



CV GENERATOR

Por Joel Matos (2021-1050)



14 DE DICIEMBRE DE 2023



Las Americas Institute of Technology

Presentación

Nombres:

Joel Javier.

Nombres:

Matos Santana.

Asignatura:

Programación 3

Matrícula:

2021-1050

Docente:

Kelyn Tejada Belliard

INTRODUCCIÓN.....	3
DESARROLLO:	4
<i>Generador de Curriculum Vitae (CV Generator Project)</i>	4
Objetivo del Proyecto:	4
Cronograma del Proyecto:	4
Metodología Agile-Scrum:	5
B- Equipo Metodología Scrum:	6
C- Acciones:.....	7
10 Historias de Usuarios con Criterios de Aceptación:.....	8
Plan de Pruebas para "CV Generator":.....	10
Conclusion:	10
Bibliografías:	10

Introducción.

Es un placer presentarles "CV Generator", Un proyecto destinado a simplificar el proceso de creación de currículos para estudiantes. Desarrollado con una interfaz intuitiva y herramientas de personalización, esta aplicación web busca ofrecer a cada usuario la capacidad de destacar sus habilidades de manera única. Con la metodología Agile-Scrum, buscamos adaptarnos rápidamente y brindar continuas mejoras. El objetivo es empoderar a los estudiantes para presentar sus perfiles de manera efectiva mientras avanzan en sus carreras académicas y profesionales. Agradecemos la oportunidad de compartir nuestro proyecto con ustedes y esperamos sus comentarios. ¡Gracias!

Desarrollo:

Generador de Curriculum Vitae (CV Generator Project).

Tecnología para Aplicar:

- HTML, CSS, JavaScript para el desarrollo web.
- Git para el control de versiones.
- Utilizar un servidor web local para pruebas.

Objetivo del Proyecto:

Desarrollar una aplicación web que permita a los usuarios generar currículos de manera rápida y personalizada.

Alcance del Proyecto:

Crear un sistema de generación de currículos que incluya secciones para información personal, educación, experiencia laboral, habilidades, etc.

Implementar una interfaz amigable e intuitiva.

Permitir la descarga del currículo en formatos populares (PDF, Word).

Cronograma del Proyecto:

Cronograma del Proyecto "CV Generator":

Semana 1-2: Diseño de la Interfaz y Estructura Básica.

Definir la apariencia y disposición de las secciones del generador de currículos.

Establecer la arquitectura del software, identificando componentes clave.

Semana 3-4: Desarrollo de Características Iniciales (Formulario de Información Personal).

Implementar la funcionalidad básica de ingreso de información personal.

Desarrollar la capacidad de guardar y editar la información del usuario.

Semana 5-6: Desarrollo de Características Adicionales (Secciones de Educación y Experiencia Laboral).

Incorporar la capacidad de agregar y gestionar secciones de educación y experiencia laboral.

Introducir funciones como la ordenación y la eliminación de elementos.

Semana 7-8: Implementación de Características Avanzadas (Generación de PDF, Personalización).

Desarrollar la funcionalidad de generación de currículos en formato PDF.

Permitir la personalización avanzada de la apariencia del currículum.

Semana 9-10: Pruebas y Depuración.

Realizar pruebas exhaustivas del sistema, identificando y corrigiendo posibles errores.

Optimizar el rendimiento y la usabilidad mediante retroalimentación interna y de usuarios beta.

Semana 11-12: Preparación para el Lanzamiento y Documentación.

- Finalizar cualquier ajuste necesario según la retroalimentación de las pruebas.
- Crear documentación completa, incluyendo manuales de usuario y desarrollador.
- Preparar la infraestructura para el lanzamiento del "CV Generator Pro" en un entorno de producción.

Definir Claramente lo que el Sistema en un Primer Release Va a Poder Hacer:

- El sistema permitirá a los usuarios ingresar información personal, educación, experiencia laboral y habilidades.
- Generará un currículum en formato PDF descargable.
- La interfaz será simple y fácil de usar.

Definir Requerimientos del Sistema para el Primer Release:

Requisitos Funcionales:

- Interfaz de usuario amigable.
- Formulario para ingresar información del usuario.
- Generación de currículum en formato PDF.
- Requisitos No Funcionales:
- Compatible con los principales navegadores web.
- Tiempo de carga rápido.
- Diseño responsivo para dispositivos móviles.

Metodología Agile-Scrum:

Product Backlog:

Lista priorizada de características y mejoras.

Sprint Planning:

Planificación detallada para cada sprint.

Daily Standups:

Reuniones diarias de seguimiento.

Sprint Review:

Revisión del trabajo al final de cada sprint.

Retrospective:

Evaluación y mejora continua al final de cada sprint.

Control de Versiones:

Utilizar Git para el control de versiones y GitHub para la colaboración.

Pruebas:

- Pruebas unitarias durante el desarrollo.
- Pruebas de aceptación al final de cada sprint.

Implementación:

- Desarrollo iterativo, enfocándose en funcionalidades clave primero.
- Actualizaciones regulares al repositorio compartido.

Documentación:

Mantener documentación actualizada, incluyendo el manual de usuario.

Despliegue:

Lanzar el sistema en un entorno de producción después de las pruebas exitosas.

Monitoreo y Mejora Continua:

Monitorear el rendimiento del sistema después del lanzamiento.

Recopilar comentarios de los usuarios y planificar mejoras futuras.

Siguiendo esta estrategia, el equipo podrá desarrollar el "CV Generator " de manera efectiva, cumpliendo con los requisitos y entregando un producto de calidad en intervalos cortos.

B- Equipo Metodología Scrum:

Definir Tareas a Ejecutar:

Identificar tareas específicas para cada sprint, como el diseño de la interfaz, la implementación de funciones clave (ingreso de información, generación de PDF), pruebas unitarias, y la integración de la seguridad.

Definir el Equipo de Trabajo (Roles, Habilidades, etc):

Product Owner: Responsable de definir y priorizar los elementos del backlog, representando las necesidades de los usuarios.

Scrum Master: Encargado de asegurar que el equipo siga la metodología Scrum, eliminando obstáculos y facilitando la comunicación.

Desarrolladores (Equipo de Desarrollo): Encargados de diseñar, implementar y probar las funcionalidades del "CV Generator Pro".

Tester: Responsable de realizar pruebas y garantizar la calidad del software.

Diseñador UX/UI: Encargado de la experiencia del usuario y del diseño de la interfaz.

Herramientas que Usarían:

- **Jira o Trello:** Para la gestión y seguimiento de las tareas del backlog.
- **Git y GitHub:** Para el control de versiones y colaboración en el código.
- **Slack o Teams:** Para la comunicación diaria entre el equipo.
- **Google Docs o Confluence:** Para la documentación compartida.

Definir las Épicas:

Épica 1 - Interfaz y Estructura:

Sub-épica 1: Diseño y maquetación de la interfaz.

Sub-épica 2: Definición de la arquitectura del software.

Épica 2 - Ingreso de Información y Funcionalidades Básicas:

Sub-épica 1: Desarrollo del formulario de información personal.

Sub-épica 2: Implementación de funciones básicas de edición y guardado.

Épica 3 - Secciones de Educación y Experiencia Laboral:

Sub-épica 1: Desarrollo de la capacidad de agregar y gestionar secciones de educación.

Sub-épica 2: Implementación de funciones para la experiencia laboral.

Épica 4 - Generación de PDF y Personalización:

Sub-épica 1: Desarrollo de la función de generación de currículos en formato PDF.

C- Acciones:

Definir Fechas para Cada Ceremonia de Scrum:

Sprint Planning:

Fecha: Inicio de cada sprint (por ejemplo, primer día hábil de la semana).

Duración: 2 horas.

Daily Standup:

Fecha: Cada día hábil.

Duración: 15 minutos.

Sprint Review:

Fecha: Final de cada sprint (último día hábil de la semana del sprint).

Duración: 1 hora.

Retrospective:

Fecha: Después de cada Sprint Review.

Duración: 1 hora.

Afinar al Menos 10 Historias de Usuarios con Criterios de Aceptación Obligatorios:

Historia de Usuario 1 - Interfaz y Estructura:

- Como usuario, quiero una interfaz intuitiva para poder navegar fácilmente por el generador de currículos.
- **Criterios de Aceptación:** La interfaz debe ser clara y tener un menú de navegación accesible desde cualquier pantalla.

Historia de Usuario 2 - Ingreso de Información Personal:

- Como usuario, quiero poder ingresar mi información personal de manera sencilla.
- **Criterios de Aceptación:** Debo poder ingresar datos como nombre, contacto, y dirección de manera clara y precisa.

Historia de Usuario 3 - Funciones Básicas de Edición:

- Como usuario, quiero poder editar y guardar mi información básica.
- **Criterios de Aceptación:** Debo poder editar y guardar cambios en mi información personal.

Historia de Usuario 4 - Secciones de Educación y Experiencia Laboral:

- Como usuario, quiero agregar y gestionar secciones de educación y experiencia laboral.
- **Criterios de Aceptación:** Debo poder agregar, editar y eliminar entradas en las secciones de educación y experiencia laboral.

Historia de Usuario 5 - Generación de PDF:

- Como usuario, quiero generar mi currículum en formato PDF.
- **Criterios de Aceptación:** Debo poder descargar un currículum en formato PDF con la información ingresada.

Historia de Usuario 6 - Personalización Avanzada:

- Como usuario, quiero poder personalizar la apariencia avanzada de mi currículum.
- **Criterios de Aceptación:** Debo poder ajustar el diseño y estilo del currículum según mis preferencias.

Historia de Usuario 7 - Pruebas Unitarias:

- Como desarrollador, quiero realizar pruebas unitarias para garantizar la funcionalidad de cada componente.
- **Criterios de Aceptación:** Debe haber una suite de pruebas unitarias que cubra todas las funciones principales.

Historia de Usuario 8 - Optimización de Rendimiento:

- Como desarrollador, quiero optimizar el rendimiento del sistema para una experiencia de usuario fluida.
- **Criterios de Aceptación:** La aplicación debe tener tiempos de carga rápidos y ser eficiente en el uso de recursos.

Historia de Usuario 9 - Retroalimentación de Pruebas Beta:

- Como equipo, queremos recopilar y analizar la retroalimentación de los usuarios beta.
- **Criterios de Aceptación:** Se debe recopilar comentarios de al menos 10 usuarios beta y realizar ajustes según sea necesario.

Historia de Usuario 10 - Documentación Completa:

- Como equipo, queremos crear documentación completa para usuarios y desarrolladores.
- **Criterios de Aceptación:** La documentación debe ser clara, detallada y accesible para todos los usuarios y miembros del equipo de desarrollo.

Asignar los Puntos de Historia:

Puntos de Historia:

- **Historia de Usuario 1:** 1 puntos.
- **Historia de Usuario 2:** 4 puntos.
- **Historia de Usuario 3:** 4 puntos.
- **Historia de Usuario 4:** 3 puntos.

- **Historia de Usuario 5:** 2 puntos.
- **Historia de Usuario 6:** 4 puntos.
- **Historia de Usuario 7:** 3 puntos.
- **Historia de Usuario 8:** 4 puntos.
- **Historia de Usuario 9:** 3 puntos.
- **Historia de Usuario 10:** 2 puntos.

Nota:

Los puntos asignados a cada historia son relativos y pueden variar según la complejidad percibida y la capacidad del equipo para completar la tarea. Es importante revisar y ajustar los puntos en base a la experiencia acumulada durante los Sprint.

2- Plan de Pruebas para "CV Generator":

1- Lista de Requerimientos Funcionales y No Funcionales:

Requerimientos Funcionales:

- El sistema debe permitir al usuario ingresar información personal.
- Debe ser posible agregar y gestionar secciones de educación y experiencia laboral.
- La aplicación debe generar currículos en formato PDF.
- Los usuarios deben poder personalizar la apariencia avanzada de sus currículos.

Requerimientos No Funcionales:

- La interfaz debe ser intuitiva y fácil de usar.
- El sistema debe ser compatible con los principales navegadores web.
- El tiempo de carga del sistema no debe exceder los 5 segundos.

2- Criterios de Aceptación de las Pruebas:

- Cada función principal debe funcionar según lo especificado en las historias de usuario.
- La interfaz debe ser coherente y fácil de navegar.
- Los currículos generados deben ser precisos y contener la información ingresada.

3- Criterios de Rechazo en las Pruebas:

- Funcionamiento incorrecto de las funciones principales.
- Interfaz confusa o incoherente.
- Generación de currículos con errores o información incorrecta.

4- Herramientas de Pruebas:

Selenium: Para pruebas unitarias y de integración.

5- Estimación del Tiempo de Ejecución de Pruebas:

Se estima que la ejecución completa de pruebas tomará aproximadamente 2 semanas, distribuidas en cada sprint.

6- Plantillas para Cada Caso de Pruebas:

Se crearán plantillas específicas para cada historia de usuario, incluyendo pasos, datos de entrada, resultados esperados y observaciones.

7- Equipos de Pruebas y Responsabilidades:

Tester Principal: Encargado de coordinar y ejecutar las pruebas.

Desarrolladores: Responsables de las pruebas unitarias y colaboración en las pruebas de integración.

Diseñador UX/UI: Evaluará la interfaz y experiencia del usuario.

8- Plan de Automatización de Pruebas:

Se priorizará la automatización de pruebas para las funciones críticas, como la generación de PDF.

9- Ejecución y Demostración del Plan de Automatización de Pruebas:

La ejecución del plan de automatización se llevará a cabo en el último sprint, y se realizará una demostración para validar la efectividad de las pruebas automatizadas.

Nota:

Las plantillas específicas para casos de pruebas, así como el cronograma de trabajo detallado, serán proporcionados según la estructura y preferencias del equipo. La flexibilidad y adaptabilidad serán clave durante la ejecución del plan de pruebas.

CONCLUSIÓN.

En resumen, el proyecto "CV Generator" ha sido cuidadosamente planificado y estructurado utilizando la metodología Agile-Scrum, lo que ha permitido abordar de manera eficiente los requisitos del sistema. La estrategia de desarrollo ha sido respaldada por un equipo multifuncional con roles claramente definidos, y se ha establecido un cronograma detallado que abarca desde el diseño inicial hasta la preparación para el lanzamiento.

El plan de pruebas diseñado se alinea directamente con las historias de usuario definidas, asegurando la calidad y funcionalidad del sistema. Se han identificado criterios de aceptación y rechazo claros, y se han seleccionado herramientas de prueba adecuadas, incluida la automatización para mejorar la eficiencia del proceso.

La conclusión general es que el proyecto está bien encaminado para cumplir con sus objetivos. La combinación de la metodología Agile-Scrum, la atención a los detalles en el plan de pruebas y la selección de herramientas adecuadas brindan una base sólida para el desarrollo y la implementación exitosa del "CV Generator". El equipo está preparado para enfrentar los desafíos, adaptarse a los cambios y entregar un producto que satisfaga las necesidades de los usuarios.

BIBLIOGRAFÍAS.

- Metodologías Ágiles y Scrum:

"Scrum: The Art of Doing Twice the Work in Half the Time" de Jeff Sutherland.

"Agile Estimating and Planning" de Mike Cohn.

Desarrollo de Software y Pruebas:

"**Clean Code**: A Handbook of Agile Software Craftsmanship" de Robert C. Martin.

"The Art of Software Testing" de Glenford J. Myers.

- Diseño de Interfaz de Usuario:

"Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability" de Steve Krug.

"Designing Interfaces: Patterns for Effective Interaction Design" de Jenifer Tidwell.

- Desarrollo Web (HTML, CSS, JavaScript):

"Eloquent JavaScript: A Modern Introduction to Programming" de Marijn Haverbeke.

"HTML and CSS: Design and Build Websites" de Jon Duckett.

- Gestión de Proyectos y Planificación:

"The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win" de Gene Kim, Kevin Behr, y George Spafford.

"Project Management for the Unofficial Project Manager: A FranklinCovey Title" de Kory Kogon, Suzette Blakemore, y James Wood.