

Aplicación Web para la Gestión de Materiales de Oficina de la Universidad de la Sierra Sur

Web Application for Office Material Management at the University of the Sierra Sur

Vicente de Jesús Méndez-Bautista^a, Everardo de Jesús Pacheco-Antonio^a, Rolando Pedro-Gabriel^a

Resumen

Las necesidades que afrontan las diversas instituciones educativas, empresas comerciales y cualquier otra dependencia que centran sus esfuerzos en la recolección de información diaria, requieren de sistemas de información que permitan agilizar estos procesos e incrementar la productividad de los recursos humanos. En este documento se presenta el desarrollo de una aplicación web que cubre la necesidad de solicitudes de material de oficina por parte de los profesores de la Universidad de la Sierra Sur. El proyecto se fundamenta en la metodología de desarrollo rápido de aplicaciones (RAD) y SCRUM, así mismo, el diseño de la arquitectura se basa en el patrón MVC. La codificación del proyecto involucra diversas herramientas de desarrollo, principalmente, el framework Spring Boot para codificar las reglas de negocio, AngularJS para la construcción de la capa de presentación y el Sistema Gestor de Bases de Datos MySQL para la persistencia de los datos.

Palabras clave: sistemas de información, aplicación web, RAD, SCRUM, MVC, framework, Spring Boot, AngularJS, MySQL.

Abstract

Several educational institutes, commercial businesses, and other dependencies focus their efforts on collecting information daily. Their needs require information systems that allow these processes to be streamlined and increase the productivity of human resources. This document presents the development of a web application that covers the need for requests for office supplies by professors of the University of the Southern Highlands. The project is based on the Rapid Application Development methodology (RAD) and SCRUM. Additionally, the architecture design is based on the MVC pattern. The codification of the project involves several development tools, mainly, the Spring Boot framework for coding business rules, Angular JS to build the presentation layer and the Database Management System MySQL for data persistence.

Keywords: information system, web application, RAD, SCRUM, MVC, framework, Spring Boot, Angular JS, MySQL.

^a Universidad de la Sierra Sur, Instituto de Informática, Guillermo Rojas Mijangos s/n, esq. Av. Universidad, C.P. 70800, Miahuatlán de Porfirio Díaz, Oaxaca, México.

Correspondencia: Rolando Pedro Gabriel
Universidad de la Sierra Sur, Instituto de Informática
Correo electrónico: rolando.pedro.gabriel@gmail.com

Introducción

Actualmente, el abanico de herramientas que existen en el área de las tecnologías de la información (TI), son diversas, tanto para el desarrollo de software de sistemas, así como, para el desarrollo de software de aplicación, que es en la que se centra y describe este documento. En el ecosistema de esta área, es posible hallar una multitud de lenguajes de programación y frameworks que facilitan aún más el desarrollo de aplicaciones. No obstante, únicamente algunas de estas herramientas se posicionan en la cúspide, tanto por su uso como por su longevidad. Para seleccionar las herramientas adecuadas que permitan la construcción de una arquitectura sólida del proyecto de software desde sus inicios, es importante realizar una evaluación exhaustiva de las características que poseen cada uno de estos frameworks y la facilidad de adaptarse al contexto de un problema en específico. Así mismo, es importante considerar la curva de aprendizaje que implica para los desarrolladores que se están iniciando en esta temática, ya que se requiere de cierta madurez, en cuanto a la experiencia de uso y adaptación en equipo. Kendall y Kendall (2011) consideran la importancia de dirigir un proyecto de desarrollo de software, bajo lineamientos bien establecidos, identificando claramente el tipo de sistema que se desea desarrollar y posteriormente llevar a cabo el proceso de análisis y diseño del sistema, proceso que persigue la comprensión del usuario final para analizar la entrada o el flujo de datos de manera sistemática.

Problemática

La captura de información diaria es una actividad que se realiza en diversas instituciones, donde figuran las áreas de educación, comercio, publicidad, entre otras. El caso específico que se presenta en este documento es la solicitud de material de oficina por parte de los profesores de la Universidad de la Sierra Sur. Esta actividad, al llevarse a cabo manualmente

requiere del consumo de papel, tinta para la impresión de las papeletas de solicitud que realizan los profesores y disponibilidad de tiempo para repartir y posteriormente recolectar cada una de las papeletas que cada profesor registra manualmente. Al efectuar la entrega de material de oficina por profesor, se deberá corroborar con su respectiva papeleta. Por lo cual, se observó la naturaleza de este mecanismo y se optó por desarrollar una aplicación web para sistematizar y agilizar este procedimiento dentro de la institución.

Metodología

Debido a que el desarrollo de proyecto de software presenta ciertas diferencias en cuanto a otros proyectos, en esta área existen diversas metodologías que proveen características muy puntuales, que se dividen en dos grandes grupos; metodologías tradicionales y metodologías ágiles. La gran mayoría de los proyectos de desarrollo de software de pequeña y mediana escala son dirigidos bajo los lineamientos de metodologías ágiles. Kendall y Kendall (2011) mencionan que los analistas de sistemas no han acordado puntualizar la cantidad de fases que se requieren para el análisis y diseño de sistemas, sin embargo, la gran mayoría sigue un esquema de siete fases, como se muestra en la Figura 1.



Figura 1. Las siete fases del ciclo de desarrollo de sistemas.

Fuente: Kendall y Kendall (2011).

La ejecución del proyecto que se expone en este documento sigue el esquema de las siete fases y se basa en los estatutos de las metodologías ágiles, específicamente SCRUM.

Identificación de los problemas, oportunidades y objetivos

La identificación de necesidades es un proceso natural que el ser humano realiza diariamente, en cualquier entorno en el que

se encuentra, en el hogar, en el trabajo, en la estancia vacacional, en el viaje, etc., buscando siempre la agilización y la sistematización de dicha necesidad. En este proyecto se observa la necesidad de implementar una aplicación web con el objetivo de agilizar y sistematizar el proceso de solicitud de material de oficina dentro del Instituto de Informática de la Universidad de la Sierra Sur, aprovechando las bondades del software libre, servicios y conocimientos que provee el área de informática dentro de la universidad (Figura 2).

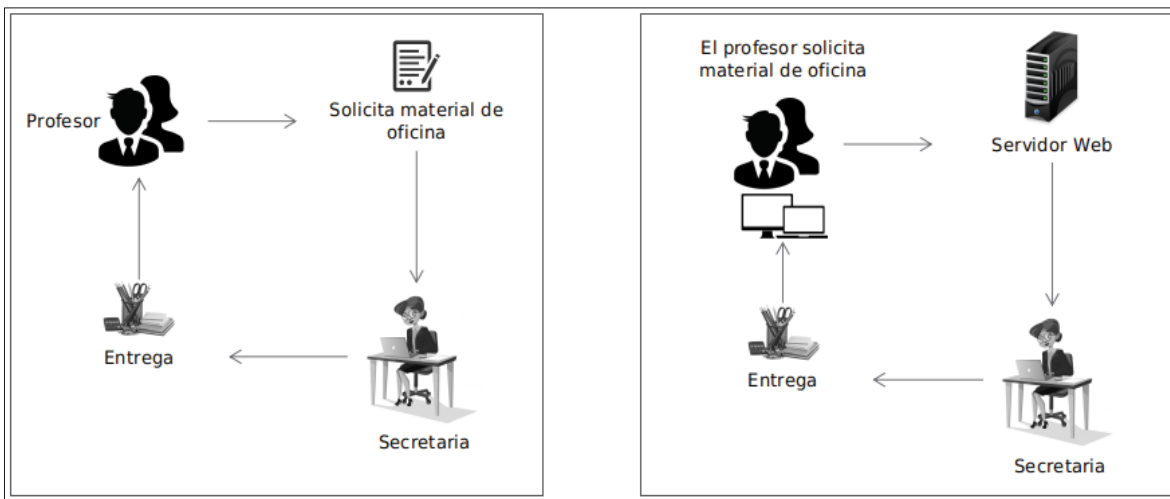


Figura 2. Solicitud de materiales.
Fuente: Elaboración propia.

Determinación de los requerimientos humanos de información

La recolección de requisitos que describen la funcionalidad del presente proyecto de software, fueron extraídos mediante un proceso práctico y rápido, utilizando entrevistas y cuestionarios. La finalidad de esta etapa es formular un documento de especificación de requisitos ERS sólido y validado previamente por el analista y los usuarios finales. En este documento se establecieron los requerimientos funcionales, no funcionales y de sistema. Pressman (2010) considera que en esta etapa se debe tener especial cuidado ya que en este documento se plasma las funcionalidades

intrínsecas del sistema final, además, en este punto se deberán plantear y responder preguntas que involucran la usabilidad del sistema.

Análisis de las necesidades del sistema

Este proceso requiere de concentración y planeación, ya que es fundamental para la construcción del proyecto. En este apartado se genera la especificación de requisitos, documento en el que se establecen las funcionalidades que deberá realizar el sistema.

El sistema de solicitudes de materiales de oficina que aquí se presenta, proveerá un mejor

control, así mismo, deberá ser fácil de manejar; el control consiste en realizar el registro de los materiales de oficina que cada profesor solicita. El sistema registrará las operacio-

nes de altas, bajas, modificaciones, reportes y administración de usuarios. En la Figura 3, se muestra a través de un diagrama una visión general del sistema y su funcionalidad.

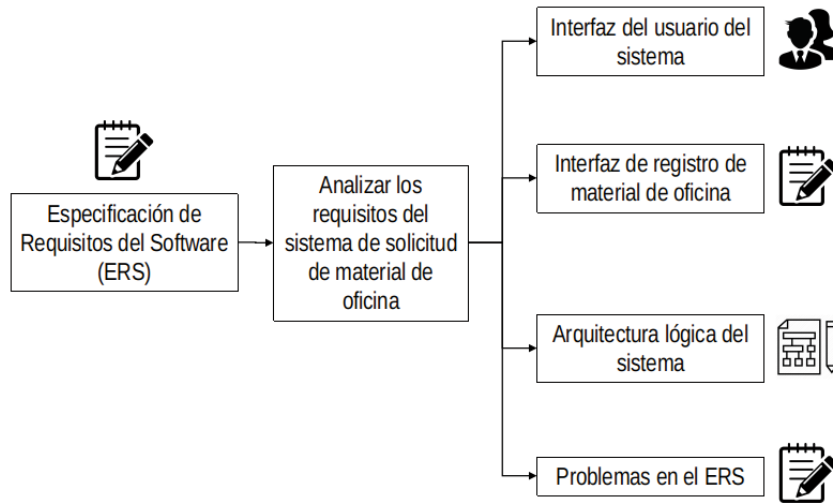


Figura 3. Diagrama para mostrar una visión general del sistema.
Fuente: Elaboración propia.

Diseño del sistema recomendado

A partir de los requisitos obtenidos en la fase previa se diseñan los prototipos de baja fidelidad que representan la funcionalidad e interfaces de usuario del sistema en general. Con la ayuda de estos prototipos se facilitó la validación de entrada de datos al sistema. Gamma, Helm, Johnson y Vlissides (2003) mencionan que las interfaces de usuario se diseñan con ayuda de usuarios para asegurar que el sistema sea perceptible, legible y seguro, además, que presenten características atractivas y divertidas al utilizarse.

A continuación se muestra un diagrama que representa el flujo general del proceso de solicitud de material, sin escatimar la autenticación del usuario.

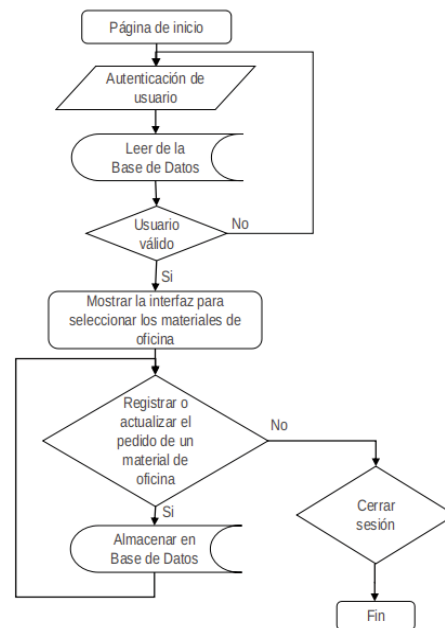


Figura 4. Diagrama para mostrar el flujo general de solicitud de material de oficina.
Fuente: Elaboración propia.

Desarrollo y documentación del software

El desarrollo del proyecto web se llevó a cabo mediante herramientas sistematizadas, desde la fase de diseño hasta la fase de despliegue, monitoreando frecuentemente algunas características como la funcionalidad y la usabilidad. La codificación se llevó a cabo mediante el lenguaje de programación Java usando el framework Spring Boot, en el que se implementaron servicios REST. Cada uno de

los servicios codificados fueron sometidos a prueba mediante peticiones POST, GET, PUT y DELETE con ayuda de la herramienta Postman.

Con ayuda de la herramienta Gradle se llevó a cabo la automatización de la construcción del código del proyecto gradualmente, y facilitó la gestión de dependencias. Otras de las herramientas importantes a resaltar es el sistema de control de versiones Git, en la Figura 5 se muestra un esquema gráfico de la arquitectura de desarrollo del proyecto.

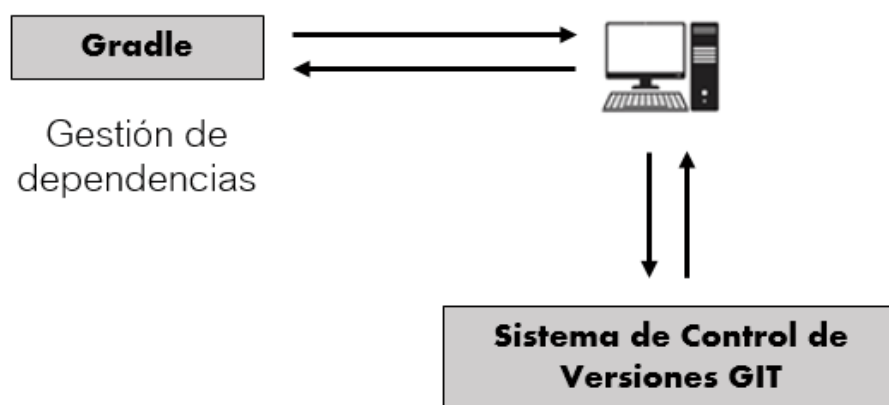


Figura 5. Construcción de proyecto.

Fuente: Elaboración propia.

Pruebas y mantenimiento del sistema

Las pruebas son técnicas que permiten aumentar la calidad del producto, y mantener el proceso de integración de manera funcional y certera. Jorgensen (2013) describe la importancia de las pruebas del producto de software, ya que hay dos razones fundamentales para llevarlas a cabo; primero para considerar el producto de calidad o la aceptabilidad y segundo para hallar los errores. Los errores tienden a propagarse, cometer un error en la fase de requerimientos puede propagarse a la fase de diseño y amplificarse en la fase de codificación y finalmente producir una falla en el producto, por lo tanto, una falla es una representación del error.

Implementación y evaluación del sistema

El despliegue de la aplicación web se realizó en un contenedor de aplicaciones Apache Tomcat en un servidor Linux a través de una intranet de la universidad. Los resultados obtenidos después de la implementación fue la aceptación por parte del usuario final.

Arquitectura de la aplicación y herramientas

El presente proyecto ostenta una arquitectura de desarrollo como el que se muestra en la Figura 6. En esta, se pueden apreciar tres

capas importantes que conforman la arquitectura Modelo – Vista – Controlador (MVC). El modelo y el controlador se presenta en la capa de codificación de las reglas de negocio,

mientras que la vista se presenta en la capa de presentación, finalmente, la persistencia de datos se lleva a cabo mediante el Sistema Gestor de Bases de Datos MySQL.

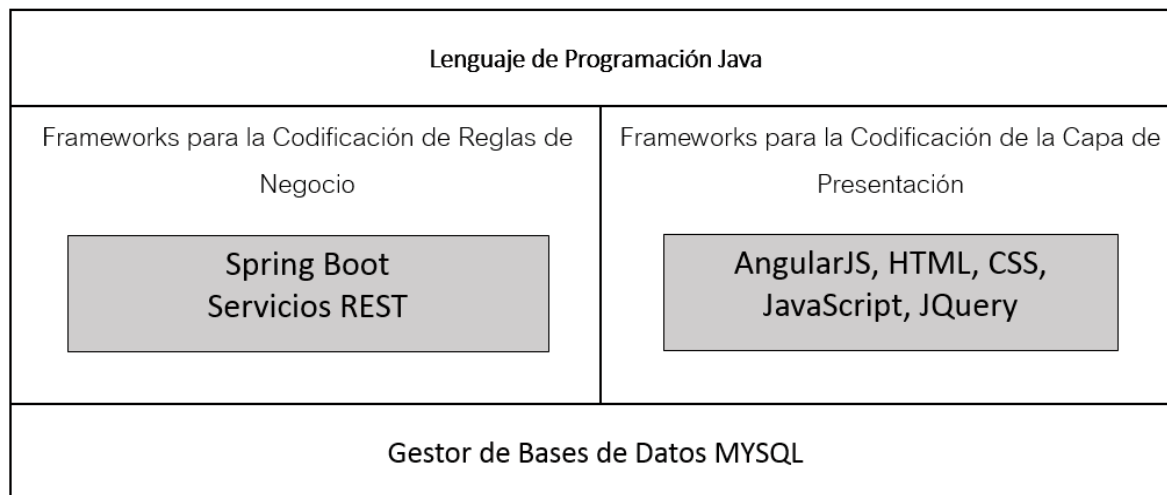


Figura 6. Arquitectura de la aplicación.

Fuente: Elaboración propia.

Con base al juicio experto se optó por construir una arquitectura de desarrollo utilizando las siguientes herramientas; el framework Spring Boot se empleó en la codificación de las reglas de negocio, Angular JS para la codificación de la capa de presentación, Gradle se utilizó como herramienta de construcción del código fuente y la gestión de dependencias, Git para mantener las diferentes versiones del proyecto en general, así mismo, se utilizaron algunos frameworks en la estilización de la capa de presentación como; AngularJS, Bootstrap 3, JQuery, CSS, HTML5 y JavaScript.

Para llevar a cabo la elección del framework que permite la codificación de la capa de la lógica de negocio y la de presentación de la aplicación, se realizaron algunas comparativas, tal como lo muestra Sánchez, Tuesta y Mejía (2015) en su análisis comparativo en cuanto a los frameworks, Spring, Struts, JSF y Angular. Finalmente, estos autores mencionan que Spring es el framework más popular y más usado en la industria del desarrollo de software, respecto a los frameworks web java.

Tabla 1. Análisis comparativo entre los frameworks Spring, Struts, JSF y AngularJS.

Framework	Año	Documentación	Funcionalidad	Usabilidad
Spring	2004	Mucha	Alta	Alta
Struts	2001	Poca	Alta	Media
JFS	2004	Poca	Media	Media
Angular	2009	Mucha	Alta	Alta

Fuente: Elaboración propia.

Spring Boot

Walls (2014) menciona que Spring Boot es uno de los frameworks más utilizados del lenguaje de programación Java y se encuentra dentro del abanico de proyectos de Spring. Sadakath (2018) considera que Spring Boot es un framework para el desarrollo de aplicaciones que sigue el paradigma de desarrollo rápido (RAD), centrándose en la codificación de la lógica de negocio. Spring Boot presenta una rápida creación del proyecto inicial, mediante un entorno de desarrollo integrado (IDE) o desde la página oficial <https://start.spring.io/> donde se pueden configurar algunas librerías como Java Persistence Application (JPA), el conector de la base de datos, entre otros.

AngularJS

Uno de los frameworks más completo y versátil para el desarrollo de aplicaciones tanto web como móviles del lado del servidor, así como para resaltar las etiquetas HTML en el navegador es AngularJS, framework desarrollado por Google. Freeman (2018) menciona que mediante el uso de este framework se construyen aplicaciones bajo el patrón de diseño llamado Modelo-Vista-Controlador (MVC), enfatizando la creación de aplicaciones que son extensibles, mantenible, testeable, y estandarizados, además, es una librería de JavaScript de Código Abierto. Mientras que Arora & Hennessy (2018) mencionan que AngularJS se basa también en otro modelo de diseño, el patrón de componentes, ya que abstrae los conceptos del mundo real, en donde se puede observar que cada objeto se compone de elementos más pequeños, y en algunos casos si esos componentes llegasen a fallar, el sistema puede mantener su funcionalidad reemplazando dicho componente por otro.

Gradle

El desarrollo de proyectos de software involucra una serie de pasos secuenciales y a la vez

repetitivas, por lo tanto, realizarlos de forma manual requiere de disponibilidad de tiempo, no obstante, afortunadamente en la actualidad existen herramientas que facilitan estos procedimientos. Ikkink (2016) menciona las etapas que pueden automatizarse mediante el uso de sistemas de construcción tales como Ant, Maven y Gradle. Las etapas para automatizar son la codificación, la compilación, las pruebas, el empaquetado, y finalmente la distribución del código.

SCRUM

El desarrollo de proyectos de software de manera tradicional conllevaría a un alto consumo de recursos, y con certeza se dirige a la culminación de un proyecto incierto. Hoy en día existe un abanico de herramientas que agilizan los procesos de construcción de proyectos de software. Esto sin descuidar, la metodología como columna vertebral de proceso de desarrollo. La metodología más demandada para el proceso de desarrollo de proyecto de software ágil es SCRUM. Bahit (2012) menciona que esta permite gestionar la construcción de proyectos relacionados con el desarrollo de software.

Conclusiones

Al implementar la aplicación web para la gestión de solicitudes de material de oficina por parte de los profesores de la Universidad de la Sierra Sur, se sustituyó el proceso de elaboración, impresión y repartición de papeletas, así mismo, al realizar las solicitudes mediante una computadora se redujo de manera significativa el tiempo que se utiliza para hacer la recolección de toda la información necesaria para generar la solicitud mensual, esto debido a que anteriormente el proceso se realizaba mediante papeletas impresas, los cálculos para obtener los totales de materiales a solicitar se realizaban manualmente y se tenía que esperar a que cada profesor terminara de realizar su solicitud de forma

escrita, esto demoraba el proceso 3 o hasta 5 días después de recabar todas las papeletas, actualmente el personal encargado de esta actividad únicamente monitorea los registros desde la aplicación web y el reporte de solicitudes se genera en 2 segundos, además se realiza la repartición de material con base a la información ingresada al sistema.

Actualmente la aplicación web se está utilizando por 24 profesores que pertenecen al Instituto de Informática. Se planea agregar los distintos institutos que se tienen en la universidad y así, tener el control de todo el material que se solicita.

Por otra parte, se logró tener un historial de las solicitudes de material y automáticamente realizar los cálculos que permiten generar la solicitud mensual evitando así, errores causados por la caligrafía o por la lectura/escritura rápida por parte del personal encargado de esta actividad.

La importancia de la automatización de ciertas actividades que ejerce una institución educativa, empresarial, publicidad, entre otras, ayudan a fortalecer el nivel de organización y la consistencia de la información. Desde la implantación de la aplicación web para llevar a cabo la solicitud de material de oficina, se ha observado que el proceso se ha llevado a cabo con mayor facilidad, y se tiene el control de la información digitalizada en las bases de datos.

De acuerdo con los resultados obtenidos en la implantación de la aplicación se puede concluir que resultaría factible utilizar este software en todos los institutos de la institución para agilizar el proceso y llevar un control automatizado de las solicitudes.

Por otra parte, la interacción entre las herramientas que se utilizaron para la construcción de la aplicación web que aquí se describe, se acoplaron adecuadamente; por un lado, el framework Spring Boot gestiona las reglas de negocio para llevar a cabo la

solicitud de material de oficina, así como las operaciones básicas del módulo de usuarios (altas, bajas y actualizaciones). Para la capa de presentación, donde interactúa directamente el usuario final, se utilizó el framework AngularJS, estableciendo el enlace de comunicación con Spring Boot y éste a su vez, establece la comunicación bilateral con la base de datos.

Finalmente, se exponen algunas imágenes que representan las interfaces de usuario de la aplicación web que se aborda en este documento. En la Figura 7 se muestra la interfaz de usuario para ingresar al sistema, posteriormente, en la Figura 8 se pueden visualizar la lista de profesores que solicitan material y el estatus en el que se encuentra dicho proceso. Así mismo, en la Figura 9 se muestra una interfaz de usuario administrador para generar el reporte mensual de las solicitudes de material realizadas por los profesores. Finalmente en la Figura 10 se observa un ejemplo de la solicitud de materiales de un profesor.

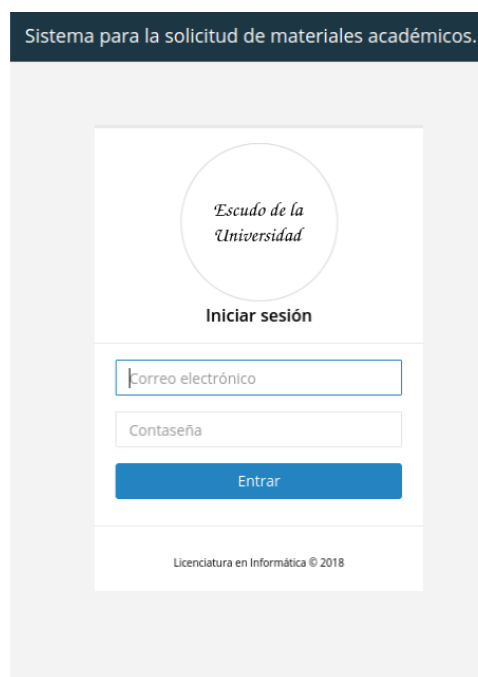


Figura 7. Login del sistema.
Fuente: Elaboración propia.

Solicitudes del mes actual

Mostrar 10 Registros Buscar:

Nombre del solicitante	Programa Educativo	Estatus	Detalle
Profesor 1	LICENCIATURA EN INFORMÁTICA	EN PROCESO	Ver
Profesor 2	LICENCIATURA EN INFORMÁTICA	EN PROCESO	Ver
Profesor 3	LICENCIATURA EN INFORMÁTICA	EN PROCESO	Ver
Profesor 4	LICENCIATURA EN INFORMÁTICA	EN PROCESO	Ver
Profesor 5	LICENCIATURA EN INFORMÁTICA	EN PROCESO	Ver
Profesor 6	LICENCIATURA EN INFORMÁTICA	EN PROCESO	Ver
Profesor 7	LICENCIATURA EN INFORMÁTICA	EN PROCESO	Ver
Profesor 8	LICENCIATURA EN INFORMÁTICA	EN PROCESO	Ver
Profesor 9	LICENCIATURA EN INFORMÁTICA	EN PROCESO	Ver

Mostrando 1 a 9 de un total de 9 registros

Primero << 1 >> Último

Figura 8. Pantalla de solicitudes de materiales.
Fuente: Elaboración propia.

Reportes de Materiales Solicitados

Tipo de reporte:

Licenciatura:

Año reporte:

Mes reporte:

[Generar PDF](#) [Generar Word](#)

Figura 9. Pantalla para generar reporte mensual.
Fuente: Elaboración propia.

Escudo de la Universidad
Nombre de la Universidad

Hola Profesor 1

Lista de materiales solicitados en este mes

+ Agregar Editar Eliminar Borrar Todos

Solicitud en progreso. Tienes hasta el 25 de este mes para enviar tu solicitud.

64% Completado...

Mostrar 10 Registros Buscar: Buscar

Material solicitado	Presentación	Cantidad solicitada	Detalles	Fecha solicitud	Estatus de la Solicitud
BOLÍGRAFO TINTA NEGRA	PIEZA	4	Bic	Aug 1, 2019 4:43:32 PM	EN PROCESO
LIBRETA FRANCESA DE RAYAS	PIEZA	1		Aug 1, 2019 4:43:45 PM	EN PROCESO
LIBRETA PROFESIONAL DE RAYAS	PIEZA	1		Aug 1, 2019 4:44:12 PM	EN PROCESO
PROTECTOR DE HOJAS	PIEZA	10		Aug 5, 2019 5:29:03 PM	EN PROCESO

Figura 10. Pantalla solicitud de profesor.
Fuente: Elaboración propia.

Escudo de la Universidad
Nombre de la Universidad

Hola Usuario Administrador

Profesores investigadores

+ Agregar Editar Eliminar

Mostrar 50 Registros Buscar: Buscar

Nombre	Email	Programa Educativo	Instituto	Cargo	Estatus
Profesor 1	Correo profesor 1	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	INACTIVO
Profesor 2	Correo profesor 2	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	INACTIVO
Profesor 3	Correo profesor 3	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	INACTIVO
Profesor 4	Correo profesor 4	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	INACTIVO
Profesor 5	Correo profesor 5	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	INACTIVO
Profesor 6	Correo profesor 6	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	INACTIVO
Profesor 7	Correo profesor 7	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 8	Correo profesor 8	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 9	Correo profesor 9	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 10	Correo profesor 10	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 11	Correo profesor 11	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 12	Correo profesor 12	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 13	Correo profesor 13	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 14	Correo profesor 14	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	JEFE DE CARRERA	ACTIVO
Profesor 15	Correo profesor 15	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 16	Correo profesor 16	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 17	Correo profesor 17	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 18	Correo profesor 18	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 19	Correo profesor 19	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 20	Correo profesor 20	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 21	Correo profesor 21	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 22	Correo profesor 22	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	PROFESOR INVESTIGADOR	ACTIVO
Profesor 23	Correo profesor 23	LICENCIATURA EN INFORMÁTICA	INSTITUTO DE INFORMÁTICA	DIRECTOR DE INSTITUTO	ACTIVO

Figura 11. Pantalla usuarios del sistema.
Fuente: Elaboración propia.

Referencias

- Arora, C., y Hennessy, K. (2018). *Angular 6 by Example: Get up and running with Angular by building modern real-world web apps*. United Kingdom: Packt Publishing.
- Bahit, E. (2012). *Scrum y eXtreme Programming para Programadores*. Argentina: Safe Creative. Disponible en: <https://www.yumpu.com/es/document/read/25629200/ug931q/3>
- Freeman, A. (2018). *Pro Angular 6*. 3rd ed. United Kingdom: Apress. Disponible en: https://www.academia.edu/38617677/Pro_Angular_6_-_2018_-_Third_Edition
- Gamma, E., Helm, R., Johnson, R., y Vlissides, J. (2003). *Patrones de diseño: elementos de software orientado a objetos reutilizables*, Madrid: Pearson Educacion. Disponible en: https://kupdf.net/download/patrones-de-dise-ntilde-o-gamma-helm-johnson-vlissides_58c04e29e12e89b00fadd375_pdf
- Ikink, H.K. (2016). *Gradle Effective Implementations Guide*. 2nd ed. United Kingdom: Packt Publishing. Disponible en: <https://gradle.org/books/packt-gradle-effective-implementations-guide.pdf>
- Jorgensen, P.C. (2013). *Software testing: a craftsman's approach*. 4th ed. United State of America: Auerbach Publications. Disponible en: <http://webpages.iust.ac.ir/azgomi/Courses/ST/eBook2%20-%20Software%20Testing%20-%20A%20Craftsman,s%20Approach.pdf>
- Kendall, K.E., y Kendall, J.E. (2011). *Análisis y Diseño de Sistemas*. 8a ed. México: Prentice Hall. Disponible en: https://www.academia.edu/7102592/Analisis_y_Diseño_de_Sistemas_8ed_Kendall_PDF
- Pressman, R.S. (2010). *Ingeniería de Software: un enfoque práctico*. 7a ed. México: McGrawHill. Disponible en: <http://cotana.informatica.edu.bo/downloads/ld-Ingenieria.de.software.enfoque.practico.7ed.Pressman.PDF>
- Sadakath, M.S. (2018). *Spring Boot 2.0 Projects: Build production-grade reactive applications and microservices with Spring Boot*. United Kingdom: Packt Publishing.
- Sánchez, A.C., Tuesta, M.V., y Mejía, C.I. (2015). *Análisis comparativo de frameworks para el desarrollo de aplicaciones web en java*, 1(2), 60-72. Disponible en: [file:///home/milan/Downloads/101-Texto%20del%20artículo-377-2-10-20151204%20\(1\).pdf](file:///home/milan/Downloads/101-Texto%20del%20artículo-377-2-10-20151204%20(1).pdf)
- Walls, C. (2014). *Spring*. 3a ed. Madrid: Anaya multimedia.
- Recibido:** 16 de agosto de 2019
Corregido: 17 de octubre de 2019
Aceptado: 04 de noviembre de 2019
- Conflicto de interés:** No existe conflicto de interés