**TI8512 – Desarrollo Ágil de Aplicaciones Web**

**Especificación del proyecto #1 del curso**

# Aspectos generales del proyecto

## Objetivo General

Evidenciar el dominio de los conceptos básicos de programación web, demostrando destreza en HTML, CSS y JavaScript mediante la implementación de una aplicación web.

## Objetivos específicos:

* Aplicar los conocimientos del curso aplicando el uso de bibliotecas de UI para acelerar el proceso de desarrollo.
* Fortalecer conceptos de integración de JS con APIs de programación.
* Mejorar la destreza de la persona estudiante, integrando bibliotecas de CSS, JavaScript mediante un ejercicio aplicado.
* Implementar una aplicación web mediante el uso de buenas prácticas de desarrollo web, como la accesibilidad, la optimización del rendimiento y la seguridad.
* Mejorar la destreza de la persona estudiante, integrando bibliotecas de CSS, JavaScript mediante un ejercicio aplicado.

## Descripción breve del proyecto:

El proyecto consiste en desarrollar una aplicación web que se integre con un API REST y permita a los usuarios navegar a través de diversas secciones. Para generar la interfaz gráfica se debe hacer uso de una biblioteca de CSS. Adicionalmente, alguno de los requerimientos a implementar requieren el consumo de APIs nativos del navegador, tales como acceso a la ubicación, acceso a notificaciones del sistema operativo, entre otros.

## Entregables del proyecto:

El proyecto #1 se compone de 5 entregables:

1. **Documento del plan de proyecto:** Este documento expone la estrategia planteada por el equipo de proyecto para su implementación. La información requerida en este entregable se detalla en el anexo de este documento.
2. **Documento de prototipo:** Envío del enlace al prototipo que deberá ser implementado como parte de este proyecto. Este debe contener información general sobre la herramienta, cuyo nivel de detalle se detalla en la sección 1.5.
3. **Enlace del backlog de componentes y tareas del proyecto:** Consiste en brindar acceso al profesor al backlog de componentes del proyecto. Este se debe crear utilizando la herramienta *Planner* de *MS Teams*. Más detalles de este entregable en la sección 2.
4. **Entrega del proyecto web:** Esta entrega se hace por medio de *Github*, realizando un último *commit* final del proyecto previo a la fecha límite. Ver sección 2 para más detalles.
5. **Exposición de funcionalidad específica:** aunque los proyectos son similares para cada grupo, cada uno tienen una o varias funcionalidades específicas que le diferencian. Este entregable consiste en la presentación al resto de integrantes del grupo sobre esta funcionalidad específica.

## Metodología del proyecto

### Documentación

Al ser una metodología ágil, las entregas de este proyecto (a excepción del plan de proyecto) se realizarán mediante herramientas que permiten colaboración y la rápida organización de componentes de software.

### Control de versiones

Para el control de versiones, se utilizará Git y GitHub. Cada cambio significativo se registrará mediante *commits* detallados, asegurando un historial claro y organizado. La entrega final del proyecto se realizará mediante un último *commit* en *GitHub* previo a la fecha límite.

La rama “main” del repositorio debe ser modificada únicamente mediante PRs (Pull Requests) que internamente el equipo debe aprobar. Se esperan commits pequeños y frecuentes, que mantengan el orden y los estándares de programación del proyecto.

Para entregar el proyecto web, debe crear un repositorio nuevo llamado: “TI-8512-Proyecto-grupo-N”, donde N será el número de grupo que usted completó en el chat “General” titulado: “Asignación de grupos de proyecto”. Debe completar la columna “Git Repo” en el mensaje, además del nombre de los integrantes. El repositorio debe ser privado.

Inmediatamente después de crear el repositorio, debe agregar el Github del profesor, invitando al usuario “varguitas” de Github.

### Diseño y prototipado

Esta sección incluye el desarrollo de wireframes y prototipos de baja fidelidad para la aplicación web. Se utilizarán herramientas como Figma, Diagrams.net o una presentación de power point para crear diseños visuales antes de proceder con la implementación.

### Evaluación

La evaluación del proyecto se realizará mediante:

* Revisión del backlog: Se analizará el cumplimiento de las tareas y componentes descritos en Planner de MS Teams.
* Defensa de proyecto de forma presencial fuera de horario de clase.
* Exposición de funcionalidad específica: Cada grupo presentará su funcionalidad única al resto de los miembros del equipo para verificar su correcta implementación.

Los pesos de cada uno de los elementos a evaluar se detallan en la rúbrica de evaluación al final de este documento.

# Detalle de los entregables

## Documento del plan de proyecto

El documento de plan de proyecto permite que el profesor conozca sobre las personas integrantes, la tecnología y división de trabajo del equipo. El documento debe ser breve (no más de 5 páginas. No debe incluir una portada y debe seguir la plantilla de “Plan de implementación” anexo en este documento.

El documento debe incluir:

* Nombre de las personas integrantes del grupo.
* Temática de la aplicación web.
* Tecnología seleccionada.
* Herramientas de prototipado a utilizar.
* Identificación preliminar del backlog.
* Plan preliminar de división de sprints.

## Documento de prototipo

Este documento puede entregarse mediante un enlace de invitación a un proyecto de Figma, Sketch, Adobe XD, o bien, una presentación de PowerPoint. No se espera un prototipo de alta fidelidad, por lo contrario, se espera únicamente un diseño estilo mock-up, donde se refleja los principales componentes del sitio web, la navegación y la asociación de los componentes web con componentes de la biblioteca de CSS seleccionada. Ver ejemplo trabajado en la clase S05P01.

Para cada una de las vistas de la aplicación debe proporcionar una propuesta de visualización en la herramienta seleccionada.

## Enlace del backlog de componentes y tareas del proyecto

Para gestionar el progreso del proyecto, los estudiantes deben utilizar la herramienta Planner de MS Teams. Esta herramienta permite la creación de proyectos, tareas y asignación de responsables por tareas. Para realizar este entregable, debe enviar la invitación al profesor a su proyecto de Planner antes de la fecha límite de entrega. Puede hacer la invitación inclusive cuando el board está vacío, siempre y cuando este tenga el backlog completo a la fecha límite de entrega.

Para la implementación del board, se recomiendan las siguientes columnas:

* Backlog: Todas las tareas a implementar en el proyecto.
* Current Sprint: Las tareas que se esperan trabajar en el sprint actual.
* In progress: Tareas que están siendo trabajadas por algún integrante del equipo.
* Done: Tareas completadas.

Todas las tareas en el Sprint actual y columnas consecutivas deben tener una estimación de peso (duración para efectos de este proyecto), deben estar asignadas a una persona integrante del equipo y deben delimitar detalladamente el alcance de la tarea.

## Proyecto web

La aplicación web a desarrollar consiste en un sitio web que permite:

* Listar entidades
* Filtrar y buscar entidades
* Visualizar una entidad específicas.

La entidad se define en la siguiente tabla según el número de grupo al que usted pertenece. Adicionalmente se indican algunas vistas y requerimientos especiales para cada grupo.

|  |  |
| --- | --- |
| **Descripción** | **Vistas** |
| **Grupo 1** | |
| Este grupo trabajará la visualización de entidades relacionadas con el universo Star Wars, haciendo consumo del API SWAPI. [SWAPI - The Star Wars API](https://swapi.dev/) | * **Personajes (10 pts)**   + Ver todos los personajes (2pts).   + Filtrar personajes por nombre (texto) (2pts).   + Enlazar películas en las que sale un personaje(3pts).   + Enlazar personajes con especies (3pts). * **Películas (5pts)**   + Ver todas las películas (1 pt).   + Filtrar películas por título (texto) (2pts).   + Enlazar personajes y especies que salen en la película. (2pts) * **Especies (5pts)**   + Ver todas las especies (1 pt).   + Filtrar especies por nombre (texto)(2 pts)   + Enlazar especies con películas. (2pts) * **Requerimiento especial (10 pts)**   + Crear una vista que despliegue un mapa (5pts).   + La aplicación web debe obtener la ubicación del usuario y mostrar un PIN en el mapa en donde se encuentre el usuario (5pts). |
| **Grupo 2** | |
| Este grupo trabajará la visualización de entidades relacionadas con el universo de el señor de los anillos, haciendo consumo del API TheOneApi. [The Lord of the Rings API - The one API](https://the-one-api.dev/) | * **Personajes (15 pts)**   + Ver todos los personajes, los cuales deben estar paginados con un máximo de 10 personajes a la vez (10 pts).   + Los personajes deben tener un filtro por especie (2 pts).   + Los personajes se pueden ordenar por nombre o por especie (3 pts).   + Esta vista no es obligatorio ser adaptable. * **Libros (5 pts)**   + Ver todos los libros. (1 pt)   + Filtrar libros por título (texto). (2 pts)   + Enlazar libros con capítulos. (2 pts) * **Capítulos (5 pts)**   + Visualizar capítulo (1 pt)   + Enlazar capítulo con libro. (2pts) * **Requerimiento especial (5 pts):**   + Cree una vista aparte con un campo de texto y un botón de enviar (2 pts).   + Al presionar el botón de enviar se debe enviar una notificación nativa del navegador al usuario, con el texto indicado en el campo de texto (3pts). |
| **Grupo 3** | |
| Este grupo trabajará la visualización de entidades relacionadas con recetas de cocina, por medio del API themealdb. [TheMealDB.com - Free Recipe API](https://www.themealdb.com/api.php) | * **Visualización por categoría (5 pts):**   + Visualizar todas las categorías de las recetas ordenadas por orden alfabético (3pts).   + Enlazar cada categoría con lista de platillos para esa categoría (2pts). * **Recetas (10 pts)**   + Visualizar recetas asignadas a una categoría (5 pts).   + Visualizar una receta (1 pt).   + Enlazar todos los ingredientes de una receta (2 pts).   + Para cada ingrediente, indicar la cantidad. (2 pt) * **Requerimiento especial (15 pts)**   + Este API presenta algunas complejidades dado que no es un API Restful (5 pts).   + Cree una vista que permita visualizar un gráfico de pastel con la cantidad de recetas disponibles por categoría (2pts).   + Integre la herramienta plotly para llevar a cabo esta tarea (3 pts).   + Realice una serie de llamadas consecutivas al API para obtener la información requerida para el pie chart. (5 pts) |

Cada uno de los grupos cuenta con diferentes retos y debería de ser resuelto de forma distinta. Asegúrese de solicitar al profesor tiempo de consulta para aclarar los requerimientos.

### Requerimientos funcionales

* Para cada entidad, debe mostrar tantos atributos como haya disponible en el API.
* En todas las vistas se debe brindar un mensaje a los usuarios en caso de que la conexión con el API falló.
* Todas las vistas deben ser adaptables a móviles (con una excepción a una vista para el grupo 02).
* Las llamadas al API deben realizarse de forma óptima, evitando duplicar las llamadas siempre que sea posible y optimizando el consumo de red.

### Requerimientos no funcionales

* El repositorio debe tener una carpeta para los archivos de CSS, JS por separado.
* Cada vista debe tener su propio archivo de HTML. (Esto no es común en un proyecto cuando se utiliza React o Angular, dado que se repite mucho código HTML en varias secciones de la página, pero para efectos de este primer entregable se realizará de esta forma).
* Para efectos de este primer proyecto, no se permite el uso de frameworks como React, Angular o Vue.
* Para todos los tipos de archivo debe utilizar la dependencia Prettier para dar formato a su código.
* En su archivo Readme.md del repositorio, debe incluir las reglas de nomenclatura a utilizar en el proyecto, por ejemplo, estándares de nombramiento de variables y clases.

# Mecanismo de evaluación

|  |  |  |  |
| --- | --- | --- | --- |
| Entregable | Detalle | Puntos a evaluar | Valor |
| Documento de plan de proyecto | Se entrega el documento al profesor subiendo en la respectiva evaluación del TEC Digital. | * Secciones generales sin incluir backlog (2pts). * Backlog preliminar (3pts). | 5 pts |
| Documento de prototipo | Se entrega al profesor el enlace o documento mediante la evaluación respectiva del TEC Digital. | * Cumple todos los requerimientos de visualización (2pts). * Se presenta el documento de forma ordenada y clara (1 * pt). * Identifica los componentes de software a utilizar en el prototipo (2pts). | 5 pts |
| Backlog de componentes y tareas | Se evidencia el uso correcto de la metodología ágil en la ejecución del proyecto. | * Las actividades de desarrollo fueron gestionadas y debidamente especificadas en la herramienta de planificación (1pt). * Se utiliza el backlog como herramienta para gestionar los pendientes y facilitar el flujo de trabajo del equipo (1pt). * Los commits y PRs del repositorio son un reflejo de las actividades por desarrollar en el backlog (3pts). | 5 pts |
| Entrega del proyecto | Ver especificación de alcance para puntuación. |  | 30 pts |
| Presentación de requerimientos especiales al grupo | El equipo presenta al grupo los requerimientos especiales. | * Brinda un contexto general del proyecto y expone las diferentes vistas (1pt) * Indica cuáles fueron las particularidades que diferenciaron el proyecto de otros (1pt) * Muestra ejemplos de código que utilizaron para implementar la particularidad del proyecto. (3pts) | 5 pts |
| Defensa del proyecto en persona | La persona estudiante es capaz de responder a las preguntas del profesor evidenciando dominio de los entregables presentados. |  | 50 pts |

# Cronograma del proyecto

|  |  |
| --- | --- |
| **Entregable** | **Fecha límite de entrega** |
| Documento de plan de proyecto | Domingo 23 de marzo. |
| Documento de prototipo | Miércoles 26 de marzo. |
| Backlog de componentes y tareas | Domingo 30 de marzo |
| Entrega del proyecto | Jueves 10 de Abril antes de las 5PM. |
| Presentación de requerimientos especiales al grupo | Jueves 10 de Abril durante la clase. |
| Defensa del proyecto en persona | Jueves 10 de Abril durante la clase. |

# Anexos

## Anexo A. Plan de implementación del proyecto

|  |  |
| --- | --- |
| **Nombre del proyecto** | Implementación de una página web para <DIGITE SU FINALIDAD AQUÍ> |
| **Integrantes del equipo** | * Nombre 1 * Nombre 2 * Nombre 3 |
| **Objetivo General del Proyecto** | *<Asuma que este proyecto ha sido asignado por su empresa, explique brevemente el objetivo del proyecto>* |
| **Fecha de inicio:** | *Completar* |
| **Fecha de fin:** | *Completar* |

**Descripción General**

*<Describir brevemente la temática de la aplicación web que se desarrollará>*

**Tecnología seleccionada**

*<Enumerar y describir las tecnologías (tecnologías web y bibliotecas seleccionadas, así como dependencias del proyecto) que se utilizarán en el desarrollo de la aplicación web. Debe justificar la elección enumerada anteriormente>*

**Herramientas de prototipado a utilizar**

*<Mencionar las herramientas que se utilizarán para crear prototipos de la aplicación, como Figma, Diagrams.net o PowerPoint.>*

**Identificación preliminar del backlog**

*<Incluir una lista preliminar de las tareas que conformarán el backlog del proyecto. Estas tareas deben ser específicas y orientadas a la funcionalidad de la aplicación. Al ser preliminar, no se espera la lista final del backlog para este proyecto.>*

|  |  |  |
| --- | --- | --- |
| **ID** | **Descripción de la tarea** | **Estimación en tiempo** |
| *T01* | *<Incluir descripción de tarea. Esta debe ser una actividad muy específica del proyecto y no debe tener una duración mayor de 4 horas. Entre más pequeña la estimación de tiempo, mejor>* |  |
| T02 |  |  |

**Plan preliminar de división de sprints**

*<Esbozar un plan preliminar de cómo se dividirán las tareas en sprints. Indicar la duración de cada sprint y las tareas que se realizarán en cada uno de ellos. Al ser una metodología ágil, este plan puede variar durante la ejecución>*

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre | Objetivo del sprint | Cronograma | Tareas |
| Sprint 01 | *<Un sprint debe tener un objetivo, por ejemplo: Implementación de backend>* | Inicio:  DD/MM/AAAA  Fin: DD/MM/AAAA | T01, T04 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |