

CAPSTONE PROJECT

I. INFORMACIÓN GENERAL

CURSO	:	Aplicaciones Open-Source
CÓDIGO	:	SI652
CICLO	:	202001
SECCIONES	:	Todas
CRÉDITOS	:	3
SEMANAS	:	16
HORAS SEMANALES	:	2 horas de teoría, 2 horas de laboratorio
ÁREA O CARRERA	:	Computación e Informática

II. LOGRO DEL CURSO

Al finalizar el curso, el estudiante desarrolla aplicaciones web que satisfacen requisitos para empresas o público en general sobre plataformas Server Side o Cloud, haciendo uso de tecnologías open source basadas en lenguaje de programación Java en un ambiente de desarrollo ágil.

En Ingeniería de Software, el logro contribuye a alcanzar el:

ABET Student Outcome (3): Capacidad de comunicarse efectivamente con un rango de audiencias.

III. OBJETIVO DEL TRABAJO

El curso de Aplicaciones Open-Source tiene una naturaleza teórico práctica, por lo que es necesario evidenciar la capacidad para desarrollar aplicaciones web aplicando los conceptos, técnicas y buenas prácticas impartidos en el curso.

Este trabajo tiene por objetivo desarrollar un producto de software que brinde soporte a un modelo de negocio independiente y sostenible. La aplicación debe contar con una interfaz web adaptable a las dimensiones del dispositivo, e incluir las características que permitan satisfacer las necesidades identificadas para los segmentos objetivo, haciendo uso de las herramientas y tecnología utilizadas en el curso, aplicando conceptos y técnicas revisados en clase ó resultado de la investigación.

Por otro lado, la arquitectura de la solución debe estar conformada por un RESTful API de elaboración interna y una Web Application integrada con dicho API, con una interfaz adaptable a las dimensiones de visualización de los dispositivos cliente. Se incluye como parte del alcance el Landing Page del modelo de negocio. Como parte de las características a considerar, además de la integración el servicio RESTful de desarrollo interno, debe acceder a algún servicio externo de terceros.

IV. COMPONENTES

Vision

Consta del enunciado de problema, y una descripción de los puntos más importante que debe resolver la solución propuesta, así como objetivos y restricciones que delimiten el alcance del proyecto. Abarca la visión del modelo de negocio que será soportado por el producto de software, incluyendo Problem Statements (incluyendo aspectos como domain, customer segments, pain points, gap, visión/strategy, e initial segment), Assumptions e Hypothesis según Lean UX Process. Se incluye el Lean UX Canvas.

Research

Se incluye el proceso de Needfinding junto con análisis de la competencia. Las entrevistas se registrarán en video y se editarán para construir el video de evidencia de entrevistas. El análisis de dichas entrevistas servirá de base para la identificación de necesidades y la construcción de los User Persona para cada segmento objetivo, así como la construcción del User Task Matrix y los User Journey Map para los User Persona identificados.

Product UX Design

Propuesta de UX para web browsers considerando tanto Desktop como Mobile UI. Para ello se tomará como base el conjunto de User Stories identificados así como el Impact Map. Se incluye los User Flow Diagrams y Wireframes para UI Desktop y Mobile que formen parte del alcance. Se incluye además un Wireframe del Landing Page.

Prototyping

Prototipos de UI para Desktop y Mobile con simulación de interacción y navegación, acorde con la propuesta de paths de User Flow Diagrams.

Server Side Service

Desarrollar y publicar un server side service cuya funcionalidad sea expuesta a través de un RESTFul API accesible desde internet, incluyendo vistas administrativas para consulta y actualización de la información.

Frontend Web Application

Desarrollar una aplicación web que incluya entre sus características la integración con el servicio RESTFul desarrollado y sea compatible con Desktop y Mobile web browsers, considerando heurísticas de usabilidad y accesibilidad.

Product Landing Page

Desarrollar un landing page que aplique responsive web design y permita presentar el modelo de negocio.

Validation

Se debe realizar entrevistas de validación en las que usuarios de los segmentos objetivo interactúen con el landing page y las aplicaciones, con los prototipos y versiones preliminares de experiencia web para desktop y mobile.

Ethics & Responsibility

Se debe redactar los términos y condiciones de servicio, evidenciando responsabilidad ética y profesional, según los principios del código de ética de software engineering de ACM/IEEE y del CIP. Del mismo modo se debe evidenciar transparencia y responsabilidad profesional en el proceso del ciclo de vida del producto, evidenciando en video el proceso de ingeniería de software que sigue el equipo, así como la aplicación de herramientas y servicios para el desarrollo colaborativo como el control de versiones den GitHub.

Objetivo del Estudiante (Student Outcome)

Cada participante del equipo debe sustentar evidencia de las actividades realizadas en el trabajo final han ayudado a desarrollar cómo las dimensiones del student outcome. Por ello en esta sección debe haber una subsección por cada alumno donde éste describa por escrito la relación entre el outcome, sus dimensiones y el trabajo que ha realizado. Esto se complementa con lo reflejado en los testimonios expuestos que forman parte del video *About The Team*.

A manera de referencia se incluye los criterios o aspectos específicos que corresponden al **ABET – EAC - Student Outcome 3**.

Comunica oralmente sus ideas y/o resultados con objetividad a público de diferentes especialidades y niveles jerárquicos, en el marco del desarrollo de un proyecto en ingeniería.

Se comunica empleando códigos apropiados de acuerdo con la audiencia a la que se dirige. Emplea los medios audiovisuales apropiados de acuerdo con el mensaje que se busca transmitir. Enfoca su comunicación hacia los objetivos específicos a los que se orienta cada presentación de resultados. Escucha a otros de manera objetiva antes de emitir su juicio crítico y objetivos.

Comunica en forma escrita ideas y/o resultados con objetividad a público de diferentes especialidades y niveles jerárquicos, en el marco del desarrollo de un proyecto en ingeniería.

Prepara informes técnicos acordes con los estándares establecidos para el desarrollo del trabajo. Demuestra capacidad de síntesis de las ideas y conceptos básicos de acuerdo con el público objetivo. Utiliza de forma sistemática un lenguaje escrito y medios acorde con el público objetivo al que se dirige.

V. EQUIPO DE TRABAJO

El equipo de desarrollo estará conformado por un grupo de estudiantes (el número de integrantes será indicado por el docente), entre quienes se distribuirá los roles y actividades a realizar como parte del proyecto. Es importante recalcar que independientemente de la colaboración en los diversos aspectos relacionados al proyecto, todos los participantes debe colaborar en la construcción de las experiencias web, evidenciando capacidad para construcción de aplicaciones web, mostrando el desarrollo de las competencias objetivo de este curso.

VI. PROCESO DE TRABAJO

El proyecto se elaborará bajo un enfoque ágil, estableciendo un Product Backlog con User Stories, aplicando un proceso de desarrollo iterativo e incremental como Scrum. Para la visión del Producto se aplicará Lean UX.

VII. TECNOLOGÍAS

Para el desarrollo del Landing Page, se utilizará HTML5, CSS3 y JavaScript.

Para elaborar los User Persona se utilizará UXPressia / Xtensio.

Para los Journey Map e Impact Map se utilizará UXPressia.

Para el desarrollo de la propuesta de UI a nivel de wireframes y prototipos con simulación de interacción se utilizará Adobe XD / JustInMind / UXPin.

Para los diagramas de ingeniería de software se utilizará LucidChart / Structurizr.

Para el desarrollo del servicio RESTful se utilizará el lenguaje de backend Java con el framework Spring Framework.

Para la Frontend Web Application se utilizará HTML5, CSS3, TypeScript y el framework Angular.

Para la documentación del RESTful API se utilizará Swagger / Postman.

Para el control de proyectos, se utilizará Pivotal Tracker / Trello.

Para el almacenamiento y control de versiones de código se utilizará GIT gestionado desde Github con aplicando GitFlow workflow.

VIII. ESPECIFICACIONES

Considerar los siguientes aspectos sobre algunos componentes para las entregas relacionadas al proyecto.

Vision

Consta del problem statement, una descripción de los puntos más importante que debe resolver la solución propuesta, así como los objetivos y restricciones que delimiten el alcance del proyecto, haciendo uso para ello de Lean UX.

Needfinding

Identificación de necesidades de los segmentos objetivos, en base a la realización de entrevistas evidenciadas en video. El análisis de dichas entrevistas permiten la identificación de los User Persona para cada segmento objetivo, así como la construcción del User Task Matrix, User Journey Map para los User Persona identificados así como Impact Map.

User Stories

Requisitos expresados como User Stories, incluyendo para cada User Story, su descripción aplicando el patrón Persona y criterios de aceptación por cada User Story (expresados como escenarios Given-When-Then ó rules-list, es decir scenario-oriented o rules-oriented).

Architecture

Detallar los módulos y componentes de la solución final apoyados en representaciones gráficas, así como indicar las tecnologías a utilizar.

Artefactos: Diagrama de clases en UML, Modelo de datos, Diccionario de Clases, Script de esquema de base de datos y script de inserción de datos de prueba (Seed).

En el diccionario de clases, considerar para cada clase: especificación de su propósito, así como el propósito de cada atributo y método. Para los métodos públicos de la clase, indicar propósito de cada parámetro y el tipo de información de retorno.

Para la documentación de los RESTful services considerar para cada RESTful Endpoint: indicar las acciones soportadas incluyendo para cada acción el verbo http (get, post, put, delete, patch), sintaxis de llamada, especificación de posibles parámetros, así como ejemplo y explicación del response.

Las pruebas deben incluir capturas de demostración en Postman con los escenarios de pruebas de aceptación del RESTful API, así como enlace a conjunto de pruebas de Postman para ilustrar escenarios de prueba según criterios de aceptación de User Stories.

Servicios

Consta de una lista de las definiciones de los servicios con los que interactuará la aplicación, los cuales estarán disponibles al público.

Artefactos: Diagrama de Arquitectura de la Solución, Relación de Tecnologías utilizadas.

Experiencia de Usuario

User Flow

Flujo de navegación de la aplicación (User Flow Diagrams), consistente de un Diagrama de flujo de interfaces, el cual debe mostrar la secuencia y navegación de las interfaces de usuario de la aplicación. Considerar Happy y Unhappy paths, es decir las rutas de navegación considerando escenarios en que la interacción del usuario es sin errores (Happy Path) y con errores (Unhappy Path) respectivamente.

Prototyping

Los prototipos deben especificar el detalle de las interfaces de usuario a implementar a diferencia del punto anterior. Su objetivo es mostrar una vista previa de la aplicación antes de implementarla, junto con la simulación de las posibles

interacciones. Considerar en los prototipos los principios y elementos de diseño visual, así como las heurísticas de usabilidad y accesibilidad.

Para el landing page considerar prototipos en baja fidelidad para tener una idea de la interfaz web para desktop y mobile.

IX. Características de Documentos

Informe de Proyecto del Curso

Carátula

Universidad, carrera, ciclo

Nombre del curso

Sección

Nombre del profesor

"Informe de Trabajo Final"

Nombre del startup

Nombre del producto

Relación de integrantes

Mes y año

Contenido

Tabla de contenidos

Capítulo I: Introducción

El Producto

Nombre del producto

Antecedentes y problemática

Antecedentes y problemática

Lean UX Problem Statement

Lean UX Assumptions (Business outcomes, Users, User outcomes, Features)

Lean UX Hypothesis

Lean UX Canvas

Capítulo II: Needfinding

Segmentos objetivo

Entrevistas

User Persona por cada Segmento objetivo

User Task Matrix

User Journey Map por cada User Persona

Capítulo III: Product Design

User Stories

Impact Map

User Flow (flujo de navegación entre Wireframes)

Wireframes de Web App (versiones Desktop y Mobile)

Architecture Overview

Architecture Container Diagrams

Diagrama de Clases del Sistema con UML

Diccionario de Clases

Diagrama de Base de Datos

Resumen de tecnologías aplicadas en el diseño y desarrollo de la solución.

Rutas de repositorios de GitHub relacionados a la solución.

Capítulo IV: Product Implementation & Validation

Implemented RESTful API.

Documentación de RESTful API .

Pruebas de Aceptación de RESTful API .

Implemented Startup Product Responsive Landing Page (Screenshots & repository).

Implemented Web Application (Screenshots & repository)

Diagrama de representación de flujo de navegación entre screenshots de vistas web (Desktop y Mobile).

Video de demostración de navegación en aplicación web (Desktop y Mobile).

Core Entities Unit Tests.

Core Behavior-Driven Development Tests.

Acceptance tests para Escenarios de interacción web (Desktop y Mobile).

Video de evidencia de validaciones, con interacciones de usuarios del segmento objetivo con la experiencia desde browser de Desktop y Mobile.

Conclusiones

Conclusiones y recomendaciones

Video About-The-Team

Informe de Participación: Proyecto del Curso

Documento donde el coordinador resume la participación de cada integrante y la asigna a cada uno una calificación entre 0 y 20.

Presentación: Proyecto del Curso

Archivo de PowerPoint

Carátula

Presentación de Miembros de Startup (con Fotos de Perfil, Nombres y Apellidos)

Antecedentes

Acerca del Producto: Nombre, Objetivo, Alcance

Needfinding

Segmentos objetivo

User Personas

User Task Matrix

User Journey Maps

User Stories

Impact Map

Wireframes

Resumen de Tecnologías utilizadas

Architecture Overview

Diagrama de Base de Datos

Diagrama de Clases

Demostración de soporte de backend para escenarios de interacción ofrecidos por la experiencia web.

Demostración de escenarios de interacción utilizando la experiencia web desde browser de desktop y móvil.

Demostración de colaboración en desarrollo según estadísticas de GitHub.

Anexo A. Estructura recomendada para la sección Objetivo del Estudiante (Student Outcome)

El curso contribuye al cumplimiento del Student Outcome ABET:

ABET – EAC - Student Outcome 3

Criterio: *Capacidad de comunicarse efectivamente con un rango de audiencias.*

En el siguiente cuadro se describe las acciones realizadas y enunciados de conclusiones por parte del grupo, que permiten sustentar el haber alcanzado el logro del ABET – EAC - Student Outcome 3.

Criterio específico	Acciones realizadas	Conclusiones
Comunica oralmente sus ideas y/o resultados con objetividad a público de diferentes especialidades y niveles jerarquicos, en el marco del desarrollo de un proyecto en ingeniería.		
Comunica en forma escrita ideas y/o resultados con objetividad a público de diferentes especialidades y niveles jerarquicos, en el marco del desarrollo de un proyecto en ingeniería..		