente por Materia y Modulo

Ingresa o elimina la hora de clase del docente actual, en cada una de las materias correspondientes a cada carrera de la facultad.

Facultad	Carrera	Materia	Docente	Modulo
FICA ▾	INGENIERÍA DE SOFTWARE ▾	FISICA I ▾	ANDRES ALEJANDRO TAPIA LOPEZ ▾	101 Lunes 7:00 - 8:00 ▾
Periodo Academico	Aula	Paralelo		
SEPTIEMBRE 2018 ▾	<input type="text"/>	<input type="text"/>	<input type="button" value="Ingresar"/>	

Figura 13. Carga Horario Docente

4.1.5 Carga horario Estudiantes

Se desarrollará el módulo de la carga manual de horario estudiante en el cual se asigna diferentes campos para poder ingresar la información.

4.1.5.1 Historia de usuario - HU-TUT003

Tabla 6.

Historia de usuario - HU-TUT003

HISTORIA DE USUARIO	
ID: HU-TUT003	Usuario: Estudiante
Nombre Historia: TUT003 -SECCIÓN DE CARGA HORARIA ESTUDIANTE	
Prioridad: Media	Obtenido por: Andrés Tapia
	Modulo Asignado: Admin
OBJETO: Estudiante	
Esta sección permite realizar las cargas horarias del estudiante mediante el usuario admin, en el horario se debe seleccionar el estudiante, en base a qué carrera esté inscrito, mostrar las materias, al seleccionar la materia se desplegara los docentes de la materia, el sistema internamente debe consultar los módulos que se dicta en el horario docente y cargar al estudiante.	

Horaio de Clases General

Ingresa o elimina la hora de clase del estudaite actual, en cada una de las materias correspondientes al mismo. Asigna un paralelo y un nu

Facultad	Carrera	Estudiante	
FICA ▾	INGENIERÍA DE SOFTWARE ▾	DAVID ARMANDO CHAGUARO VASCONEZ ▾	
Materia	Paralelo	Docente	Numero de Matricula
FISICA I ▾	1 ▾	SANTIAGO RAMIRO VILLARREAL NARVAEZ ▾	1 ▾
			Ingresar

Figura 14. Carga Horario Estudiante

4.1.6 Carga automática horario Docente

Se desarrolla la funcionalidad de carga automática mediante un archivo para el horario del docente.

4.1.6.1 Historia de usuario - HU-TUT014

Tabla 7.

Historia de usuario - HU-TUT014

HISTORIA DE USUARIO	
ID: HU-TUT014	Usuario: Docente
Nombre Historia: TUT014 - CARGA AUTOMATICA HORARIO DOCENTE	
Prioridad: Media	Obtenido por: Andrés Tapia
	Modulo Asignado: Admin
OBJETO: Docente	
Esta sección permite realizar las cargas horarias del docente mediante el usuario admin, el cual se ingresa mediante un archivo de Excel los datos en nuestra base de datos para formar el horario del docente	

Cargar Horarios de Docentes desde Excel

Selecciona el Docente Ingresado para consultar su horario

Docente:
 Materia:

Hora	Lunes	Martes	Miercoles	Jueves	Viernes	Sabado
07:00 - 08:00						
08:05 - 09:05						
09:10 - 10:10				FIS100-2/456		
10:15 - 11:15				FIS100-2/456		
11:20 - 12:20						
12:25 - 13:25						
ALMUERZO						
13:30 - 14:30						
14:35 - 15:35						
15:40 - 16:40						
16:45 - 17:45						
17:50 - 18:50						
18:50 - 19:50						
19:50 - 20:50						
20:50 - 21:50						

Figura 15. Carga Automática Horario Docente

4.1.7 Carga automática horario Estudiante

Se desarrolla la funcionalidad de carga automática mediante un archivo para el horario del estudiante.

4.1.7.1 Historia de usuario - HU-TUT015

Tabla 8.

Historia de usuario - HU-TUT015

HISTORIA DE USUARIO	
ID: HU-TUT015	Usuario: Docente
Nombre Historia: TUT015 - CARGA AUTOMATICA HORARIO ESTUDIANTE	
Prioridad: Media	Obtenido por: Andrés Tapia
	Modulo Asignado: Admin
OBJETO: Estudiante	
Esta sección permite realizar las cargas horarias del docente mediante el usuario admin, el cual se ingresa mediante un archivo de Excel los datos en nuestra base de datos para formar el horario del estudiante	

Cargar Horarios de Estudiantes desde Excel

Selecciona el Estudiante Ingresado para consultar su horario

Estudiante:
 Materia:

Horario de Clases

Conoce las materias y horas de clases, no te descuides de tu horario.

Hora	Lunes	Martes	Miercoles	Jueves	Viernes	Sabado
07:00 - 08:00						
08:05 - 09:05						
09:10 - 10:10						
10:15 - 11:15						
11:20 - 12:20						
12:25 - 13:25						
ALMUERZO						
13:30 - 14:30						
14:35 - 15:35						
15:40 - 16:40						
16:45 - 17:45						
17:50 - 18:50						
18:50 - 19:50						
19:50 - 20:50						
20:50 - 21:50						

Figura 16. Carga Automática Horario Estudiante

4.2 Sprint 3.

En este sprint vamos a trabajar en el perfil del docente lo que implica el desarrollo de login del docente, datos del docente, notificaciones del docente, horario del docente que formara parte del perfil.

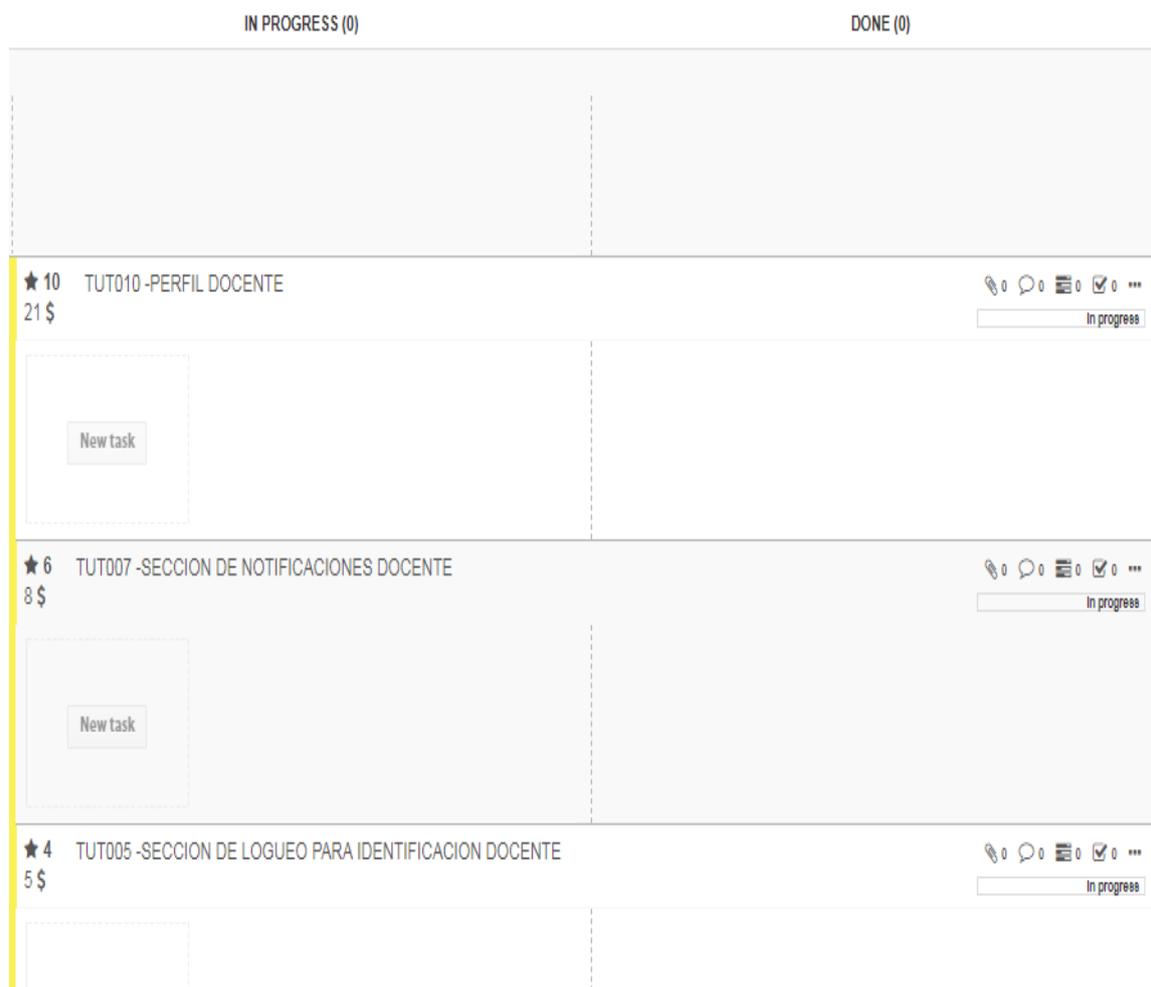


Figura 17. Sprint 3

4.2.1 Sección de Login para identificación del docente.

En este módulo se va desarrollar la función del login usando el api de office 365 para autenticación y asignación de rol.

4.2.1.1 Historia de usuario - HU-TUT005

Tabla 9.

Historia de usuario - HU-TUT005

HISTORIA DE USUARIO	
ID: HU-TUT005	Usuario: Docente
Nombre Historia: TUT005 -SECCION DE LOGUEO PARA IDENTIFICACION DOCENTE	
Prioridad: Media	Obtenido por: Andrés Tapia
	Modulo Asignado: Docente
OBJETO: Docente-Login	
Esta sección se trabajó en la sección del login con el api de office 365, el cual se validara con el registro que este ingresado en la base de datos, el usuario ingresa el correo y contraseña lo cual le dirige a su perfil respectivo	



Figura 18. Login Docente

4.2.2 Perfil Docente

En este módulo se trabaja en el desarrollo del perfil del Docente en donde se desarrolló parte del horario, que el usuario pueda agregar- modificar sus horas libres y horas obligatorias de tutorías.

4.2.2.1 Historia de usuario - HU-TUT010

Tabla 10.

Historia de usuario - HU-TUT010

HISTORIA DE USUARIO	
ID: HU-TUT010	Usuario: Docente
Nombre Historia: TUT010 –PERFIL DOCENTE	
Prioridad: Media	Obtenido por: Andrés Tapia
	Modulo Asignado: Docente
OBJETO: Docente	
Con la autenticación del docente la información del usuario se despliega en el perfil del docente, adicional en este módulo muestra el horario actual de clases, el docente podrá interactuar con el mismo par así asignar horas libres y horas de tutorías obligatorias.	

Datos Personales

ANDRES ALEJANDRO Nombre	TAPIA LOPEZ Apellidos
andres.tapia.lopez@udla.edu.ec Email	988865478 Celular
1720645223 Cedula	

Ingresar/Eliminar Horas libres

Horas Disponibles Lunes 9:10 - 10:10

Ingresar/Elim Horas Obligatorias

Horas Disponibles Lunes 9:10 - 10:10

Hora	Lunes	Martes	Miercoles	Jueves	Viernes	Sabado
07:00 - 08:00	MAT330-2/504					
08:05 - 09:05	MAT330-2/415					
09:10 - 10:10						
10:15 - 11:15						
11:20 - 12:20						
12:25 - 13:25						
ALMUERZO						
13:30 - 14:30						
14:35 - 15:35						
15:40 - 16:40						
16:45 - 17:45						
17:50 - 18:50						
18:50 - 19:50						
19:50 - 20:50						
20:50 - 21:50						

Figura 19. Perfil Docente

4.2.3 Sección de notificación Docentes.

Se desarrolló el módulo de notificaciones dentro del perfil del docente.

4.2.3.1 Historia de usuario - HU-TUT007

Tabla 11.

Historia de usuario - HU-TUT007

HISTORIA DE USUARIO	
ID: HU-TUT007	Usuario: Docente
Nombre Historia: TUT007 – SECCION DE NOTIFICACIONES DOCENTE	
Prioridad: Media	Obtenido por: Andrés Tapia
	Modulo Asignado: Docente
OBJETO: Docente Dentro de esta funcionalidad el docente podrá entrar a su perfil y revisar las notificaciones-mensajes de sus estudiantes para la solicitud de tutorías, la misma que servirá para revisión y aprobación de la tutoría.	



Figura 20. Notificación Docente

4.3 Sprint 4.

En este sprint vamos a trabajar en el perfil del estudiante lo que implica el desarrollo de login del estudiante, datos del estudiante, notificaciones del estudiante, horario del estudiante que formara parte del perfil y el desarrollo de la petición y reserva de tutoría del estudiante.

Task ID	Description	Estimate	Status
★ 1	TUT001-SECCION DE BUSQUEDA Y PETICIÓN DE TUTORIA	21 \$	Estimated
★ 3	TUT004-SECCIÓN DE LOGUEO PARA IDENTIFICACIÓN SUPER ADMIN-ADMIN	5 \$	Estimated
★ 5	TUT006-SECCION DE LOGUEO PARA IDENTIFICACION ESTUDIANTE	5 \$	Estimated
★ 7	TUT008- SECCIÓN DE NOTIFICACIONES ESTUDIANTE	8 \$	Estimated
★ 11	TUT011 -SECCION DE REPORTES ADMIN-SUPERADMIN	21 \$	Estimated
★ 4	TUT005-SECCION DE LOGUEO PARA IDENTIFICACION DOCENTE	5 \$	Done
★ 16	TUT016 -CARGA-FACULTAD-CARRERA-MATERIAS-PERIDO-MODULOS	8 \$	Done
★ 10	TUT010 -PERFIL DOCENTE	21 \$	Done
★ 12	TUT012 -CARGA AUTOMATICA EXCEL DOCENTES	5 \$	Done
★ 6	TUT007 -SECCION DE NOTIFICACIONES DOCENTE	8 \$	Done
★ 13	TUT013 -CARGA AUTOMATICA EXCEL ESTUDIANTES	5 \$	Done
★ 2	TUT002 -SECCIÓN DE CARGA HORARIA DOCENTE	21 \$	Done
★ 9	TUT003 -SECCIÓN DE CARGA HORARIA ESTUDIANTE	5 \$	Done
★ 14	TUT014 - CARGA AUTOMATICA HORARIO DOCENTE EXCEL	5 \$	Done
★ 15	TUT015 - CARGA AUTOMATICA HORARIO ESTUDIANTE EXCEL	5 \$	Done

Figura 21. Sprint 4

4.3.1 Sección de Login para identificación del estudiante.

En este módulo se va desarrollar la función del login usando el api de office 365 para autenticación y asignación de rol.

4.3.1.1 Historia de usuario - HU-TUT006

Tabla 12.

Historia de usuario - HU-TUT005

HISTORIA DE USUARIO	
ID: HU-TUT006	Usuario: Estudiante
Nombre Historia: TUT006 -SECCION DE LOGUEO PARA IDENTIFICACION ESTUDIANTE	
Prioridad: Media	Obtenido por: Andrés Tapia
	Modulo Asignado: Estudiante
OBJETO: Estudiante-Login	
Esta sección se trabajó en la sección del login con el api de office 365, el cual se validara con el registro que este ingresado en la base de datos, el usuario ingresa el correo y contraseña lo cual le dirige a su perfil respectivo en este caso el del estudiante	



Figura 22. Login Docente

4.3.2 Perfil Estudiante.

En este módulo se trabaja en el desarrollo del perfil del estudiante en donde se desarrolló la información y horario del estudiante.

4.3.2.1 Historia de usuario - HU-TUT017

Tabla 13.

Historia de usuario - HU-TUT017

HISTORIA DE USUARIO	
ID: HU-TUT017	Usuario: Docente
Nombre Historia: TUT017 –PERFIL ESTUDIANTE	
Prioridad: Media	Obtenido por: Andrés Tapia
	Modulo Asignado: Docente
OBJETO: Estudiante	
Con la autenticación del estudiante la información del usuario se despliega en el perfil del estudiante, adicional en este módulo muestra el horario actual de clases.	

 **Joan Sebastian Andrade Casco** Perfil Horario Tutorías ▲

Mi Perfil
Hechale un vistazo a tu información personal

Nombres JOAN SEBASTIAN	Apellidos ANDRADE CASCO	Email joan.andrade@udla.edu.ec
Celular 978616500	Matricula 501735	Carrera INGENIERÍA DE SOFTWARE

Figura 23. Perfil Estudiante

Horario de Clases

Conoce las materias y horas de clases, no te descuides de tu horario.

Hora	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
07:00 - 08:00						
08:05 - 09:05						
09:10 - 10:10						
10:15 - 11:15						
11:20 - 12:20						
12:25 - 13:25						
ALMUERZO						
13:30 - 14:30						
14:35 - 15:35						
15:40 - 16:40						
16:45 - 17:45						
17:50 - 18:50						
18:50 - 19:50						
19:50 - 20:50						
20:50 - 21:50						

Figura 24. Horario Estudiante

4.3.3 Notificaciones Estudiante.

Se desarrolló el módulo de notificaciones dentro del perfil del docente.

4.3.3.1 Historia de usuario - HU-TUT008

Tabla 14.

Historia de usuario - HU-TUT008

HISTORIA DE USUARIO	
ID: HU-TUT008	Usuario: Estudiante
Nombre Historia: TUT008 – SECCION DE NOTIFICACIONES ESTUDIANTE	
Prioridad: Media	Obtenido por: Andrés Tapia
	Modulo Asignado: Estudiante
OBJETO: Estudiante	
Dentro de esta funcionalidad el estudiante podrá verificar y ser notificado que tutorías tiene pendientes de realizar para su posterior asistencia, las mismas serán reflejadas en una tabla de tutorías	

Tutorías

No te descuides de tus tutorías, revisa el cronograma constantemente.

Cod Materia	Materia	Docente	Modulo	Periodo	Estado Obligatorio	Fecha	Aula
FIS100	FISICA I	SANTIAGO RAMIRO VILLARREAL NARVAEZ	110 Lunes 15:40-16:40	PEP2018	SI	04/01/2019	465

Figura 25. Notificación Estudiante

4.3.4 Búsqueda y petición de tutorías.

En este módulo se desarrollará la petición de la tutoría en el perfil del estudiante hacia el docente.

4.3.4.1 Historia de usuario - HU-TUT001

Tabla 15.

Historia de usuario - HU-TUT001

HISTORIA DE USUARIO	
ID: HU-TUT001	Usuario: Estudiante
Nombre Historia: TUT001-SECCION DE BUSQUEDA Y PETICIÓN DE TUTORIA	
Prioridad: Alta	Obtenido por: Andrés Tapia
	Modulo Asignado: Estudiante
OBJETO: Estudiante Dentro de esta funcionalidad el estudiante podrá realizar la petición de la tutoría de acuerdo a las materias en las cuales se encuentre matriculado, cada materia cuenta con varios docentes por lo que el estudiante deberá elegir un docente, y validando con el horario del docente podrá seleccionar el modulo en el cual desea realizar la petición-.	

Aprende

Busca un docente de la materia que deseas recibir una tutoria, programa el día y la hora que el docente tenga disponible, y espera su confirmacion.

SOLICITUD DE TUTORIA

Llena la informacion solicitada

Elige una materia

▼

Elige al docente de tu carrera que tengas mas confianza
 Deseas conocer un poco mas de tu docente, selecciona su nombre.

Horario de

Selecciona un modulo, estos modulos estan disponibles para el Docente

▼

Fecha de la tutoria
 05/01/2019

Escribe un mensaje explicando el motivo de la tutoria con el tema que tienes problemas.

Enviar

Figura 26. Búsqueda y petición de tutoría

4.4 Sprint 5.

En este Sprint se va desarrollar el módulo de reportes, el cual permitirá a los usuarios tener datos históricos de asistencias a tutorías por parte de los estudiantes.



Figura 27. Sprint 5

4.4.1 Sección de reportes

En este módulo se desarrollará el todo el contenido de reportes y resultados que muestra resultados del uso de la aplicación.

4.4.1.1 Historia de usuario - HU-TUT011

Tabla 16

Historia de usuario - HU-TUT011

HISTORIA DE USUARIO	
ID: HU-TUT011	Usuario: Estudiante
Nombre Historia: TUT001-SECCION DE REPORTES	
Prioridad: Alta	Obtenido por: Andrés Tapia
	Modulo Asignado: Admin
OBJETO: Módulos	
Dentro del perfil administrador contamos con la sección de reportes clasificados en 5 tipos, los mismo nos indican resultados y estadísticas que se dan con el uso de la aplicación a lo largo de un periodo académico.	



Figura 28. Tipos de reportes

4.5 Sprint 6.

En este sprint se realizarán las pruebas funcionales de cada una de las historias de usuario.

4.5.1 Pruebas

4.5.1.1 PA_APT01

Tabla 17.

Prueba de aceptación - PA_APT01

Nombre:	Prueba de Aceptación Carga faculta, carreras, materias, módulos, periodo	
ID Caso de Prueba:	PA_APT01	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT016	
Nombre Requerimiento:	CARGA-FACULTAD-CARRERA-MATERIAS-PERIODO-MODULOS	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP01 Carga facultad-carreras-módulos-periodo" para el proceso de pruebas el cual debe cargar dicha información.		
PREREQUISITO		
HU-017 Perfil Admin		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "Error"		
Tipo de Error: "Error conexión Servidor"		
Tipo de Error : "Error con la base de datos"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Ingreso Facultad	Ingreso exitoso
2	Ingreso Carrera	Ingreso exitoso
3	Ingreso Materias	Ingreso exitoso
4	Ingreso Módulos	Ingreso exitoso
5	Ingreso Periodo	Ingreso exitoso
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Ingreso exitoso	

4.5.1.2 PA_APT02

Tabla 18.

Prueba de aceptación - PA_APT02

Nombre:	Prueba de Aceptación Carga automática Excel Docentes	
ID Caso de Prueba:	PA_APT02	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT012	
Nombre Requerimiento:	CARGA AUTOMATICA EXCEL DOCENTES	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP02 Carga automática Excel docentes" para el proceso de pruebas el cual debe cargar dicha información.		
PREREQUISITO		
HU-017 Perfil Admin		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "Campos Inválidos"		
Tipo de Error : "Error de archivo"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Seleccionar archivo	
2	Cargar lista	
3	Validar Campos	
4	Ingreso Docentes	Ingreso exitoso
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Ingreso exitoso	

4.5.1.3 PA_APT03

Tabla 19.

Prueba de aceptación - PA_APT03

Nombre:	Prueba de Aceptación Carga automática Excel Estudiantes	
ID Caso de Prueba:	PA_APT03	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT013	
Nombre Requerimiento:	CARGA AUTOMATICA EXCEL ESTUDIANTES	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP03 Carga automática Excel Estudiantes" para el proceso de pruebas el cual debe cargar dicha información.		
PREREQUISITO		
HU-012 Perfil Admin		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "Campos Inválidos"		
Tipo de Error : "Error de archivo"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Seleccionar archivo	
2	Cargar lista	
3	Validar Campos	
4	Ingreso Estudiantes	Ingreso exitoso
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Ingreso exitoso	

4.5.1.4 PA_APT04

Tabla 20.

Prueba de aceptación - PA_APT04

Nombre:	Prueba de Aceptación Carga horaria Docentes	
ID Caso de Prueba:	PA_APT04	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT002	
Nombre Requerimiento:	SECCIÓN DE CARGA HORARIA DOCENTE	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP04 Carga horaria Docentes" para el proceso de pruebas el cual debe asignar un horario al docente ingresando los campos requerid.		
PREREQUISITO		
HU-002 Carga horaria docentes		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "Docentes no ingresado"		
Tipo de Error : "Error modulo invalido"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Seleccionar Facultad	
2	Seleccionar Carrera	
3	Seleccionar Materia	
4	Seleccionar Docente	
5	Seleccionar Modulo	
6	Ingresar Aula Paralelo	
7	Ingresar Horario	Ingreso exitoso
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Ingreso exitoso	

4.5.1.5 PA_APT05

Tabla 21.

Prueba de aceptación - PA_APT05

Nombre:	Prueba de Aceptación Carga horaria Estudiante	
ID Caso de Prueba:	PA_APT05	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT003	
Nombre Requerimiento:	SECCIÓN DE CARGA HORARIA ESTUDIANTE	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP05 Carga horaria Estudiante" para el proceso de pruebas el cual debe asignar un horario al docente ingresando los campos requerid.		
PREREQUISITO		
HU-002 Carga horaria docentes		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "Docente no ingresado"		
Tipo de Error: "Error modulo invalido"		
Tipo de Error: "Modulo duplicado"		
Tipo de Error : "Dato no existe"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Seleccionar Facultad	
2	Seleccionar Carrera	
3	Seleccionar Estudiante	
4	Seleccionar Materia	
5	Seleccionar Paralelo	
6	Seleccionar Docente	
7	Ingresar número de matricula	
8	Ingresar Horario	Ingreso exitoso
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Ingreso exitoso	

4.5.1.6 PA_APT06

Tabla 22.

Prueba de aceptación - PA_APT06

Nombre:	Prueba de Aceptación Carga automática Excel horario Docentes	
ID Caso de Prueba:	PA_APT06	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT014	
Nombre Requerimiento:	CARGA AUTOMATICA HORARIO DOCENTE	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP06 Carga automática Excel horario Docentes" para el proceso de pruebas el cual debe cargar dicha información.		
PREREQUISITO		
HU-012 Perfil Súper Admin		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "Campos Inválidos Tipo de Error : "Error de archivo"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Seleccionar archivo	
2	Cargar lista	
3	Validar Campos	
4	Ingreso horario Docentes	Ingreso exitoso
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Ingreso exitoso	

4.5.1.7 PA_APT07

Tabla 23.

Prueba de aceptación - PA_APT07

Nombre:	Prueba de Aceptación Carga automática Excel horario Estudiantes	
ID Caso de Prueba:	PA_APT07	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT015	
Nombre Requerimiento:	CARGA AUTOMATICA HORARIO ESTUDIANTE	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP07 Carga automática Excel horario Estudiantes" para el proceso de pruebas el cual debe cargar dicha información.		
PREREQUISITO		
HU-012 Perfil Súper Admin		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "Campos Inválidos Tipo de Error : "Error de archivo"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Seleccionar archivo	
2	Cargar lista	
3	Validar Campos	
4	Ingreso horario Docentes	Ingreso exitoso
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Ingreso exitoso	

4.5.1.8 PA_APT08

Tabla 24.

Prueba de aceptación - PA_APT08

Nombre:	Prueba de Aceptación Sección de Login para identificación del docente	
ID Caso de Prueba:	PA_APT08	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT015	
Nombre Requerimiento:	SECCION DE LOGUEO PARA IDENTIFICACION DOCENTE	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP08 Login Perfil Docente" para el proceso de pruebas el cual debe ingresar con el api de office 365 el usuario registrado en la base de datos y acceder a su rol-perfil.		
PREREQUISITO		
HU-012 Carga automática Docentes		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "usuario no registrado"		
Tipo de Erro: "Recarga de página de login"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Ingresar con Microsoft	
2	Ingreso correo office 365	
3	Ingreso Contraseña	
4	Acceso al perfil y rol del usuario	Ingreso exitoso
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Ingreso exitoso	

4.5.1.9 PA_APT09

Tabla 25.

Prueba de aceptación - PA_APT09

Nombre:	Prueba de Aceptación Perfil Docente	
ID Caso de Prueba:	PA_APT09	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT010	
Nombre Requerimiento:	PERFIL DOCENTE	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP09 Perfil Docente" para el proceso de pruebas el usuario debe navegar por los diferentes módulos que dispone el perfil.		
PREREQUISITO		
HU-012 Carga automática Docentes		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "usuario no registrado"		
Tipo de Error: "error servidor"		
Tipo de Error: "error rol usuario"		
Tipo de Error: "error api office 365"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Ingresar con Microsoft	
2	Ingreso correo office 365	
3	Navegación datos personales	
4	Navegación mis estudiantes	
5	Navegación Notificaciones	
6	Navegación exitosa	Ingreso exitoso
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Ingreso exitoso	

4.5.1.10 PA_APT10

Tabla 26.

Prueba de aceptación - PA_APT10

Nombre:	Prueba de Aceptación Sección de notificación Docentes	
ID Caso de Prueba:	PA_APT10	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT007	
Nombre Requerimiento:	SECCION DE NOTIFICACIONES DOCENTE	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP10 Perfil Docente" para el proceso de pruebas el cual debe dentro del perfil del docente validar la función de notificaciones-mensajes.		
PREREQUISITO		
HU-010 Perfil Docente		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "error cargar modulo mensajes"		
Tipo de Error : "error cargar mensajes"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Ingresar con Microsoft	
2	Ingreso correo office 365	
3	Navegación mensajes-notificaciones	
4	Confirmación de mensajes exitosa	Ingreso exitoso
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Ingreso exitoso	

4.5.1.11 PA_APT11

Tabla 27.

Prueba de aceptación - PA_APT11

Nombre:	Prueba de Aceptación Sección de Login para identificación del estudiante	
ID Caso de Prueba:	PA_APT08	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT006	
Nombre Requerimiento:	SECCION DE LOGUEO PARA IDENTIFICACION ESTUDIANTE	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP11 Login Perfil Estudiante" para el proceso de pruebas el cual debe ingresar con el api de office 365 el usuario registrado en la base de datos y acceder a su rol-perfil.		
PREREQUISITO		
HU-013 Carga automática Estudiantes		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "usuario no registrado"		
Tipo de Erro: "Recarga de página de login"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Ingresar con Microsoft	
2	Ingreso correo office 365	
3	Ingreso Contraseña	
4	Acceso al perfil y rol del usuario	Ingreso exitoso
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Ingreso exitoso	

4.5.1.12 PA_APT12

Tabla 28.

Prueba de aceptación - PA_APT12

Nombre:	Prueba de Aceptación Perfil Estudiante	
ID Caso de Prueba:	PA_APT09	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT017	
Nombre Requerimiento:	PERFIL DOCENTE	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP12 Perfil Docente" para el proceso de pruebas el usuario debe navegar por los diferentes módulos que cuenta el perfil		
PREREQUISITO		
HU-013 Carga automática Estudiantes		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "usuario no registrado" Tipo de Error: "error servidor" Tipo de Error: "error rol usuario" Tipo de Error: "error api office 365"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Ingresar con Microsoft	
2	Ingreso correo office 365	
3	Navegación datos personales	
4	Navegación Perfil	
5	Navegación Horario	
6	Navegación tutorías	
7	Navegación exitosa	Navegación exitosa
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Navegación exitosa	

4.5.1.13 PA_APT13

Tabla 29.

Prueba de aceptación - PA_APT13

Nombre:	Prueba de Aceptación Notificación estudiantes	
ID Caso de Prueba:	PA_APT13	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT008	
Nombre Requerimiento:	SECCION DE NOTIFICACIONES ESTUDIANTE	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP13 Perfil Docente" para el proceso de pruebas el usuario debe tener el correcto funcionamiento de la notificación de la tutoría.		
PREREQUISITO		
HU-013 Carga automática Estudiantes		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "no registro de tutoría por parte del docente"		
Tipo de Error: "error al cargar el modulo"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Ingresar con Microsoft	
2	Navegación módulo de tutoría	
3	Confirmación recepción de tutoría	Confirmación exitosa
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Confirmación exitosa	

4.5.1.14 PA_APT14

Tabla 30.

Prueba de aceptación - PA_APT14

Nombre:	Prueba de Aceptación Búsqueda y petición de tutorías.	
ID Caso de Prueba:	PA_APT14	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT008	
Nombre Requerimiento:	SECCION DE BUSQUEDA Y PETICIÓN DE TUTORIA	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP14 Perfil Docente" para el proceso de pruebas el usuario debe realizar la búsqueda y solicitud de tutoría.		
PREREQUISITO		
HU-012 Carga automática Docentes HU-013 Carga automática Estudiantes		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "módulos duplicados" Tipo de Error: "error usuario docentes" Tipo de Error: "error seleccionar materia" Tipo de Error: "error ingresar cuadro de texto" Tipo de Error: "error petición al docente de tutoría"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Ingresar con Microsoft	
2	Navegación módulo de tutoría	
3	Navegación al form Solicitud de tutoría	
4	Selección de materia	
5	Selección de docente	
6	Selección de modulo	
7	Selección fecha de tutoría	
8	Escribir mensaje de texto	
9	Enviar petición	Envió Exitoso
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
29/12/2018	Envió exitosa	

4.5.1.15 PA_APT15

Tabla 31

Prueba de aceptación - PA_APT15

Nombre:	Prueba de Aceptación Reportes.	
ID Caso de Prueba:	PA_APT15	
Responsable Caso de Prueba:	Andrés Tapia	
Fecha Def. Caso de Prueba:	27/12/2018	
ID Requerimiento:	HU-TUT008	
Nombre Requerimiento:	SECCION DE BUSQUEDA Y PETICIÓN DE TUTORIA	
Responsable Requerimiento:	Andrés Tapia	
ALCANCE		
Establecer el caso de prueba de aceptación "PA_ATP12 Modulo Reportes" Se plantea el desarrollo de 5 reportes con diferentes datos, los mismos que deben entregar históricos y estadísticas del uso de la aplicación.		
PREREQUISITO		
HU-012 Carga automática Docentes HU-013 Carga automática Estudiantes		
DATOS GENERALES DE LA PRUEBA		
Tipo de Error: "reporte no generado" Tipo de Error: "grafico inconsistente" Tipo de Error: "estadísticas y efectividad incorrecta"		
CASO DE PRUEBA		
Orden	Descripción	Resultado Esperado
1	Seleccionar Reporte	
2	Generar Reporte	Generación Exitosa
RESULTADOS OBTENIDOS		
Fecha	Resultado Obtenido	
15/01/2018	Reporte Generado	

5. CONCLUSIONES Y RECOMENDACIONES

En mi proyecto de tesis “SISTEMA DE GESTIÓN DE TUTORÍAS ACADÉMICAS”, elaborado con el fin de integrar un sistema de software que busca una nueva solución a la problemática de gestión de tutorías de la UDLA, se ha llegado a las siguientes conclusiones y recomendaciones:

5.1 Conclusiones

Se llegó a implementar el sistema de gestión de tutorías conociendo los aspectos administrativos, así como las necesidades tanto del docente como del estudiante.

El sistema permite mejorar el tiempo de actualización de la información gracias a la carga de información del sistema administrativo de la facultad, enfocando los procedimientos para procesos de cargas masiva de horarias, docentes, estudiantes, materias.

Se desarrolló un sistema personalizado para que el estudiante ahora cuente con la facilidad de solicitar tutorías al docente que más confianza tenga pero que imparta la materia de su horario.

Se desarrolló un sistema personalizado para el docente, este pudo confirmar las solicitudes de diferentes estudiantes que necesiten realizar una tutoría. Caso contrario que el docente no pueda la fecha solicitada por el estudiante, el sistema permitió cambiar la fecha y hora de la tutoría enviando una notificación automática al estudiante y cargándolo en su horario de tutorías.

Se desarrolló una sección del docente, en la cual este pueda revisar el estado actual de matrícula en cada una de sus materias, brindándole la capacidad de poder asignar tutorías a estudiantes que estén cruzando segunda y tercera

matricula de manera obligatoria, dándole una facilidad de obtener información, controlar y preocuparse más por sus estudiantes.

El docente ahora puede cargar sus horas libres u obligatorias de tutorías, con la facilidad de llevar mejor su cronograma de actividades, así como tener pendiente sus compromisos estudiantiles. Esto facilitó al estudiante conocer las horas libres u obligatorias que el docente tiene asignado en su horario, para realizar una solicitud de tutoría directamente con el docente.

El sistema mejoró la coordinación administrativa, así como la comunicación entre docente y estudiante.

Se concluyó que la plataforma si es necesaria para la universidad, debido a los beneficios que anteriormente se describen, siempre y cuando se realice una etapa de pruebas para corregir los errores a grane escala.

El sistema nos ayudado a entender la problemática de los estudiantes como de los docentes en el momento de interactuar entre sí para llegar a establecer horas de tutorías, así como entender el sistema de carga horaria universitario. Esto nos permitió realizar un sistema que une más el lazo de amistad entre docente y estudiante, así como dar seguimiento y entender la problemática de cada uno de los estudiantes, en su periodo académico actual.

El sistema cuenta con la tecnología de Microsoft ASP.NET, la cual es catalogada como una de herramientas de software más utilizadas y con gran desempeño en el mercado.

A pesar del déficit de enseñanza de diseño en la carrera de Sistemas, la aplicación está pensada para que haiga la mejor experiencia de UX en los usuarios, dándole la facilidad de reconocimiento de las acciones, así como una experiencia amigable y única.

5.2 Recomendaciones

El sistema está en fase de desarrollo, por lo tanto, es recomendable usarlo a modo de pruebas en los servidores universitarios hasta que el sistema permanezca estable y pueda tener los requerimientos necesarios para ponerlo en modo de producción.

El sistema tiene un grado de complejidad medio, por lo tanto, se recomienda realizar una capacitación al administrador de la aplicación como a los usuarios (docente y estudiante), con la finalidad que tenga una correcta manipulación, y el desempeño sea óptimo.

Se recomienda utilizar ASP.NET MV5, así como utilizar la documentación oficial de Microsoft, para la manipulación o modificación de la aplicación.

Recomendaría en futuras generaciones integrar un poco de diseño en la malla de la carrera, con la finalidad de que el desarrollo de software de la universidad no se base solo en la funcionalidad, sino también en la experiencia visual del usuario.

El sistema cuenta con una base de datos de carácter avanzado en el almacenamiento de los datos, por lo cual se recomienda realizar backups periódicamente para no perder información vital de la universidad, en el caso que se llegue a implementar el sistema en la universidad.

El sistema con tecnología está pensado en servidores con tecnología Windows, por lo tanto, si se desea compatibilidad con un sistema Linux o Mac se recomienda actualizar a la nueva versión de ASP que es ASP.NET CORE para herramientas multiplataforma.

REFERENCIAS

- Alaimo, D. (2013). *Proyectos Ágiles con Scrum*. Recuperado el 20 de enero de 2019, de <http://www.elkinforero.com/joomdocs/kleer-proyecto-agiles-con-scrum.pdf>
- Badrul, E., & Mohamed, K. (2015). *International Handbook of E-Learning Volume 1: Theoretical Perspectives and Research*. Recuperado el 15 de enero de 2019, de <https://books.google.com.ec/books?id=HHzABgAAQBAJ&printsec=frontcover&hl=es#v=onepage&q&f=false>
- Botella, A. (2018). ASP.NET MVC 4. Recuperado el 15 de enero de 2019, de <https://si.ua.es/es/documentacion/asp-net-mvc-4/documentos/1-dia/contacto-con-mvc.pdf>
- Freeman, A. (2013). Pro Asp.net MVC 5. Recuperado el 14 de enero de 2019, de <https://www.apress.com/us/book/9781430265290>
- Gallegos, M. (2018). Metodología Scrum. Recuperado el 18 de enero de 2019, de <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/17885/1/mtrigastFC0612memoria.pdf>
- Menzinsky, A., & Gertrudis, j. (2016). Scrum (Guía SBOK™). Recuperado el 22 de enero de 2019, de <https://www.tenstep.ec/portal/servicios-de-formacion/por-que-scrum/sbok-guide>
- Microsoft. (2015). Interacción a las consultas LINQ. Recuperado el 19 de enero de 2019, de <https://docs.microsoft.com/es-es/dotnet/csharp/programming-guide/concepts/linq/introduction-to-linq-queries>
- Oppel, A., & Sheldon, R. (2010). Fundamentos de SQL. *Recuperado* el 20 de enero de 2019, de <https://codigosdeprogramacion.com/2017/05/29/libro-fundamentos-de-sql/>

Universidad Internacional de Valencia. (2018). Seguridad Informática.

Recuperado el 12 de enero de 2019, de

<https://www.universidadviu.es/la-seguridad-informatica-puede-ayudarme/>

ANEXOS

Anexos 1. Código de la arquitectura Modelo Vista Controlador (MVC): Modelo

```
metodoLoginUsuario.cs  estudiante.cs  admin.cs  metodoEstudiante.cs  Index.cshtml  moduloHorarioEstudiante.cshtml
AppTutorias
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5
6  namespace AppTutorias.Models.Admin
7  {
8      public class admin
9      {
10         public int Id_Docente { get; set; }
11         public string Nombre { get; set; }
12         public string NombreDos { get; set; }
13         public string Apellido { get; set; }
14         public string ApellidoDos { get; set; }
15         public string Email { get; set; }
16     }
17 }
18
```

Modelo usuario Admin.

```
admin.cs  estudiantesDocente.cs  horarioClaseDocente.cs  superAdmin.cs  carrera.cs  docente.cs
AppTutorias
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5
6  namespace AppTutorias.Models.SuperAdmin.Docente
7  {
8      public class docente
9      {
10         public int Id_Docente { get; set; }
11         public string Nombre { get;set; }
12         public string NombreDos { get; set; }
13         public string Apellido { get; set; }
14         public string ApellidoDos { get; set; }
15         public string Email { get; set; }
16         public string Celular { get; set; }
17         public string Cedula { get; set; }
18         public HashSet<string> ConexionId { get; set; }
19     }
20 }
21
```

Modelo usuario Docente.

```
docente.cshtml  Estudiante.cshtml
horarioClaseDocente.cs  superAdmin.cs  carrera.cs  docente.cs  estudiante.cs  horarioClaseEstudiante.cs
AppTutorias
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5
6  namespace AppTutorias.Models.SuperAdmin.Estudiante
7  {
8      public class estudiante
9      {
10         public int Id_Estudiante{ get; set; }
11         public string Nombre { get; set; }
12         public string NombreDos { get; set; }
13         public string Apellido { get; set; }
14         public string ApellidoDos { get; set; }
15         public string Email { get; set; }
16         public string Celular { get; set; }
17         public string Matricula { get; set; }
18         public string CodCarrera { get; set; }
19         public string CodFacultad { get; set; }
20     }
21 }
22
23
```

Modelo usuario Estudiante.

```
horarioClaseEstudiante.cs  horarioTutorias.cs  repitencia.cs  facultad.cs  horarioClases.cs  modulo.cs  loginUsuario.cs
AppTutorias
AppTutorias.Models.LoginUsuario
Id_LoginUsuario
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5
6  namespace AppTutorias.Models
7  {
8  public class loginUsuario
9  {
10     public int Id_LoginUsuario { get; set; }
11     public int Id_Rol { get; set; }
12     public string EmailUsuario { get; set; }
13     public string Pass { get; set; }
14 }
15 }
16
```

Modelo Login

```
ioClaseEstudiante.cs  horarioTutorias.cs  repitencia.cs  facultad.cs  horarioClases.cs  AdalTokenCache.cs
AppTutorias
AppTutorias.Models.ADALTokenCache
db
6  using System.Web;
7  using System.Web.Security;
8  using Microsoft.IdentityModel.Clients.ActiveDirectory;
9
10 namespace AppTutorias.Models
11 {
12     public class ADALTokenCache : TokenCache
13     {
14         private ApplicationDbContext db = new ApplicationDbContext();
15         private string userId;
16         private UserTokenCache Cache;
17
18         public ADALTokenCache(string signedInUserId)
19         {
20             // asociar la caché al usuario actual de la aplicación web
21             userId = signedInUserId;
22             this.AfterAccess = AfterAccessNotification;
23             this.BeforeAccess = BeforeAccessNotification;
24             this.BeforeWrite = BeforeWriteNotification;
25             // buscar la entrada en la base de datos
26             Cache = db.UserTokenCacheList.FirstOrDefault(c => c.webUserUniqueId == userId);
27             // colocar la entrada en la memoria
28             this.Deserialize((Cache == null) ? null : MachineKey.Unprotect(Cache.cacheBits, "ADALCache"));
29         }
30
31         // limpiar la base de datos
32         public override void Clear()
```

Modelo token cuenta office 365

```
ClaseEstudiante.cs  horarioTutorias.cs  repitencia.cs  facultad.cs  horarioClases.cs  ApplicationDbContext.cs
AppTutorias
AppTutorias.Models.ApplicationDbContext
ApplicationDbContext()
3  using System.ComponentModel.DataAnnotations;
4  using System.Data.Entity;
5  using System.Linq;
6  using System.Web;
7
8  namespace AppTutorias.Models
9  {
10     public class ApplicationDbContext : DbContext
11     {
12         public ApplicationDbContext()
13             : base("DefaultConnection")
14         {
15         }
16
17         public DbSet<UserTokenCache> UserTokenCacheList { get; set; }
18     }
19
20     public class UserTokenCache
21     {
22         [Key]
23         public int UserTokenCacheId { get; set; }
24         public string webUserUniqueId { get; set; }
25         public byte[] cacheBits { get; set; }
26         public DateTime LastWrite { get; set; }
27     }
28 }
```

Modelo conexión a la base office365 token

```
docente.cshtml  Estudiante.cshtml
rrera.cs  docente.cs  estudiante.cs  horarioClaseEstudiante.cs  horarioTutorias.cs  repitencia.cs  facultad.cs
AppTutorias
AppTutorias.Models.SuperAdmin.facultad
Id_Facultad
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5
6  namespace AppTutorias.Models.SuperAdmin
7  {
8     public class facultad
9     {
10
11         public int Id_Facultad { get; set; }
12         public stringCodigoFacultad { get; set; }
13         public stringNombreFacultad { get; set; }
14     }
15 }
16 }
```

Modelo Facultad

```
AppTutorias
  AppTutorias.Models.SuperAdmin.Carrera.carrera
  Id_Carrera
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5
6  namespace AppTutorias.Models.SuperAdmin.Carrera
7  {
8      public class carrera
9      {
10         public int Id_Carrera { get; set; }
11         public stringCodigoFacultad { get; set; }
12         public string NombreCarrera { get; set; }
13         public stringCodigoCarrera { get; set; }
14     }
15 }
16
17
```

Modelo Carrera

```
Docente.cshtml # Estudiante.cshtml #
estudiante.cs horarioClaseEstudiante.cs horarioTutorias.cs repitencia.cs facultad.cs horarioClases.cs materia.cs
AppTutorias
  AppTutorias.Models.SuperAdmin.Materia.materia
  Id_Carrera
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5
6  namespace AppTutorias.Models.SuperAdmin.Materia
7  {
8      public class materia
9      {
10         public int Id_Carrera { get; set; }
11         public stringCodigoFacultad { get; set; }
12         public stringCodigoCarrera { get; set; }
13         public string NombreMateria { get; set; }
14         public stringCodigoMateria { get; set; }
15         public char Repitencia { get; set; }
16     }
17 }
18
```

Modelo Materia

```
rioClaseEstudiante.cs  horarioTutorias.cs  repitencia.cs  facultad.cs  horarioClases.cs  modulo.cs
AppTutorias
AppTutorias.Models.SuperAdmin.Modulo.modulo  Id_Modulo
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5
6  namespace AppTutorias.Models.SuperAdmin.Modulo
7  {
8      public class modulo
9      {
10         public int Id_Modulo { get; set; }
11         public stringCodigoModulo { get; set; }
12         public string HoraInicio { get; set; }
13         public string HoraFin { get; set; }
14         public string Dia { get; set; }
15     }
16 }
17
18
```

Modelo Modulo

```
rioClaseEstudiante.cs  horarioTutorias.cs  repitencia.cs  facultad.cs  horarioClases.cs  modulo.cs  periodo.cs
AppTutorias
AppTutorias.Models.SuperAdmin.Periodo.periodo  Id_Periodo
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5
6  namespace AppTutorias.Models.SuperAdmin.Periodo
7  {
8      public class periodo
9      {
10         public int Id_Periodo { get; set; }
11         public stringCodigoPeriodo { get; set; }
12         public string NombrePeriodo { get; set; }
13     }
14 }
15
```

Modelo Periodo

```
metodoDocente.cs  DocenteController.cs  Solicitud.cs  Notificaciones.cshtml  MasterDocente.cshtml  SuperAdminjs
Docente.cshtml  Estudiante.cshtml
superAdmin.cs  carrera.cs  docente.cs  estudiante.cs  horarioClaseEstudiante.cs  horarioTutorias.cs  repitencia.cs
AppTutorias
AppTutorias.Models.SuperAdmin.Estudiante.repitenc  Matricula
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5
6  namespace AppTutorias.Models.SuperAdmin.Estudiante
7  {
8      public class repitencia
9      {
10         public int Matricula { get; set; }
11         public string NombreEstudiante { get; set; }
12         public string Materia { get; set; }
13         public stringCodigoMateria { get; set; }
14         public char Repitencia { get; set; }
15     }
16 }
17
```

Modelo Repitencia

```
Docente.cshtml # Estudiante.cshtml #
admin.cs Index.cshtml moduloHorarioEstudiante.cshtml estudiantesDocente.cs horarioClaseDocente.cs x
AppTutorias
AppTutorias.Models.Docente.horarioClaseDocente CodigoFacultad
3 using System.Linq;
4 using System.Web;
5
6 namespace AppTutorias.Models.Docente
7 {
8     public class horarioClaseDocente
9     {
10
11
12         public string CodigoFacultad { get; set; }
13         public string NombreFacultad { get; set; }
14         public string CodigoCarrera { get; set; }
15         public string NombreCarrera { get; set; }
16         public string CodigoMateria { get; set; }
17         public string NombreMateria { get; set; }
18         public string Seccion { get; set; }
19         public string Paralelo { get; set; }
20         public string Docente { get; set; }
21         public string CodigoModulo { get; set; }
22         public string Dia { get; set; }
23         public string HoraInicio { get; set; }
24         public string HoraFin { get; set; }
25         public string Periodo { get; set; }
26         public string CodigoPeriodo { get; set; }
27         public string Fecha { get; set; }
28
29     }
30 }
```

Modelo Horario clases Docente

```
Docente.cshtml # Estudiante.cshtml #
estudiante.cs admin.cs metodoEstudiante.cs Index.cshtml moduloHorarioEstudiante.cshtml estudiantesDocente.cs x
AppTutorias
AppTutorias.Models.Docente.estudiantesDocente CodigoMateria
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5
6 namespace AppTutorias.Models.Docente
7 {
8     public class estudiantesDocente
9     {
10
11
12         public string CodigoMateria { get; set; }
13         public string NombreMateria { get; set; }
14         public string Matricula { get; set; }
15         public string Estudiante { get; set; }
16         public string EstudianteDatos { get; set; }
17         public string Repitencia { get; set; }
18         public string Horario { get; set; }
19         public string Periodo { get; set; }
20
21     }
22 }
```

Modelo Horario clases Estudiante

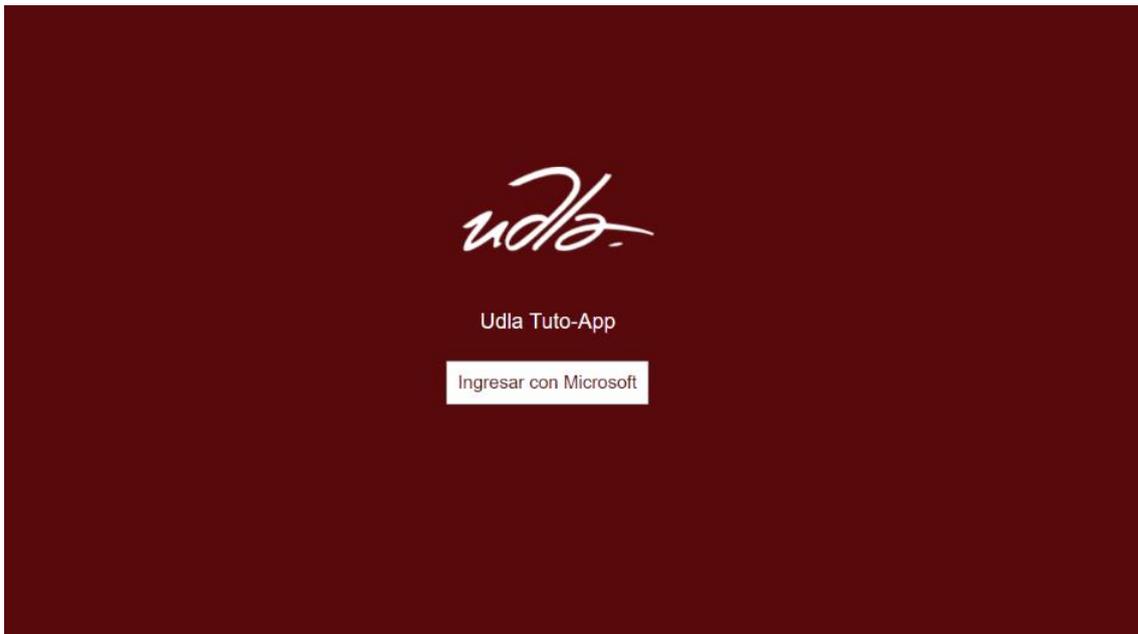
```
metodoDocente.cs # DocenteController.cs # Solicitud.cs # Notificaciones.cshtml # MasterDocente.cshtml # SuperAdmin.js #
Docente.cshtml # Estudiante.cshtml #
superAdmin.cs # carrera.cs # docente.cs # estudiante.cs # horarioClaseEstudiante.cs # horarioTutorias.cs # X
AppTutorias
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5
6 namespace AppTutorias.Models.SuperAdmin.Estudiante
7 {
8     public class horarioTutorias
9     {
10         public int Id_HorarioTutorias { get; set; }
11         public string Matricula { get; set; }
12         public string NombreDocente { get; set; }
13         public string NombreEstudiante { get; set; }
14         public string CedulaDocente { get; set; }
15         public stringCodigoModulo { get; set; }
16         public string NombreMateria { get; set; }
17         public stringCodigoMateria { get; set; }
18         public stringCodigoPeriodo { get; set; }
19         public string Modulo { get; set; }
20         public string EstadoObligatorio { get; set; }
21         public string Fecha { get; set; }
22     }
23 }
24
25 }
26
```

Modelo Horario tutorías

```
Reportes.cshtml # Reportes.cshtml # Comandos.cshtml # EstudianteController.cs # MasterSuperAdmin.cshtml # estadisticaTutorias.cs # PerfilDocente.cshtml #
Modelos # Estadistica.cshtml # TutoriasEstudiante.cshtml # Notificaciones.cshtml # Docente.cshtml #
.cshtml # moduloHorarioTutoriasEstudiante.cshtml # moduloHorarioEstudiante.cshtml # perfilEstudiante.cshtml # Inicio.cshtml # Estudiante.cshtml # Facultad.cshtml
Modelos # AppTutorias.Models.estadisticaTutorias # rep.docente
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using AppTutorias.Models.SuperAdmin.Estudiante;
using AppTutorias.Models.SuperAdmin.Docente;
namespace AppTutorias.Models
{
    public class estadisticaTutorias
    {
        public docente Docente { get; set; }
        public estudiante Estudiante { get; set; }
        public int HorasPlanificadas { get; set; }
        public int HorasDadas { get; set; }
        public int HorasLibres { get; set; }
        public int HorasObligatorias { get; set; }
        public int numeroEstudiantes { get; set; }
        public int horasCarreras { get; set; }
        public int horasOtrasCarreras { get; set; }
        public string codigoMateria { get; set; }
        public string rep.docente { get; set; }
    }
}
```

Modelo Reportes

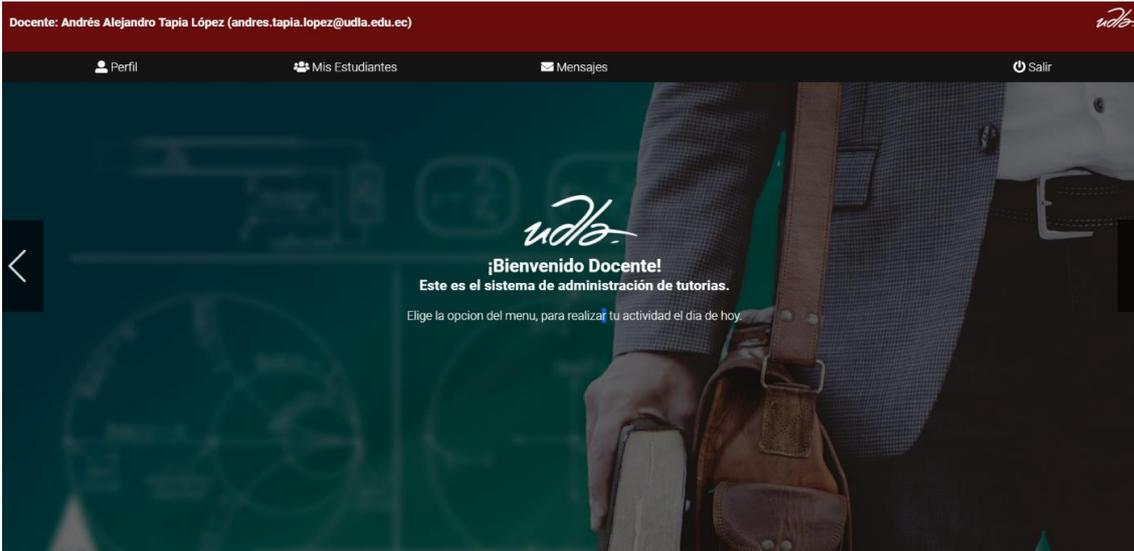
Vista



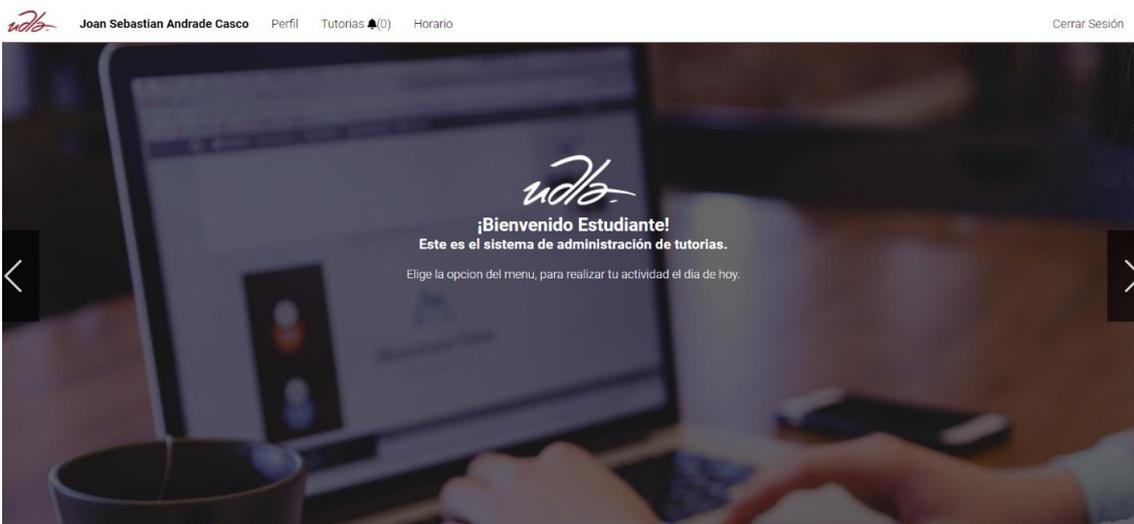
Vista Login Office 365



Vista Administración Central Tutorías



Vista Perfil Docente



Vista Perfil Estudiante

Reportes de Tutorías

Registro de reportes de tutorías de los estudiantes

Tutorías por docente Tutorías por carrera Tutorías por matrícula de estudiante Tutorías por número de matrícula Tutorías por materia

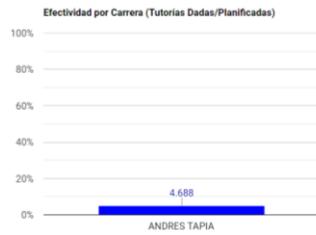
Analiza la efectividad de cada docente en su cronograma de tutorías

Vista reportes por categoría

Estadísticas por docente

Período: **SEPTIEMBRE 2018** Docente: **1720765533** [Generar Reporte](#)

Docente	Horas Planificadas	Horas Dadas	Efectividad(Total Tutorías/Asistidas)
ANDRES TAPIA	128 horas	6 horas	4.6875 %



Vista reporte 1

- Tutorías por docente
- Tutorías por carrera
- Tutorías por matrícula de estudiante
- Tutorías por número de matrícula
- Tutorías por materia

Analiza la efectividad de cada docente en su cronograma de tutorías

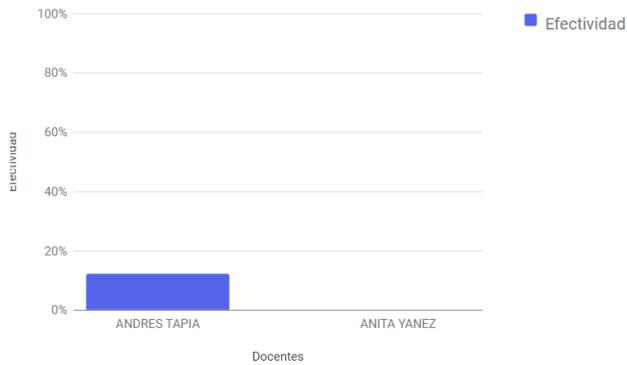
Estadísticas de docentes por Carrera

Período: **SEPTIEMBRE 2018** Facultad: **FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS** Carrera: **ING SISTEMAS** [Generar Reporte](#)

Carrera: ING SISTEMAS

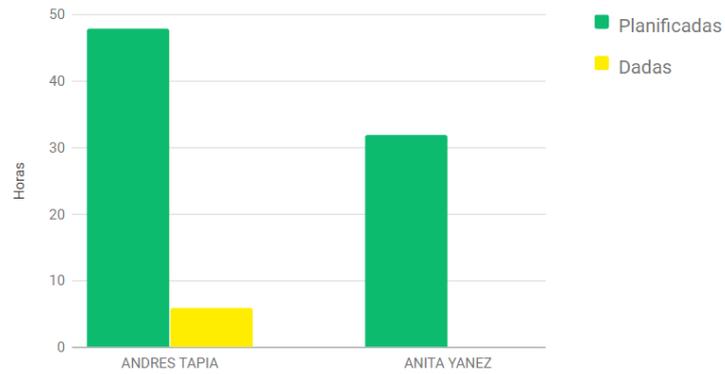
Docentes	Tutorías Planificadas	Tutorías Dadas	Tutorías Carrera	Tutorías Otras Carreras	Número Estudiantes	Efectividad
ANDRES TAPIA	48	6	5	1	3	12.50%
ANITA YANEZ	32	0	0	0	0	0.00%

Efectividad Tutorías Dadas/Planificadas



Horas Planificadas vs Horas Dadas

Horas Planificadas vs Horas Dadas



Vista reporte 2

[Tutorías por docente](#)
[Tutorías por carrera](#)
[Tutorías por matrícula de estudiante](#)
[Tutorías por número de matrícula](#)
[Tutorías por materia](#)

Analiza la efectividad de cada docente en su cronograma de tutorías

Estadísticas de tutorías por repetencia de estudiante

Período:
 Facultad:
 Carrera:
 Número de Matrícula:

Carrera: ING SISTEMAS

Estudiante	Materia	Número tutorías Planificadas	Número de tutorías Asistidas
JOAN ANDRADE	FIS100	2	2
JOAN ANDRADE	BDD	3	0
PEDRO NAVAJA	SIS1	1	1

Vista reporte 3

Estadística de repitencia por carrera

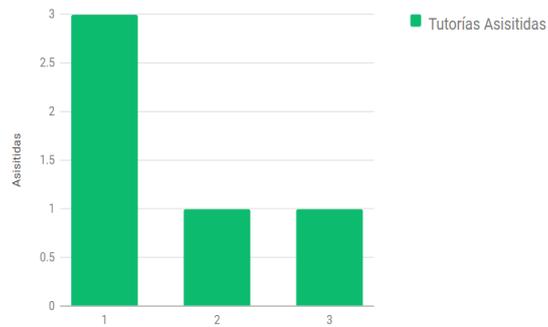
Período:
 Facultad:
 Carrera:

Carrera: ING SISTEMAS

Carrera: ING SISTEMAS

Número Matrícula(Repitencia)	Tutorías Asistidas(SI)
1	3
2	1
3	1

Tutorías por tipo de Matrículas



Vista reporte 4

Estadística de repitencia por Materia

Período: SEPTIEMBRE 2018 | Facultad: FACULTAD DE INGENIERIA Y CIENCIAS APLICADAS | Carrera: ING SISTEMAS | Materia: FISICA I | [Generar Reporte](#)

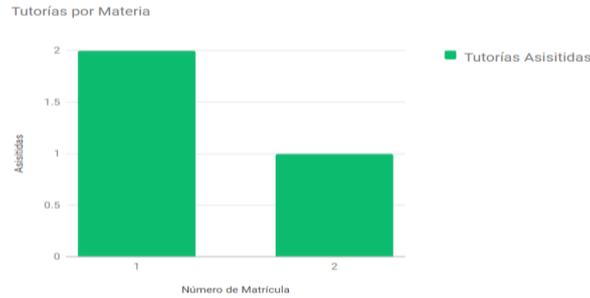
Carrera: ING SISTEMAS

Materia: FISICA I

arrera: ING SISTEMAS

lateria: FISICA I

Número Matrícula(Repitencia)	Tutorías Asistidas(SI)
1	2
2	1



Vista reporte 5

Controlador

```

11 using Microsoft.Azure.ActiveDirectory.GraphClient;
12
13 namespace AppTutorias.Controllers
14 {
15     public class AccountController : Controller
16     {
17         public void SignIn()
18         {
19             // Enviar una solicitud de inicio de sesión a OpenID Connect.
20             if (!Request.IsAuthenticated)
21             {
22                 HttpContext.GetOwinContext().Authentication.Challenge(new AuthenticationProperties { RedirectUri = "/Account/Usuar
23                     OpenIdConnectAuthenticationDefaults.AuthenticationType);
24             }
25         }
26
27         public void SignOut()
28         {
29             string callbackUrl = Url.Action("SignOutCallback", "Account", routeValues: null, protocol: Request.Url.Scheme);
30
31             HttpContext.GetOwinContext().Authentication.SignOut(
32                 new AuthenticationProperties { RedirectUri = callbackUrl },
33                 OpenIdConnectAuthenticationDefaults.AuthenticationType, CookieAuthenticationDefaults.AuthenticationType);
34         }
35
36         public ActionResult SignOutCallback()
37         {
38             if (Request.IsAuthenticated)
39             {
40                 // Redirigir a la página principal si el usuario está autenticado.
41                 return RedirectToAction("Usuario", "Account");
42             }
43         }
44
45         return View();

```

Controlador Account valida office 365

```

31 using System.IO;
32 using System.Runtime.InteropServices;
33
34 namespace TutoriasApp.Controllers
35 {
36
37     [Authorize]
38     public class SuperAdminController : Controller
39     {
40         //login
41
42         private ApplicationDbContext db = new ApplicationDbContext();
43         private string clientId = ConfigurationManager.AppSettings["ida:ClientId"];
44         private string appKey = ConfigurationManager.AppSettings["ida:ClientSecret"];
45         private string aadInstance = EnsureTrailingSlash(ConfigurationManager.AppSettings["ida:AADInstance"]);
46         private string graphResourceID = "https://graph.windows.net";
47
48
49
50         public async Task<ActionResult> Index()
51         {
52
53             string tenantID = ClaimsPrincipal.Current.FindFirst("http://schemas.microsoft.com/identity/claims/tenantid").Value;
54             string userObjectID = ClaimsPrincipal.Current.FindFirst("http://schemas.microsoft.com/identity/claims/objectidentifier").Value;
55             try
56             {
57                 Uri servicePointUri = new Uri(graphResourceID);
58                 Uri serviceRoot = new Uri(servicePointUri, tenantID);
59                 ActiveDirectoryClient activeDirectoryClient = new ActiveDirectoryClient(serviceRoot,
60                     async () => await GetTokenForApplication());
61
62                 // utiliza el token para consultar el gráfico y obtener detalles sobre el usuario
63
64                 var result = await activeDirectoryClient.Users
65                     .Where(u => u.ObjectId.Equals(userObjectID))

```

Controlador Súper Admin.

```

13 using AppTutorias.Metodos.SuperAdmin.Estudiante;
14
15 namespace AppTutorias.Controllers
16 {
17     // [Authorize]
18     public class AdminController : Controller
19     {
20         // GET: Admin
21         public ActionResult Index()
22         {
23             return View();
24         }
25
26
27         //-----TUTORIAS-----
28         //-----
29
30         public ActionResult Tutorias()
31         {
32
33             metodoHorarioTutorias metodoHorarioTutorias = new metodoHorarioTutorias();
34             List<horarioTutorias> listaHorariosTutorias = new List<horarioTutorias>();
35
36             listaHorariosTutorias = metodoHorarioTutorias.consultaHorarioTutorias();
37
38             return PartialView("Tutorias", listaHorariosTutorias);
39         }
40         //LISTA Docentes del Estudiante
41
42         public ActionResult listaDocentesEstudiante(string Matricula, string CodigoMateria)
43         {
44             metodoDocente metodoDocente = new metodoDocente();
45             List<docente> listaDocentes = new List<docente>();
46
47

```

Controlador Admin.

```
DocenteController.cs Solicitud.cs Notificaciones.cshtml MasterDocente.cshtml SuperAdminjs Docente.cshtml Estudiante.cshtml
periodo.cs metodoHorarioClases.cs HorarioClases.cshtml AccountController.cs AdminController.cs SuperAdminController.cs
AppTutorias
AppTutorias.Controllers.DocenteController RefreshSession()
25 using Microsoft.Owin.Security.OpenIdConnect;
26 using AppTutorias.Models.Estudiante;
27
28 namespace AppTutorias.Controllers
29 {
30
31
32
33 [Authorize]
34
35 public class DocenteController : Controller
36 {
37     //login
38
39     private ApplicationDbContext db = new ApplicationDbContext();
40     private string clientId = ConfigurationManager.AppSettings["ida:ClientId"];
41     private string appKey = ConfigurationManager.AppSettings["ida:ClientSecret"];
42     private string aadInstance = EnsureTrailingSlash(ConfigurationManager.AppSettings["ida:AADInstance"]);
43     private string graphResourceID = "https://graph.windows.net";
44
45     // GET: UserProfile
46     public async Task<ActionResult> Index()
47     {
48         if (Session["usuarioLogeado"] != null)
49         {
50             string tenantID = ClaimsPrincipal.Current.FindFirst("http://schemas.microsoft.com/identity/claims/tenantid").Value;
51             string userObjectID = ClaimsPrincipal.Current.FindFirst("http://schemas.microsoft.com/identity/claims/objectidentifier").Value;
52             try
53             {
54                 Uri servicePointUri = new Uri(graphResourceID);
55                 Uri serviceRoot = new Uri(servicePointUri, tenantID);
56                 ActiveDirectoryClient activeDirectoryClient = new ActiveDirectoryClient(serviceRoot,
57                     async () => await GetTokenForApplication());
58
59                 // utiliza el token para consultar el gráfico y obtener detalles sobre el usuario
60
61             }
62             catch { }
63         }
64     }
65 }
```

Controlador Admin.

```
UserProfileController.cs periodo.cs metodoHorarioClases.cs HorarioClases.cshtml AccountController.cs AdminController.cs
AppTutorias
AppTutorias.Controllers.UserProfileController db
9 using Microsoft.Azure.ActiveDirectory.GraphClient;
10 using Microsoft.IdentityModel.Clients.ActiveDirectory;
11 using Microsoft.Owin.Security;
12 using Microsoft.Owin.Security.Cookies;
13 using Microsoft.Owin.Security.OpenIdConnect;
14 using AppTutorias.Models;
15
16 namespace AppTutorias.Controllers
17 {
18     [Authorize]
19     public class UserProfileController : Controller
20     {
21         private ApplicationDbContext db = new ApplicationDbContext();
22         private string clientId = ConfigurationManager.AppSettings["ida:ClientId"];
23         private string appKey = ConfigurationManager.AppSettings["ida:ClientSecret"];
24         private string aadInstance = EnsureTrailingSlash(ConfigurationManager.AppSettings["ida:AADInstance"]);
25         private string graphResourceID = "https://graph.windows.net";
26
27         // GET: UserProfile
28         public async Task<ActionResult> Index()
29         {
30             string tenantID = ClaimsPrincipal.Current.FindFirst("http://schemas.microsoft.com/identity/claims/tenantid").Value;
31             string userObjectID = ClaimsPrincipal.Current.FindFirst("http://schemas.microsoft.com/identity/claims/objectidentifier").Value;
32             try
33             {
34                 Uri servicePointUri = new Uri(graphResourceID);
35                 Uri serviceRoot = new Uri(servicePointUri, tenantID);
36                 ActiveDirectoryClient activeDirectoryClient = new ActiveDirectoryClient(serviceRoot,
37                     async () => await GetTokenForApplication());
38
39                 // utiliza el token para consultar el gráfico y obtener detalles sobre el usuario
40
41                 var result = await activeDirectoryClient.Users
42                     .Where(u => u.ObjectId.Equals(userObjectID))
43                     .ExecuteAsync();
44                 //var user = result.FirstOrDefault().Client();
45             }
46             catch { }
47         }
48     }
49 }
```

Controlador Estudiante.

```
Controladora * * * Reportes.cshtml * * * controladora * * * EncabezadoControladora * * * MasterSuperAdmin.cshtml * * * estadisticaHoras * * * PerfilDocente.cshtml * * *
Valores * * * Estadisticas.cshtml * * * tutoriasEstudiante.cshtml * * * Notificaciones.cshtml * * * Docente.cshtml * * *
horas * * * moduloHorarioTutoriasEstudiante.cshtml * * * moduloHorarioEstudiante.cshtml * * * perfilEstudiante.cshtml * * * Index.cshtml * * * Estudiante.cshtml * * * Facultad.cshtml * * *
138 - * * * TutoriasApp\Controllers\SuperAdmin\Controlador - * * * docentesCedula(string Cedula)

int semanas = 16;

//TUTORIAS DADAS POR DOCENTE
public ActionResult BarnesDocente(string Cedula, string CodigoPeriodo)
{
    metodoModulo metodoModulo = new metodoModulo();
    metodoHorarioTutorias metodoHorarioTutorias = new metodoHorarioTutorias();
    metodoDocente metodoDocente = new metodoDocente();
    docente docente = new docente();

    docente = metodoDocente.consultaDocente(Cedula);

    int horasLibre = metodoModulo.consultaHorasLibres(Cedula).ToList().Count * semanas;
    int horasObligatorias = metodoModulo.consultaHorasObligatorias(Cedula).ToList().Count * semanas;
    int horasPlanificadas = horasLibre + horasObligatorias;

    int horasDadas = metodoHorarioTutorias.consultaHorarioTutoriasAsistencia("SI").Where(x => x.CedulaDocente == Cedula && x.CodigoPeriodo == CodigoPeriodo).Count;

    var data = new
    {
        HorasPlanificadas = horasPlanificadas,
        HorasDadas = horasDadas,
        Docente = docente
    };

    return Json(data, JsonRequestBehavior.AllowGet);
}

//TUTORIAS DADAS POR DOCENTE POR CARRERA
```

Controlador Reportes

Anexo 2. Manual de uso del sistema de tutorías UDLA

1. Descripción del sistema

Tutorías UDLA, es un sistema web que permite administrar de manera digital todo el proceso de asignación de tutorías entre estudiantes y docentes de la universidad, así como el ingreso de docentes, estudiantes y sus respectivas cargas académicas el periodo actual.

El sistema actualmente cuenta con tres secciones, las cuales son:

- ✓ Súper Administrador
- ✓ Docente
- ✓ Estudiante

Cada sección está diseñada y estructuradamente para el rol que la persona ejerza en el sistema universitario.

2. Recomendaciones a su previo uso

Para la utilización del sistema de tutorías UDLA, se recomienda tener un navegador Firefox o Chrome actualizado, así como una velocidad conexión óptima para el manejo de los datos correctamente.

Adicionalmente el sistema solo traba con correos de la plataforma Office 365 de la UDLA, por lo que intentar navegar con un correo que no pertenezca a esta plataforma no permitirá su funcionamiento.

Nota: El enlace de direccionamiento dependerá del servidor local de la universidad, y como está asigne el nombre de sub-dominio. Ejemplo:

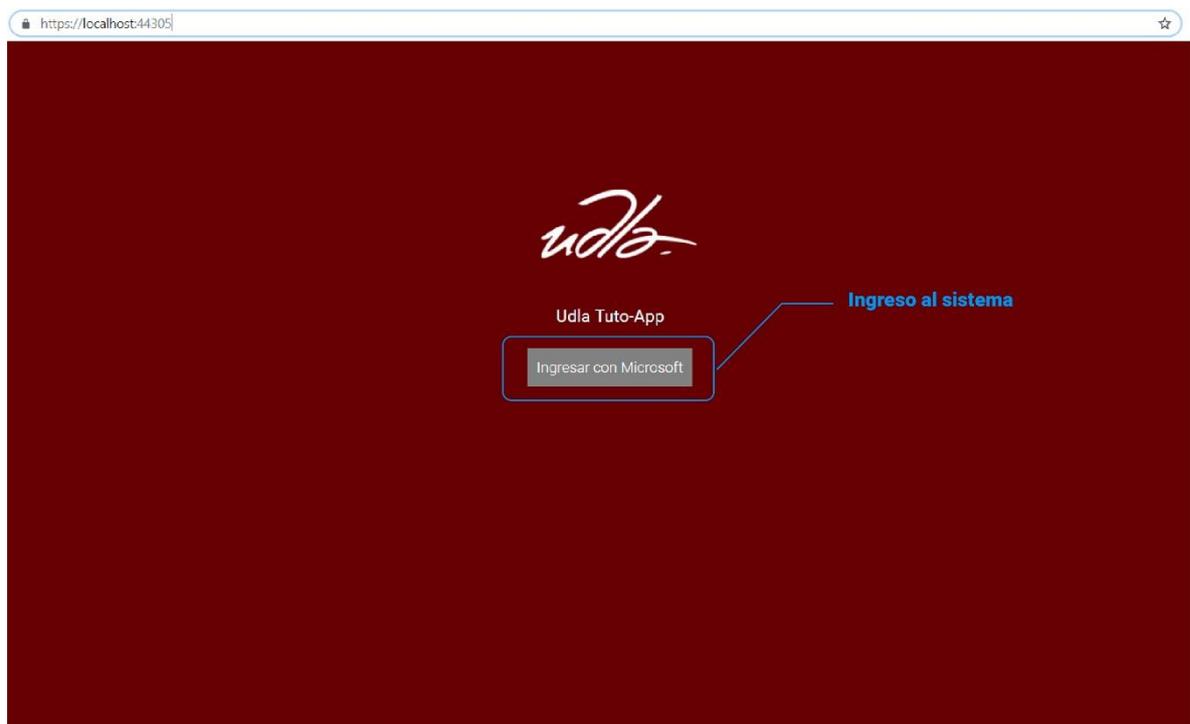
<https://tutorias.udla.edu.ec>

3. Uso del Sistema

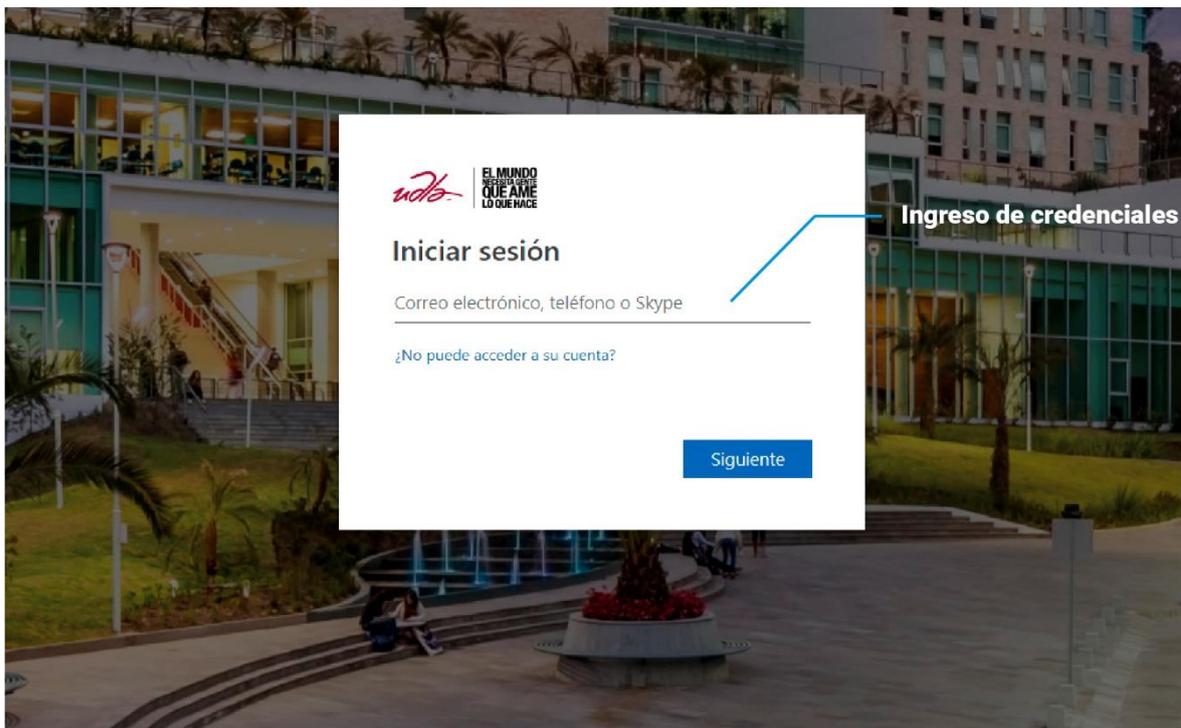
Para ingresar acceda a la dirección web asignada por la universidad, en este caso como usaremos de manera local a la dirección <https://localhost:44305/>.

3.1. Pantalla de acceso principal

Como pantalla de ingreso con un botón principal el cual nos permite dirigirnos a la plataforma office 365 de la universidad UDLA, para poder ingresar correctamente.



Ingrese las credenciales al rol que pertenezca la cuenta y acceda.



3.2. SUPER ADMIN

Ingrese las credenciales asignadas al súper administrador, estas serán asignadas a la persona la cual manipulará esta sección del sistema y será responsable de la misma.

Estas credenciales se asignan en la implementación del sistema. Estas credenciales tienen que ser un correo universitario de la plataforma office 365 con su respectiva contraseña universitaria.

3.2.1. Panel Administrativo

Al ingresar a la página principal del usuario administrador, nos encontramos con un mensaje de bienvenida al administrador actual.

En la parte izquierda de la página podemos observar un panel o menú.



Este panel cuenta con los siguientes módulos:

- ✓ Facultad
- ✓ Carrera
- ✓ Materia
- ✓ Módulos/Periodo Académico
- ✓ Docente
- ✓ Horario Docentes
- ✓ Estudiante
- ✓ Horario Estudiantes
- ✓ Reportes Tutorías

Adicional cuenta su respectivo botón para cerrar la sesión actual.

Cerrar Sección

3.2.2. Módulos

Cada módulo permite realizar las diferentes operaciones del sistema en la sección del administrador, para lo cual vamos a conocer el funcionamiento de cada uno de los mismos.

3.2.2.1. Facultad

En esta sección el administrador podrá:

1. Ingresar facultades

2. Observar las Facultades existentes en el sistema
3. Eliminar la Facultad seleccionada
4. Actualizar la Facultad seleccionada

Facultad
Ingresa, actualiza o elimina las Facultades actuales de la Universidad

1 Ingresar Facultad

Ingrese código de la Facultad Ingrese Facultad **Ingresar** ✓

2 Facultades Actuales

Cod. Facultad	Nombre Facultad	Acción	
FICA	FACULDADE DE INGENIERIA Y CIENCIAS APLICADAS	Eliminar	Actualizar
FACEA	FACULTAD DE ADMINISTRACION	Eliminar	Actualizar
MED	MEDICINA	Eliminar	Actualizar

3 Eliminar la Facultad

4 Actualiza la Facultad

Al actualizar una facultad, se mostrará un panel lateral derecho el cual permite modificar el nombre de la facultad seleccionada.

Facultad
Ingresa, actualiza o elimina las Facultades actuales de la Universidad

Ingrese código de la Facultad Ingrese Facultad **Ingresar** ✓

Actualizando Facultad

Editar Facultad

Código Facultad: FICA

Nombre Facultad: FACULDADE DE INGENIERIA Y CIENCIAS

Aceptar ✓ **Cancelar** ✕

Tabla de Facultades Actuales

Cod. Facultad	Nombre Facultad	Acción	
FICA	FACULDADE DE INGENIERIA Y CIENCIAS APLICADAS	Eliminar	Actualizar
FACEA	FACULTAD DE ADMINISTRACION	Eliminar	Actualizar
MED	MEDICINA	Eliminar	Actualizar

3.2.2.2. Carrera

En esta sección el administrador podrá:

1. Ingresar Carreras:

Seleccione la facultad existente a la cual va a pertenecer la carrera a ingresar.

2. Filtrar por Carreras existentes en el sistema
3. Observar las Carreras Filtradas

4. Eliminar la Carrera seleccionada
5. Actualizar la Carrera seleccionada

Carrera
Ingresa, actualiza o elimina las Carreras actuales de la Universidad

1 Ingresar Carrera

Facultad: FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS | Código Carrera: | Ingrese Carrera: | Ingresar

Selecciona la Facultad para mostrar las carreras

Facultad: FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS | Consultar

2 Filtro por Carrera

3 Carreras Filtradas

Tabla de Carreras Actuales

Nombre Facultad	Cod. Carrera	Nombre Carrera	Acción
FICA	511	ING SISTEMAS	Eliminar Actualizar
FICA	611	ING ELECTRONICA	Eliminar Actualizar

4 Eliminar Carrera

5 Actualiza Carrera

Al actualizar una Carrera, se mostrará un panel lateral derecho el cual permite modificar el nombre de la carrera seleccionada.

Carrera
Ingresa, actualiza o elimina las Carreras actuales de la Universidad

Facultad: FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS | Código Carrera: | Ingrese Carrera: | Ingresar

Selecciona la Facultad para mostrar las carreras

Facultad: FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS | Consultar

Actualizando Carrera

Editar Carrera

Código Carrera: 511

Nombre Carrera: ING SISTEMAS

Aceptar ✓ | Cancelar ↻

Tabla de Carreras Actuales

Nombre Facultad	Cod. Carrera	Nombre Carrera	Acción
FICA	511	ING SISTEMAS	Eliminar Actualizar
FICA	611	ING ELECTRONICA	Eliminar Actualizar

3.2.2.3. Materia

Materia
Ingresa, actualiza o elimina las Materias actuales de la Universidad

1 Ingresar Materia

Facultad: FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS
Carrera: ING SISTEMAS
Código Materia:
Ingrese Materia:
Ingresar

2 Carga Masiva de Materias

Cargar Materias desde Excel
Seleccionar archivo | No se eligió archivo | **Cargar Lista**

Selecciona la Facultad y Carrera para mostrar las materias

3 Filtro Materias por Carrera

Facultad: FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS
Carrera: ING SISTEMAS
Consultar

4 Materias Filtradas

Tabla de Materias Actuales

Cod. Facultad	Cod. Carrera	Cod. Materia	Nombre Materia	Acción	
FICA	511	FIS100	FISICA I	Eliminar	Actualizar
FICA	511	SIS1	INTRODUCCIÓN A LA PROGRAMACIÓN	Eliminar	Actualizar
FICA	511	BDD	BASE DE DATOS I	Eliminar	Actualizar

5 Eliminar Materia

6 Actualizar Materia

En esta sección el administrador podrá:

1. Ingresar Materias por carrera:

Seleccione la facultad existente, seleccione la carrera de la facultad seleccionada, ingrese los datos de la nueva materia a ingresar.

Materia
Ingresa, actualiza o elimina las Materias actuales de la Universidad

1 Ingresar Materia

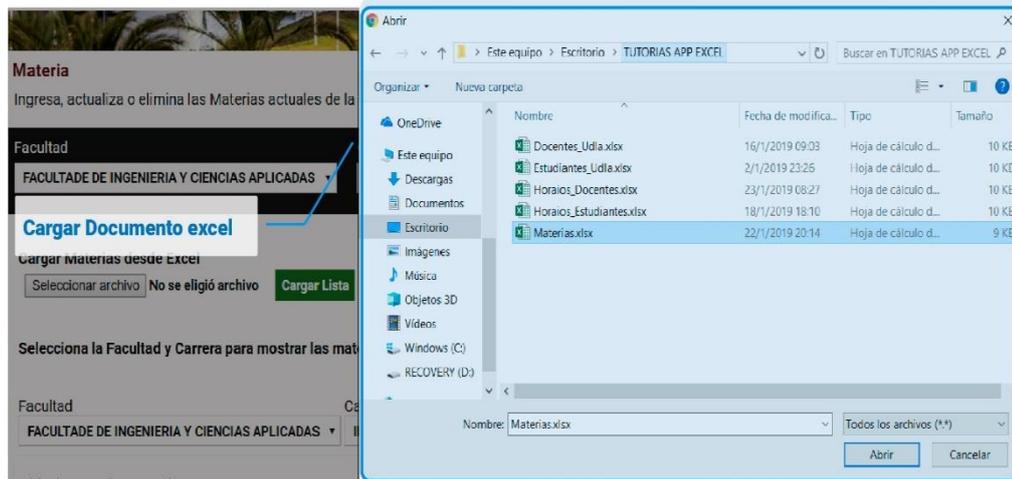
Facultad: FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS
Carrera: ING SISTEMAS
Código Materia:
Ingrese Materia:
Ingresar

2. Cargar masivamente materias desde un documento de Excel

2 Carga Masiva de Materias

Cargar Materias desde Excel
Seleccionar archivo | No se eligió archivo | **Cargar Lista**

Presione el botón seleccionar archivo y seleccione el documento correspondiente a las materias a ingresar.



El documento tiene que tener el siguiente formato adecuado para el correcto ingreso de los datos.

CodigoCarrera	CodigoMateria	NombreMateria
511	FIS100	FISICA I
511	MAT330	ECUACIONES DIFERENCIALES
511	EMP	EMPRESAS
511	BDD	BASE DE DATOS I

3. Filtrar materias por carrera

Seleccione la facultad y carrera de la cual desea consultar las materias.



4. Observar las materias filtradas por carrera

Tabla de Materias Actuales

4 Materias Filtradas

Cod. Facultad	Cod. Carrera	Cod. Materia	Nombre Materia	Acción	
FICA	511	FIS100	FISICA I	Eliminar	Actualizar
FICA	511	SIS1	INTRODUCCIÓN A LA PROGRAMACIÓN	Eliminar	Actualizar
FICA	511	BDD	BASE DE DATOS I	Eliminar	Actualizar

5 Eliminar Materia

6 Actualizar Materia

5. Eliminar la Materia seleccionada
6. Actualizar la Materia seleccionada

Al actualizar una Materia, se mostrará un panel lateral derecho el cual permite modificar el nombre de la carrera seleccionada.

Editar Materia

Codigo Materia: FIS100

Nombre Materia: FISICA I

Aceptar ✓ Cancelar ↻

Actualizando Materia

Tabla de Materias Actuales

Cod. Facultad	Cod. Carrera	Cod. Materia	Nombre Materia	Acción	
FICA	511	FIS100	FISICA I	Eliminar	Actualizar
FICA	511	SIS1	INTRODUCCIÓN A LA PROGRAMACIÓN	Eliminar	Actualizar
FICA	511	BDD	BASE DE DATOS I	Eliminar	Actualizar

3.2.2.4. Módulos/Periodo Académico

En esta sección módulos el administrador podrá:

1. Ingresar Módulos:
2. Observar Módulos Actuales
3. Eliminar módulo seleccionado

Módulos/Período Académico

Esta sección contiene todos los módulos(horas de clase) de la universidad, así como el período actual de la misma.

Módulos

Ingresar o elimine los módulos actuales de la Universidad

1 Ingresar Módulo

Código Modulo	Día	Hora Inicio	Hora Fin	Ingresar
<input type="text"/>	Lunes	--:--	--:--	

2 Módulos Actuales

Tabla de Módulos(horas de clase) Actuales

Código Módulo	Día	Hora Inicio	Hora Fin	Acción
101	Lunes	7:00	8:00	Eliminar
102	Lunes	8:05	9:05	Eliminar
103	Lunes	9:10	10:10	Eliminar
104	Lunes	10:15	11:15	Eliminar
105	Lunes	11:20	12:20	Eliminar
106	Lunes	12:25	13:25	Eliminar
108	Lunes	13:30	14:30	Eliminar
109	Lunes	14:35	15:35	Eliminar

3 Eliminar Módulo

En esta sección periodo académico el administrador podrá:

1. Ingresar periodo académico:
2. Observar los periodos académicos actuales
3. Eliminar periodo seleccionado

Período Académico

Ingrese el período actual de clases de la universidad

1 Ingresar Periodo Académico

Código Período Académico	Nombre Período Académico	Ingresar
<input type="text"/>	<input type="text"/>	

2 Periodos Académicos

Período actual de estudio de la universidad

Código Período Académico	Nombre Período Académico	Acción
PEP2018	SEPTIEMBRE 2018	Eliminar

3 Eliminar Periodo Académico

3.2.2.5. Docente



DOCENTE
Ingresa, actualiza o elimina los Docentes actuales de la Universidad

1 Ingresar Docente

Primer Nombre Segundo Nombre Apellido Paterno Apellido Materno Cédula Email
Celular **Ingresar**

Cargar Docentes desde Excel
Seleccionar archivo No se eligió archivo **Cargar Lista**

2 Carga Masiva de Docentes

3 Docentes Actuales

Tabla de Docentes Actuales

Primer Nombre	Segundo Nombre	Apellido Paterno	Apellido Materno	Cédula	Email	Celular	Acción
ANDRES	ALEJANDRO	TAPIA	LOPEZ	1720765533	andres.tapia.lopez@udla.edu.ec	087896587	Eliminar Actualizar
ANITA		YANEZ		1802462216	anita.yanez@udla.edu.ec		Eliminar Actualizar

4 Eliminar Docente

5 Actualizar Docente

En esta sección el administrador podrá:

1. Ingresar Docentes

DOCENTE
Ingresa, actualiza o elimina los Docentes actuales de la Universidad

1 Ingresar Docente

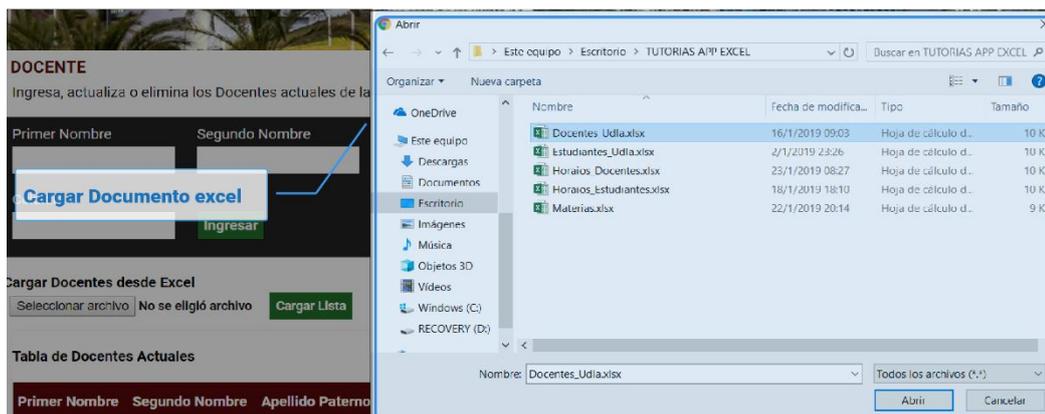
Primer Nombre Segundo Nombre Apellido Paterno Apellido Materno Cédula Email
Celular **Ingresar**

2. Cargar masivamente docentes desde un documento de Excel

Cargar Docentes desde Excel
Seleccionar archivo No se eligió archivo **Cargar Lista**

2 Carga Masiva de Docentes

Presione el botón seleccionar archivo y seleccione el documento correspondiente a ingresar.



El documento tiene que tener el siguiente formato adecuado para el correcto ingreso de los datos.

The screenshot shows an Excel spreadsheet with the following data:

Nombre	NombreDos	Apellido	ApellidoDos	Cedula	Correo	Celular
EDDY		ARMAS		1711715803	eddy.armas@udla.edu.ec	
VERONICA		FALCONI		0602395270	veronica.falconi@udla.edu.ec	

3. Observar las docentes actuales

The screenshot shows the 'Tabla de Docentes Actuales' table with three rows of data. Annotations point to the table title and the 'Eliminar' and 'Actualizar' columns.

Primer Nombre	Segundo Nombre	Apellido Paterno	Apellido Materno	Cédula	Email	Celular	Acción
ANDRES	ALEJANDRO	TAPIA	LOPEZ	1720765533	andres.tapia.lopez@udla.edu.ec	087896587	Eliminar Actualizar
ANITA		YANEZ		1802462216	anita.yanez@udla.edu.ec		Eliminar Actualizar

4. Eliminar docente seleccionado

5. Actualizar docente seleccionado

Al actualizar un Docente, se mostrará un panel lateral derecho el cual permite modificar los datos ingresados del docente a excepción de la cédula, ya que es su identificador primario.

DOCENTE
Ingresa, actualiza o elimina los Docentes actuales de la Universidad

Primer Nombre Segundo Nombre Apellido Paterno Apellido Materno Cédula

Celular

Actualizar Docente

Cargar Docentes desde Excel
Seleccionar archivo No se eligió archivo

Tabla de Docentes Actuales

Primer Nombre	Segundo Nombre	Apellido Paterno	Apellido Materno	Cédula	Email
ANDRES	ALEJANDRO	TAPIA	LOPEZ	1720765533	andres.tapia.lopez@udla.edu.ec
ANITA		YANEZ		1802462216	anita.yanez@udla.edu.ec

Editar Docente

Cédula: 1720765533

Primer Nombre
ANDRES

Segundo Nombre
ALEJANDRO

Apellido Paterno
TAPIA

Apellido Materno
LOPEZ

Email
andres.tapia.lopez@udla.edu.ec

Celular
087896587

Eliminar Actualizar

Aceptar Eliminar Cancelar

3.2.2.6. Horario Docente

Horario de Clases del Docente por Materia y Módulo

Ingresa o elimina la hora de clase del docente actual, en cada una de las materias correspondientes a cada carrera de la facultad.

1 Ingresar Horario Docente

Facultad Carrera Materia Docente

FACULTAD DE INGENIERIA Y CIENCIAS APLICADAS ING SISTEMAS FISICA I

Módulo Período Académico Aula Paralelo

101 Lunes 7:00 - 8:00 SEPTIEMBRE 2018

Cargar Horarios de Docentes desde Excel

Seleccionar archivo No se eligió archivo

2 Carga Masiva Horario de Docentes

Selecciona el Docente Ingresado para consultar su horario

Docente Materia Paralelo

1720765533 BASE DE DATOS I 1

3 Eliminar Materia del horario

HORARIO DE ANDRES ALEJANDRO TAPIA LOPEZ

Hora	Lunes	Martes	Miércoles	Jueves	Viernes	Sabado
07:00 - 08:00	FIS100-1/200	SIS1-1/457		EST78-1/457		
08:05 - 09:05	LIBRE					
09:10 - 10:10	LIBRE					
10:15 - 11:15						
11:20 - 12:20	TUT.ESTU.					
12:25 - 13:25	TUT.ESTU.					
ALMUERZO						
13:30 - 14:30		LIBRE				
14:35 - 15:35						
15:40 - 16:40						
16:45 - 17:45	BDD-1/415	LIBRE				
17:50 - 18:50						
18:50 - 19:50						
19:50 - 20:50						
20:50 - 21:50	TUT.ESTU.		LIBRE			

4 Horario Actual del Docente

En esta sección el administrador podrá:

1. Ingresar Docentes:

Para ingresar el horario de clases del docente realiza lo siguiente

- ✓ Selecciona una facultad existente
- ✓ Selecciona la carrera de la facultad seleccionada
- ✓ Selecciona la materia de la carrera seleccionada
- ✓ Busca y selecciona un docente ingresado en el sistema
- ✓ Selecciona un módulo existente
- ✓ Selecciona el periodo académico actual
- ✓ Ingresar un número de aula a impartir la clase
- ✓ Ingresar el paralelo de la clase que imparte el docente en la materia seleccionada

Horario de Clases del Docente por Materia y Módulo

Ingresar o eliminar la hora de clase del docente actual, en cada una de las materias correspondientes a cada carrera de la facultad.

1 Ingresar Horario Docente

Facultad	Carrera	Materia	Docente
FACULTAD DE INGENIERIA Y CIENCIAS APLICADAS	ING SISTEMAS	FISICA I	
Módulo	Período Académico	Aula	Paralelo
101 Lunes 7:00 - 8:00	SEPTIEMBRE 2018		

Ingresar

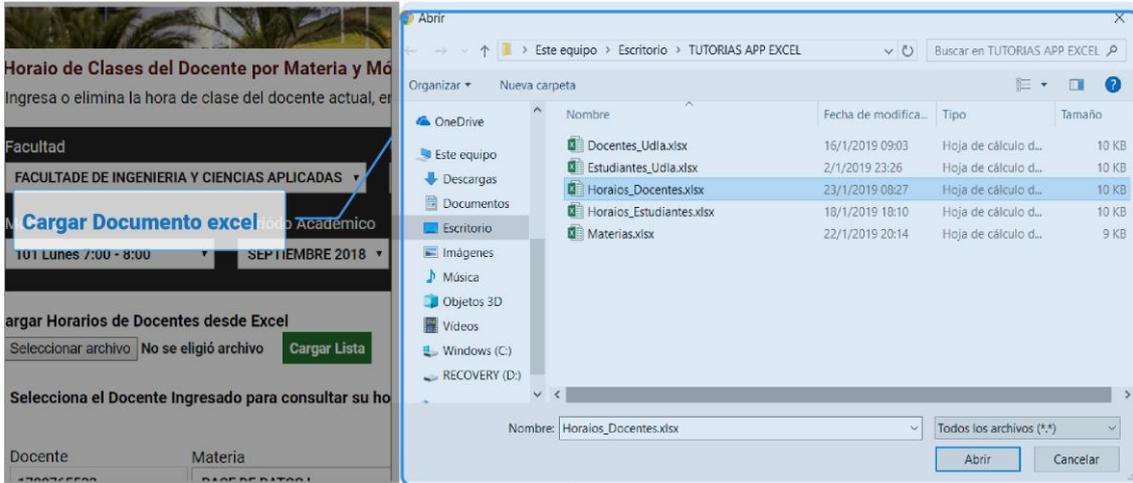
2. Cargar masivamente horarios de los docentes desde un documento de Excel

Cargar Horarios de Docentes desde Excel

Seleccionar archivo No se eligió archivo Cargar Lista

2 Carga Masiva Horario de Docentes

Presione el botón seleccionar archivo y seleccione el documento correspondiente a ingresar.



El documento tiene que tener el siguiente formato adecuado para el correcto ingreso de los datos.

	A	B	C	D	E	F	G
1	CodMateria	CedulaDocente	Modulo	Periodo	Aula	Paralelo	
2	FIS100	1500829427	101	PEP2018	503	1	
3	FIS100	1500829427	102	PEP2018	200	2	
4	ACI520	1711715803	403	PEP2020	456	2	
5	ACI220	0502395270	506	PEP2021	421	1	
6	EMP	0502395270	507	PEP2018	602	2	
7	MAT110	0502395270	202	PEP2018	603	1	
8	BDD	1711715803	2013	PEP2018	604	1	
9							
10							
11							
12							
13							
14							

3. Elimina Materias del horario del docente:

- ✓ Busca el docente por nombre o cédula
- ✓ busca la materia que se encuentre en su horario
- ✓ selecciona el paralelo de la misma
- ✓ elimina.

The image shows a portion of the web application interface. It features three dropdown menus: 'Docente' with the value '1720765533', 'Materia' with the value 'BASE DE DATOS I', and 'Paralelo' with the value '1'. To the right of these dropdowns is a green button labeled 'Eliminar Materia'.

3 Eliminar Materia del horario

4. Consulta el Horario del docente ingresando su nombre o cédula

HORARIO DE ANDRES ALEJANDRO TAPIA LOPEZ

Hora	Lunes	Martes	Miercoles	Jueves	Viernes	Sabado
07:00 - 08:00	FIS100-1/200	SIS1-1/457		EST78-1/457		
08:05 - 09:05	LIBRE					
09:10 - 10:10	LIBRE					
10:15 - 11:15						
11:20 - 12:20	TUT.ESTU.					
12:25 - 13:25	TUT.ESTU.					
ALMUERZO						
13:30 - 14:30		LIBRE				
14:35 - 15:35						
15:40 - 16:40						
16:45 - 17:45	BDD-1/415	LIBRE				
17:50 - 18:50						
18:50 - 19:50						
19:50 - 20:50						
20:50 - 21:50	TUT.ESTU.		LIBRE			

4 Horario Actual del Docente

3.2.2.7. Estudiante

Estudiante

1 Ingresar Estudiante

Facultad	Carrera	Matrícula	Primer Nombre	Segundo Nombre
FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS	ING SISTEMAS			
Apellido Paterno	Apellido Materno	Email	Celular	Ingresar

Cargar Estudiantes desde Excel

Seleccionar archivo No se eligió archivo Cargar Lista

2 Carga Masiva de Estudiantes

Selecciona la Facultad y Carrera para mostrar las materias

Facultad	Carrera	Consultar
FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS	ING SISTEMAS	

3 Filtro Estudiantes por Carrera

4 Estudiantes Actuales

Facultad	Carera	Matrícula	Primer Nombre	Segundo Nombre	Apellido Paterno	Apellido Materno	Email	Celular	Acción
FICA	511	501735	JOAN	SEBASTIAN	ANDRADE	CASCO	joan.andrade@udla.edu.ec	0978616500	Eliminar Actualizar
FICA	511	14789	PEDRO		NAVAJA		pedro@udla.edu.ec	0747854785	Eliminar Actualizar
FICA	511	789654	MARIA		CAZA		maria@udla.edu.ec	0854785698	Eliminar Actualizar

5 Eliminar Estudiantes

6 Actualizar Estudaintes

En esta sección el administrador podrá:

1. Ingresar Estudiantes
- ✓ Seleccione la facultad
 - ✓ Seleccione la carrera de la facultad
 - ✓ Ingrese los datos del estudiante

Estudiante

1 Ingresar Estudiante

Facultad	Carrera	Matrícula	Primer Nombre	Segundo Nombre
FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS	ING SISTEMAS			
Apellido Paterno	Apellido Materno	Email	Celular	Ingresar

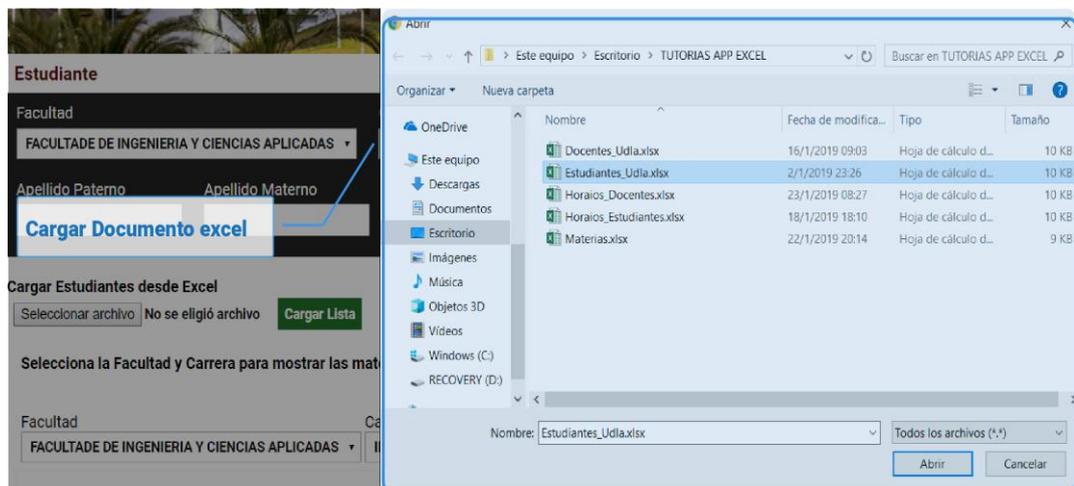
2. Cargar masivamente estudiantes desde un documento de Excel

Cargar Estudiantes desde Excel

2 Carga Masiva de Estudiantes

Seleccionar archivo No se eligió archivo Cargar Lista

Presione el botón seleccionar archivo y seleccione el documento correspondiente a ingresar.



El documento tiene que tener el siguiente formato adecuado para el correcto ingreso de los datos.

Nombre	NombreDos	Apellido	ApellidoDos	Matricula	Correo	Celular	CodCarrera
DAVID	ARMANDO	CHAGUARO	VASCONEZ	504789	david.chaguaro@udla.edu.ec	983361439	511
JOAN	SEBASTIAN	ANDRADE	CASCO	501735	joan.andrade@udla.edu.ec	978616500	511

3. Filtro de Estudiantes por carrera

3 Filtro Estudiantes por Carrera

Facultad: FACULDADE DE INGENIERIA Y CIENCIAS APLICADAS | Carrera: ING SISTEMAS | **Consultar**

4 Estudiantes Actuales

Facultad	Carera	Matrícula	Primer Nombre	Segundo Nombre	Apellido Paterno	Apellido Materno	Email	Celular	Acción
FICA	511	501735	JOAN	SEBASTIAN	ANDRADE	CASCO	joan.andrade@udla.edu.ec	0978616500	Eliminar Actualizar
FICA	511	14789	PEDRO		NAVAJA		pedro@udla.edu.ec	0747854785	Eliminar Actualizar
FICA	511	789654	MARIA		CAZA		maria@udla.edu.ec	0854785698	Eliminar Actualizar

5 Eliminar Estudiantes

6 Actualizar Estudiantes

4. Observa los Estudiantes actuales por carrera
5. Eliminar estudiante seleccionado
6. Actualizar estudiante seleccionado

Al actualizar un Estudiante, se mostrará un panel lateral derecho el cual permite modificar los datos ingresados del estudiante a excepción de la matrícula, ya que es su identificador primario.

Estudiante

Facultad: FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS | Carrera: ING SISTEMAS | Matrícula: | Primer Nombre: |

Apellido Paterno: | Apellido Materno: | Email: | Celular: | **Ingresar**

Actualizar Estudiante

Editar Estudiante

Matrícula 501735

Primer Nombre: JOAN

Segundo Nombre: SEBASTIAN

Apellido Paterno: ANDRADE

Apellido Materno: CASCO

Email: joan.andrade@udla.edu.ec

Celular: 0978616500

Cargar Estudiantes desde Excel

Seleccionar archivo | No se eligió archivo | **Cargar Lista**

Selecciona la Facultad y Carrera para mostrar las materias

Facultad: FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS | Carrera: ING SISTEMAS | **Consultar**

Facultad	Carera	Matrícula	Primer Nombre	Segundo Nombre	Apellido Paterno	Apellido Materno	Email
FICA	511	501735	JOAN	SEBASTIAN	ANDRADE	CASCO	joan.andrade@udla.edu.ec
FICA	511	14789	PEDRO		NAVAJA		pedro@udla.edu.ec
FICA	511	789654	MARIA		CAZA		maria@udla.edu.ec

Acción: **Aceptar** | **Cancelar**

0747854785 | Eliminar | Actualizar

0854785698 | Eliminar | Actualizar

3.2.2.8. Horario Estudiante

Horario de Clases Estudiantes

Ingresar o elimina la hora de clase del estudiante actual, en cada una de las materias correspondientes al mismo. Asigna un paralelo y un número de aula de clase.

1 Ingresar Horario Estudiantes

Facultad: FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS | Carrera: ING SISTEMAS | Materia: FISICA I | Paralelo: 1

Docente: ANDRES ALEJANDRO TAPIA LOPEZ | Estudiante: | Número de Matrícula: 1 | **Ingresar**

Cargar Horarios de Estudiantes desde Excel

Seleccionar archivo | No se eligió archivo | **Cargar Lista**

2 Carga Masiva Horario de Estudiantes

3 Eliminar Materia del horario

Selecciona el Estudiante Ingresado para consultar su horario

Facultad: FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS | Carrera: ING SISTEMAS | Estudiante: | Materia: | **Eliminar Materia**

HORARIO DE JOAN SEBASTIAN ANDRADE CASCO

Horario de Clases

Conoce las materias y horas de clases. ¡No te descuides de tu horario!

4 Horario Actual del Estudiante

Hora	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
07:00 - 08:00	FIS100-1/200	SIS1-1/457		EST78-1/457		
08:05 - 09:05						
09:10 - 10:10						
10:15 - 11:15						
11:20 - 12:20						
12:25 - 13:25						
ALMUERZO						
13:30 - 14:30						
14:35 - 15:35						
15:40 - 16:40						
16:45 - 17:45	BDD-1/415	BDD-1/304				
17:50 - 18:50						
18:50 - 19:50						
19:50 - 20:50						
20:50 - 21:50						

En esta sección el administrador podrá:

1. Ingresar Horario Estudiante:

Para ingresar el horario de clases del estudiante realiza lo siguiente

- ✓ Selecciona una facultad existente
- ✓ Selecciona la carrera de la facultad seleccionada
- ✓ Selecciona la materia de la carrera seleccionada
- ✓ Selecciona un paralelo de la materia
- ✓ Selecciona el docente de la materia seleccionada
- ✓ Busca y selecciona un estudiante por su nombre o matrícula
- ✓ Ingresas el número de matrícula en la cual está cruzando el estudiante dicha materia

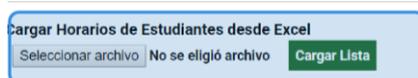
Horario de Clases Estudiantes

Ingresas o elimina la hora de clase del estudiante actual, en cada una de las materias correspondientes al mismo. Asigna un paralelo y un número de aula de clase.

1 Ingresar Horario Estudiantes



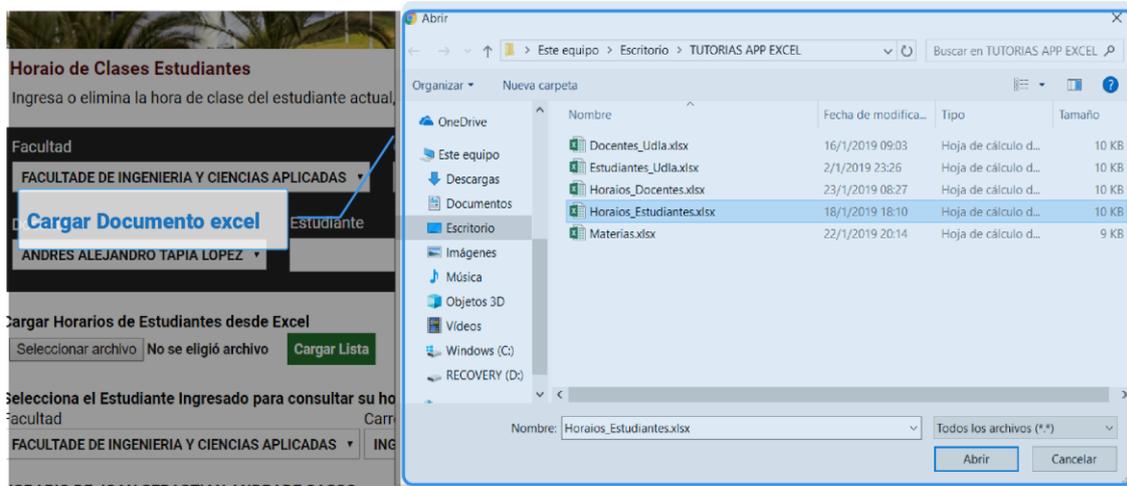
2. Cargar masivamente horarios de los estudiantes desde un documento de Excel



2 Carga Masiva Horario de Estudiantes

3 Elimina

Presione el botón seleccionar archivo y seleccione el documento correspondiente a ingresar.



El documento tiene que tener el siguiente formato adecuado para el correcto ingreso de los datos.

	A	B	C	D	E	F
1	MATRICULA ESTUDIANTE	CEDULA DOCENTE	CODIGO MATERIA	Repitencia	Paralelo	
2	504789	1500829427	FIS100	1	1	
3	504789	1500829427	FIS100	2	2	
4	504789	1720645223	MAT330	1	2	
5	504789	1720645223	MAT330	1	2	
6						
7						
8						
9						
10						

3. Elimina Materias del horario del docente:

- ✓ Selecciona la Facultad
- ✓ Selecciona la carrera de la facultad
- ✓ Busca el estudiante por nombre o matrícula
- ✓ Selecciona la materia del estudiante
- ✓ elimina.

Facultad	Carrera	Estudiante	Materia
FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS	ING SISTEMAS		Eliminar Materia

4. Consulta el Horario del estudiante ingresando su nombre o matrícula

HORARIO DE JOAN SEBASTIAN ANDRADE CASCO

Horario de Clases

Conoce las materias y horas de clases. ¡No te descuides de tu horario.!

4 Horario Actual del Estudiante

Hora	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
07:00 - 08:00	FIS100-1/200	SIS1-1/457		EST76-1/457		
08:05 - 09:05						
09:10 - 10:10						
10:15 - 11:15						
11:20 - 12:20						
12:25 - 13:25						
ALMUERZO						
13:30 - 14:30						
14:35 - 15:35						
15:40 - 16:40						
16:45 - 17:45	BDD-1/415	BDD-1/304				
17:50 - 18:50						
18:50 - 19:50						
19:50 - 20:50						
20:50 - 21:50						

3.2.2.9. Reportes



Reportes de Tutorías

Registro de reportes de tutorías de los estudiantes

Panel Reportes

Tutorías por docente Tutorías por carrera Tutorías por matrícula de estudiante Tutorías por número de matrícula Tutorías por materia

Analiza la efectividad de cada docente en su cronograma de tutorías

En este módulo el administrador podrá consultar reportes de:

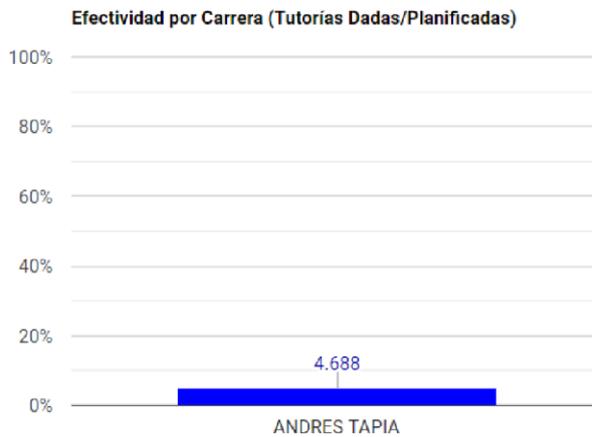
1. Tutorías por docente

Período	Docente	Generar Reporte
SEPTIEMBRE 2018 ▾	1720765533	

Generar Reporte Docente

Docente	Horas Planificadas	Horas Dadas	Efectividad(Total Tutorías/Asisitidas)
ANDRES TAPIA	128 horas	6 horas	4.6875 %

Análisis Docente



2. Tutorías por carrera

Estadísticas de docentes por Carrer

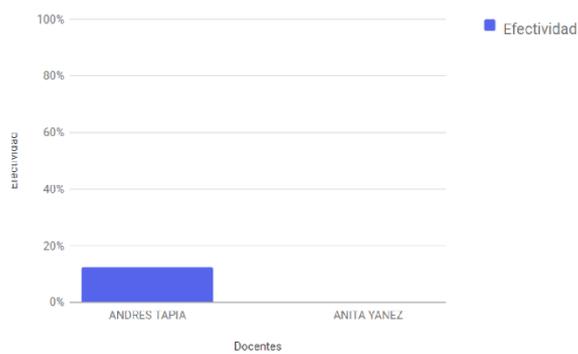
Período	Facultad	Carrera	Generar Reporte
SEPTIEMBRE 2018	FACULDADE DE INGENIERIA Y CIENCIAS APLICADAS	ING SISTEMAS	

Generar Reporte

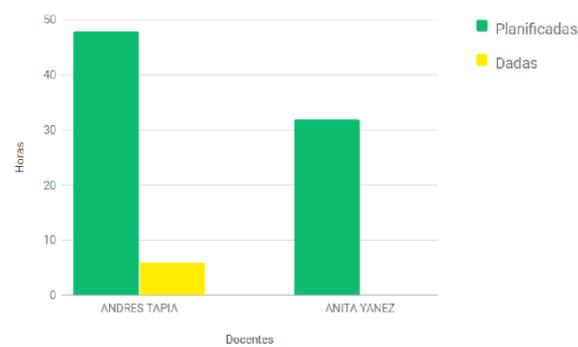
Carrera: ING SISTEMAS

Docentes	TutoríasPlanificadas	TutoríasDadas	Tutorías Carrera	Tutorías Otras Carreras	Número Estudiantes	Efectividad
ANDRES TAPIA	48	6	5	1	3	12.50%
ANITA YANEZ	32	0	0	0	0	0.00%

Efectividad Tutorías Dadas/Planificadas



Horas Planificadas vs Horas Dadas



3. Tutorías por matrícula del estudiante

Estadísticas de tutorías por repitencia de estudiante

Período	Facultad	Carrera	Número de Matrícula	Generar Reporte
SEPTIEMBRE 2018	FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS	ING SISTEMAS	1	

Generar Reporte

Carrera: ING SISTEMAS

Estudiante	Materia	Número tutorías Planificadas	Número de tutorías Asistidas
JOAN ANDRADE	FIS100	2	2
JOAN ANDRADE	BDD	3	0
PEDRO NAVAJA	SIS1	1	1

4. Tutorías por número de matrícula

Estadística de repitencia por carrera

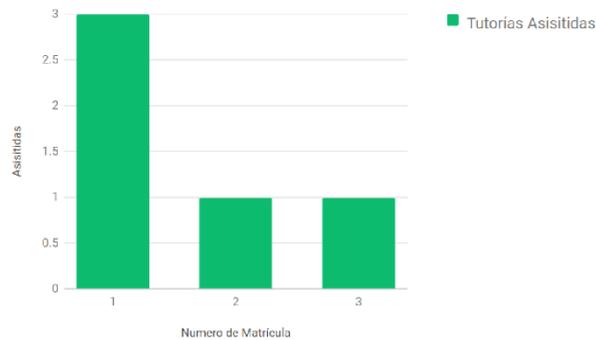
Período	Facultad	Carrera	Generar Reporte
SEPTIEMBRE 2018	FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS	ING SISTEMAS	

Generar Reporte

Carrera: ING SISTEMAS

Número Matrícula(Repitencia)	Tutorías Asistidas(SI)
1	3
2	1
3	1

Tutorías por tipo de Matrículas



5. Tutorías por materia

Estadística de repitencia por Materia

Período	Facultad	Carrera	Materia	
SEPTIEMBRE 2018	FACULTADE DE INGENIERIA Y CIENCIAS APLICADAS	ING SISTEMAS	FISICA I	Generar Reporte

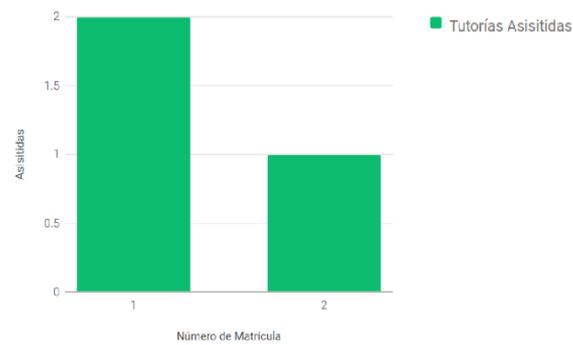
Generar Reporte

Carrera: ING SISTEMAS

Materia: FISICA I

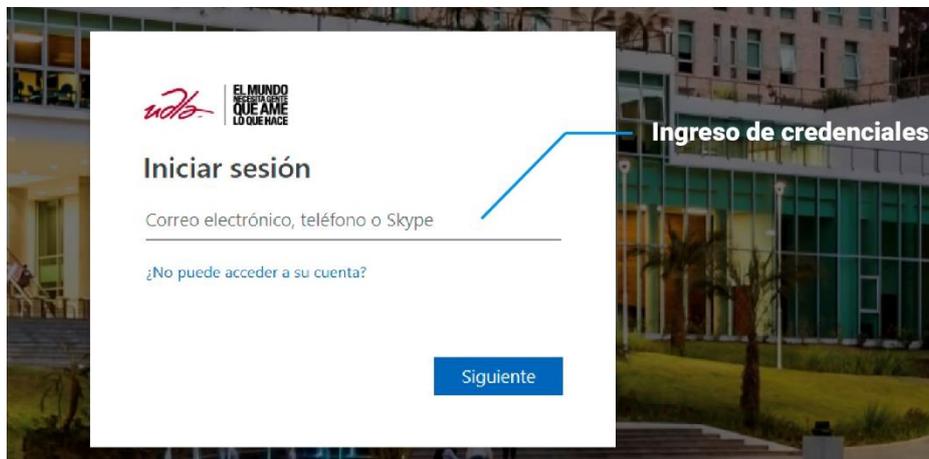
Número Matrícula(Repitencia)	Tutorías Asistidas(SI)
1	2
2	1

Tutorías por Materia



3.3. DOCENTE

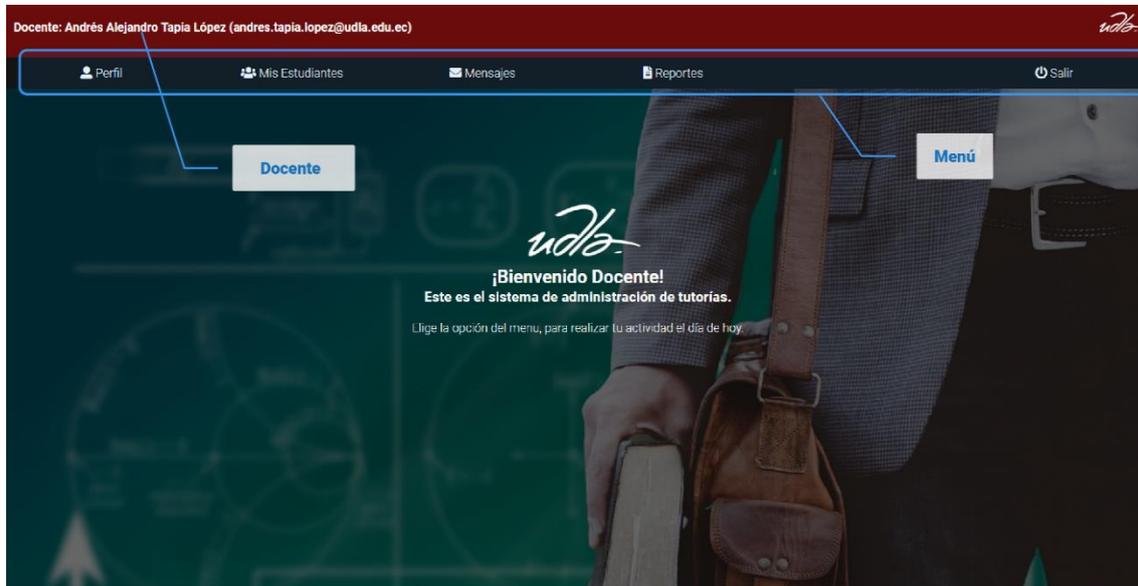
Ingrese las credenciales del docente ingresado en el sistema.



3.3.1. Menú Docente

Al ingresar a la página principal del usuario docente, nos encontramos con un mensaje de bienvenida al docente actual.

En la parte superior de la página podemos observar un panel o menú.



Este panel cuenta con los siguientes módulos:

- ✓ Perfil
- ✓ Mis Estudiantes
- ✓ Mensajes
- ✓ Reportes

Adicional cuenta su respectivo botón salir para cerrar la sesión actual.

Salir

3.3.2. Módulos

Cada módulo permite realizar las diferentes actividades del docente, para lo cual vamos a conocer el funcionamiento de cada uno de los mismos.

3.3.2.1. Perfil

Esta sección cuenta con los datos personales del docente, así como el ingreso de horas obligatorias o disponibles para realizar tutorías a los estudiantes.

Datos Personales

ANDRÉS ALEJANDRO	TAPIA LOPEZ
Nombre	Apellidos
andres.tapia.lopez@udla.edu.ec	087896587
Email	Celular
1720765533	
Cédula	

Datos Docente

Ingresar/Eliminar Horas libres

Período	Horas Disponibles		Horas Disponibles	
SEPTIEMBRE 2018	104 Lunes 10:15 - 11:15	Ingresar	102 Lunes 8:05 - 9:05	Eliminar

Ingresar/Eliminar Horas Obligatorias

Período	Horas Disponibles		Horas Disponibles	
SEPTIEMBRE 2018	104 Lunes 10:15 - 11:15	Ingresar	105 Lunes 11:20 - 12:20	Eliminar

Ingrear hora libres u obligatorias

Hora	Lunes	Martes	Miercoles	Jueves	Viernes	Sabado
07:00 - 08:00	FIS100-1/200	SIS1-1/457		EST78-1/457		
08:05 - 09:05	LIBRE					
09:10 - 10:10	LIBRE					
10:15 - 11:15						
11:20 - 12:20	TUT.ESTU.					
12:25 - 13:25	TUT.ESTU.					
ALMUERZO						
13:30 - 14:30		LIBRE				
14:35 - 15:35						
15:40 - 16:40						
16:45 - 17:45	BDD-1/415	LIBRE				
17:50 - 18:50						
18:50 - 19:50						
19:50 - 20:50						
20:50 - 21:50	TUT.ESTU.		LIBRE			

Horario del docente

Nota: Es necesario ingresar las horas, para continuar con el proceso de tutorías.

3.3.2.2. Mis Estudiantes

En esta sección el docente puede:

- ✓ Observar sus estudiantes actuales por materia de su horario.
- ✓ Conocer el número de matrícula en la cual se encuentra el estudiante en cada una de las materias.
- ✓ Asignar una tutoría a los estudiantes.
- ✓ Conocer sus tutorías pendientes.
- ✓ Controlar la asistencia de los estudiantes a las tutorías.

Estudiantes actuales

Conoce tus estudiantes actuales, revisa el número de matrícula que contienen en cada una de tus materias y asigna tutorías necesarias para los mismos.

Materias

Selecciona una materia, analiza los estudiantes que contienen un número de matrícula superior a uno (color rojo).

Filtro por Materia del Docente

ESTADÍSTICA

Número de Matrícula de los estudiantes pertenecientes a la materia seleccionada.

Matrícula Estudiante	Datos	Cod Materia	Materia	Repetencia
501735	JOAN SEBASTIAN ANDRADE CASCO joan.andrade@udla.edu.ec 0978616500	EST78	ESTADÍSTICA	3
789654	MARIA CAZA maria@udla.edu.ec 0854785698	EST78	ESTADÍSTICA	3
14789	PEDRO NAVAJA pedro@udla.edu.ec 0747854785	EST78	ESTADÍSTICA	3

Asignar Tutoría

Selecciona un estudiante de la materia que impartes para establecer una tutoría con el mismo, selecciona un paralelo una aula y una fecha para que el estudiante pueda asistir.

Asignar Tutoría al estudiante

Estudiante
JOAN SEBASTIAN ANDRADE CASCO

Materia
BASE DE DATOS I

Período
SEPTIEMBRE 2018

Módulo
102 Lunes 8:05 - 9:05

Obligatorio
SI

Aula/Oficina
Aula

Fecha
05/02/2019

Ingresar

Estudiantes actuales

Para confirmar la asistencia, una vez pasada la fecha de la tutoría, la tutoría pasará a la sección de tutorías pendientes.

Tutorías Pendientes

Cod Materia	Materia	Módulo	Período	Estado Obligatorio	Fecha	Aula	Asistencia	
SIS1	INTRODUCCIÓN A LA PROGRAMACIÓN	208 Martes 13:30-14:30	PEP2018	SI	12/02/2019	Oficina	SI	Confirmar Asistencia
FIS100	FISICA I	105 Lunes 11:20-12:20	PEP2018	SI	18/02/2019	356	SI	Confirmar Asistencia

Confirmar Asistencia del Estudiante a la tutoría

3.3.2.3. Mis Estudiantes

En esta sección el docente puede:

- ✓ Observar las solicitudes de tutorías por parte de los estudiantes
- ✓ Confirmar la tutoría en la fecha establecida o reasignar una nueva fecha con el lugar de encuentro (aula/oficina)

Mensajes

=>Revisa las nuevas solicitudes de estudiantes que necesitan tu ayuda.
=>Activa la tutoría que creas que estarás disponible, así el estudiante pueda observar en su cronograma, y asistir.
=>Si no dispones de tiempo a la solicitud enviada, envía un mensaje al estudiante respondiendo a su correo.

Buzon de Mensajes

Nombre	Mensaje	Confirmado
JOAN SEBASTIAN ANDRADE CASCO	Hola	NO

Mensajes

¿Deseas confirmar la tutoría solicitada?, selecciona una fecha y presiona en confirmar, o cancelar.

Materia BASE DE DATOS I	Período PEP2018
Código Materia BDD	Módulo 102
Estudiante JOAN SEBASTIAN ANDRADE CASCO	Hora 8:05 - 9:05
Matrícula 501735	Aula/Oficina <input type="text" value="Lugar"/>
Código Período SEPTIEMBRE 2018	Fecha <input type="text" value="30/01/2019"/>

Confirmar la solicitud del estudiante

3.3.2.4. Reportes

En esta sección el docente puede:

- ✓ Filtrar las tutorías asistidas o no asistidas por periodo académico
- ✓ Observar todas las tutorías realizadas

Filtro de Reportes

Encuentra todos los reportes de tutorías que has tenido en los diferentes períodos o por asistencia.

Período Académico Asistencia

SEPTIEMBRE 2018 NO

Todos

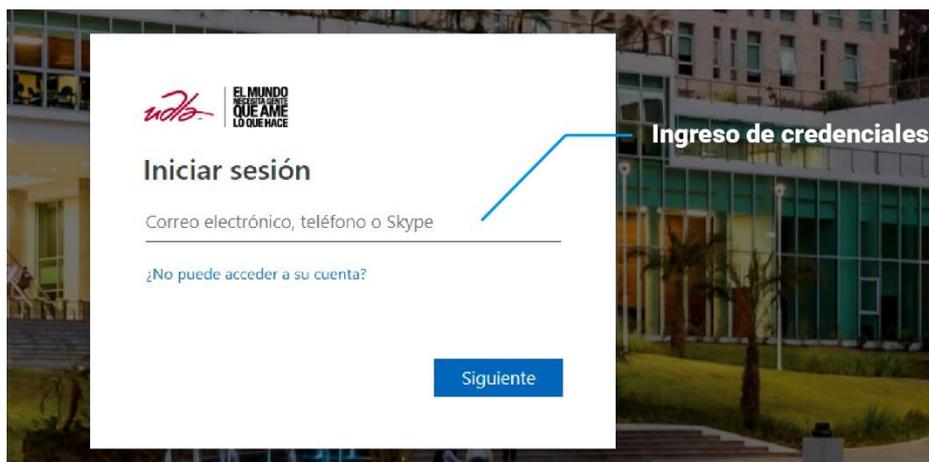
Filtro de Reportes

Tabla de tutorías general

Matrícula	Estudiante	Cod Materia	Materia	Módulo	Período	Estado Obligatorio	Fecha	Aula	Asistencia	
501735	JOAN SEBASTIAN ANDRADE CASCO	BDD	BASE DE DATOS I	102	Lunes 8:05-9:05	PEP2018	SI	04/02/2019	Oficina	NO
501735	JOAN SEBASTIAN ANDRADE CASCO	BDD	BASE DE DATOS I	315	Miercoles 20:50-21:50	PEP2018	SI	29/01/2019	Oficina	NO

3.4. ESTUDIANTE

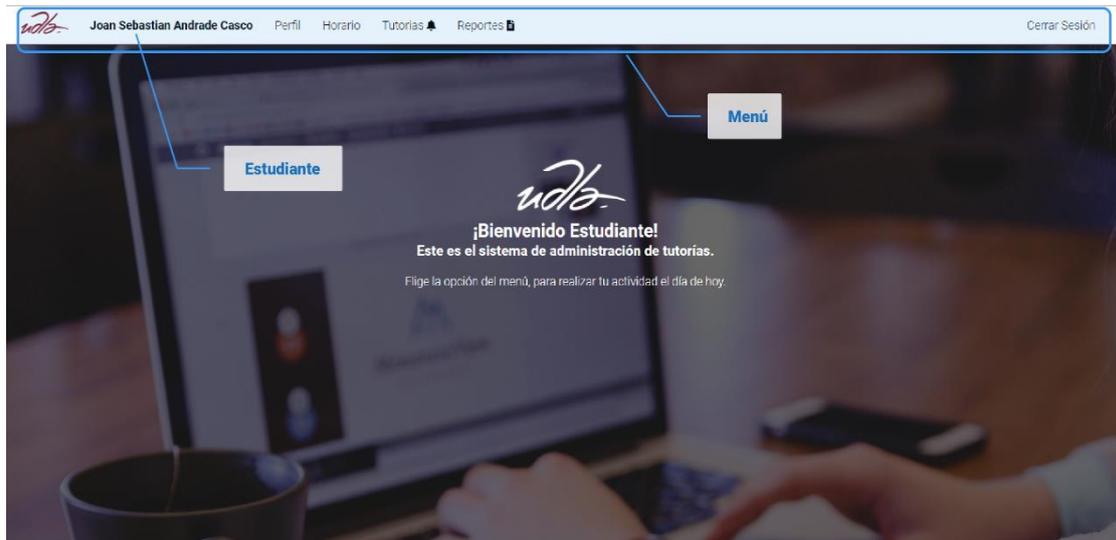
Ingrese las credenciales del estudiante ingresado en el sistema.



3.4.1. Menú Estudiante

Al ingresar a la página principal del usuario estudiante, nos encontramos con un mensaje de bienvenida al estudiante actual.

En la parte superior de la página podemos observar un panel o menú.



Este panel cuenta con los siguientes módulos:

- ✓ Perfil
- ✓ Horario
- ✓ Tutorías
- ✓ Reportes

Adicional cuenta su respectivo botón salir para cerrar la sesión actual.

Cerrar Sesión

3.4.2. Módulos

Cada módulo permite realizar las diferentes actividades del estudiante, para lo cual vamos a conocer el funcionamiento de cada uno de los mismos.

3.4.2.1. Perfil

Esta sección cuenta con los datos personales del estudiante.

Mi Perfil

Hechale un vistazo a tu información personal

Nombres
JOAN SEBASTIAN

Apellidos
ANDRADE CASCO

Email
joan.andrade@udla.edu.ec

Celular
0978616500

Matrícula
501735

Carrera
ING SISTEMAS

3.4.2.2. Horario

Esta sección contiene el horario del periodo actual del estudiante.

Horario de Clases

Conoce las materias y horas de clases. ¡No te descuides de tu horario!

Hora	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
07:00 - 08:00	FIS100-1/200	SIS1-1/457		EST78-1/457		
08:05 - 09:05						
09:10 - 10:10						
10:15 - 11:15						
11:20 - 12:20						
12:25 - 13:25						
ALMUERZO						
13:30 - 14:30						
14:35 - 15:35						
15:40 - 16:40						
16:45 - 17:45	BDD-1/415	BDD-1/304				
17:50 - 18:50						
18:50 - 19:50						
19:50 - 20:50						
20:50 - 21:50						

3.4.2.3. Tutorías

En esta sección el estudiante puede:

- ✓ Conocer las tutorías pendientes que tiene que asistir
- ✓ Solicitar una tutoría a cualquiera docente que de las materias de su horario actual

Tabla de Tutorías actuales del estudiante

Tutorías
No te descuides de tus tutorías, revisa el cronograma constantemente.

[Tutorías Actuales](#)

Cod Materia	Materia	Docente	Módulo	Período	Estado Obligatorio	Fecha	Aula	Asistencia
DDD	BASC DE DATOS I	ANDRES ALEJANDRO TAPIA LOPEZ	102 Lunes 8:05-9:05	PLP2018	SI	11/02/2019	oficina	NO

Solicitud de Tutoría

SOLICITUD DE TUTORÍA

Llena la información solicitada

Elige una materia
 BASE DE DATOS I

Elige al docente de tu carrera que tengas mas confianza
 Deseas conocer un poco mas de tu docente, selecciona su nombre.

ANDRES TAPIA ✓ ANITA YANEZ

Horario de **ANDRES TAPIA**

Hora	Lunes	Martes	Miercoles
07:00 - 08:00	FIS100-1/200	SIS1-1/457	
08:05 - 09:05	LIBRE		
09:10 - 10:10	LIBRE		
10:15 - 11:15			
11:20 - 12:20	TUT.ESTU.		
12:25 - 13:25	TUT.ESTU.		
ALMUERZO			
13:30 - 14:30		LIBRE	
14:35 - 15:35			
15:40 - 16:40			
16:45 - 17:45	BDD-1/415	LIBRE	
17:50 - 18:50			
18:50 - 19:50			
19:50 - 20:50			
20:50 - 21:50	TUT.ESTU.		LIBRE

Período Académico
 SEPTIEMBRE 2018

Selecciona un módulo, estos módulos estan disponibles para el Docente
 102 Lunes: 8:05 9:05

Fecha de la tutoría
 05/02/2019

Escribe un mensaje explicando el motivo de la tutoría con el tema que tienes problemas.

Enviar

Filtro Materia

Docentes que imparten la materia

Horario del docente seleccionado

Periodo Académico

Módulo de Clases

Fecha de Tutoría

Mensaje

Una vez llenada la solicitud y enviada, el estudiante debe estar al pendiente de la confirmación o respuesta del docente.

