

SELECTOR DE MATERIAS

PROYECTO FINAL



Metodología Agile-Scrum



Presentación

Nombre:

Rony Rafael De Mota Herrera

Matricula:

2020-10472

Asignatura:

Programación 3

Maestro:

Kelyn Tejada Belliard

Tema:

Selector de Materias (Proyecto final)

Índice

PRESENTACION	Error! Bookmark not defined.
Metodología de Trabajo Agile-Scrum	3
Requisitos iniciales	3
Tecnología para aplicar:	3
Objetivo del proyecto:	3
Alcance del proyecto:.....	3
Definición de lo que el sistema en un primer Reléase va a poder hacer:.....	3
Definición de requerimientos del sistema para el primer Release.....	3
Equipo Metodología Scrum	4
Definición de tareas a ejecutar:	4
Definición del equipo de trabajo (roles, habilidades, etc.):.....	4
Herramientas por utilizar:.....	5
Definición de épicas:	5
Acciones:	5
Definición de fechas para cada ceremonia de Scrum.....	5
Historias de usuarios con sus puntos y criterios de asignación	6
Demostración.....	6
Plan de pruebas	7
Lista de requerimientos funcionales y no funcionales de acorde a las historias de usuarios.	7
Herramientas de pruebas que se estarán usando.....	8
Definir cuáles son los criterios de aceptación de las pruebas.....	8
Definir cuáles son los criterios de rechazo en las pruebas.....	8
Estimación del tiempo de la ejecución de pruebas. (elaborar cronograma de trabajo).	9
Plantillas para caso de pruebas.....	10
Plantilla con los equipos de pruebas y sus responsabilidades.....	11
Plan de automatización de pruebas.....	12
Ejecución y demostración del plan de automatización de pruebas.	12
Bibliografía	13

Metodología de Trabajo Agile-Scrum

Selector de Materias

Requisitos iniciales

Tecnología para aplicar:

En esta ocasión para realizar nuestro proyecto acerca de las metodologías de trabajo agile-scrum, estaré utilizando el lenguaje de programación JavaScript.

Objetivo del proyecto:

El objetivo principal de este proyecto será el de elaborar una página web en la cual podamos hacer una selección de las materias a elegir, con la finalidad de aplicarle la metodología de trabajo Agile-Scrum para un mayor desarrollo de este proyecto.

Alcance del proyecto:

Este proyecto tiene como meta ser utilizado en un futuro no muy lejano para ser utilizado en escuelas ya que la virtualidad es una realidad que en pocos años llegara a las escuelas y colegios en república dominicana.

Definición de lo que el sistema en un primer Reléase va a poder hacer:

Este sistema tendrá varias funciones fundamentales que posee un selector de materias, entre ellas podemos encontrar:

- Registrar los datos del estudiante
- Carrera a elegir del estudiante, en esta opción encontraras:
 - Software
 - Mecatrónica y redes
 - Redes
- Selección de los horarios de clase
- Ver el horario seleccionada por día y hora

Definición de requerimientos del sistema para el primer Release.

Como primera instancia el sistema va a necesitar de diferentes métodos de códigos para su elaboración, como, por ejemplo:

- Creación de la estructura principal de la página.
- Separar las vistas.
- Creación de los campos.
- Añadirle la parte visual con Bootstrap.

Equipo Metodología Scrum

Definición de tareas a ejecutar:

Las tareas por ejecutar que tendrá nuestro selector de materias van de la mano con los requerimientos establecidos, estas tareas serán las siguientes:

- Creación de las vistas principales: esta parte es para ir definiendo los principales campos que necesitaremos.
- Creación de elementos: En esta parte se crearán los campos, labels, botones, etc. Los cuales se utilizarán para el manejo efectivo de la página según los requerimientos.
- Darle los arreglos visuales: En este paso iremos mejorando la parte visual de nuestra página web para darle a los usuarios una mejor experiencia de uso.
- Correctos funcionamientos de los elementos: validar que cada uno de los campos y botones funcionen correctamente.
- Realizar validaciones: realizar la validación para que los usuarios no puedan dejar campos vacíos a la hora de registrarse.

Definición del equipo de trabajo (roles, habilidades, etc.):

Scrum Máster:

- Rony Rafael De Mota Herrera

Va a hacer la persona de tiempo completo, facilitador o (Coach) de Scrum.

PO (Product Owner):

- Ariel Jose Brito Bautista

Será la persona responsable de asegurar que el equipo aporte valor al negocio, y apoyar las necesidades de todos los usuarios en el negocio, así como también las necesidades y el funcionamiento del Equipo Scrum.

DEV (Developer):

- Geronimo Guayara Rodríguez – DEV
- Yohan Humberto Rijo Astacio - DEV

Serán las personas expertas en escribir código, es decir, idear el conjunto de secuencias de órdenes que llevan a un sistema informático a realizar una acción concreta.

QA (Quality Assurance):

- Juan Pablo Céspedes - QA

Plantear la estrategia de pruebas. Diseñar los escenarios y casos de prueba. Revisar, planificar, preparar y ejecutar las pruebas. Reportar errores, plantear soluciones y validar la corrección.

Herramientas por utilizar:

Para este proyecto se estarán utilizando diferentes herramientas para la elaboración de este, entre ellas estarán:

- **HTML-5:** son códigos utilizados para "marcar" el texto de una página web, con el fin de dar instrucciones al navegador sobre cómo mostrarlo. Es decir, las etiquetas HTML son el lenguaje utilizado para estructurar y definir el contenido en un documento HTML.
- **JavaScript:** es un lenguaje de programación que los desarrolladores utilizan para hacer páginas web interactivas. Desde actualizar fuentes de redes sociales a mostrar animaciones y mapas interactivos, las funciones de JavaScript pueden mejorar la experiencia del usuario de un sitio web.
- **Bootstrap:** es una biblioteca multiplataforma o conjunto de herramientas de código abierto para diseño de sitios y aplicaciones web.

Definición de épicas:

Dentro del sistema se busca que los usuarios puedan realizar procesos básicos que posee un selector de materias.

Acciones:

Definición de fechas para cada ceremonia de Scrum.

El grupo SCRUM tendrá un total de 4 reuniones las cuales se estarán realizando en un rango de tres semanas en las siguientes fechas:

➤ **Martes 14 de marzo, 11:30 AM:**

- Creación de Historias de usuario
- Asignación de puntuaciones de las historias de usuario
- Reasignación de fechas y reuniones.
- Realización de los criterios de aceptación.

➤ **Viernes 17 de marzo, 5:00 PM**

- Confirmación de las estimaciones de historias para el sprint planning.
- Evaluación e incorporación de nuevas historias o ajuste de historias.
- Se replantearán aquellas historias con altos puntos por historias más pequeñas.

➤ **Miércoles 22 de marzo, 9:30 PM**

- Inspección de los puntos débiles en las historias de usuarios.
- Realización del plan de prueba

➤ **Lunes 27 de marzo, 10:00 AM**

- Chequeo del proyecto en general.

Historias de usuarios con sus puntos y criterios de asignación

Enlace de Azure:

https://dev.azure.com/202010472/Selector%20de%20materias/_boards/board/t/Selector%20de%20materias%20Team/Issues

Demostración

Enlace del video del incremento:

https://drive.google.com/drive/u/1/folders/1UIBwdJWxqvycJYZRhW_bPf31yPEHGmN1

Plan de pruebas

Lista de requerimientos funcionales y no funcionales de acorde a las historias de usuarios.

1. Pruebas de funcionalidad básica:

- Verificar que se pueden crear nuevas cuentas de usuario y que se guardan correctamente en la base de datos.
- Verificar que se puede acceder a la aplicación con una cuenta de usuario existente.
- Verificar que se pueden agregar datos por defecto a la cuenta de usuario, como la provincia y la ciudad.
- Verificar que los campos de registro se muestran correctamente en la vista principal de la aplicación.
- Verificar que los botones funcionan correctamente y realizan las acciones correspondientes.
- Verificar que las animaciones se muestran correctamente y no afectan la funcionalidad de la aplicación.
- Verificar que los campos se colocan correctamente en la última vista de la aplicación.
- Verificar que se validan correctamente todos los campos y que se muestra un mensaje de error si algún campo está incompleto o incorrecto.
- Verificar que el menú para volver a las vistas anteriores funciona correctamente y no afecta la funcionalidad de la aplicación.

2. Pruebas de validación de campos:

- Verificar que se validan correctamente los campos obligatorios, como el nombre, la provincia, la ciudad y la carrera a estudiar.
- Verificar que se validan correctamente los campos de texto, como el sector y la calle, y que no se permiten caracteres inválidos.
- Verificar que se validan correctamente las selecciones de carrera y que solo se permiten seleccionar tres carreras.
- Verificar que se validan correctamente las selecciones de horario y que no se permiten horarios duplicados o superpuestos.

3. Pruebas de funcionamiento:

- Verificar que la aplicación funciona correctamente en diferentes navegadores y dispositivos.
- Verificar que la aplicación se carga y se responde rápidamente, incluso cuando se han seleccionado muchas materias y horarios.
- Verificar que la aplicación no se bloquea o se cuelga durante el uso normal.
- Verificar que los datos se guardan correctamente en la base de datos y que se pueden recuperar correctamente después de cerrar y volver a abrir la aplicación.

Herramientas de pruebas que se estarán usando.

En este sistema se estarán utilizando una herramienta para la elaboración de pruebas, las cuales son:

➤ Selenium IDE

Es un conjunto de utilidades que facilita la labor de obtener juegos de pruebas para aplicaciones web. Para ello nos permite grabar, editar y depurar casos de prueba, que podrán ser ejecutados de forma automática e iterativa posteriormente.

Definir cuáles son los criterios de aceptación de las pruebas.

- Validación de la información: se debe validar que se hayan completado todos los campos obligatorios antes de continuar con la selección de materias.
- Selección de carreras: se debe validar que el estudiante solo pueda seleccionar las 3 carreras disponibles y no se le permita seleccionar más.
- Selección de materias: se debe validar que el estudiante solo pueda seleccionar las materias correspondientes a la carrera que eligió y no se le permita seleccionar materias de otras carreras.
- Horarios de materias: se debe validar que cada materia tenga 3 horarios disponibles para que los estudiantes puedan elegir el más adecuado para ellos.
- Generación del horario: se debe validar que el horario generado no tenga conflictos de horario entre las materias seleccionadas.
- Mostrar información del estudiante: se debe validar que se muestren correctamente los datos del estudiante al finalizar el proceso de selección

Definir cuáles son los criterios de rechazo en las pruebas.

- Campos obligatorios incompletos: si el estudiante no completa todos los campos obligatorios, no se le permitirá continuar con la selección de materias y se mostrará un mensaje de alerta solicitando que complete todos los campos.
- Selección de más de 3 carreras: si el estudiante intenta seleccionar más de 3 carreras, no se le permitirá continuar con la selección y se mostrará un mensaje de alerta indicando que solo puede seleccionar 3 carreras.
- Selección de materias fuera de la carrera: si el estudiante intenta seleccionar materias que no corresponden a la carrera que eligió, no se le permitirá continuar con la selección y se mostrará un mensaje de alerta indicando que solo puede seleccionar materias de la carrera elegida.

- Materias sin horarios disponibles: si todas las secciones de una materia ya están llenas, no se permitirá seleccionar esa materia y se mostrará un mensaje de alerta indicando que no hay horarios disponibles.
- Conflictos de horario: si hay conflictos de horario entre las materias seleccionadas, no se permitirá continuar con la selección y se mostrará un mensaje de alerta indicando que hay conflictos en los horarios seleccionados.
- Funcionalidad de guardar e imprimir: si la funcionalidad de guardar e imprimir el horario no funciona correctamente, se mostrará un mensaje de alerta indicando que ha habido un error al guardar o imprimir el horario.

Estimación del tiempo de la ejecución de pruebas. (elaborar cronograma de trabajo).

	AGOSTO			
PRUEBAS	SEMANA 1	SEMANA 2	SEMANA 3	SEMANA 4
Creación de las vistas principales				
Creación los campos de registro				
Agregar los datos por defecto				
Creación de los botones				
Añadirle las animaciones				
Crear las conexiones para que los campos se coloquen en la última vista				
Validar que todos los campos estén correctamente				
Hacer un menú para volver a las vistas anteriores				
Hacer validaciones de campos				
Hacer pruebas de funcionamiento				

Plantillas para caso de pruebas

PLANTILLA PARA CASOS DE PRUEBA	
PRUEBAS	CASO DE PRUEBA
Creación de las vistas principales	Validar que se cree según lo requerido
	Que estén los campos necesarios
Creación los campos de registro	Crear los campos del formulario de registro
	Crear los títulos
Agregar los datos por defecto	Agregar las informaciones que vienen por defecto al cargar la página.
Creación de los botones	Crear los botones de registro
	Crear botón de guardar
Añadirle las animaciones	Crear animaciones de alerta
	Crear animaciones a botones
Crear las conexiones para que los campos se coloquen en la última vista	Agregar las relaciones entre campos
	Agregar funcionamiento de botones
Validar que todos los campos estén correctamente	Validar que todo esté según lo requerido
Hacer un menú para volver a las vistas anteriores	Hacer menú para movernos entre vistas
Hacer validaciones de campos	Hacer validaciones y alertas en caso de error
Hacer pruebas de funcionamiento	Probar todo el funcionamiento

Plantilla con los equipos de pruebas y sus responsabilidades.

PLANTILLA PARA EQUIPOS DE PRUEBAS		
PRUEBAS	EQUIPOS DE PRUEBAS	RESPONSABILIDADES
Creación de las vistas principales	DEV	Desarrollar las vistas de la página.
Creación los campos de registro	DEV	Colocar todos los campos requeridos para la página.
Agregar los datos por defecto	DEV	Añadir todos los datos requeridos.
Creación de los botones	DEV	Diseñar los botones que ese utilizaran para el correcto funcionamiento.
Añadirle las animaciones	DEV	Agregar animaciones para que la pagina sea mas agradable para el usuario
Crear las conexiones para que los campos se coloquen en la última vista	DEV	Darle el correcto funcionamiento a los botones y formularios.
Validar que todos los campos estén correctamente	QA	Verificar que todos los campos funciones.
Hacer un menú para volver a las vistas anteriores	DEV	Crear un menú para poder movernos entre las vistas.
Hacer validaciones de campos	QA	Verificar que todos funcionen
Hacer pruebas de funcionamiento	QA	Validar que todo este según los requerimientos

Plan de automatización de pruebas

- Identificación de los flujos críticos: Se deben identificar los flujos críticos de la página, es decir, aquellos flujos que son más importantes para el usuario y que deben ser probados con mayor detalle. En este caso, los flujos críticos podrían incluir el registro del estudiante, la selección de las carreras y materias, la validación de horarios, la generación del horario de clases y la funcionalidad de guardar e imprimir el horario.
- Selección de la herramienta de automatización: Se debe seleccionar una herramienta de automatización de pruebas que sea adecuada para el proyecto y que tenga las funcionalidades necesarias para cubrir los flujos críticos. Algunas herramientas populares incluyen Selenium, Cypress y Appium.
- Diseño de los casos de prueba: Se deben diseñar los casos de prueba para los flujos críticos. Estos casos de prueba deben incluir los pasos que debe seguir el usuario, los datos de entrada que se deben utilizar y los resultados esperados. También se deben incluir los criterios de aceptación y rechazo para cada caso de prueba.
- Implementación de los casos de prueba: Se deben implementar los casos de prueba utilizando la herramienta de automatización seleccionada. Es importante asegurarse de que los casos de prueba sean claros y fáciles de seguir para que otros miembros del equipo puedan entenderlos.
- Ejecución de los casos de prueba: Se deben ejecutar los casos de prueba para verificar que la página funciona correctamente. Es importante asegurarse de que todos los flujos críticos se prueben adecuadamente y de que se capturen los resultados de cada prueba.
- Reporte de resultados: Se deben generar informes de resultados que indiquen el estado de las pruebas y los resultados obtenidos. Estos informes deben ser claros y fáciles de entender para que los miembros del equipo puedan actuar en consecuencia.
- Mantenimiento de las pruebas: Se deben mantener las pruebas actualizadas y relevantes. Esto incluye actualizar los casos de prueba según los cambios en la página y agregar nuevos casos de prueba cuando sea necesario.
- Integración con el proceso de desarrollo: Se deben integrar las pruebas automatizadas con el proceso de desarrollo para asegurarse de que se prueben todas las funcionalidades antes de que se libere una nueva versión de la página.

Ejecución y demostración del plan de automatización de pruebas.

Link de repositorio con las pruebas y el proyecto:

<https://github.com/RonyHerrera/SubjectSelector-master>

Bibliografía

- <https://www.softwaretestingbureau.com/crear-un-buen-plan-de-pruebas/#:~:text=El%20plan%20de%20pruebas%20tiene,cambios%20que%20tiene%20el%20proyecto>
- <https://spa.myservername.com/sample-test-plan-document>
- <https://docs.microsoft.com/en-us/azure/devops/test/perform-exploratory-tests?view=azure-devops>
- https://es.wikipedia.org/wiki/Requisito_no_funcional
- <http://www.pmoinformatica.com/2015/05/requerimientos-no-funcionales-ejemplos.html>
- <http://www.pmoinformatica.com/2017/02/requerimientos-funcionales-ejemplos.html>
- <https://scrum.menzinsky.com/2020/06/una-historia-de-usuario-puede-dividirse.html>
- <http://www.lecciones-aprendidas.info/2013/11/como-es-una-historia-de-usuario-un.html>
- <https://www.sinnaps.com/blog-gestion-proyectos/scrum-epica>
- <https://marketplace.visualstudio.com/items?itemName=ms.vss-exploratorytesting-web#supportedbrowsers>