



Universidad Central de Venezuela

Facultad de Ciencias

Escuela de Computación

**ANÁLISIS DE LOS REQUERIMIENTOS PARA EL DESARROLLO DE UN SISTEMA
WEB PARA LA GESTIÓN DE CONTROL DE ESTUDIOS DE LA FACULTAD DE
ARQUITECTURA Y URBANISMO DE LA UNIVERSIDAD CENTRAL DE
VENEZUELA – MODULO DE INSCRIPCIONES.**

**Seminario presentado ante la ilustre
Universidad Central de Venezuela**

Por los Bachilleres:

Jesús Enrique Mambie Machado

José Valentin Salina Peña

Con la asesoría del tutor:

Profa. Yosly Hernández

Profa. Filia Suárez (Facultad de Arquitectura)

Caracas, abril de 2018

RESUMEN

La tecnología ha tenido un auge y un impacto muy grande en los avances de hoy en día. Cada vez son más los problemas que se resuelven con sistemas que automatizan y mejoran gran cantidad de procesos en todas las áreas alrededor del mundo. Los Sistemas Web son uno de esos avances y permiten que, con tan solo contar un computador o un teléfono inteligente con acceso a la red de internet, se puede hacer uso de todas sus ventajas y las facilidades que ofrecen.

En la actualidad, la Coordinación Docente de la Escuela de Arquitectura Carlos Raúl Villanueva (EACRV) de la Facultad de Arquitectura y Urbanismo (FAU) de la Universidad Central de Venezuela (UCV), ha tenido que llevar la mayoría de sus procesos académicos de manera manual, guardando toda la información en almacenamiento físico, en especial el proceso de inscripciones. Motivo por el cual el presente trabajo de investigación se basa principalmente en la realización de un análisis de los requerimientos para el desarrollo de una plataforma que automatice y sistematice el mencionado proceso que servirá para la optimización y precisión en la gestión de los procesos académicos de los estudiantes.

Para el desarrollo de este sistema se propone el uso del framework Scrum de manera que se puedan utilizar sus herramientas para el desarrollo de las distintas fases del proyecto por Sprint y la identificación de las historias de usuario dentro del Product Backlog. En cuanto a las tecnologías se propone una arquitectura Cliente-Servidor, tomando para el desarrollo del cliente los principales lenguajes para el Desarrollo Web: HTML5, CSS3 y JavaScript, utilizando a la vez un framework del lado cliente llamado Angular en su versión 7. Para el desarrollo de la lógica del sistema en el lado del servidor se utilizará el entorno de ejecución NodeJS debido a las bondades del uso de JavaScript como lenguaje de programación tanto en el servidor como en el cliente, a su vez utilizando framework de desarrollo a Express.js y como manejador de Bases de Datos a PostgreSQL.

Palabras Claves: Sistema Web, inscripción, JavaScript, Scrum.

ÍNDICE GENERAL

RESUMEN	i
INTRODUCCIÓN	1
CAPÍTULO I: PROPUESTA DE TRABAJO ESPECIAL DE GRADO	3
1.1 Contexto de la investigación	3
1.2 Descripción del proceso de inscripción.....	4
1.3 Planteamiento del problema	5
1.4 Justificación de la investigación.....	8
1.5 Objetivo general	9
1.6 Objetivos específicos.....	9
1.7 Alcance	10
1.8 Antecedentes	12
1.9 Propuesta de solución.....	15
1.10 Método de desarrollo.....	19
1.11 Plan de trabajo	21
CAPÍTULO II: SISTEMAS WEB.....	23
2.1 Definición de Sistema Web.....	23
2.2 Características	23
2.3 Arquitectura.....	24
2.4 Ejemplos.....	27
2.5 Importancia.....	30
CAPITULO III: TECNOLOGÍAS DE DESARROLLO	32
3.1 Concepto.....	32
3.2 Tecnologías de desarrollo.....	32
3.2.1. Tecnologías del Lado Cliente	32
3.2.2. Tecnologías del Lado del Servidor	35
3.3 Bases de Datos	37
3.4 Sistema de control de versiones	38
3.5 Editores de código	40
CONSIDERACIONES FINALES.....	42
REFERENCIAS.....	43

ANEXO.....	46
------------	----

ÍNDICE DE FIGURAS

FIGURA 1. DIAGRAMA DE FLUJO DEL PROCESO ACTUAL DE INSCRIPCIONES	4
FIGURA 2. DIAGRAMA DE CASOS DE USO	16
FIGURA 3. ARQUITECTURA TECNOLÓGICA DEL SISTEMA WEB	19
FIGURA 4. MARCO DE TRABAJO SCRUM	21
FIGURA 5. ARQUITECTURA CLIENTE – SERVIDOR	25
FIGURA 6. MODELO DE SERVICIOS	27
FIGURA 7. INTERFAZ DE USUARIO FINAL DE CONTRACT ROOM.....	28
FIGURA 8. INTERFAZ DE SUSCRIPCIÓN A CLASE DE TABATADESK	29
FIGURA 9. INTERFAZ DE COMPRA DE ARTÍCULO DE AMAZON.....	30
FIGURA 10. MODELO DE DATOS UTILIZADO POR GIT	39

ÍNDICE DE TABLAS

TABLA 1. CUADRO DE ESPECIFICACIONES DE CASOS DE USO	16
TABLA 2. PLAN DE TRABAJO	22
TABLA 3. ESPECIFICACIÓN - UC1 INICIO DE SESIÓN.....	46
TABLA 4. ESPECIFICACIÓN - UC2. INSCRIBIR SEMESTRE.....	47
TABLA 5. ESPECIFICACIÓN - UC3 CONSULTAR HORARIO.....	49
TABLA 6. ESPECIFICACIÓN – UC4 RETIRAR MATERIA.....	50
TABLA 7. ESPECIFICACIÓN – UC5 CONSULTAR LISTADO DE ESTUDIANTES.....	52
TABLA 8. ESPECIFICACIÓN – UC6 ADMINISTRAR DOCENTES	53
TABLA 9 ESPECIFICACIÓN – UC7 CARGAR NUEVOS ESTUDIANTES	54
TABLA 10. ESPECIFICACIÓN – UC8 CARGAR OFERTA DOCENTE.	56
TABLA 11 ESPECIFICACIÓN – UC9. ASIGNAR MATERIAS A DOCENTES.	58

INTRODUCCIÓN

Hoy en día, la tecnología ha tenido un auge y un impacto muy grande por los avances que se vienen haciendo. Cada día son más los problemas que se resuelven con sistemas que automatizan y no solo facilitan, sino que mejoran gran cantidad de procesos en todas las áreas y alrededor de todo el mundo.

Una de las soluciones más importantes que se han venido creando a lo largo de los últimos años son los Sistemas Web, los cuales, con tan solo contar un computador o un teléfono inteligente con acceso a la red de internet, se puede hacer uso de todas sus ventajas y las facilidades que ofrecen.

En la actualidad, la Coordinación Docente de la Escuela de Arquitectura Carlos Raúl Villanueva de la Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela, ha tenido que llevar la mayoría de sus procesos académicos de manera manual, guardando toda la información en almacenamiento físico. Dicha gestión, hace que la realización de las actividades sea muy compleja, lenta, y que haya poca confiabilidad en los datos.

Entre los diversos procesos de gestión académica que se realizan actualmente en la Facultad, el principalmente afectado, es el de inscripción de los estudiantes, en la cual es de suma importancia la información histórica de las notas de las asignaturas inscritas a lo largo de su carrera. La decisión de permitir la inscripción por parte de los estudiantes, ya sea de una determinada asignatura o de un semestre en general, depende en gran medida de dichos datos históricos, por lo que se separa en dos vertientes que hay que verificar.

El primero de los puntos importantes son las prelación. Las carreras poseen un sistema de prelación entre asignaturas, el cual consiste en que algunas materias dependen de otras para seguir avanzando. Existen materias básicas, cuya aprobación es indispensable para la inscripción de las correspondientes asignaturas posteriores, así como el orden en que debe ir establecido mediante un nivel de prioridad en el que se deben obtener los conocimientos a lo largo de la carrera.

A su vez, el otro aspecto importante que hay que tomar en cuenta con respecto a la información histórica de los estudiantes, es que la facultad, y la universidad en general, posee

distintas leyes y normas que rigen el proceso de inscripción tomando en cuenta el desempeño de los estudiantes en el semestre inmediato transcurrido. Dichas normas se establecen en distintos niveles, Ley de Universidades, Normas de rendimiento mínimo, Condiciones de Permanencia, por parte de la Universidad Central de Venezuela y normativas internas dentro de cada Facultad y Escuela en particular.

Es por estos motivos, que se planteó como objetivo General del Trabajo de Seminario: Analizar los requerimientos para el desarrollo del módulo de inscripciones del Sistema Web para la gestión de control de estudios de la Facultad de Arquitectura y Urbanismo (FAU) de la Universidad Central de Venezuela (UCV), que permita automatizar el proceso de inscripciones, buscando así, mejorar los tiempos de respuesta y ejecución del mismo. Entre los objetivos específicos del Seminario se mencionan:

1. Analizar los requerimientos del proceso de inscripción de la Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela.
2. Describir la arquitectura del sistema web acorde con los requerimientos del proceso inscripción de la Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela.
3. Examinar las tecnologías de desarrollo para el sistema web propuesto para el módulo de inscripción de la Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela.

El presente documento se encuentra estructurado por tres capítulos. El primero de ellos está relacionado con la propuesta de Trabajo Especial de Grado a realizar, donde se presenta el contexto y el problema con la situación actual de la EACRV-FAU-UCV, y de esta manera justificar la realización del proyecto, así como proponer el método de desarrollo y plan de trabajo que se seguirá para la satisfactoria realización de las actividades. El segundo capítulo consta de la definición y explicación de las aplicaciones web, la cual es el tipo de herramienta que se desarrollará como solución del problema. El tercer y último capítulo define las tecnologías específicas con las que se procederá a implementar la solución planteada, seleccionadas por los estudiantes en base a las características y requerimientos del sistema.

CAPÍTULO I

PROPUESTA DE TRABAJO ESPECIAL DE GRADO

En este capítulo se presenta la propuesta para el Trabajo Especial de Grado, en el que se muestra la problemática actual del proceso de inscripciones dentro de la Escuela de Arquitectura Carlos Raúl Villanueva de la Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela (EACRV-FAU-UCV), y se describen los objetivos y la propuesta de solución para la implementación de una solución en forma de un sistema web para optimizar los procesos de la Coordinación Docente. Adicionalmente, se identifica el plan a trabajo y la metodología a seguir para su desarrollo.

1.1 Contexto de la investigación

La presente investigación se desarrolla en la EACRV, que fue fundada en el año 1941, siendo el primer centro de enseñanza de arquitectura en Venezuela. En 1944 inicia sus labores docentes; en ese momento no existía la FAU, por lo que la EACRV era un Departamento de la Facultad de Ingeniería. Luego, entre 1946 y 1953 estuvo adscrita a la Facultad de Ciencias Físicas y Matemáticas, periodo durante el cual egresó su primera promoción en 1948. El 20 de octubre de 1953, por Resolución Electoral, se elevó la Escuela de Arquitectura y se fundó la FAU. En el 2000 obtuvo el nombre con el cual se le conoce actualmente en honor a Carlos Raúl Villanueva, arquitecto del siglo XX venezolano, cuya obra más importante es la Ciudad Universitaria de Caracas, Patrimonio Mundial de la Humanidad (FAU-UCV, 2015).

En la actualidad, la FAU cuenta con aproximadamente 1300 estudiantes y 250 docentes. En cuanto a su estructura, la Asamblea de la Facultad representa la máxima autoridad en la toma de decisiones. Le sigue, en orden jerárquico, el Consejo de Facultad, que es presidido por el Decano, quien se apoya en la Coordinación Académica y en la Coordinación Administrativa para llevar adelante las directrices fundamentales de la gestión. En las áreas de docencia, investigación y extensión, la Facultad cuenta con las siguientes unidades académicas: la EACRV, el Instituto de Urbanismo, el Instituto de Desarrollo Experimental de la Construcción, la Dirección de Estudios de Postgrado, y otros centros con proyectos y programas específicos (FAU-UCV, 2015).

1.2 Descripción del proceso de inscripción

El proceso de inscripción de la EACRV es el punto de partida de todos los semestres en la FAU UCV, ya que es obligatorio para todos los estudiantes, debido a que, en éste, cada uno se encarga de seleccionar las asignaturas que cursará durante el siguiente periodo académico.

Anteriormente, el sistema utilizado para la gestión académica de los estudiantes, llamado UXXI, era utilizado por la mayoría de las unidades de control de estudios de las facultades de la UCV (exceptuando las Facultad de Ciencias y la Facultad de Ingeniería), pero dejó de prestar servicio. A raíz de esto se generó una profunda crisis en la población universitaria, ya que no se pudo seguir utilizando para los procesos que realizaba, como, por ejemplo, el proceso de inscripciones de un nuevo período académico.

En consecuencia, las unidades de control de estudios afectadas tuvieron que empezar a tomar medidas de emergencia para afrontar dicha crisis y tratar de seguir prestando los servicios a los estudiantes de alguna manera. Específicamente la EACRV continuó realizando el proceso de inscripción de los estudiantes en los semestres I-2017, I-2018 y II-2018 de manera manual, al igual que la administración de las inscripciones. En la figura 1 se describe cómo se realiza actualmente el proceso.

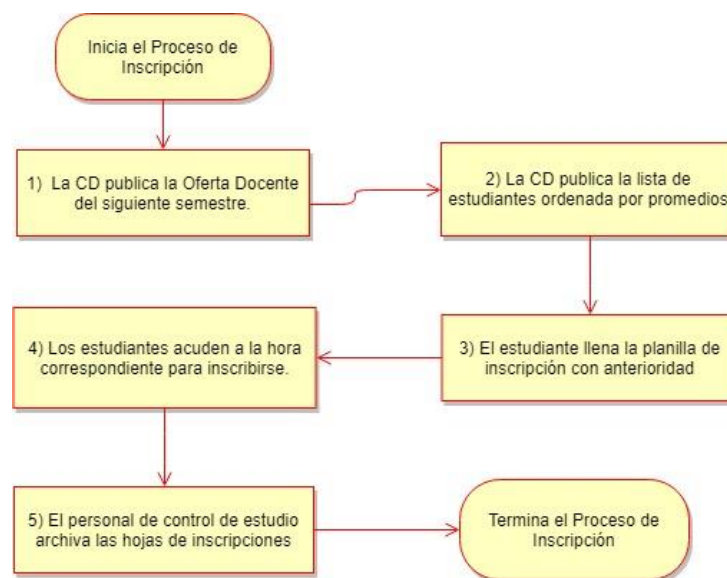


Figura 1. Diagrama de flujo del Proceso Actual de Inscripciones. Elaboración Propia, 2019.

Del cual podemos destacar lo siguiente:

1. La Coordinación Docente publica en la página de la FAU la Oferta Docente del siguiente semestre, que consta de las materias a ofertar junto a los docentes, horarios y las secciones de cada una. Posteriormente esta información es divulgada por las redes sociales y publicada en la cartelera de la oficina de control de estudios.
2. Una semana antes de la inscripción, se publica la lista de estudiantes ordenada de acuerdo al promedio obtenido por éstos en el semestre inmediatamente anterior.
3. Cada estudiante luego de revisar la oferta, hace su horario de manera manual, intentando todas las posibles combinaciones con las materias disponibles y que puedan inscribir según su historial académico. Existe un formato de planilla de inscripción que el estudiante debe descargar de la página web de la FAU, imprimir y luego plasmar las asignaturas, horarios y secciones seleccionadas.
4. El día de la inscripción los estudiantes acuden según les corresponda por su cita horaria, el personal de control de estudios junto a algunos voluntarios son los encargados de realizar las inscripciones, manejando la disponibilidad de materias en una computadora de escritorio con conexión a internet, ya que utilizan las hojas de cálculo de Google, mientras que las inscripciones de los estudiantes se van anotando en físico y se almacenan las hojas del formato de inscripción. Cabe destacar que este paso es sumamente tedioso tanto para los que inscriben, por la presión de los estudiantes, como para los mismos estudiantes en caso de que el proceso se retrase por factores externos (falla en conexión de internet, demora de personal, entre otros). Este paso puede llegar a tomar entre 2 a 3 días si todo sale con normalidad.
5. Finalmente, cuando se terminan todas las inscripciones, el personal de control de estudios archiva las hojas de inscripciones debidamente identificadas por semestres.

1.3 Planteamiento del problema

La Universidad Central de Venezuela (UCV) disponía de un software llamado UXXI, que permitía manejar los procesos académicos que incluía la inscripción de los estudiantes, la

generación de los kárdex, constancias de estudios, entre otros, de manera sistematizada. Sin embargo, según el secretario de la UCV “la Secretaría de la UCV se vio obligada a no renovar las licencias del sistema, debido a la falta de presupuesto para realizar dicha actualización, la cual en diciembre del año 2015 estaba valorada en 600.000 dólares” (Belmonte, 2016).

A raíz de esto, por la falta de mantenimiento, el estado en el que se encontraba UXXI para el año 2016 era totalmente obsoleto, pero sumado a esto, y “ante algunas denuncias en contra del acceso directo al Kardex con el número de identidad del usuario, la Secretaría tomó medidas para generar mayor privacidad, obligando a los usuarios a crear un registro con una cuenta y una contraseña para acceder al récord académico” (Belmonte, 2016). Esta medida, lamentablemente, no fue soportada por esa versión del software, generando un colapso del sistema que impidió que la Secretaría pudiera seguir prestando los servicios anteriormente mencionados a los estudiantes de las distintas facultades de la universidad.

Aun así, la universidad pudo generar un software propio, llamado CONEST, el cual representa una alternativa ante la problemática. Sin embargo, para hacer efectivo su uso en toda la universidad, la UCV debe invertir una alta suma de dinero en: equipos, instalaciones, personal y conectividad, lo que también hace inviable su implementación inmediata, pues es conocida la grave crisis presupuestaria por la cual atraviesa la UCV (Belmonte, 2016).

En consecuencia, como no se tiene una solución a corto plazo para el problema principal ocasionado con el cese de prestación de servicios del sistema UXXI, la EACRV optó por hacer el proceso de las inscripciones de manera manual con el apoyo de herramientas ofimáticas, pero para la gestión de la información de cada uno de los estudiantes se dificulta al realizar las validaciones correspondientes antes de inscribir alguna materia, ya que es de suma importancia tomar en cuenta su información histórica para poder cumplir con el sistema de prelación entre materias o con el cumplimiento de las leyes y normas de la UCV y de la FAU, entre las que destacan los siguientes artículos de la Ley de Universidades:

Artículo 156.- Los alumnos que resulten aplazados por primera vez en exámenes de reparación en no más de una asignatura pueden inscribirse condicionalmente en todas las del curso inmediato superior y podrán presentar exámenes finales de una y otras en el período ordinario de exámenes. Si la asignatura pendiente tiene prelación sobre alguna o

algunas del curso superior no podrán rendirse los exámenes de éstas sin haber aprobado previamente aquella. El Consejo de cada Facultad determinará el orden de prelación de las asignaturas.

Artículo 25°.- Cuando el aspirante favorecido con la equivalencia le hayan sido fijadas asignaturas que corresponden a diferentes cursos, podrá presentar los exámenes en las fechas señaladas conforme al Reglamento de Exámenes. El aspirante queda sujeto al régimen de prelaciones establecido para cada Facultad, y, en consecuencia, se considerará como no aprobada la asignatura de un curso superior cuando el aspirante fuere aplazado en una o más asignaturas de un curso inferior, si tales asignaturas fueren prelativas a las del curso inferior.

Así como también, es importante en el proceso de inscripción considerar las Normas sobre el rendimiento mínimo y condiciones de permanencia de los alumnos en la U.C.V, en torno a los estudiantes que no estén incursos en los siguientes artículos:

Artículo 1°. Todo alumno de la Universidad Central de Venezuela deberá lograr un rendimiento académico no inferior a los límites mínimos establecidos en las presentes normas como condición para permanecer con tal carácter en la Universidad.

Artículo 3°. Todo alumno que en un período no apruebe el 25% de la carga académica que curse o que, en todo caso no apruebe por lo menos una asignatura, deberá participar obligatoriamente en el procedimiento especial de recuperación establecido en estas normas.

Artículo 6°. El alumno que al final del semestre de recuperación no alcance nuevamente a aprobar el 25% de la carga académica que cursa o en todo caso a aprobar por lo menos una asignatura, no podrá reinscribirse en la Universidad Central de Venezuela, en los dos semestres siguientes. Pasados éstos, tendrá el derecho de reincorporarse en la Escuela en la que cursaba sin que puedan exigir otros requisitos que los trámites administrativos usuales. Igualmente, podrá inscribirse en otra Escuela diferente con el Informe favorable del Profesor Consejero y de la Unidad de Asesoramiento Académico de la Escuela a la

cual pertenecía, y la aprobación por parte del Consejo de Facultad a la cual solicita el traslado.

Artículo 7°. El alumno que, habiéndose reincorporado conforme al artículo anterior, dejare nuevamente de aprobar el 25% de la carga que curse, o en todo caso, el que no apruebe ninguna asignatura durante dos períodos consecutivos, no podrá incorporarse más a la misma Escuela o Facultad, a menos que el Consejo de Facultad, previo estudio del caso, autorice su reincorporación.

Artículo 8°. El Consejo de Facultad podrá autorizar, en casos excepcionales que a su juicio lo ameriten, la reinscripción inmediata del alumno que se encuentre en el supuesto previsto en el artículo 6° de estas normas, previo informe favorable y razonado del respectivo Profesor Consejero y a recomendación del Consejo de la Escuela si lo hubiera.

Artículo 9°. Las Facultades y Escuelas que lo estimen conveniente podrán organizar, con la autorización del Consejo Universitario, exámenes especiales de recuperación, cuya aprobación permita la reincorporación inmediata de quienes se hallen dentro del plazo de separación de la Universidad por aplicación de las presentes normas.

Debido a la problemática planteada, no existe una forma expedita de poder velar por el cumplimiento de las mencionadas normas, motivado a que todo el proceso se realiza de forma manual con uso de algunas herramientas ofimáticas, y el respaldo de las inscripciones se tiene de forma física en las respectivas planillas que consignan. Lo que trae como consecuencia que algunos estudiantes no cumplan con las sanciones respectivas.

Con base a lo anterior se plantean como pregunta de investigación:

¿Qué mecanismo permite sistematizar y automatizar el proceso de inscripción de la Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela?

1.4 Justificación de la investigación

En vista de las problemáticas mencionadas anteriormente es necesario llevar a cabo un proceso de análisis y diseño de un Sistema Web que permita la sistematización y automatización

de dichas actividades de gestión académica, destacando el proceso de inscripción. De esta manera, se busca acelerar los tiempos de culminación de cada proceso, y así evitar demoras, y en el peor de los casos, producir una suspensión forzada de los servicios prestados, lo cual afectaría gravemente a todos los miembros de la comunidad, ya sea estudiante, docente, o personal administrativo.

El empleo de esta solución les permitirá a los miembros de la EACRV optimizar el tiempo del proceso de inscripciones, desde el momento en que se carga la Oferta Docente al Sistema Web todos los usuarios podrán ver con facilidad desde cualquier lugar dentro o fuera de la universidad las materias ofertadas en cualquier momento con su correspondiente horario y profesor.

Una vez llegue el día de las inscripciones, los estudiantes no tendrán la necesidad de acudir a las instalaciones de la EACRV en el horario de su cita horaria, ya que por tratarse de una solución Web estos podrán acceder desde cualquier lugar siempre y cuando cuenten con una conexión a internet, evitando así problemas que puedan surgir con el transporte. En caso de que el estudiante quiera igualmente acudir a la sede de la EACRV para inscribirse, allí también podrá acceder realizar su correspondiente inscripción, pero en este caso, la carga de estudiantes será menor para el personal administrativo y mucho más rápida al contar con el sistema.

Además, gracias al proceso de autenticación de un usuario dentro del sistema se obtendrá una validación confiable de los datos involucrados, lo que permitirá velar por el cumplimiento de las distintas leyes mencionadas en el apartado anterior, mantener el orden de inscripciones según la cita horaria generada, entre otros beneficios.

1.5 Objetivo general

Desarrollar el Módulo de Inscripciones del Sistema Web para la Gestión de Control de Estudios de la Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela.

1.6 Objetivos específicos

1. Analizar los requerimientos de los procesos de inscripciones de la Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela.

2. Seleccionar la Arquitectura del Sistema Web a desarrollar que cumpla con los requerimientos previamente analizados.
3. Determinar las tecnologías de desarrollo que mejor se adapten a los requerimientos y necesidades del Sistema Web a desarrollar.
4. Diseñar interfaces de usuarios, accesibles y usables para el fácil e intuitivo uso por parte de los diferentes tipos de usuarios involucrados con la aplicación.
5. Modelar el esquema lógico y físico de la base de datos donde será alojada toda la información solicitada por el servidor el correcto funcionamiento del sistema.
6. Aplicar pruebas de aceptación, usabilidad y funcionalidad a la aplicación con el fin de la verificación de su calidad.
7. Desplegar el Sistema Web en un servidor de producción perteneciente a la Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela.

1.7 Alcance

Se desarrollará un sistema web escalable para la incorporación de futuros módulos, y de esta manera, adopte mayor robustez a través del tiempo. Se busca analizar la mejor manera de realizar el desarrollo de la aplicación, la cual tendrá como objetivo principal la automatización y sistematización del proceso de inscripciones de la Facultad, adicionalmente, se implementarán otros procesos importantes de administración y gestión académica que actualmente son llevados a cabo de manera manual, entre el que resalta la carga al sistema web de la Oferta Docente. Se tendrán cuatro distintos roles de usuario, a continuación, se describen las funcionalidades que podrá ejercer cada uno:

Estudiante : es el usuario que cursará las distintas materias ofertadas.

- a. Inscribir el semestre: el estudiante podrá inscribir las asignaturas que le corresponden en el determinado semestre que esté cursando al momento de hacerlo. El sistema proporcionará todas las validaciones pertinentes con respecto a la permisología de la

inscripción, tomando en cuenta su información histórica previamente almacenada en la base de datos.

- b. Consultar horario: en caso de estar inscrito en el transcurso del periodo lectivo el estudiante podrá visualizar su horario, junto con la descripción de las asignaturas, docentes y aulas asignadas.
- c. Retirar asignaturas: será capaz de hacer el retiro de asignaturas que haya inscrito en un determinado período lectivo que no esté en disposición de seguir cursando. Para esto, deberá estar dentro de las fechas necesarias que le permitan hacerlo, previamente establecidas por el personal administrativo de la EACRV.

Docente: es el encargado de dictar una o más materias a lo largo del semestre.

- a. Consultar listado de estudiantes: el docente contará con la funcionalidad de generar un listado con los estudiantes inscritos en cualquiera de las asignaturas que dicta en un período lectivo, que a su vez cuente con su información principal como, por ejemplo: horarios, código de la materia, coordinador y estudiantes inscritos.
- b. Ver horario: al igual que los estudiantes, los docentes podrán ver en la página principal el horario con la información de las materias que curso en el semestre en curso.

Coordinador de Área: Es el usuario encargado de supervisar a los docentes a cargo, pero al ser también un uno, podrá realizar las mismas funcionalidades que estos. Entre las funciones dentro del sistema se encuentra:

- a. Administrar docentes: tiene el trabajo de velar y supervisar el correcto desenvolvimiento de los docentes a cargo, por ende, será el encargado de administrar su información. Adicionalmente también podrá asignarles materias según considere.

Administrador: será el usuario encargado de administrar el sistema, por lo que será capaz de ejercer las siguientes funcionalidades:

- a. Cargar Oferta Docente: el administrador deberá cargar la información necesaria para el correcto desenvolvimiento de los semestres, las diferentes asignaturas ofertadas, así como los docentes encargados de dictar cada una de ellas, con sus respectivos horarios,

secciones, aulas, y máxima cantidad de alumnos. Dicha carga deberá hacerse con anticipación al inicio del proceso de inscripciones.

- b. Retirar asignaturas: el administrador será capaz de hacer el retiro de asignaturas que haya inscrito un estudiante. Para esto, deberá estar dentro de las fechas necesarias que le permitan hacerlo, previamente establecidas por el personal administrativo de la facultad.
- c. Cargar nuevos estudiantes: esta funcionalidad le permitirá a los administradores cargar a estudiantes que no estén registrados dentro del sistema, esto permitirá cargar los estudiantes por primera vez al sistema y posteriormente ingresar los nuevos ingresos en los futuros períodos.
- d. Administrar docentes: al igual que los coordinadores, los administradores podrán de administrar la información de los docentes. Adicionalmente también podrá asignarles materias para dictar.

Dada la identificación de los distintos tipos de usuarios que le darán uso a la plataforma, adicionalmente a las funcionalidades previamente explicadas, se desarrollará un módulo de autenticación en el que los usuarios deberán iniciar sesión según el rol que tengan.

1.8 Antecedentes

A continuación, se describen los antecedentes de la investigación que tienen la intención de explicar los principales sistemas automatizados para la gestión de los procesos académicos utilizados dentro de la Universidad Central de Venezuela en los últimos 10 años.

a) UXXI

UXXI por sus siglas de UNIVERSITAS XXI, es un sistema o software propietario de origen colombiano para la gestión académica que proporciona a las Universidades e Instituciones de Educación Superior una completa y eficiente automatización de sus procesos administrativos. Mejora la calidad en el tratamiento de la información sobre los estudiantes y permite realizar una gestión eficaz y coordinada entre todas las áreas y servicios universitarios (OCU, 2017).

Dispone de una serie de componentes y capacidades de parametrización que permiten a las universidades una implantación flexible y por fases, adaptándose así a sus prioridades y a sus procesos de gestión.

Entre los componentes funcionales que dispone según OCU, destacan los siguientes:

- Acceso.
- Calificaciones y firma digital.
- Historia académica.
- Inscripciones.
- Títulos.
- Planes de estudios.
- Recursos docentes y asignación de espacios y horarios.
- Gestión económica.
- Cursos cortos.
- Gestión de propuestas de programas.
- Generador de informes.
- Admisión.
- Graduaciones.
- Evaluación docente.
- Portal académico.

b) FIUCV - Control de Estudios

Es un sistema web desarrollado en PHP por la Facultad de Ingeniería de la Universidad Central de Venezuela para la gestión de los servicios estudiantiles. Permite ejecutar, controlar y supervisar los procesos de inscripción y registro de estudiantes y el proceso para otorgar títulos en las diversas carreras que ofrece la Facultad por parte del personal de control de estudios. (Facultad de Ingeniería UCV, 2017).

Entre los componentes funcionales que dispone según la página de la Facultad de Ingeniería, destacan los siguientes:

- Acceso directo al Kardex.

- Acceso a Control de Estudio.
- Programación Docente.
- Programas de Asignaturas.
- Estudiantes de Nuevo Ingreso.
- Solicitud de Correo UCV.
- Servicio Comunitario
- Graduando
- Planillas
- Calendario Académico

c) **CONEST**

Es un proyecto operativo desde el año 2007, que ha tenido como primer objetivo desarrollar un Sistema que automatice la gestión académica de la Facultad de Ciencias de la Universidad Central de Venezuela, incentivando la participación de forma activa de estudiantes, docentes y el personal administrativo que forman parte de esta comunidad. El 28 de abril de 2014, se lanzó la nueva versión evolucionada del sistema: CONEST 3.0.0. (Grupo CONEST UCV, 2014)

CONEST permite que antes de acceder a las principales funcionalidades, se pueda identificar a cada uno de los usuarios con un mecanismo de autenticación y dependiendo del rol asignado habilitar las acciones que puede realizar.

Entre las acciones más destacadas del sistema se encuentran:

- Inscripciones.
- Visualización de Kardex
- Generación de constancias
- Visualización de horario.
- Generación de Reportes.

- Programación Docente.

1.9 Propuesta de solución

En vista de la problemática causada por el manejo manual que se utiliza en la actualidad para llevar el control tanto del proceso de inscripción, como de la data histórica de las notas en semestres anteriores de los estudiantes de la EACRV FAU UCV, se decidió plantear como solución el desarrollo de un Sistema Web con el uso de las tecnologías. De esta manera se busca automatizar y sistematizar el proceso de inscripciones para facilitar en gran medida las tareas tanto de los estudiantes y docentes, como del personal administrativo.

Adicionalmente, este sistema permitirá agilizar el proceso de inscripción y disminuir en un alto porcentaje la cantidad de errores que pueden surgir durante su ejecución.

El sistema se encargará de realizar todas las validaciones pertinentes cuando un estudiante no deba inscribir alguna materia, ya sea por no haber aprobado las asignaturas que la prelen o por no cumplir alguna de las leyes o normas de la UCV mencionadas en el presente trabajo; y además brindará a estudiantes, docentes, y administradores, una interfaz amigable al momento de gestionar toda esta información, como carga y visualización de asignaturas y horarios ofertados durante un semestre, así como qué docente sería el encargado del curso en ese determinado período.

Luego de analizar el problema anteriormente señalado, en la Figura 2 se puede observar el diagrama de casos de uso propuesto para la solución, en el cual se identifican los requerimientos funcionales.

Cada una de las elipses contiene una funcionalidad del sistema identificada con la abreviatura de UC por sus siglas en inglés Use Case, traducido al español como Caso de Uso, seguido de un número secuencial que lo identifica unívocamente. Además, se pueden visualizar los distintos actores mencionados anteriormente, que tendrán acceso a las distintas funcionalidades dentro del sistema, unidos por medio de las líneas rectas a los UC que tendrá acceso el actor.

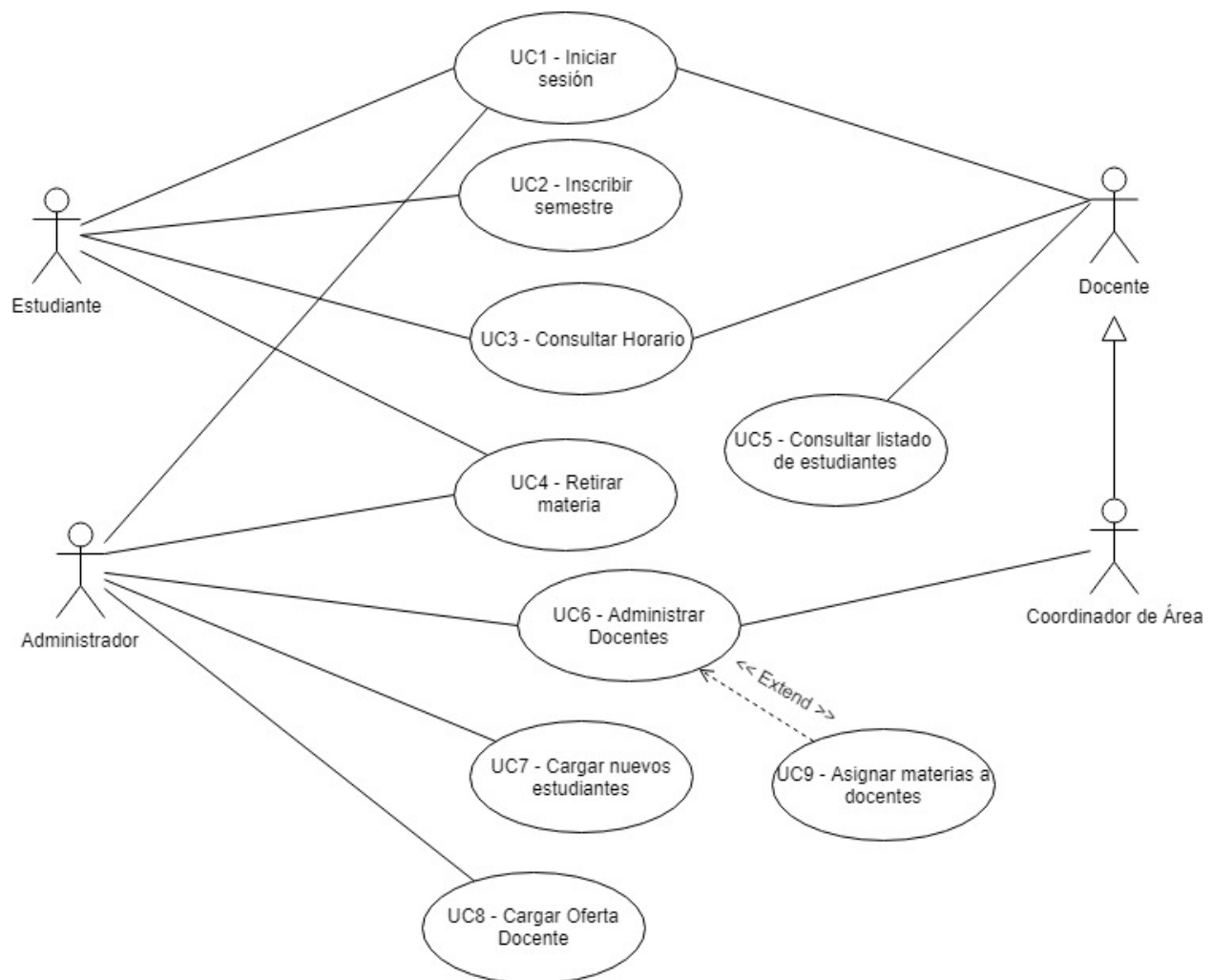


Figura 2. Diagrama de casos de uso. Fuente: (Elaboración propia, 2019).

A continuación, se puede observar la Tabla 1, en la que se muestra el formato donde se plasman las especificaciones de caso de uso, en los que se describe brevemente su funcionalidad, los actores que pueden acceder a él, las dependencias, precondiciones y el flujo básico del UC describiendo las acciones que puede realizar el actor paso a paso, los flujos alternos que podrían surgir del flujo básico, la postcondición y los comentarios adicionales.

Tabla 1. Cuadro de especificaciones de Casos de uso

UC# - Nombre caso de uso	
Descripción	Breve descripción.

Actor/es	Actor 1, ..., Actor N.	
Dependencias	● UC#	
Precondición	Precondición antes de efectuar el UC	
Flujo básico	Paso	Acción
	# Paso	Acción a realizar
Flujos Alternos		
Postcondición	Postcondición luego de efectuar el UC.	
Comentarios	Comentarios adicionales	

Fuente: (Junta de Andalucía, 2009)

Las tablas descriptivas de los casos de uso se presentan en el anexo A.

En cuanto a los Requerimientos No Funcionales tendremos:

- **Seguridad:** Proveer de mecanismos de cifrado de contraseñas durante el proceso de autenticación para que las contraseñas de los usuarios no sean vulneradas. Adicionalmente proveer un usuario y contraseña seguros para el acceso a la base de datos y que de esta manera evitar que cualquier persona externa pueda modificar los datos de los usuarios sin conocer esta información.
- **Usabilidad:** Utilizar los principales estándares de usabilidad en cuanto a la interacción humano computador de manera que los usuarios pueda interactuar con el sistema con facilidad desde cualquier dispositivo (incluyendo los móviles).
- **Disponibilidad:** Emplear tecnologías y mecanismos que permitan mantener el acceso a lo largo del proceso de inscripción según la demanda de usuarios.
- **Rendimiento y Escalabilidad:** Permitir que el sistema pueda ejecutarse correctamente con la cantidad actual de usuarios dentro de la EACRV y que pueda soportar la incorporación de nuevos módulos a lo largo del tiempo, manteniendo la escalabilidad horizontal (varios

servidores concurrentes) y vertical (mayores capacidades de hardware) en caso de ser requerida.

- **Mantenimiento:** Proveer de recomendaciones para el respaldo de la información de los estudiantes y de herramientas para que la administración de los servicios sea más sencilla.

La arquitectura de solución propuesta para el Sistema Web será Cliente-Servidor, en la que del Lado del Cliente se dispondrá el manejo de interfaces y eventos a través de los navegadores web modernos, en cuanto al Lado del Servidor, es el encargado de ejecutar la lógica de negocio, adicionalmente tendrá la capa de Datos que será la encargada de almacenar la información indispensable a ser utilizada. En la Figura 3 se presenta una descripción gráfica de la propuesta. En la que destacan tres capas principales con diferentes tecnologías:

- **Capa del cliente o Front-End:** se utilizarán las tecnologías web con mayor uso y soporte en los navegadores modernos como lo son HTML5 (Lenguaje de Marcas de Hipertexto), CSS3 (Hojas de estilo en cascada) y JS (JavaScript), de esta manera el sistema podrá ser accedido desde cualquier ordenador. Adicionalmente se utilizará un framework, o marco de trabajo, llamado Angular, para agilizar los tiempos de desarrollo y ventajas que nos proporcionan las SPA (Aplicaciones de una sola página).
- **Capa del servidor o Back-End:** enlace de la capa del cliente con el servidor mediante conexiones a través del protocolo HTTP (Protocolo de transferencia de hipertexto) hacia un servidor, utilizando como principal lenguaje de programación a JS a través del entorno de ejecución llamado Node.js y su microframework Express.js. Por ser un lenguaje que utiliza JS se podrá aprovechar de utilizar un solo lenguaje tanto para el cliente como para el servidor, además de los beneficios propios del lenguaje del lado del servidor.
- **Capa de datos o Bases de Datos:** el servidor web se conectará a una base de datos relacional PostgreSQL donde se almacenará la información que utilizará el sistema web. De esta manera la administración de la base de datos será mucho más sencilla gracias al uso del lenguaje SQL que hoy en día es mucho más conocido que los de las bases de datos no relacionales.

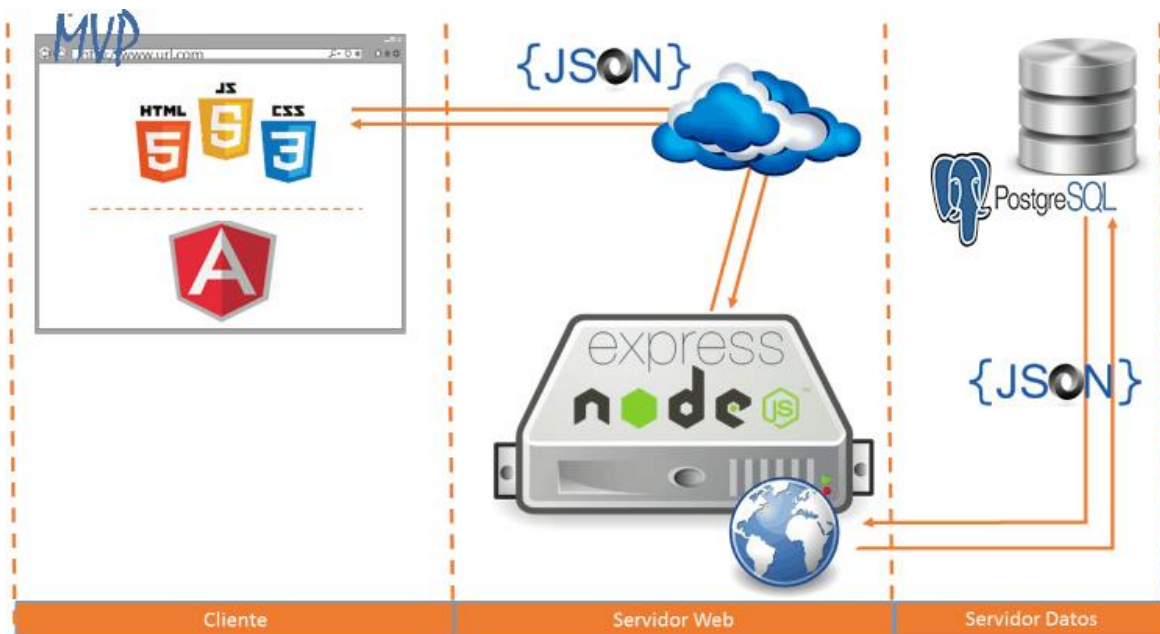


Figura 3. Arquitectura tecnológica del Sistema Web. Fuente: (Alarcón, 2015)

1.10 Método de desarrollo

Durante el desarrollo de la solución propuesta se trabajará con una adaptación del framework Scrum el cual es un marco de trabajo dentro del cual se pueden emplear varios procesos y técnicas para mejorar la eficacia con relación a la gestión del producto y las técnicas de trabajo (Schwaber & Sutherland, 2018).

Entre algunos eventos que incluye el framework Scrum, que se tomarán en cuenta para su uso dentro de la realización de la propuesta consideramos:

- **Sprint:** es el corazón de Scrum, es un intervalo de tiempo, generalmente entre 2 y 4 semanas, en el cual se espera realizar las historias de usuario del Sprint Backlog con el fin de generar un incremento de producto en el proyecto. El mismo comienza con una planificación, en donde se discute que historias de usuario que se tomarán en este sprint, y termina con una retrospectiva en la que se evalúa el resultado del proceso. Para el desarrollo del proyecto se trabajará con un sprint de 2 semanas de duración.

- Reunión de Planificación del Sprint o Sprint Planning: reunión de trabajo previa al inicio de cada sprint en la que se determina cuál va a ser el objetivo del sprint y las tareas necesarias para conseguirlo.
- Revisión del Sprint o Sprint Review: análisis e inspección del incremento generado, y adaptación de la pila del producto si resulta necesario. (Schwaber & Sutherland, 2018)

En cuanto a los artefactos que se emplearan durante el desarrollo, destacamos los siguientes:

- Product Backlog: es un listado de todas las historias de usuario (por sus siglas en ingles US) previamente analizadas y discutidas con el cliente, que se pretenden desarrollar durante el desarrollo de un proyecto.
- Sprint Backlog: es el conjunto de elementos del Product Backlog seleccionados para el Sprint, más un plan para entregar el Incremento del producto y lograr el objetivo del Sprint.
- Incremento de Producto: es la suma de todos los elementos de Product Backlog completados durante un Sprint y el valor de los incrementos de todos los Sprints anteriores. El incremento debe estar en condiciones de uso en producción, independientemente de si el propietario del producto decide liberarlo. (Schwaber & Sutherland, 2018)

Adicionalmente, Scrum posee una cantidad de roles que diferencian las distintas asignaciones que debe realizar cada persona dentro del proyecto. En este caso, los tesisas formarán parte de rol de SDT (equipo de desarrollo de Scrum por sus siglas en inglés) y serán los encargados de realizar las distintas historias de usuario durante cada Sprint. En cuanto a las tutoras tendrán el rol de Product Owners (dueños de producto) que se encargarán de velar por el cumplimiento de lo acordado. En cuanto al resto de los roles, estos no podrán ser asignados por contar con un equipo pequeño de trabajo, por lo que se decidió prescindir de ellos, aunque aún así se puede hacer uso perfectamente del framework sin ningún problema dada su naturaleza flexible.

En la Figura 4 podrán evidenciar en modo de resumen todo el conjunto de eventos, roles, artefactos del marco de trabajo de Scrum.

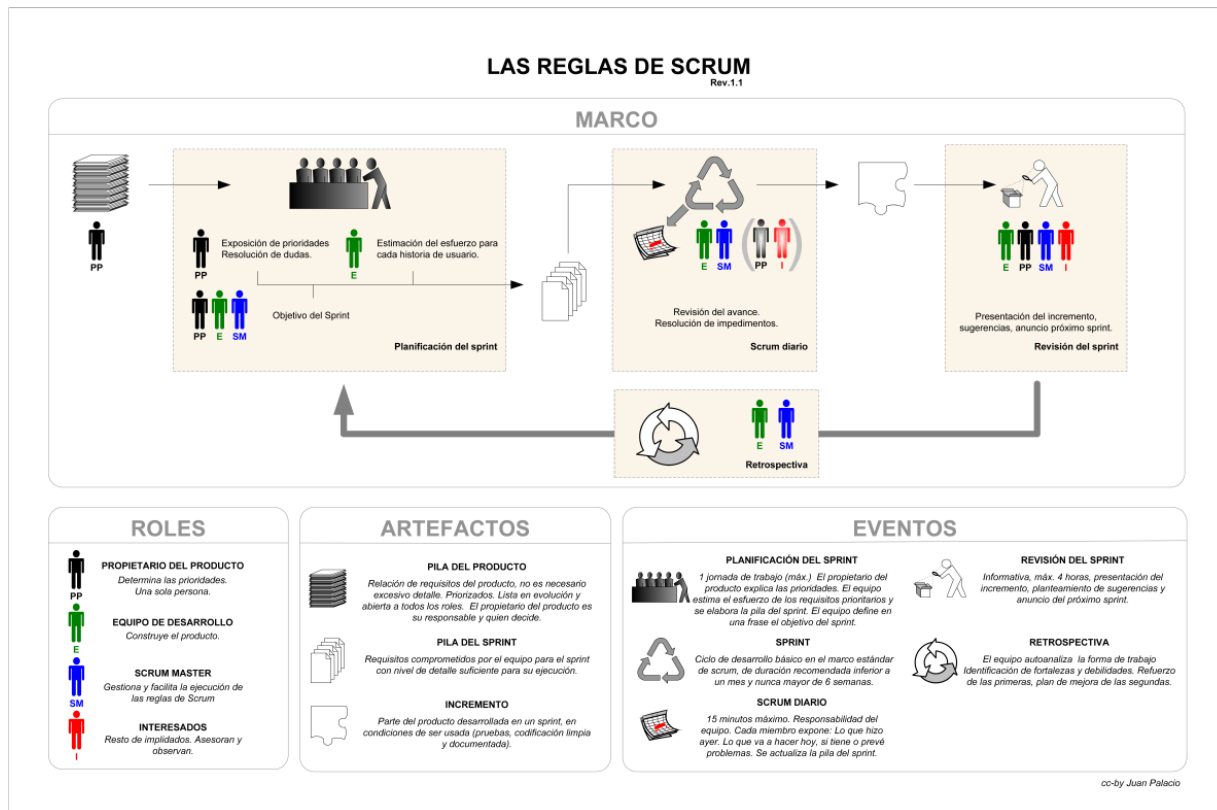


Figura 4. Marco de trabajo Scrum. Fuente: (Palacio, 2014)

1.11 Plan de trabajo

Siguiendo el planteamiento expuesto en el apartado anterior con respecto al uso del marco de trabajo Scrum, se procederá a definir el plan de trabajo conformado por los sprint que conformarán la duración del tiempo de desarrollo del proyecto. Estará compuesto por un total de cinco sprint con una duración de 1 a 3 semanas. El Sprint 0 por ser fase de diseño y del Sprint 4 por ser fase de entrega que tendrán una duración de una semana cada uno, en cuanto a los Sprint 1, 2 y 3 tendrán una duración de 3 semanas por ser parte de la fase de desarrollo del sistema, y se obteniendo un resultado de tres meses para la realización de la totalidad de los casos de uso propuestos.

En la Tabla 2 se podrá evidenciar la planificación de las entregas de cada sprint con su duración estimada y los objetivos respectivamente:

Tabla 2. Plan de trabajo. Elaboración propia (2019).

Sprint	Duración Estimada	Objetivos
0 – Diagramas, diseño y ambiente de desarrollo	1 semana: desde el 12/04/2019 hasta el 19/04/2019	<p>Confección de diagramas entidad relación que representen gráficamente los diferentes entes del Sistema y como se relacionan entre sí. Confección de diagramas de flujo para visión gráfica de la secuencia de las funcionalidades.</p> <p>Elaboración de mockups para el diseño de las interfaces.</p> <p>Preparación de ambiente de desarrollo, descarga de diferentes tecnologías y editor de texto usado para la implementación.</p>
1 – Lado del servidor (primera parte)	3 semanas: desde el 19/04/2019 hasta el 10/05/2019	<p>Implementación de arquitectura del lado del servidor (Backend). Definición de modelos con sus respectivos atributos. Implementación de la integración con la base de datos.</p> <p>Desarrollo del módulo de autenticación.</p>
2 – Lado del servidor (Segunda parte)	3 semanas: desde el 10/05/2019 hasta el 31/05/2019	Desarrollo de las funcionalidades planteadas para cada tipo de usuario, Estudiante, Docente, Coordinador y Administrador.
3 – Lado del cliente	3 semanas: desde el 31/05/2019 hasta el 21/06/2019	Implementación del lado del cliente (Frontend), todas las vistas necesarias para el correcto uso del Sistema a través del Cliente Web.
4 – Despliegue en ambiente de pruebas	1 semanas: desde el 21/06/2019 hasta el 28/06/2019	Despliegue de plataforma en ambiente de pruebas para certificación de requerimientos y correcto funcionamiento.

Teniendo en cuenta el contexto y las necesidades en la que se encuentra la EACRV-FAU-UCV, es de suma importancia hacer uso de las tecnologías para resolver gran parte de los problemas que se ven reflejados al realizar los procesos de gestión académica de los estudiantes y del personal trabajador. Con los objetivos planteados y el alcance bien definido, será posible desarrollar un sistema que facilite las tareas diarias de las personas que hacen vida en la Facultad.

CAPÍTULO II

SISTEMAS WEB

En el presente capítulo presenta la definición, características e importancia de los Sistemas Web, lo cual es el tema central del trabajo que estamos desarrollando. Así mismo, presentaremos algunos ejemplos que podrán ayudar a entender mejor a qué se hace referencia cuando hablamos de Sistemas Web.

2.1 Definición de Sistema Web

Un Sistema Web es una herramienta que se usa para resolver un problema haciendo uso de la tecnología. Funcionan mediante la comunicación entre un cliente (usualmente un navegador web, en un computador o un teléfono inteligente) y un servidor, el cual proporciona toda la información necesaria a ser desplegada para el usuario.

Los Sistemas Web han venido obteniendo cada vez más importancia con el paso de los últimos años gracias al valor que ofrecen como solución a una inmensa cantidad de problemas, así como automatización de largos y tediosos procesos. A su vez, por la facilidad que brindan al momento de hacer actualizaciones, mantenimiento y distribución hacia los usuarios. Sin mencionar una de sus ventajas más importantes como es la ubicuidad que le brinda el hecho de estar en la red de internet, de manera que se puede acceder a ellas desde cualquier parte del mundo.

Es importante mencionar que una página web puede contener elementos que permiten una comunicación activa entre el usuario y la información. Esto permite que el usuario acceda a los datos de modo interactivo, gracias a que la página responderá a cada una de sus acciones, como por ejemplo rellenar y enviar formularios, participar en juegos diversos y acceder a gestores de base de datos de todo tipo.

2.2 Características

Según Rivas (2011), los Sistemas Web tienen las siguientes características:

- **Multiplataforma:** ya que toda su información proviene de un servidor, el cual tiene compatibilidad con cualquier explorador web instalado sobre cualquier dispositivo sin

importar el tipo de hardware o software que posea. Esta característica se extiende a que los Sistemas Web tienen menos problemas y errores (en informática conocidos como bugs) ya que no están desarrolladas de forma tal que corran sobre una tecnología específica, de esta manera, una misma versión puede correr sobre distintos sistemas operativos lo cual disminuye notablemente la cantidad de los mencionados bugs que pueden surgir durante su utilización.

- **Actualización y mantenimiento:** es muy fácil realizar actualizaciones y mantenimiento de Sistemas Web, ya que, al estar alojados en la nube, solo requieren que se despliegue la nueva versión sobre el servidor en el que están alojadas, de manera que los usuarios no deben hacer ningún tipo de descarga para acceder a los cambios.
- **Acceso:** al estar alojadas en la red de internet, los usuarios pueden acceder fácilmente a ellas desde cualquier dispositivo mediante un navegador, esto provee una infinita cantidad de posibilidades para hacer uso de estas herramientas.
- **Seguridad de los datos:** al ser herramientas que tienen toda la información alojada en un servidor en la nube, los datos están seguros y tienen un fácil acceso, ya que no se pierden en caso de posibles daños en un determinado hardware. Si un disco duro sufre de fallas se puede acceder a la herramienta, y por ende a la información, desde cualquier otro dispositivo con un navegador.
- **Requiere de acceso a internet:** aunque la gran mayoría de las características son ventajas, este tipo de sistemas requieren que el dispositivo que accede a ellos esté conectado a la red de internet. Sin embargo, con lo importante que se ha vuelto la tecnología a lo largo de los últimos años, prácticamente hay acceso a internet en casi todos los lugares del mundo.

2.3 Arquitectura

Se decidió usar la arquitectura Cliente-Servidor, que como explica Marini (2012), es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Las aplicaciones Clientes realizan peticiones a una o varias aplicaciones Servidores, que deben encontrarse en ejecución para atender dichas demandas.

El modelo Cliente-Servidor permite diversificar el trabajo que realiza cada aplicación, de forma que los Clientes no se sobrecarguen, cosa que ocurriría si ellos mismos desempeñan las funciones que le son proporcionadas de forma directa y transparente. En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema. Tanto el Cliente como el Servidor son entidades abstractas que pueden residir en la misma máquina o en máquinas diferentes.

En la Figura 5 se puede observar gráficamente el modelo de la arquitectura Cliente-Servidor:

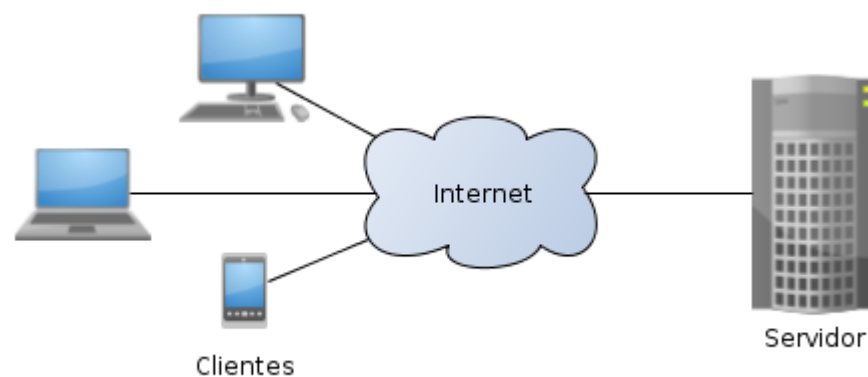


Figura 5. Arquitectura Cliente – Servidor. Fuente (Baul de Linux, 2019)

Acorde al Instituto Tecnológico de Matehuala (2015), es el tipo de arquitectura que comúnmente mejor se adapta a los Sistemas Web. Realizando la comunicación entre el Cliente y el Servidor por medio de peticiones ejecutadas sobre los protocolos HTTP.

Los navegadores piden páginas (almacenadas o creadas dinámicamente) con información a los servidores web. En la mayoría de los ambientes de desarrollo de Sistemas Web, las páginas contienen código HTML y scripts dinámicos, que son ejecutados por el servidor antes de entregar la página.

Una vez que se entrega una página, la conexión entre el navegador y el servidor Web se rompe, es decir que la lógica del negocio en el servidor solamente se activa por la ejecución de

los scripts de las páginas solicitadas por el navegador (en el servidor, no en el cliente). Cuando el navegador ejecuta un script en el cliente, éste no tiene acceso directo a los recursos del servidor.

Se usará un modelo de servicio de tres capas, el cual, de nuevo citando a Marini (2012), es una vista abstracta que establece las definiciones, reglas y relaciones entre las estructuras relacionadas con la aplicación. Sirve de base para el intercambio de ideas durante el desarrollo lógico de la aplicación, determina exactamente lo que debe hacer y cómo será el resultado. El modelo de servicios establece tres grandes conjuntos de funcionalidades, en cada uno de los cuales se encuadran las distintas tareas en las que se ve involucrado cualquier tipo de proyecto de desarrollo. Dicho modelo establece los siguientes conjuntos que comúnmente denominamos lógicas.

1. Lógica de negocio: define los procesos que involucra el sistema, y contiene el conjunto de operaciones requeridas para proveer el servicio. Controla la secuencia de acciones y fuerza el cumplimiento de las reglas del negocio; además, asegura la integridad de las operaciones necesarias para su cumplimiento. La lógica del negocio también transforma una serie de datos en información útil para el usuario mediante la aplicación de las reglas apropiadas.
2. Administración de los datos: en este conjunto entran los procesos encargados de la gestión de los datos propiamente dicha, es decir, los procesos encargados del mantenimiento de los datos, de garantizar las reglas de integridad referencial establecidas, así como de la gestión de las transacciones. Estas tareas son realizadas, generalmente por un Sistema de Gestión de Bases de Datos.
3. Interfaz: es la lógica responsable del control de todos los aspectos relacionados con la interacción entre el usuario y la aplicación. Para llevar a cabo esta tarea de control, es necesario conocer qué tipos de usuarios utilizarán la aplicación, qué actividades tienen que realizar y, teniendo en cuenta estos datos, cuáles son los mejores estilos de interfaz para que esos usuarios realicen sus tareas. En esta lógica se engloban todas las tareas que deben ser realizadas por la parte Cliente del modelo general.

En la Figura 6 se puede evidenciar un ejemplo del modelo de servicios de tres capas:



Figura 6. Modelo de Servicios. Fuente (Marin, 2012)

2.4 Ejemplos

- **Contract Room**

Es un sistema de gestión de contratos (de todo tipo) en línea, que permite centralizar los procesos que involucra generar este tipo de documento. Este sistema además facilita el manejo posterior de dichos contratos permitiendo acciones como modificaciones, seguimiento de historial, firma en línea, manejo de plantillas, entre otros; reduciendo así los gastos operativos que genera contar con la presencia de las partes involucradas y la inversión de tiempo que esto demanda, lo que se traduce en mayor productividad para los usuarios gracias al uso de la tecnología.

En la Figura 7 se puede observar una de las funcionalidades más importantes del sistema Contract Room, en la interfaz de configuración de colaboración, en la que se puede agregar tantos usuarios como se desee para colaborar en un mismo contrato, con el fin de tener acceso a la información y sus respectivas ediciones por parte de los integrantes de un contrato que desean realizarlo en equipo. Adicionalmente, se puede observar que el Sistema ofrece distintos roles para ser asignados al usuario que se quiere agregar a la colaboración.

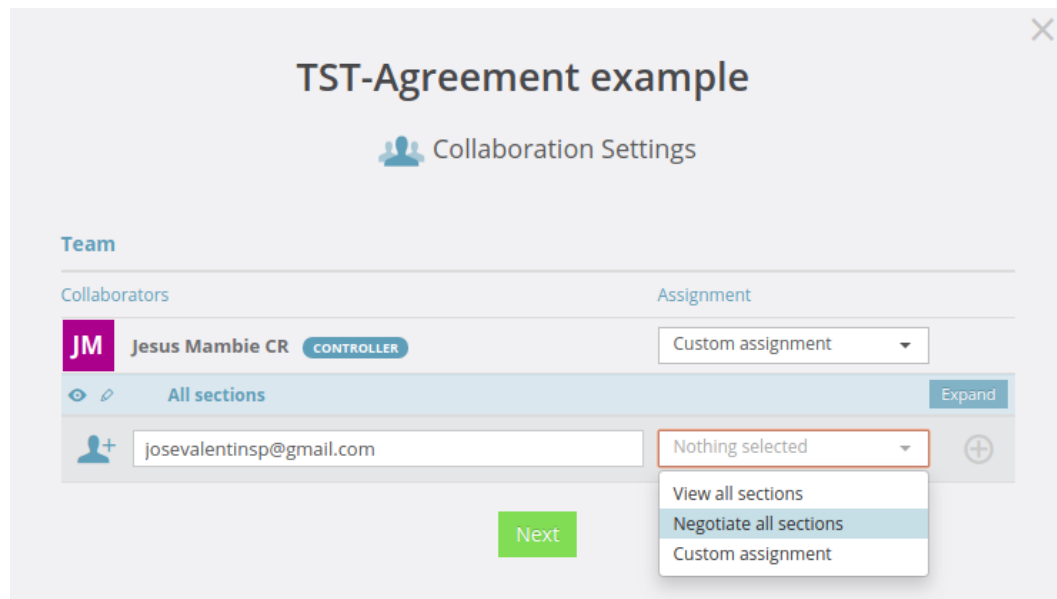


Figura 7. Interfaz de Usuario Final de Contract Room. Fuente: Elaboración Propia, 2019.

- **Tabata Desk**

Es una aplicación diseñada para establecimientos que ofrecen servicios de actividades físicas de todo tipo alrededor del mundo. La misma, permite tanto a los administradores como a los usuarios, verificar la disponibilidad de las clases en tiempo real e inscribirse o retirarse de las mismas a través de un sistema centralizado en línea sin necesidad de tener ningún tipo de contacto. Esta herramienta facilita el manejo del tiempo de los beneficiarios, además de permitir la emisión de reportes que pueden ser relevantes a nivel de estadísticas para los administradores.

En la figura 8 se puede evidenciar la interfaz que ven los usuarios al momento de suscribirse una clase, en la cual obtienen información con respecto al entrenador, duración, tipo, y cupos disponibles de la misma.

INTENSITY

Modalidad del Crossfit que implica todos los movimientos del Crossfit exceptuando el Levantamiento Olímpico.

Fecha: 22/03/2019 5:00pm

Duración: De: 40 min

Entrenador: Ruben Alvitez

Cupos Disponibles: 10

ATLETAS

Nombre
Sergio Castillo
Salem Homsani
Andres Alberti
Vanessa Noguero
Carolina Gonzalez
Lucila Rivas

RUTINA

No hay un worout definido para esta clase

Suscribirse

Atrás

Figura 8. Interfaz de suscripción a clase de Tabatadesk. Fuente: Elaboración Propia, 2019.

- **Amazon**

Es un sistema que permite realizar compras desde casi cualquier parte del mundo con sólo tener un dispositivo con acceso a internet. El mismo permite la búsqueda de cualquier tipo de producto, ofrece diferentes opciones de vendedores, opera las 24 horas del día y realizan entregas hasta la dirección que el usuario desee.

Adicionalmente, es una herramienta que minimiza errores relacionados con estafas al momento de realizar los pagos, ya que la información que el usuario suministra es totalmente confidencial y también permite el rastreo de los productos. Esta aplicación web reduce considerablemente los gastos de tiempo y dinero para los beneficiarios, y además es una ventana para los cientos de vendedores que ofrecen sus productos en una plataforma que les permite operar sin los gastos que implica un establecimiento comercial.

En la Figura 9 se puede apreciar una interfaz de usuario que presenta Amazon, cuya funcionalidad permite a los usuarios realizar una compra, incluyendo un breve resumen de la información relevante del producto. Así mismo se puede observar un botón de “Agregar al Carro”, el cual guarda la información de la compra en el caso de que el usuario desee realizarla en el futuro.

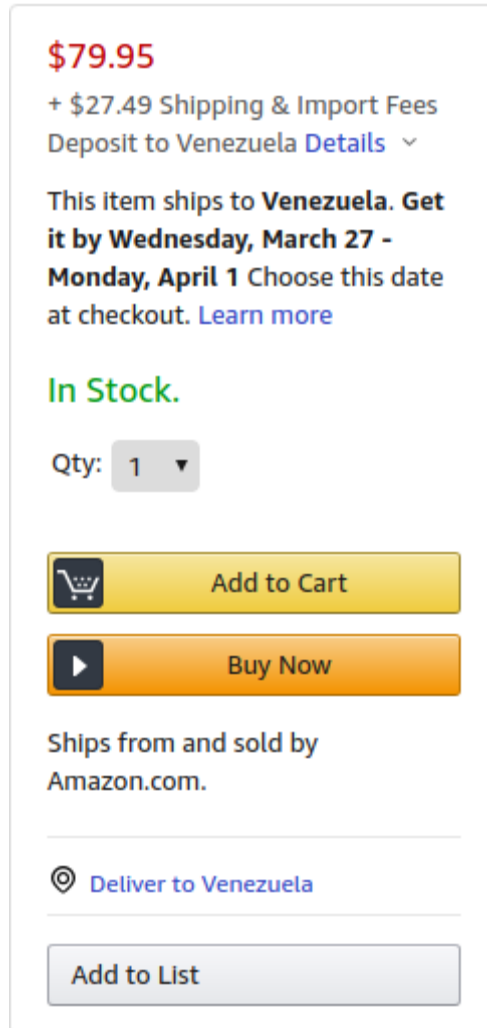


Figura 9. Interfaz de compra de artículo de Amazon. Fuente: Elaboración Propia, 2019.

2.5 Importancia

Los Sistemas Web han cobrado una gigante importancia en los últimos años por diversas razones:

- Se pueden utilizar desde cualquier dispositivo móvil o computador con acceso a internet, esto le da una gran facilidad de acceso alrededor de todo el mundo.
- Permiten la comunicación directa e inmediata sin importar la distancia o el tiempo.
- Facilitan la optimización de los procesos y disminuyen considerablemente gastos de tiempo y energía en relación al trabajo que es realizado de manera manual, lo que se traduce en reducción de costos y manejo eficiente del tiempo.

- Permiten el almacenamiento de información masiva y posteriormente facilitan el acceso y manejo de la misma; y adicionalmente aumentan la confiabilidad ya que, según las funcionalidades que tenga el sistema, se puede lograr que la información se encuentre resguardada.
- En la mayoría de los casos son escalables, lo cual posibilita que sean modificados y se realicen actualizaciones y mejoras que se adapten a las necesidades de los usuarios.

Finalmente, en este capítulo se definieron conceptos relacionados con los Sistemas Web, así como sus principales características e importancia. Su auge a nivel mundial ha sido contundente y totalmente justificado ya que los beneficios que ofrece a los usuarios son numerosos. Tanto a nivel corporativo y comercial, como a nivel personal, la automatización y sistematización de los procesos logran una enorme simplificación en la realización de las tareas, por lo que cada día más personas quieren sumarse a la era digital y disfrutar de las múltiples ventajas que los Sistemas Web tienen para ofrecer.

CAPITULO III

TECNOLOGÍAS DE DESARROLLO

En este tercer capítulo se podrán definir en detalle las tecnologías seleccionadas para el desarrollo de la solución. Se procederá a describir la arquitectura, que constará tanto de tecnologías del lado del cliente como del servidor, la base de datos a utilizar, y las herramientas necesarias para la implementación del código.

3.1 Concepto

Al momento de realizar un desarrollo web, la decisión del tipo de tecnología a usar es de suma importancia. Con el auge que han cobrado los Sistemas Web en la última década, se han creado multitud de aplicaciones, frameworks, librerías, arquitecturas y sistemas de publicación en diferentes versiones que a su vez reciben cambios o mejoran con el tiempo. Hay que tomar en cuenta los requerimientos que demanda el sistema a desarrollar, ya que existen tecnologías que aplican mejor para ciertas características que otras.

Citando a Baquero (2015), a la hora de seleccionar el lenguaje con el cual trabajar, es importante conocer las diferencias y singularidades de cada uno de ellos, sus ventajas e inconvenientes, en función de la plataforma para la que estemos desarrollando nuestro proyecto. Además, también conviene familiarizarse con el concepto framework, cada vez más extendido, y que hace referencia al conjunto de buenas prácticas, módulos y librerías de cada lenguaje para abordar proyectos y necesidades concretas.

3.2 Tecnologías de desarrollo

Se describen las tecnologías que se utilizarán para el desarrollo, seleccionadas en base a su adaptación a los requerimientos del Sistema Web:

3.2.1. Tecnologías del Lado Cliente

Son todas aquellas tecnologías usadas para desarrollar un Sistema Web del lado del cliente (Front-End), que en este caso son los navegadores web. Mediante ellas, el usuario es capaz de interactuar con el sistema, es decir, tener acceso a todas las vistas y realizar la carga de datos

necesarios que posteriormente serán almacenados o procesados. Los lenguajes y frameworks propuestos para el desarrollo del sistema serán los siguientes:

HTML: Según Lamarca (2018) el lenguaje de marcas de hipertexto, HTML (HyperText Markup Language) se basa en el metalenguaje SGML (Standard Generalized Markup Language) y es el formato de los documentos de la World Wide Web.

HTML fue concebido como un lenguaje para el intercambio de documentos científicos y técnicos adaptado para su uso por no especialistas en tratamiento de documentos. HTML resolvió el problema de la complejidad de SGML sirviéndose de un reducido conjunto de etiquetas estructurales y semánticas apropiadas para la realización de documentos relativamente simples, además del soportar el hipertexto.

En un corto período de tiempo, HTML se hizo muy popular y rápidamente superó los propósitos para los que había sido creado. Desde sus albores, ha habido una constante invención de nuevos elementos para usarse dentro de HTML como estándar y adaptarlo a las nuevas posibilidades de la Web, como la posibilidad de usar elementos multimedia o la utilización de elementos dinámicos que hacen las páginas web mucho más llamativas e interactivas para el usuario (Lamarca, 2018).

CSS3: por sus siglas en español, Hojas de Estilo en Cascada, son un lenguaje de diseño simple destinado a simplificar el proceso de presentación y manejo de apariencia de páginas web.

Usando CSS, puede controlar el color del texto, el estilo de las fuentes, el espacio entre los párrafos, cómo se dimensionan y disponen las columnas, qué imágenes de fondo o colores se usan, variaciones en la visualización para diferentes dispositivos y tamaños de pantalla, así como una variedad de otros efectos. Entre las ventajas del uso de CSS tenemos:

- Tiempo: permite la reutilización de la misma hoja en varias páginas HTML. Puede definir un estilo para cada elemento HTML y aplicarlo a tantas páginas web como desee.

- Las páginas se cargan más rápido: al utilizar CSS, no se necesita escribir múltiples atributos de etiquetas HTML. Basta escribir una sola regla para aplicarla en todas sus apariciones. De este modo, menos código significa tiempos de descarga más rápidos.
- Fácil mantenimiento: para realizar un cambio global, simplemente cambie el estilo, y todos los elementos en todas las páginas web se actualizarán automáticamente.
- Compatibilidad con múltiples dispositivos: las hojas de estilo permiten optimizar el contenido para más de un tipo de dispositivo. Al utilizar el mismo documento HTML, se pueden presentar diferentes versiones de un sitio web para dispositivos de mano como PDA, teléfonos celulares o para imprimir. (W3C, 2016).

JavaScript: basado en Navarrete (2007) es un lenguaje de programación que presenta una característica especial: sus programas, llamados comúnmente scripts, se leen en páginas HTML y se ejecutan en los navegadores web. Estos scripts normalmente consisten en unas funciones que son llamadas desde el propio HTML cuando algún evento sucede. De ese modo, podemos añadir efectos como que un botón cambie de forma al pasar el ratón por encima, o abrir una ventana nueva al pulsar en un enlace, etc. Fue desarrollado por Netscape a partir del lenguaje Java.

Tiene como principales características que es interpretado, orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico. Aunque se utiliza principalmente en su forma del lado del cliente como lo define Navarrete (2007), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas. Últimamente ha venido cobrando gran importancia y cada vez mayor uso del lado del servidor en todo el mundo, con tecnologías que se expondrán más adelante en el documento.

Angular: como explica Robles (2018), es un framework de desarrollo para JavaScript creado por Google. La finalidad de Angular es facilitarnos el desarrollo de aplicaciones web SPA y además darnos herramientas para trabajar con los elementos de una web de una manera más sencilla y óptima.

SPA: acorde a Baquero (2017), por sus siglas en inglés Single Page Application, que en español significan aplicación de una sola página, consiste en el tipo de aplicaciones web, en el

que todo lo que se muestra y se procesa está dentro de la misma página, de manera que al pasar de una opción a otra no hace falta recargar el navegador, lo normal es que sea un único archivo desde el que se reproduce absolutamente todo.

Se decidió usar Angular como framework para el desarrollo del Front-End (lado del cliente) por ser uno de los marcos de trabajo más estables que existe en la actualidad. Provee una biblioteca con una serie de herramientas que permiten una sencilla organización y estructura del código, para optimizar y acelerar su implementación. Adicionalmente, se busca hacer una herramienta que sea fácilmente escalable para su futuro crecimiento, Angular divide su código en porciones llamadas Componentes Web que permiten su reutilización con poco esfuerzo. Ha cobrado mucho valor con el paso de los últimos meses y se ha vuelto una apuesta a largo plazo, ideal para proyectos grandes que se espera estén vigentes por mucho tiempo.

Como se explicó previamente, Angular facilita el desarrollo de Aplicaciones Web SPA (Aplicación de una sola Página), las cuales acorde a Baquero (2017), consisten en el tipo de Aplicaciones Web, en el que todo lo que se muestra y se procesa está dentro de la misma página, de manera que al pasar de una opción a otra no hace falta recargar el navegador, lo normal es que sea un único archivo desde el que se reproduce absolutamente todo.

3.2.2. Tecnologías del Lado del Servidor

Acorde a Arjonilla (2016), es el componente del desarrollo web que se encarga de que toda la lógica para que una página web funcione correctamente. Se trata de un conjunto de acciones realizadas en una web pero que no se ven, como, por ejemplo, la comunicación con el servidor o el acceso a la base de datos para buscar información. Se presentarán las tecnologías que se van a utilizar del lado del servidor (Back-End) así como las razones por las cuales fueron seleccionadas:

Node.js: es un entorno JavaScript del lado del servidor, basado en eventos. Node.js ejecuta JavaScript utilizando el motor V8, desarrollado por Google para uso de su navegador Chrome. Aprovechando el motor V8 permite a Node.js proporciona un entorno de ejecución del lado del servidor que compila y ejecuta JavaScript a altas velocidades (Abernethy, 2011).

Existen diversas razones por las cuales se seleccionó Node.js como tecnología de desarrollo. A continuación, se presentarán algunas ventajas extraídas de una noticia del sitio web de Universia España (2017) donde se podrá ver en detalle las características más importantes, que a su vez son ventajas con respecto a otras tecnologías:

- Velocidad: una de las características de Node.js más importantes es su rapidez, lo que ayuda a desarrollar más rápido, ejecutar test de unidad de forma veloz, las aplicaciones se ejecutan mucho más ágiles.
- Flexibilidad: porque puede ejecutarse en una variedad de servidores, entre los que destacan Microsoft Windows, Mac OS X y Unix.
- Rendimiento: permite crear trabajos de gran calidad y disminuye el margen de experimentar errores técnicos.
- Estándar: el hecho de permitir a los desarrolladores escribir en JavaScript tanto del lado del servidor como del lado del cliente hace que la transferencia de datos entre estos puntos sea más rápida y por lo tanto reduce los tiempos de trabajo. Así como el uso de JSON como estructura de datos para el intercambio de información.
- Concurrencia: Node.js está orientado a eventos, por lo que es ideal para aplicaciones con alto tráfico de usuarios en tiempo real y el manejo concurrente de las ejecuciones. Esta característica también refleja una disminución de los costos de infraestructura, al no necesitar más de un servidor ya que uno solo puede soportar decenas de miles de conexiones concurrentes.

Es importante destacar el hecho de hacer uso de JavaScript como lenguaje de desarrollo, que no sólo establece un estándar para todo el proyecto, a su vez proporciona una curva de aprendizaje que acelerará los tiempos de producción, y, por ende, la culminación rápida y eficiente del mismo.

Adicionalmente, se decidió utilizar el marco de trabajo Express.js, que según su Web oficial (2019), es una infraestructura de Aplicaciones Web en Node.js mínima y flexible que proporciona un conjunto sólido de características y herramientas para facilitar y agilizar el desarrollo.

3.3 Bases de Datos

Son repositorios centralizados donde se encuentra almacenada toda la información perteneciente a un mismo contexto, en este caso un Sistema Web, ofrece seguridad, fácil gestión, actualización y consulta sobre los datos.

Se decidió utilizar PostgreSQL como base de datos para el Sistema, a continuación, se presenta su definición, y las razones de su selección:

PostgreSQL: como explica Torres (2016) es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiar a otras bases de datos comerciales.

PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando.

Entre las principales ventajas de PostgreSQL tenemos:

- **Flexibilidad:** con el tipo de datos JSONB, los usuarios no tienen que escoger entre almacenes de datos relacionales y no-relacionales: pueden tener los dos al mismo tiempo. JSONB soporta búsquedas rápidas y consultas de búsqueda con expresiones simples usando Generalized Inverted Indexes (GIN). Varias funciones de apoyo les permiten a los usuarios extraer y manipular datos JSON, con un rendimiento que iguala o supera las bases de datos documentales más populares. Con JSONB, los datos de tablas pueden ser fácilmente integrados con los datos documentales, obteniendo un entorno de base de datos completamente integrado.
- **Escalabilidad:** provee una API para leer, filtrar y manipular el flujo de replicación. Esta interfaz es la base para nuevas herramientas de replicación, como la Replicación Bi-Direccional, la cual soporta la creación de clústeres de PostgreSQL multi-maestros.
- **Rendimiento:** PostgreSQL incluye manejo de índices, vistas Materializadas actualizables de forma concurrente, para reportes más rápidos y actualizados, recarga rápida del caché

de la base de datos en un reinicio y escritura paralela altamente rápida en el log transaccional.

- **Alta Concurrencia:** Mediante un sistema denominado MVCC (Acceso concurrente multiversión, por sus siglas en inglés) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases de datos, eliminando la necesidad del uso de bloqueos explícitos.
- **Amplia variedad de tipos nativos:** PostgreSQL provee nativamente soporte para: números de precisión arbitraria, texto de largo ilimitado, figuras geométricas (con una variedad de funciones asociadas), direcciones IP (IPv4 e Ipv6), bloques de direcciones estilo CIDR, direcciones MAC, arreglos (Torres, 2016).

Se decidió seleccionar PostgreSQL como base de datos principal por todas las ventajas que ofrece, por ser de código abierto, y por adaptarse a la perfección al tipo de proyecto que se quiere realizar, al tener estructuras de datos que están directamente relacionadas entre sí. Adicionalmente existen múltiples ORM que permiten una sencilla integración con Node.js.

3.4 Sistema de control de versiones

El control de versiones es un sistema que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, de modo que puedas recuperar versiones específicas más adelante (Straub & Chacon, 2014). Te permite revertir archivos a un estado anterior, revertir el proyecto entero a un estado anterior, comparar cambios a lo largo del tiempo, ver quién modificó por última vez algo que puede estar causando un problema, quién introdujo un error y cuándo, y mucho más. Utilizar un Sistema de Control de Versiones también significa generalmente que, si pierdes archivos, puedes recuperarlos fácilmente. Además, obtienes todos estos beneficios a un coste muy bajo.

A continuación, se describe el Sistema de Control de Versiones seleccionado que se usará para el desarrollo colaborativo de la solución:

Git: Es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente y son trabajadas por diferentes personas concurrentemente.

Git modela sus datos más como un conjunto de instantáneas de un mini sistema de archivos. Cada vez que confirmas un cambio, o guardas el estado de tu proyecto en Git, él básicamente hace una foto del aspecto de todos tus archivos en ese momento, y guarda una referencia a esa instantánea. Para ser eficiente, si los archivos no se han modificado, Git no almacena el archivo de nuevo, sólo un enlace al archivo anterior idéntico que ya tiene almacenado (Straub & Chacon, 2014). En la figura 10 se muestra como Git almacena la información como instantáneas del proyecto a lo largo del tiempo.

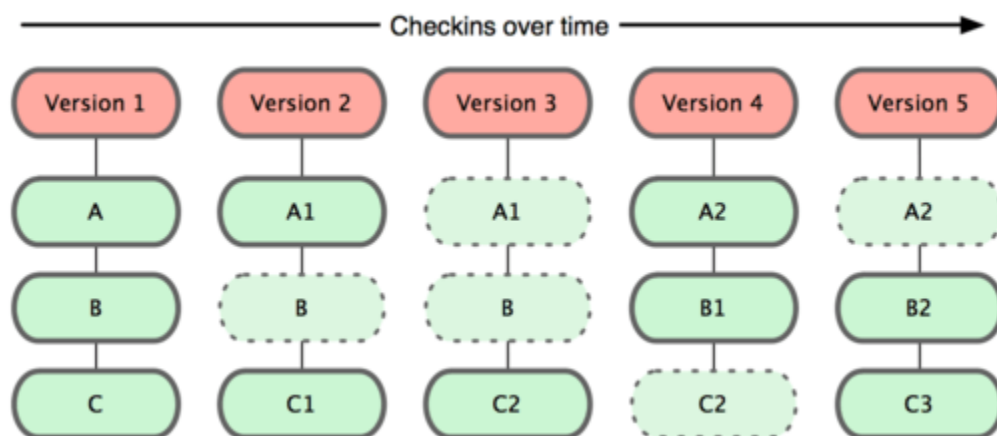


Figura 10. Modelo de Datos utilizado por Git. Fuente: (Straub & Chacon, 2014).

En cuanto al repositorio en la nube para alojar el código de la aplicación se empleará la siguiente herramienta:

GitHub: es un servicio de alojamiento que ofrece a los desarrolladores repositorios de software usando el sistema de control de versiones, Git. Existen varios aspectos que hacen que GitHub sea una opción eficiente para el control y gestión de tus proyectos, estos son algunos de ellos:

- GitHub permite que alojemos proyectos en repositorios de forma gratuita.

- Actualmente permite almacenar repositorios privados sin costo alguno.
- Permite que puedas compartir tus proyectos de una forma mucho más fácil.
- Ofrece todas las ventajas del sistema de control de versiones, Git, pero también tiene otras herramientas que ayudan a tener un mejor control de nuestros proyectos. (Cuadra, 2018)

Se decidió utilizar GitHub como herramienta de desarrollo colaborativo por ser la plataforma web más grande y estable que existe actualmente al momento de trabajar con la tecnología Git. Permite alojar repositorios de código de una manera sencilla, hacer control de versiones y realizar un trabajo colaborativo eficiente, además de, por ser una plataforma web, ser compatible con cualquier sistema operativo.

3.5 Editores de código

Alvarez (2017) afirma que son procesadores de texto orientados a la escritura de código fuente de aplicaciones en lenguajes de programación. Generalmente los editores de código soportan varios lenguajes y son capaces de abrir varios archivos a la vez, resaltar su sintaxis y ofrecer ayudas contextuales a la hora de escribir o visualizar el código fuente de las aplicaciones.

Se decidió usar el siguiente editor para la implementación del código:

VS Code: Es un editor de código fuente desarrollado por Microsoft para Windows, Linux y MacOS. Incluye soporte para depuración, control de Git integrado, resaltado de sintaxis, finalización de código inteligente, fragmentos de código y refactorización de código. También es personalizable, de modo que los usuarios pueden cambiar el tema del editor, los métodos abreviados de teclado y las preferencias. Es gratuito y de código abierto, aunque la descarga oficial está bajo una licencia de propietario. Visual Studio Code se basa en Electron, un marco que se utiliza para implementar aplicaciones Node.js para el escritorio que se ejecuta en el motor de diseño Blink (Wikipedia, 2019).

A lo largo del Capítulo III se presentaron las tecnologías que serán usadas para el desarrollo de la solución planteada, las cuales están conformadas por lenguajes y software que proporcionarán todas las herramientas necesarias para la implementación. Las mismas, fueron

seleccionadas cuidadosamente tomando en cuenta diversas razones entre ellas las necesidades requeridas por el Sistema, la curva de aprendizaje de cada tecnología en base a los conocimientos de los desarrolladores, las mejores prácticas del mercado y el auge que han alcanzado en los últimos años, lo cual las ha convertido en referencias mundiales.

CONSIDERACIONES FINALES

Luego de realizar las investigaciones pertinentes y corroborar cómo son llevados a cabo los procedimientos administrativos en la Facultad de Arquitectura y Urbanismo de la Universidad Central de Venezuela, se pudo constatar la necesidad inminente de generar una solución para agilizar los procesos y reducir el alto riesgo de errores que son generados por el manejo manual de la información. Esta situación es una realidad ya que para la Universidad es muy difícil asumir todo lo que representa la automatización de dichos procesos, a nivel de gastos, tiempo y el personal capacitado que se requiere.

Es así como luego de la detección de esta necesidad, surgió la idea de diseñar un Sistema Web que permitirá realizar algunos de los procedimientos administrativos y académicos más importantes de manera computarizada.

Para la realización de dicho Sistema Web se seleccionaron las tecnologías que se considera que son las que mejor se adaptan a los requerimientos que se buscan cubrir con la implementación del mismo, que a su vez permitirán que el trabajo sea escalable y pueda seguir expandiéndose exitosamente en el futuro.

Con el desarrollo de dicho Sistema en la FAU-UCV, se busca solucionar los inconvenientes más relevantes que genera la forma en que son llevados a cabo los procesos actualmente, minimizando errores y permitiendo un manejo eficiente y eficaz del tiempo del personal que labora en la Facultad. Adicionalmente, el hecho de realizar las tareas y operaciones de forma automatizada permitirá a la FAU-UCV aprovechar los innumerables beneficios que la sistematización digital ofrece hoy en día.

REFERENCIAS

- Abernethy, M. (2011). *¿Simplemente qué es Node.js?* Recuperado de <https://www.netconsulting.es/blog/nodejs/>
- Alarcón, J. M. (15 de enero de 2015). *campusmvp*. Recuperado de: <https://www.campusmvp.es/recursos/post/Que-es-el-stack-MEAN-y-como-escoger-el-mejor-para-ti.aspx>
- Alvarez, M. (2017). *Editor de código*. Recuperado de <https://desarrolloweb.com/wiki/editor-de-codigo.html>
- Arjonilla, R. (2016). *BackEnd*. Recuperado de <https://rafarjonilla.com/que-es/backend/>
- Baquero, J. (2015). *¿Cómo elegir el lenguaje de programación más adecuado para cada proyecto web?* Recuperado de <https://www.arsys.es/blog/programacion/elegir-lenguaje-programacion-web/>
- Baquero, J. (2017). *Single-Page Application, todo un website desde única página*. Recuperado de <https://www.arsys.es/blog/programacion/diseno-web/spa-unica-pagina/>
- Baul de Linux. (2019). *Baul de Linux*. Recuperado de: <https://baulderasec.wordpress.com/linux/6-configurar-impresion-x-windows-y-localizacion/6-4-utilizar-x-para-el-acceso-remoto/6-4-1-fundamentos-clienteservidor-de-x/>
- Cuadra, K. d. (2018). *Que es github*. Recuperado de: <https://platzi.com/blog/que-es-github/>
- Drayer, Alberto. (1970). *Ley de Universidades*. Recuperado de: http://www.ucv.ve/fileadmin/user_upload/documentos/ley_de_universidades.pdf
- Express. (2018). *Express*. Recuperado de: <https://expressjs.com/>
- Facultad de Ingeniería UCV. (2017). *Facultad de Ingeniería*. Recuperado de: <http://www.ing.ucv.ve/ce/index.php>

FAU UCV. (2015). *Facultad de Arquitectura y Urbanismo*. Recuperado de <https://www.fau.ucv.ve/>

Grupo CONEST UCV. (2014). *CONEST*. Recuperado de CONEST: <http://conest.ciens.ucv.ve/webapp>

Junta de Andalucía. (2 de julio de 2009). *Junta de Andalucía*. Recuperado de Junta de Andalucía: <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/416>

Lamarca, M. (2018). *Hipertexto: El nuevo concepto de documento en la cultura de la imagen*. Recuperado de <http://www.hipertexto.info/>

(N/A), Instituto Tecnológico de Matehuala. (2015). *Arquitectura de las aplicaciones Web*. Recuperado de <https://programacionwebisc.wordpress.com/2-1-arquitectura-de-las-aplicaciones-web/>

(N/A), Universia España. (2017). *10 motivos para usar Node.js para desarrollar aplicaciones web*. Recuperado de: <http://noticias.universia.es/ciencia-tecnologia/noticia/2017/07/10/1154033/10-motivos-usar-nodejs-desarrollar-aplicaciones-web.html>

(N/A), Wikipedia. (2019). *Visual Studio Code*. Recuperado de: https://es.wikipedia.org/wiki/Visual_Studio_Code

Navarrete, T (2007). *El lenguaje Javascript*. Recuperado de: <https://www.dtic.upf.edu/~tnavarrete/fcsig/javascript.pdf>

OBS. (2019). *OBS Business School*. Recuperado de: <https://www.obs-edu.com/int/blog-project-management/temas-actuales-de-project-management/te-conviene-utilizar-la-metodologia-scrum-en-tus-proyectos>

OCU. (2017). *Oficina de Cooperación Universitaria*. Recuperado de: <http://www.ocu.es/productos/universitas-xxi-academico/>

Palacio, J. (abril de 2014). *Scrum Manager*. Recuperado de: https://www.scrummanager.net/bok/images/8/84/Marco_estandar_scrum.png

Rivas, I. (2011). *Características de Aplicaciones Web*. Recuperado de <http://estudiantealdeunare3irmadj.blogspot.com/p/caracteristicas-de-aplicaciones-web.html>

Schwaber, K., & Sutherland, J (2008). *ScrumGuides*. Recuperado de: <https://www.scrumguides.org/scrum-guide.html>

Straub , B., & Chacon, S. (2014). *Pro Git*. Nueva York: Apress.

Universidad Central de Venezuela. (2017). *Normas sobre el rendimiento mínimo y condiciones de permanencia de los alumnos en la UCV*. Recuperado de <http://www.ucv.ve/organizacion/facultades/facultad-de-ciencias-economicas-y-sociales/escuelas/estudios-internacionales/reglamentos/normas-sobre-el-rendimiento-minimo-y-condiciones-de-permanencia-de-los-alumnos-en-la-ucv.html>

ANEXO A

En este anexo se presentan las tablas descriptivas de las especificaciones de los casos de uso definidos en el documento:

Tabla 3. Especificación - UC1 Inicio de sesión. Fuente: (Elaboración propia, 2019)

UC1 - Iniciar Sesión		
Descripción	Mecanismo para identificar al usuario según su documento de identidad y contraseña con el que se le concederá el acceso al sistema según su rol.	
Actor/es	Estudiante, Docente, Coordinador de Área y Administrador	
Dependencias	-	
Precondición	El usuario está registrado dentro del sistema.	
Flujo básico	Paso	Acción
	1	El actor ingresa en la página de inicio del sistema.
	2	El sistema muestra la página de inicio con la sección de inicio de sesión.
	3	El actor introduce la cédula de identidad y su contraseña. Presiona el botón de “Inicio de sesión”.
	4	El sistema valida los datos enviados por el actor y de existir coincidencia lo redirige al dashboard de usuario. El usuario sólo tiene un rol definido.
Flujos Alternos	Caso 1	Paso 4 - Usuario incorrecto.
		Paso Acción
		4 Usuario o contraseña incorrecta, el sistema niega el

			acceso al dashboard de usuario.
	Caso 2	Paso 4 - Usuario autenticado pero con más de un rol	
		Paso	Acción
		4	El sistema valida los datos enviados por el actor exitosamente y encuentra que tiene más de un rol disponible. Solicita al actor que seleccione el rol a utilizar.
		5	El actor ingresa el rol para la sesión
		6	El sistema según el rol seleccionado redirige al dashboard correspondiente.
Postcondición	El usuario posee un token de sesión, por lo que puede acceder al sistema		
Comentarios	-		

Tabla 4. Especificación - UC2. Inscribir semestre. Fuente: (Elaboración propia 2019).

UC2 – Inscribir semestre		
Descripción	Funcionalidad para permitirle al estudiante poder inscribir las materias que cursará el siguiente semestre	
Actor/es	Estudiante.	
Dependencias	● UC1	
Precondición	El usuario inició sesión dentro del sistema.	
Flujo básico	Paso	Acción
	1	El estudiante ingresa en la opción de “Inscribir Semestre”.

	2	El sistema muestra la página de la inscripción, con las materias que puede inscribir según su historial académico y las leyes universitarias.	
	3	El estudiante selecciona las materias que desea inscribir.	
	4	El sistema realiza la inscripción del estudiante y deshabilita que pueda volver a inscribirse para ese semestre.	
Flujos Alternos	Caso 1	Paso 2 - El estudiante no tiene permisos para inscribirse.	
		Paso	Acción
		2	El sistema niega la solicitud de inscripción.
	Caso 2	Paso 4 - Al menos una de las materias seleccionadas se quedó sin cupo.	
		Paso	Acción
		4	El sistema indica mensaje de error en la petición y retornar las materias que aún tiene disponible para inscribir.
		5	El estudiante vuelve a seleccionar las materias y envía la solicitud
		6	El sistema realiza la inscripción del estudiante y deshabilita que pueda volver a inscribirse para ese semestre.
Postcondición	El usuario queda inscrito en el semestre.		
Comentarios	Dentro del paso 2 se realizan todas las validaciones con respecto a las leyes descritas en el documento.		

Tabla 5. Especificación - UC3 Consultar horario. Fuente: (Elaboración propia 2019).

UC3 - Consultar horario		
Descripción	Funcionalidad que le permite al usuario visualizar su horario con las materias asignadas (según su rol), donde podrá consultar el código y nombre de la materia al igual que el aula en el que se imparte la clase.	
Actor/es	Estudiante, Docente, Coordinador de Área.	
Dependencias	<ul style="list-style-type: none"> ● UC1 	
Precondición	El usuario inició sesión dentro del sistema.	
Flujo básico	Paso	Acción
	1	El actor ingresa en la página de principal del dashboard del sistema.
	2	El sistema busca las materias según el rol de actor. En caso que sea estudiante buscará las inscritas. En caso que sea coordinador de área o docente buscará las que dicte. Luego mostrará un horario con dichas materias según la hora académica.
Flujos Alternos	-	
Postcondición	-	
Comentarios	-	

Tabla 6. Especificación – UC4 Retirar materia. Fuente: (Elaboración propia 2019).

UC4 – Retirar materia			
Descripción	Funcionalidad que le permite al usuario retirar una materia.		
Actor/es	Estudiante, Administrador.		
Dependencias	● UC1		
Precondición	El usuario inició sesión dentro del sistema.		
Flujo básico	Paso	Acción	
	1	El sistema muestra el dashboard de usuario luego de que haya iniciado sesión, en el están las materias inscritas y si tiene permisos y el periodo de retiro de materias está habilitado podrá ver la opción de retirarlas.	
	2	El actor selecciona la materia a retirar.	
	3	El sistema muestra un mensaje de alerta para confirmar el retiro de la materia.	
	4	El usuario confirma la acción.	
	5	El sistema guarda el retiro de la materia.	
Flujos Alternos	Caso 1	Paso 1 - El periodo de retiro no está habilitado o no tiene permisos de retirar materia en ese momento.	
		Paso	Acción
		1	El sistema muestra el dashboard de usuario luego de que haya iniciado sesión, en el están las materias inscritas sin la opción de retiro de sesión.
Postcondición	Si el usuario retira todas sus materias se evalúan las condiciones de las leyes		

	universitarias.
Comentarios	El administrador tendrá permisos para retirar materias en todo momento durante el semestre, pero el estudiante solo lo tendrá mientras se encuentre en el período de retiro de materias.

Tabla 7. Especificación – UC5 Consultar listado de estudiantes.

Fuente: (Elaboración propia 2019).

UC5 - Consultar listado de estudiantes		
Descripción	Permitir a los usuarios descargar la lista con los datos básicos de los estudiantes que ven la materia dicho usuario imparte.	
Actor/es	Docente y Coordinador de Área.	
Dependencias	<ul style="list-style-type: none"> ● UC1 	
Precondición	El usuario inició sesión dentro del sistema.	
Flujo básico	Paso	Acción
	1	El actor ingresa en la página del dashboard del sistema.
	2	El sistema muestra como dashboard principal la sección de listado de usuarios con enlace a los archivos descargables del listado de estudiantes según la materia.
	3	El actor selecciona el listado que desea generar
	4	El sistema comienza la descarga del archivo seleccionado
Flujos Alternos	-	
Postcondición	El usuario obtiene el listado de los estudiantes inscritos en su materia en un archivo con formato Excel.	
Comentarios	-	

Tabla 8. Especificación – UC6 Administrar docentes. Fuente: (Elaboración propia 2019).

UC6 – Administrar docentes		
Descripción	Funcionalidad que permite administrar los docentes que se encuentran registrados en el sistema, en el se implementa un CRUD (Crear, Leer, actualizar y borrar) de usuarios para que estos puedan ser agregados en cualquier momento.	
Actor/es	Administrador, Coordinador de Área	
Dependencias	<ul style="list-style-type: none"> ● UC1 	
Precondición	El usuario inició sesión dentro del sistema.	
Flujo básico	Paso	Acción
	1	El usuario ingresa a la sección de docentes
	2	El sistema muestra una página con la opción de agregar nuevos usuarios, además también trae un cuadro de los docentes disponibles y un menú de opciones para modificar su información básica, cambiar su rol, otra para asignarle una materia y borrar el usuario.
	3	El usuario selecciona la opción correspondiente, llena los datos en caso de ser necesario y confirma la operación.
	4	El sistema actualiza o agrega al docente.
Flujos Alternos	-	
Postcondición	-	
Comentarios	El flujo de asignar materia se comenta en el UC9	

Tabla 9 Especificación – UC7 Cargar nuevos estudiantes. Fuente: (Elaboración propia 2019).

UC7 – Cargar nuevos estudiantes		
Descripción	<p>Funcionalidad que permite cargar nuevos estudiantes por lotes para que se puedan inscribir antes del inicio de un semestre. Se recomienda utilizar en los siguientes casos:</p> <ul style="list-style-type: none"> • Primeras cargas de estudiantes para poblar bases de datos. • Estudiantes nuevo ingresos. 	
Actor/es	Administrador	
Dependencias	<ul style="list-style-type: none"> ● UC1 	
Precondición	<p>- El usuario inició sesión dentro del sistema.</p> <p>- El usuario debe tener un Excel con la nueva data de los estudiantes en el formato establecido por el sistema.</p>	
Flujo básico	Paso	Acción
	1	El usuario ingresa a la sección de estudiantes
	2	El sistema carga la página de estudiantes, esta contendrá un botón para agregar estudiante, uno para agregar estudiantes en lote, una tabla con los datos básicos de los estudiantes y una barra de búsqueda.
	3	El usuario selecciona la opción de agregar estudiantes en lote y agrega el archivo Excel con la data de los estudiantes.
	4	El sistema procesa el archivo y agrega los nuevos estudiantes.
Flujos Alternos	-	
Postcondición	<p>- Se genera un informe indicando el número de éxitos y los registros tuvieron errores.</p>	

Comentarios	-
--------------------	---

Tabla 10. Especificación – UC8 Cargar Oferta Docente.

Fuente: (Elaboración propia 2019).

UC8 - Cargar Oferta Docente			
Descripción	Funcionalidad que permite la carga de la Oferta Docente del semestre antes de realizar las inscripciones		
Actor/es	Administrador		
Dependencias	● UC1		
Precondición	El usuario inició sesión dentro del sistema.		
Flujo básico	Paso	Acción	
	1	El actor ingresa a la opción del menú “Crear nuevo semestre”.	
	2	El sistema muestra una página donde se encontrarán los formatos requeridos para montar la planificación del semestre. A su vez también incluye una sección para cargar el archivo que contiene la planificación.	
	3	El actor sube al sistema la planificación.	
	4	El sistema verifica el formato del archivo y procede a crear las materias para el nuevo semestre según lo descrito en la planificación. Luego crea la inscripción del nuevo semestre según los plazos establecidos.	
Flujos Alternos	Caso 1	Paso 4 - El formato del archivo es incorrecto	
		Paso	Acción
		2	El sistema niega la solicitud de creación del semestre y muestra el motivo por el que fue rechazada.

Postcondición	La inscripción del semestre se habilita según lo especificado en el archivo de planificación.
Comentarios	-

Tabla 11 Especificación – UC9. Asignar materias a docentes.

Fuente: (Elaboración propia 2019).

UC9 – Asignar materias a docentes		
Descripción	Funcionalidad extendida de del UC6, en la que se le asigna a las materias a los docentes.	
Actor/es	Administrador, Coordinador de Área	
Dependencias	<ul style="list-style-type: none"> ● UC1 	
Precondición	<ul style="list-style-type: none"> - El usuario inició sesión dentro del sistema. - El usuario accedió a la sección de docentes. 	
Flujo básico	Paso	Acción
	1	El usuario selecciona la opción de asignar materias a docentes.
	2	El sistema carga un formulario que contiene un listado de docentes, un listado de materias, y la sección de la materia
	3	El usuario agrega la información del formulario y confirma la operación.
	4	El sistema procesa la operación y se le guarda la asignación del docente.
Flujos Alternos	-	
Postcondición	-	
Comentarios	-	