Diagrama

Descripción generada automáticamente con confianza media

**00/00/2021**

**M. en A. Graciela Margarita Cetina Quijano**

**M. en C. Homer Alberto Lara Luis**

**Universidad Politécnica de Tecámac**

**Anacleto Ángeles Aritzi Denalyh**

**Estrada Mejía Raúl**

**Herrera Santiago Ulises**

**Sistema de Administración Para Educación Dual**

# **AGRADECIMIENTOS**

ARITZI:

Le quiero dar gracias especialmente a mi madre por su apoyo espiritual, físico, económico y sobre todo moral, y que gracias a ello hoy me brinda la oportunidad de estar en un nivel superior y esto me motiva a ser una mejor estudiante cada día y a superarme como persona, sobre todo porque siempre ha creído en mí.

Posteriormente agradecer a los docentes y directivos que laboran dentro de ella que me apoyan en esta formación académica que forma parte de la Universidad Politécnica de Tecámac, pero sobre todo por brindarme sus conocimientos a lo, largo de este proceso y poder salir adelante con mis estudios.

Agradezco a la Universidad Politécnica de Tecámac por brindarme la oportunidad de colaborar con ellos, pero sobre todo porque gracias a los que forman parte de la institución he obtenido nuevos conocimientos y habilidades, con las cuales he podido aplicarlas en diferentes momentos de mi vida académica, especialmente a mi asesor Hernández M. en A. Graciela Margarita Cetina Quijano por brindarme su valiosa colaboración y orientación en el desarrollo de este trabajo. Su labor se enfoca en cuidar los saberes de la institución, y permitirles a otros aprende de ellos. Nos ayudan a cumplir el sueño de superarnos y cumplir nuestras expectativas, y de siempre ir por la constate mejora, para ser excelentes seres humanos por esta ocasión no ha sido la excepción, y le agradezco con creces por ayudarme a lograr este propósito en mi vida académica.

Raúl:

Ulises:

Primero quiero agradecer a mi familia por el apoyo incondicional y la confianza que han depositado en mi para estar estudiando esta carrera, de igual manera a amigos por siempre dar ánimos en seguir adelante en todos los proyectos en los que he estado trabajando.

De igual manera a nuestra casa de estudios la universidad politécnica de Tecámac por proporcionarme las herramientas de trabajo para así poder lograr de mejor manera los objetivos que nos hemos propuesto para poder culminar la carrera de igual manera por habernos dado una educación de calidad.

De igual manera a los docentes que han estado presente en cada enseñanza, así como el brindarnos las herramientas de trabajo y conocimiento para poder desempeñar de mejor manera las competencias que desarrollamos durante nuestra estadía en la universidad.

De igual manera a la asesora académica M. en A. Graciela Margarita Cetina Quijano quien ha estado presente en la elaboración de este proyecto guiándonos de una manera que este trabajo pueda ser elaborado con las mejores practicas en cuanto a redacción, de igual manera por haber estado apoyándome en cada proceso de mi estancia por la universidad ya que me proporciono los conocimientos en cuanto a ética profesional y las enseñanzas fundamentales para poder desempeñarme en el mundo laboral.

Índice general

[**INTRODUCCIÓN** 2](#_Toc118831730)

[**ANTECEDENTES** 3](#_Toc118831731)

[**PLANTEAMIENTO DEL PROBLEMA** 5](#_Toc118831732)

[**JUSTIFICACIÓN** 8](#_Toc118831733)

[**OBJETIVO GENERAL** 9](#_Toc118831734)

[**MARCO TEÓRICO** 10](#_Toc118831735)

[**METODOLOGÍA** 18](#_Toc118831736)

[**CRONOGRAMA** 20](#_Toc118831737)

[**DESARROLLO DEL PROYECTO** 21](#_Toc118831738)

[**RESULTADOS** 52](#_Toc118831739)

[**CONCLUSIONES Y RECOMENDACIONES** 53](#_Toc118831740)

[**REFERENCIAS BIBLIOGRÁFICAS** 55](#_Toc118831741)

[**ANEXOS** 57](#_Toc118831742)

**INDICE DE TABLAS**

[Tabla 1 Ejemplo de la plantilla de levantamiento de requerimientos Fuente: Original. 40](#_Toc119347056)

[Tabla 2 Acceso al docente, Fuente: Original. 41](#_Toc119347057)

[Tabla 3 Acceso al estudiante, Fuente: original. 42](#_Toc119347058)

[Tabla 4 Llenado de datos, Fuente: original 43](#_Toc119347059)

[Tabla 5 Descarga de formatos en Excel, Fuente: original 44](#_Toc119347060)

[Tabla 6............, Fuente: original. 45](#_Toc119347061)

**INDICE DE FIGURAS**

[Figura 1Entorno de desarrollo integrado, Fuente: https://visualstudio.microsoft.com/es/vs](#_Toc118831530) 29

[Figura 2 Complementos y herramientas de desarrollo para Visual Studio 2022. 30](#_Toc118831531)

[Figura 3 Complemento para administrar bases de datos. 24](#_Toc118831532)

Figura 4 Gestor de base de datos Microsoft SQL, Fuente: https://www.microsoft.com/es-mx/sql-server/sql-server-downloads 31

[Figura 5 Documentación del gestor de base de datos, Fuente: learn.microsoft.com 32](#_Toc118831534)

[Figura 6 Crear y elegir el proyecto. 32](#_Toc118831535)

[Figura 7 Carpetas y archivos del sistema. 33](#_Toc118831536)

[Figura 8 Paquetes NuGet, Fuente: Servidor localhost del proyecto. 34](#_Toc118831537)

[Figura 9 Comando CLI para instalar dotnet. 34](#_Toc118831538)

[Figura 10 Conexion a la Base de datos, Fuente:Servidor localhost del proyecto. 35](#_Toc118831539)

[Figura 11 Clase Empresa y sus atributos. 35](#_Toc118831540)

[Figura 12 Uso de clases en la Base de Datos. 36](#_Toc118831541)

[Figura 13 Comando CLI para crear una migración. 36](#_Toc118831542)

[Figura 14 Comando CLI para actualizar la Base de Datos. . 37](#_Toc118831543)

[Figura 15 Creación del repository. 37](file:///C:\Users\hp\Downloads\avance%2003%20(1).docx#_Toc118831544)

[Figura 16 Carpetas Implementation e Interface. 38](#_Toc118831545)

[Figura 17 Mapeo de ViewModels con AutoMapper. 39](#_Toc118831546)

[Figura 18 ViewModels. 39](#_Toc118831547)

[Figura 19 Servicios y Controladores. 40](#_Toc118831548)

[Figura 20 Ajax para mostrar los datos. . 41](#_Toc118831549)

[Figura 21 Paquetería NuGet, Fuente: Servidor localhost del proyecto. 57](file:///C:\Users\hp\Downloads\avance%2003%20(1).docx#_Toc118831550)

[Figura 22 Botón para correr la API, Fuente: Servidor localhost del proyecto. 57](file:///C:\Users\hp\Downloads\avance%2003%20(1).docx#_Toc118831551)

[Figura 23 Estilos y colores, Fuente: Servidor localhost del proyecto. 58](file:///C:\Users\hp\Downloads\avance%2003%20(1).docx#_Toc118831552)

[Figura 24 Ventana principal del sistema, Fuente: Servidor localhost del proyecto. 59](file:///C:\Users\hp\Downloads\avance%2003%20(1).docx#_Toc118831553)

[Figura 25 Vista registro Universidad, Fuente: Servidor localhost del proyecto. 60](file:///C:\Users\hp\Downloads\avance%2003%20(1).docx#_Toc118831554)

# **RESUMEN**

En la Universidad Politécnica de Tecámac es una institución educativa de nivel medio superior, en donde actualmente no se cuenta con un sistema de registro al programa de educación dual, en la actualidad este proceso se lleva manualmente y esto hace que su proceso no se lleve correctamente y sea un poco largo ya que debe entregar la información solicitada, para poder realizar el procedimiento de selección a ED según sea el caso ya que los profesores y estudiantes no tienen un control total en el área de la documentación, para el respaldo y guardado de información , este proceso se lleva a cabo de manera manual, llenando algunos formatos y es un proceso un poco largo ya que debe de pasa por varios pasos, hasta llegar al encargado del área correspondiente para validar documentos y finalmente terminar el trámite . Es por eso que este proyecto es muy importante para agilizar este procedimiento, en esta primera fase, se realiza el análisis de los requerimientos con el objetivo de cubrir las necesidades de automatización del mismo, así como también se realiza la base de datos para guardar la información necesaria y finalmente el sistema completo de la navegación de lo que sería la aplicación terminada, el desarrollo del software para su administración, conlleva una navegación que sea transparente y fácil para el o los usuarios finales. Con el sistema se pretende generar el respaldo de información necesaria el fin de evitar errores y tener mayor eficacia en tiempo y forma, así como la generación de los formatos en XML que pueda ser generado y descargado desde esta API.

# **ABSTRACT**

The “Universidad Politécnica de Tecámac” is an educational institution of higher education, which currently does not have a registration system to the dual education program, currently this process is carried out manually and this makes the process is not carried correctly and is a bit long as it must deliver the requested information, This process is carried out manually, filling out some forms and it is a long process because it must go through several steps, until it reaches the person in charge of the corresponding area to validate documents and finally finish the process. That is why this project is very important to streamline this procedure, in this first phase, the analysis of the requirements in order to meet the needs of automation of the same, as well as the database is made to store the necessary information and finally the complete system of navigation of what would be the finished application, the development of software for administration, involves a navigation that is transparent and easy for the end user or users. The system is intended to generate the necessary information backup in order to avoid errors and have greater efficiency in time and form, as well as the generation of XML formats that can be generated and downloaded from this API.

# **INTRODUCCIÓN**

La educación dual es una modalidad dónde la escuela y la empresa le ofrecen al estudiante una opción educativa alterna, esta modalidad de estudio tiene como objetivo introducir al estudiante en un ámbito laboral.

En esta modalidad el estudiante es evaluado con actividades, las cuales son asignadas en conjunto por la escuela y la empresa.

El proyecto se refiere llevar a cabo la elaboración de un sistema software para la administración de información para llevar a cabo este proyecto se necesita mencionar las causas una de ellas es el extravió o mala organización de documentos. Dada esta situación como primer paso se debe saber el motivo o la causa de por qué el usuario entrega la documentación y el proceso con lleva varios filtros porque es un poco tedioso. Anteriormente para llevar a cabo este paso el estudiante se necesitaba poner en contacto con el encargado del área para poder hacer su trámite correspondiente en donde se tomaban datos generales y personales del usuario para así poder llenar el formato correspondiente.

Con el sistema se pretende generar respuestas preestablecidas para el llenado de los formatos con el fin de evitar errores y tener mayor eficacia en tiempo y forma. Actualmente el proceso de inscripción a este proceso los documentos se entregaban personalmente en el área correspondiente de la Universidad corriendo el riesgo de pérdidas de datos por diversas situaciones el cual es un proceso no confiable ya que se corre el riesgo de pérdida de la información y de la necesidad de volver solicitar la documentación. Se busca generar una base de datos la cual nos ayudara a llevar una mejor administración de los registros con la cual tendremos acceso mediante una interfaz del sistema generado en la cual podremos buscar los registros por ID del usuario dado de alta en el sistema.

Así mismo lleva un análisis por acciones de acompañamiento de todos los usuarios dados de alta en el sistema, realizadas en cierto periodo para así poder obtener ciertos indicadores importantes para la mejora continua y tener un mejor manejo de información de los usuarios. Actualmente acciones de acompañamiento realizan estos análisis de manera semiautomática corriendo el riesgo de una mala captura de datos, o pérdidas de estos. Con el sistema se pretende obtener estos análisis de forma automatizada con una interfaz la cual tendrá un generador de tablas con los datos de los usuarios ya registrados como el sistema o base de datos. La cual contara con filtros (fecha, ID Estudiante, calificaciones, Empresas, etc.) teniendo así un mejor análisis de los usuarios dados de alta en el sistema y poder generar tablas con la información que se requiere a si mismo generando archivos en XML para presentar los datos.

# **ANTECEDENTES**

El uso de sistemas informativos ha ido en constante crecimiento, el manipular la información y tener acceso a datos específicos es de vital importancia para las empresas hoy en día. Un sistema de información (SI), hace referencia a un conjunto de procesos que tiene como objetivo la administración de datos e información (Equipo editorial, 2021).

Así como el desarrollo de software ha evolucionado con el tiempo, los SI también lo han hecho, siendo más complejos hasta ser denominados sistemas de información estratégicos. Los SI que usaban las empresas eran considerados como un instrumento simplificador de actividades, siendo una herramienta la cual facilitaba los tramites y reducía la burocracia (Hernández Trasobares, 2016). Con el pasar del tiempo se percataron como los SI les permitía obtener mejores resultados en sus procesos a comparación de sus competidores.

El rápido crecimiento tecnológico, así como el enfoque de las empresas hacia estos SI, ha provocado que estos adquieran características más específicas, ocasionando que sean clasificados en distintos grupos.

El compartir información permite organizar y administrar proyectos de forma más eficiente, además de incrementar la productividad, por esta razón las empresas buscan a través de nuevos tecnológicas mantener el alcance de sus empleados. “Un sistema de colaboración es una aplicación que hace más fácil la tarea de compartir información entre usuarios y dentro de los equipos de trabajo, ayudándolos a comunicarse y a trabajar unidos de manera más efectiva y eficiente” (Corporation, 2001).

La ingeniería de Software basada en Componentes surgió a finales de los 90 como un enfoque basado en la reutilización para el desarrollo de software. Bajo este enfoque, los desarrolladores buscan primero los componentes antes de diseñarlos, de esta forma basan el diseño en componentes disponibles. Dada la versatilidad de producción de estos métodos de trabajo, los desarrolladores fueron creando modelos, siendo estos esquemas en la producción de software.

Un modelo que fomenta el desarrollo de sistemas de software basado en componentes es el Componente Object Model (COM), ya que es un modelo de programación basado en objetos, diseñado para promover interoperabilidad del software (Williams & Kindel, 1994). Esto significa, permitir que dos o más componentes puedan cooperar entre ellos.

Las áreas empresariales no han sido las únicas en aprovechar el modelado de software, pues el constante desarrollo del software ha hecho que este pueda ser implementado en diversos sectores de la sociedad, siendo el área de la educación una de las más sobresalientes en los últimos años con respecto a la implementación de sistemas informativos. “Los catálogos de software educativo suele agrupar programas bajo áreas curriculares: ciencias, matemáticas, música, etc.” (Begoña, 2000). Pero con el tiempo las etiquetas han sido más variadas y complejas, dando como resultado distintos ámbitos de modelos educativos, siendo el acceso a la información uno de los más utilizados.

# **PLANTEAMIENTO DEL PROBLEMA**

La UPT (Universidad Politécnica de Tecámac) cuenta con un modelo educativo denominado ED (Educación Dual), el cual tiene como primer modelo educativo el 09 de junio del 2015 en Alemania. En México (Modelo Mexicano de Formación Dual) ,el 22 de febrero 2017 se lleva a cabo la firma del Convenio SEP-CCE para impulsar el MMFD. La ED consiste en la formación en la empresa se lleva a cabo mediante un plan de formación en el que se definen las competencias que se desarrollarán y/o fortalecerán en la empresa, con incidencia curricular variable, las fases de aprendizaje en la empresa y en la escuela se interrelacionan y existe una valoración de los desempeños. Por esta razón se necesita la implementación de un sistema el cual sea capaz de automatizar los procesos para la administración de los datos que se ingresen.

Se necesita que el sistema centre sus procesos en optimizar lo más posible sus funciones, esto es porque se cuenta con acceso a un sistema funcional, pero debido a que ciertas funciones no satisfacen completamente al usuario, es que se tomó la decisión de crear un sistema el cual sea administrado por la UPT y así este pueda ser adaptado a las medidas que se requieren.

Los estudiantes que sean aceptados en la ED deben tener acceso a ciertas funciones del sistema, como poder ingresar los datos que le sean solicitados, así como el seguimiento de las asignaturas que realicen dentro de la empresa, por esto el mentor empresarial a cargo del proyecto tendrá a disposición un apartado en el cual podrá evaluar al estudiante, de esta manera sustituyendo el uso de papel en este proceso

Se propone una aplicación web, la cual realice la mayoría de las funciones que tiene el sistema actual para la ED, con la diferencia que este cuente con procesos específicos los cuales sean capaces de satisfacer la obtención de cierta información, así como clasificarla dependiendo a sus características, esto para que el usuario pueda tener una clara visualización de los datos que se requieren.

# **JUSTIFICACIÓN**

El sistema de software contará con un formulario el cual nos ayudará a obtener información acerca de los alumnos de la Universidad Politécnica de Tecámac y de tal forma poder generar la hoja automáticamente con dicho sistema sin la necesidad de realizarlo manualmente y así el proceso sea más fácil y no ocurran errores ortográficos, la cual podrá ser bajada en formato xlsx.

El sistema de software estará conectado a una base de datos esto con el fin que todos los procesos tengan un resguardo de la información ingresada, de esta manera se facilita el proceso del sistema dual en la UPT, así mismo para evitar la pérdida de información.

Dado que el software estará conectado a una base de datos, se mostrará una interfaz la cual va a constar de un motor de búsqueda para facilitar al usuario buscar información ya registrada y de esta forma pueda ser consultado rápidamente. Otras de las herramientas importantes que tendrá este es la generación de reportes con todas las características de los estudiantes que están activos e inactivos del sistema dual, cuales obtienen beca y quienes fueron contratados en la empresa.

Este proyecto permitirá demostrar los conocimientos adquiridos y las habilidades desarrolladas a lo largo de la formación académica en la UPT, con el fin de aplicar los principios teóricos y prácticos de cada una de las asignaturas cursadas, como también ofrecer una solución tecnológica a problemas reales enfrentados en el campo laboral.

# **OBJETIVO GENERAL**

* Desarrollar un software para la administración de estudiantes en el sistema de educación dual para agilizar la interacción con este, mediante la recreación digital de los documentos y llenado de los mismo, en una interfaz con lenguaje C# y .NET y así evitar el extravío de documentos físicos y tener el almacenamiento electrónico de estos como de los datos.

**OBJETIVOS ESPECÍFICOS**

* Identificar los elementos que debe contener de las vistas para integrar un levantamiento de requerimientos completo y exitoso.
* Definir los requerimientos funcionales del sistema para desarrollar las vistas adecuadas y que brinden información efectiva.
* Diseñar la maquetación de la vista y base de datos, para el flujo del manejo del sistema con el fin de crear una interfaz amigable para el usuario.
* Diseñar el modelo de base de datos relacional adecuada para la interfaz de usuario para evitar la pérdida y duplicación de datos.
* Documentar el estudio de factibilidad para la administración de tiempos, generando un sistema sin errores, y con la optimización de recursos necesarios.

# **MARCO TEÓRICO**

Para el desarrollo de una aplicación es necesario especificar, como conocer qué tipo de patrón de arquitectura es más adaptable al proyecto y este obtenga mejores resultados durante su ejecución.

El sistema tiene un acceso a base de datos, introducción de datos, despliegue de información, procesos de búsqueda y salida de archivos, por lo que se optó por el patrón de diseño Modelo-Vista-Controlador. “MVC es un patrón de diseño que considera dividir una aplicación en tres módulos claramente identificables y con funcionalidad bien definida: El Modelo, las Vistas y el Controlador” (Bascón Pantoja, 2004).

Modelo: es el conjunto de clases que representan la información del mundo real que el sistema debe procesar, sin tomar en cuenta la forma en que será mostrado ni los mecanismos que hacen que esos datos estén dentro del modelo. El modelo desconoce la existencia de las vistas y el controlador.

Vista: son el conjunto de clases que se encargan de mostrar al usuario la información contenida en el modelo. Una vista está asociada a un modelo, pudiendo existir varias vistas asociadas al mismo modelo.

Controlador: es la clase que se encarga de dirigir el flujo del control de la información que proviene del usuario hacia la aplicación, como datos introducidos u opciones del menú que han sido seleccionadas (Bascón Pantoja, 2004).

Lenguaje de programación: se define como el lenguaje formal o artificial, el cual se basa en reglas gramaticales y bien definidas, es decir la capacidad que tiene una persona para escribir una serie de instrucciones o secuencia de algoritmos a una maquina con el don de encontrar el comportamiento físico y lógico del sistema informático.

Bases de datos dinámicas: principalmente una base de datos es aquel lugar en donde se almacenan diferentes tipos de datos que pueden cambiar con el paso del tiempo, las bases dinámicas son las que nos permiten realizar operaciones de cambio, edición, actualización o borrador de información.

Metodología Scrum: es una serie de pasos para el desarrollo de un proyecto con la finalidad de no cometer errores y saber que secuencia debe seguir este proyecto y en cada fase nos habla un poco de lo que debe realizar.

Programación visual: es una serie de instrucciones que suele implicar o establecer relaciones sistemáticas o geométricas entre las partes del diseño del sistema a desarrollar y una mejor interacción para el usuario con dicho sistema.

La programación visual cumple igualmente los paradigmas clásicos de los lenguajes de programación textuales tales como:

* *Programación imperativa*: cada programa consta de una serie de bloques, cada uno de los cuales realiza una operación simple y cuya ejecución se sucede siguiendo un determinado orden.
* *Programación orientada a objetos:* las aplicaciones se componen de objetos cada uno de los cuales constan de sus propios programas y variables que determinan su comportamiento.

Programación orientada a eventos: los programas pueden activarse ante ciertos sucesos que inicien su ejecución (pulsaciones de teclado, ratón, detección de sensores en dispositivos externos, etc.).

HTML: es el lenguaje con el que se define el contenido de las páginas web. Que se trata de un conjunto de etiquetas que sirven para definir el texto y otros elementos que compondrán una página web, como imágenes, listas, vídeos, etc.

CSS: (en inglés Cascading Style Sheets) es lo que se denomina lenguaje de hojas de estilo en cascada y se usa para estilizar elementos escritos en un lenguaje de marcado como HTML. CSS separa el contenido de la representación visual del sitio.

C#: es un lenguaje de programación desarrollado por Microsoft, orientado a objetos, que ha sido diseñado para compilar diversas aplicaciones que se ejecutan en .NET Framework. Permiten desarrollar aplicaciones rápidamente y mantener la expresividad y elegancia de los lenguajes de estilo de C.

.NET: Se trata de una plataforma para el desarrollo de software que fue lanzada por Microsoft con la finalidad de fusionar su amplio catálogo de productos, que va desde sus múltiples sistemas operativos hasta herramientas de desarrollo o un framework o conjunto de herramientas, tecnologías y servicios que facilita la construcción de todo tipo de aplicaciones (Warren, Coulter, & Dykstra, 2022).

Visual Studio 2022: es el IDE (Integrated Development Environment) de 64 bits más reciente de Microsoft, ofrece un entorno de desarrollo productivo para trabajos de cualquier tamaño. Este IDE junta las tecnologías de desarrollo Microsoft en un solo entorno, facilitando al programador la obtención de estas herramientas.

Microsoft SQL Server: es un sistema de gestión de base de datos relacional, fue desarrollado por Microsoft. El lenguaje de desarrollo que utiliza es Transact-SQL, una implementación del estándar ANSI del lenguaje SQL, es utilizado para manipular, recuperar, crear tablas y definir relaciones entre ellas. (Mishra, 2011)

SSMS: SQL Server Management Studio (SSMS) es un entorno integrado que administra cualquier infraestructura de SQL, desde SQL Server hasta Azure SQL Database. Proporciona herramientas para configurar, monitorear y administrar instancias de SQL Server y bases de datos. Utilizada para monitorear los componentes de nivel de datos que usan las aplicaciones (Randolph, Rabeler, & Leavitt, 2022).

Front-end: Es la parte de una web que conecta e interactúa con los usuarios que la visitan. Es la parte visible, la que muestra el diseño, los contenidos y la que permite a los usuarios navegar por las diferentes páginas. Esta parte engloba y muestra todo el trabajo de diseño web, por lo general reúne en su interior hasta tres leguajes diferentes: HTML, CSS y JavaScript. Cada uno orientado a determinados fines en concreto, se suman para conseguir el resultado final que aparece en la pantalla de cada usuario que entra en una web.

Back-end: es aquella parte encargada de que la lógica de una página web funcione. Son un conjunto y acciones que pasan en una web, pero no son visibles a la vista. El encargado de desarrollar el back-end se encarga de realizar las pruebas necesarias para que el sitio web funcione de manera correcta, el no realizar estas pruebas puede ocasionar fallas al momento de usar página web. Alguna de las funciones que realiza el back-end son: desarrollo de funciones que simplifiquen el proceso de desarrollo, acciones y lógica, conexión con una base de datos, uso de librerías.

Entity Framework Core: Es una versión liviana de Entity Framework que permite a los desarrolladores trabajar con una base de datos utilizando objetos de .NET. Elimina la necesidad de la mayor parte del código de acceso a datos que normalmente debe escribirse. El acceso a datos se realiza mediante un modelo. El objeto de contexto permite consultar y guardar datos (Dykstra & Seders, 2021).

NuGet: es un administrador de paquete diseñado para que los desarrolladores puedas compartir código reutilizable. Es una solución de software como servicio cuya aplicación de cliente es gratuita y de código abierto.

CLI de .NET: es una cadena de herramientas multiplataforma que sirve para desarrollar, compilar, ejecutar y publicar aplicaciones .NET. Se incluye con el SDK de .NET.

Norma ISO 25000: También llamadas como SQuaRE (System and Software Quality Requirements and Evaluation), cuyo propósito es guiar el desarrollo con los requisitos y la evaluación de atributos de calidad, principalmente: la adecuación funcional, eficiencia de desempeño, compatibilidad, capacidad de uso, fiabilidad, seguridad, mantenibilidad y portabilidad (Alfonso, 2012).

**Autorizar a los usuarios: l**a autorización es garantizar que un usuario autenticado tiene los derechos para acceder y modificar datos o servicios. Esto es generalmente administrado ofreciendo algunas pautas de control de acceso dentro de un sistema. Control de acceso puede ser por usuario o por clase de usuario. Clases de usuarios pueden ser definidos por grupos de usuarios, por los roles de usuario, o por listas de las personas y así con cada una de las vistas y contar con una experiencia de usuario agradable y segura.

Calidad de software: un software de calidad debe satisfacer los requisitos dados por el usuario. La obtención de un software con calidad implica la utilización de metodologías o procedimientos estándares para el análisis, diseño, programación y prueba del software que permitan uniformar la filosofía de trabajo, y lograr una mayor confiabilidad, mantenibilidad y facilidad de prueba, a la vez que eleven la [productividad](https://www.ecured.cu/Productividad), tanto para la labor de desarrollo como para el control de la calidad del software.

GitHub: es un sitio web y un servicio en la nube que ayuda a los desarrolladores a almacenar y administrar su código, al igual que llevar un registro y control de cualquier cambio sobre este código del proyecto de software.

Nube**:** es un término que se utiliza para describir una red mundial de servidores, cada uno con una función única. La nube no es una entidad física, sino una red enorme de servidores remotos de todo el mundo que están conectados para funcionar como un único ecosistema.

**Caso de estudio Sistema desarrollo de vistas del software para Sistema Dual**

En la Universidad Politécnica de Tecámac se quiere desarrollar, diseñar e implementar un sistema que facilite el proceso de documentos y registro a dicho sistema, que sea simple, sencillo, eficiente en beneficio para el usuario y al administrador de documentos. El sistema es un punto en donde los administrativos estén enterados de la situación actual de dichos documentos del registro.

Las herramientas para el desarrollo de estos sistemas serian:

* Lenguaje de programación: HTML5, C#, CSS.
* Base de datos: MYSQL (SQL server Management Studio).
* programación visual: (CSS).

**PASO 1: Identificar requisitos funcionales y no funcionales**

Lista de requisitos funcionales para sistema Dual de la Universidad Politécnica de Tecámac:

REQUERIMIENTOS FUNCIONALES

1. Acceso al Docente.
2. Acceso al estudiante.
3. Llenado de datos.
4. Descargar formatos en Excel.

REQUERIMIENTOS NO FUNCIONALES

1. Base de datos relacional que soporte la información que se desea guardar mediante un formulario para la creación del registro y visualización del sistema dual.
2. Servidor o dominio para la implementación de la base de datos.

# **METODOLOGÍA**

Metodología Scrum: Para el desarrollo de este proyecto se lleva a cabo mediante el scrum que es un marco de trabajo colaborativo entre equipos que manejan proyectos complejos especialmente para satisfacer las necesidades del proyecto, donde se les anima para que aprendan a través de las experiencias, a que se auto organicen mientras se trabaja en un determinado problema y reflexionar acerca de sus victorias y derrotas para la mejora continua. Por ende, se puede definir como un método de trabajo ágil que contribuye a realizar la entrega de valor en periodos de tiempo cortos, obteniendo el mejor resultado posible. La gerencia y los equipos de Scrum trabajan juntos alrededor de requisitos y tecnologías para entregar productos funcionando de manera incremental usando el empirismo. Scrum es una gran colección de partes y componentes para definir el proyecto (ED) de forma creativa para afrontar las tareas que día a día el software pueda enfrentar.

ETAPAS DE LA METODOLOGIA SCRUM

1.- Preparación para el juego

Es el instante en donde se establecen la visión y la planificación conforme a los objetivos específicos del proyecto de educación dual. Para poder realizar el análisis, diseño e implementación un poco más detallado y con sus características del sistema y poder llevar a cabo la siguiente fase.

2. Juego

El sprint es un mini-proyecto con una duración no mayor a un mes que se interconecta con otros mini-proyectos para dirigirnos a los objetivos generales y específicos del proyecto general (ED). La etapa del juego es el momento clave dentro del juego de la metodología, así como esprintar es lo básico del scrum. En donde se comienza a desarrollar el proyecto conforme a los tiempos ya planteados como se muestra en el cronograma más adelante, en cada etapa del sistema Dual donde se requiere procesos de terminación en donde se pueda validar y corregir el avance de dicho sistema.

3. Post-juego

Una vez que el producto está terminado (Sistema ED), es necesario entregar toda la documentación que avale el proceso de trabajo y llevar a cabo las pruebas finales y su funcionamiento. Aparente en donde se encuentra toda la evidencia y desarrollo detallado de cómo se fue elaborando y las dificultades con las que se encuentra durante su proceso de desarrollo.

# 

# **CRONOGRAMA**

Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente

# 

# **DESARROLLO DEL PROYECTO**

Antes de iniciar con el proceso de codificación se tuvo una charla, en la cual el cliente especificó las características que el sistema requiere. Se hizo enfoque en automatizar el ingreso y la obtención de datos, por lo que se propuso darle al estudiante un acceso en el cual pueda ingresar o editar los datos que se le solicitan, porque el asesor institucional no siempre cuenta con la información que el sistema solicita. De igual manera el asesor empresarial necesita un acceso al sistema, esto se requiere porque la ED cuenta con asignaturas las cuales son evaluadas semanalmente, por lo que este apartado del sistema se tiene que adaptar para que el asesor empresarial suba la calificación del estudiante, evitando el uso del papel y la firma será de manera digital.

También se solicita que el sistema agrupe a estudiantes dependiendo de ciertas características: activo-inactivo, beca dual, egresado, en empresa. Estas características son mostradas en tablas y cada una tiene la opción de descargar en Excel los datos que en esta se muestran.

Se le otorgo al equipo de desarrollo un archivo en el que se especifica cuáles son las entidades que complementan el sistema, así como los atributos que se requieren de cada uno de ellos.

Dada esta información, se llevó a cabo un análisis más rápido, lo que permitió definir como se iba a realizar el modelado de datos y de qué manera se implementan el sistema, no se cuenta con una base de datos para la ED por parte de la UPT.

Por último, en la charla se propuso desarrollar con .NET Core, esto porque la UPT cuenta con sistemas desarrollados en esta tecnología, lo cual les va a permitir realizar mantenimiento.

En la siguiente figura 1 se muestra una captura de pantalla, en la que se visualiza la herramienta que se utiliza para desarrollar el código en el Back-End, como en Front-End. Para el desarrollo del proyecto es necesario un IDE (Entorno de Desarrollo Integrado) que se propuso trabajar con .NET Core, se opta descargar Visual Studio 2022, el cual cuenta con herramientas para el desarrollo de aplicaciones con esta tecnología. El IDE puede ser descargado desde la página oficial de Visual Studio (<https://visualstudio.microsoft.com/es/vs/>).

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Figura Entorno de desarrollo integrado, Fuente: https://visualstudio.microsoft.com/es/vs

Una vez descargado e instalado el programa, se descarga manualmente algunos de los complementos con los que cuenta este IDE, los cuales son de gran importancia, porque cuentan con los módulos y la lógica de con la que se desarrolla en esta tecnología.

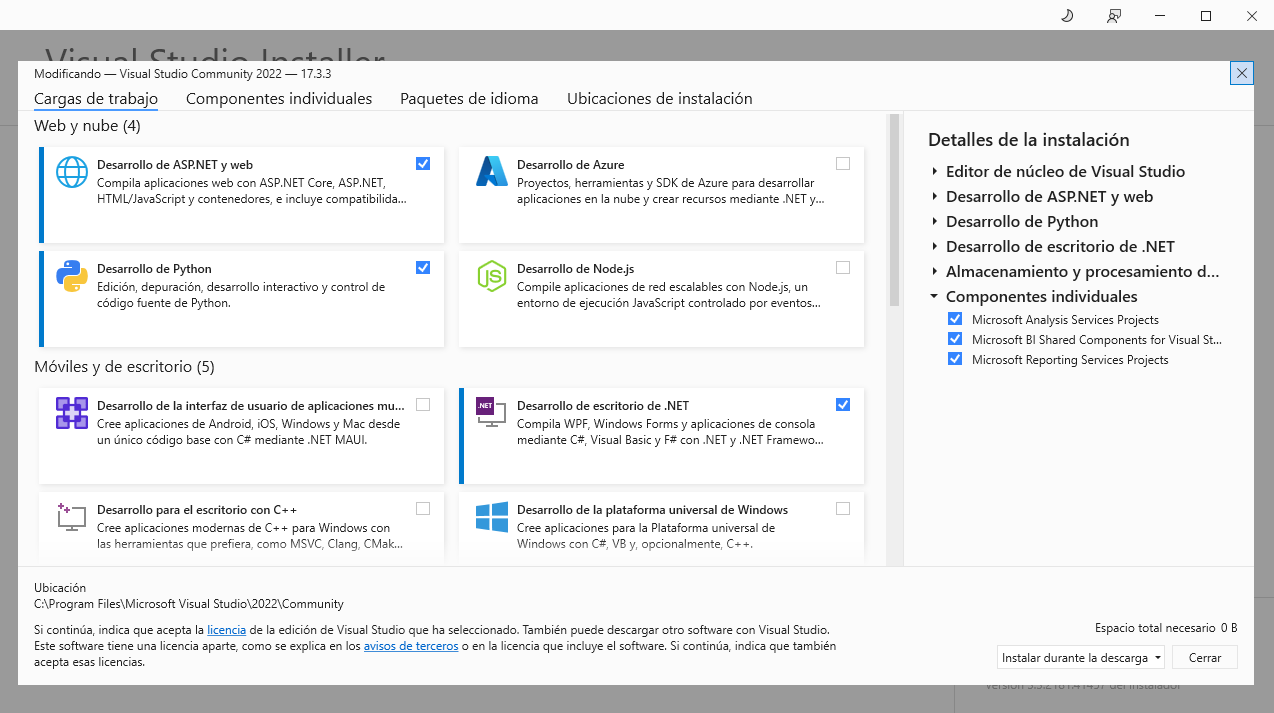


Figura Complementos y herramientas de desarrollo para Visual Studio 2022

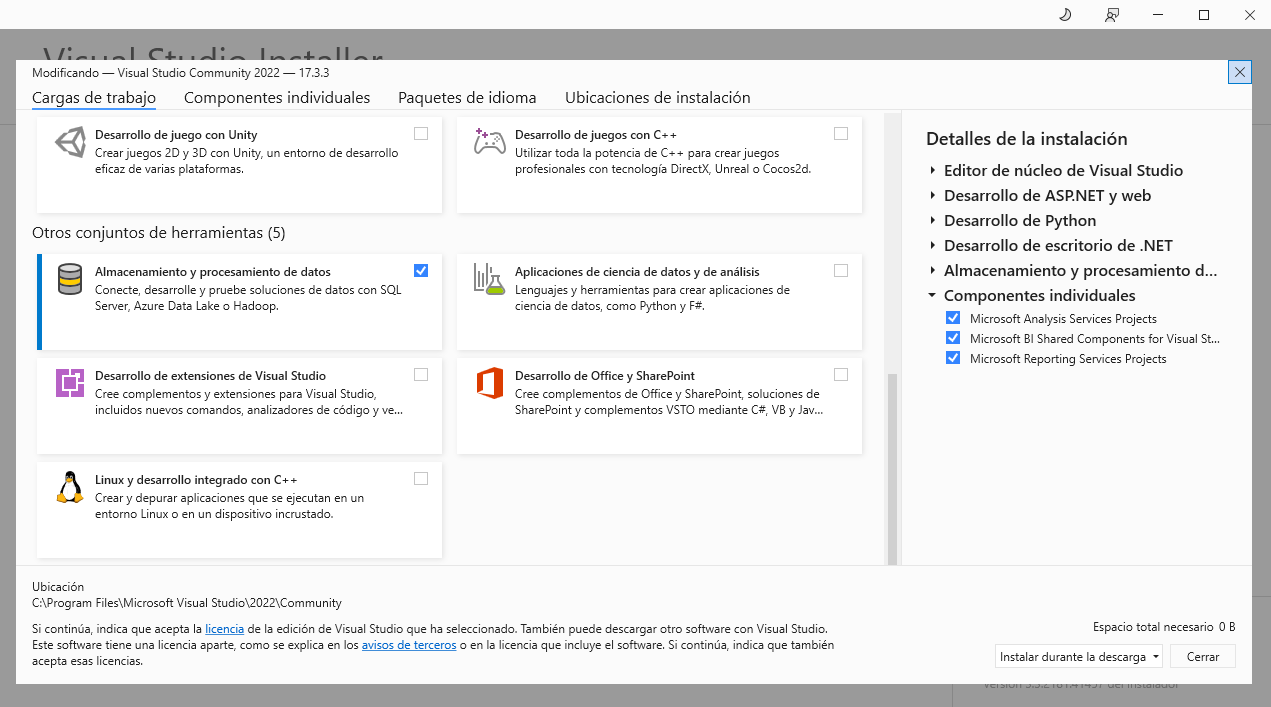


Figura Complemento para administrar bases de datos

La naturaleza de este software requiere de especificaciones medias a altas para poder ejecutarse fluidamente y no siempre se dispone de un ordenador con tales requisitos, por lo que se llevó a cabo una investigación sobre cómo ahorrar lo más posible de memoria durante la ejecución de este programa. Se llegó a una versión de SQL la cual es orientada para el desarrollo de aplicaciones, siendo Express la versión que se utiliza el en desarrollo del sistema. Esta herramienta pude ser descargada desde su página oficial (<https://www.microsoft.com/es-mx/sql-server/sql-server-downloads>).

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura Gestor de base de datos Microsoft SQL, Fuente: <https://www.microsoft.com/es-mx/sql-server/sql-server-downloads>

SSMS es una herramienta la cual permite visualizar e interactuar con bases de datos de SQL, esto hace que se tenga una correcta evaluación de todos los datos entrantes durante el proceso de desarrollo del sistema. Esta herramienta puede ser descargada de manera gratuita desde la página oficial de Microsoft (<https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16>).

Texto

Descripción generada automáticamente

Figura Documentación del gestor de base de datos, Fuente: learn.microsoft.com

Una vez instaladas las herramientas que se utilizan para desarrollar el sistema se empezó con la creación del proyecto, esto es desde el acceso directo que se crea cuando se instala Visual Studio: Nuevo proyecto -> Crear un proyecto ->Aplicación web de ASP.NET Core (Modelo-Vista-Controlador).

Texto

Descripción generada automáticamente

Figura Crear y elegir el proyecto.

Ya que se asignó un nombre al proyecto, el programa pasa a una nueva venta que muestra el editor de código, así como las carpetas y archivos, estos son creados automáticamente por el programa.

Pantalla de computadora con letras

Descripción generada automáticamente con confianza media

Figura Carpetas y archivos del sistema.

Todos los participantes en el desarrollo del sistema contaban con un bajo conocimiento en desarrollo de aplicaciones web con esta tecnología, así que se asigna un tiempo para realizar una investigación acerca de los componentes y funciones que permite realizar. Durante esa investigación se encuentra la documentación de Microsoft (<https://learn.microsoft.com/es-es/aspnet/core/data/ef-mvc/intro?view=aspnetcore-3.1>), la información que se obtuvo sirve para obtener la base de desarrollo con la que se trabaja en esta tecnología.

Para desarrollar una aplicación web se necesitar de “Entity-Framework”, es un paquete que se instala a través de NuGet, este es la base en la cual se desarrollan los métodos para que la aplicación tenga comunicación entre las distintas clases. Además, se instalan otros paquetes que son de utilidad durante el proceso.

Texto

Descripción generada automáticamente

Figura Paquetes NuGet.

Debido a no contar con una base datos previa, se realiza un método de trabajo denominado “First Code”, que permite crear la base de datos en base a los modelos que se programan. Primero se necesita instalar por medio de una línea de comando la herramienta que va a permitir trabajar con migraciones.

Interfaz de usuario gráfica, Aplicación, Sitio web

Descripción generada automáticamente

Figura Comando CLI para instalar dotnet

Para asignar la ruta donde el proyecto se conecta con la base de datos se tiene que ubicar en el archivo *‘appsettings.json’*, donde se le asigna tanto el servidor, nombre de la base de datos, así como un par de características adaptadas de la documentación.

Texto

Descripción generada automáticamente

Figura Conexión a la Base de datos, Fuente: Servidor localhost del proyecto.

Para empezar a trabajar con la base de datos se crean los modelos, siendo estas clases que almacenan la información correspondiente de cada entidad. En estas se especifica cuáles son las distintas relaciones que tiene cada clase y todas las propiedades que se requiere.

Texto

Descripción generada automáticamente

Figura Clase Empresa y sus atributos

Una vez creadas éstas que se requieren para el sistema se necesita una clase, que especifica el nombre que se representa en la base de datos, así como el método con el cual se está haciendo uso de sus propiedades, para esto se crea una carpeta llamada *‘Data’* y un archivo llamado *‘ProgramaDualContext’*.

Texto

Descripción generada automáticamente

Figura Uso de clases en la Base de Datos.

Para hacer uso de las migraciones se necesita un comando para crear la migración hacia la base de datos. Una vez realizado se hace uso de otro comando, que actualiza la base de datos dentro del SSMS.

Interfaz de usuario gráfica, Texto, Sitio web

Descripción generada automáticamente

Figura Comando CLI para crear una migración.

Interfaz de usuario gráfica, Texto, Sitio web

Descripción generada automáticamente

Figura Comando CLI para actualizar la Base de Datos.

Texto

Descripción generada automáticamente El enfoque de la investigación es implementar métodos para agilizar la construcción de funciones para cada modelo, es así como se crea un par de carpetas: *‘Implementation’, ‘Interfaces’*, la carpeta interfaces con tiene la interfaz en la cual se le asigna los métodos que utiliza los modelos *‘IGenericRepository’*.

Figura Creación del repository.

En Implementación se desarrolla la lógica la cual es de gran ayuda que permite reutilizar funciones ahorrando tiempo. Los métodos implementados dentro de la clase *‘GenericRepository’* son: Crear, Editar, Consultar, Eliminar y Obtener.

Para que cada modelo tenga sus respectivos métodos, se crean dos clases en cada carpeta, así como GenericRepository, cada modelo tiene su interfaz en la cual se le asigna los métodos que cada uno usa.

Se priorizo comenzar con el desarrollo de la entidad del Estudiante, esta cuenta con un requerimiento de datos mayor, esto también para agilizar el proceso de desarrollo de las demás.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Figura Carpetas Implementation e Interface.

Antes de empezar a desarrollar el controlador de cada clase es necesario crear “ViewModels”, estas son clases que reflejan los datos de cada modelo, pero añadiendo o conservando datos para interactuar ampliamente con todo el sistema.

El paquete NuGet “AutoMapper” es una herramienta que mapea los datos relacionados entre el ViewModel y el Modelo original, ahorrando la codificación de cada atributo, también cuenta con métodos que permiten heredar desde otras clases, así como enviar información e ignorar ese mapeo para enviar solo los datos correspondientes.

Texto

Descripción generada automáticamente

Figura Mapeo de ViewModels con AutoMapper.

Texto

Descripción generada automáticamente

Figura ViewModels.

Se especifica tanto ViewModels como el AutoMapper, se continua con el controlador, en este se ejecuta la lógica dentro del servidor y se hace uso de los métodos ya realizados. Principalmente se crea el controlador del estudiante con los métodos de: Lista, Crear, Editar, Eliminar.

Texto

Descripción generada automáticamente

Figura Servicios y Controladores

Para realizar la comunicación entre la vista que tiene el usuario con el controlador, se desarrolló un archivo JavaScript ubicado en *‘wwwroot/js/vistas’*, esta carpeta contiene las vistas de cada modelo requerido. Cada script contiene los datos que se le muestran al usuario, así como las funciones que se comunican con el controlador.

Esto permite hacer uso de herramientas fuera de .NET Core, ampliando la versatilidad del sistema y otorgando a los desarrolladores habilidad para desarrollar con fluidez los métodos básicos de un sistema de información.

Texto

Descripción generada automáticamente

Figura Ajax para mostrar los datos.

**PASO 1: Identificar requisitos funcionales y no funcionales**

REQUERIMIENTOS FUNCIONALES

1. Acceso al Docente.
2. Acceso al estudiante.
3. Llenado de datos.
4. Descargar formatos en Excel.

REQUERIMIENTOS NO FUNCIONALES

1. Seguridad en el proceso de desarrollo del sistema esto conlleva a seguir buenas prácticas de programación, así como una correcta creación de la base de datos que a su vez esto conlleva a seguir una correcta relación entre los datos y las relaciones que tenga la base de datos, además de poder tener un dominio confiable y un buen servidor al momento de pasar a la fase de implementación del sistema
2. Flexibilidad para poder ser utilizado en cualquier dispositivo cuando sea presentado al usuario final, de igual forma se busca que pueda al sistema se le puedan añadir tecnologías distintas al momento que este sistema tenga una mayor demanda de usuarios.
3. Base de datos relacional que soporte la información que se desea guardar mediante un formulario para la creación del registro y visualización del sistema dual.
4. Servidor o dominio para la implementación de la base de datos.

Posteriormente se muestra un ejemplo de una plantilla para el levantamiento de requerimientos funcionales y no funciones, en la que es necesaria para el vaciando de información, los requerimientos son fundamentales para conocer el inicio del desarrollo del proyecto que este contiene a detalle las características que son fundamentales para su desarrollo e implementación del proyecto.

|  |  |
| --- | --- |
| **N°RF** | **Nombre del requerimiento funcional o no funcional** |
| **Versión** |  |
| **Autor(es)** |  |
| **Objetivos** |  |
| **Requisitos** |  |
| **Descripción** |  |
| **Precondición** |  |
| **Secuencia normal** |  |
|  |
|  |
|  |
|  |
| **Excepciones** |  |
|  |
|  |
| **Postcondición** |  |
| **Importancia** |  |
| **Prioridad/urgencia** |  |
| **Comentarios** |  |

Tabla Ejemplo de la plantilla de levantamiento de requerimientos Fuente: Original.

|  |  |
| --- | --- |
| **RF-01** | **Acceso al Docente** |
| **Versión** | 1.0 |
| **Autor(es)** | Anacleto Angeles Aritzi Denalyh Estrada Mejía Raúl y Herrera Santiago Ulises |
| **Objetivos** | Controlar el acceso de usuarios a al sistema |
| **Requisitos** | Registrar información del administrador: Correo y/o Matricula, Contraseña. |
| **Descripción** | El usuario deberá seguir los pasos descritos aquí para poder realizar el acceso al sistema. |
| **Precondición** | El administrador deberá contar con su número de matrícula y una contraseña para poder ingresar al sistema. |
| **Secuencia normal** | 1. El administrador realiza formulario de registro mediante el permiso de las autoridades y es que no cuenta con un usuario. |
| 1. El administrador solicita el ingreso al sistema contando con su número de matrícula y su contraseña |
| 1. El sistema valida los datos (autenticar usuario y contraseña) |
| 1. Si la validación es correcta el usuario podrá ingresar al sistema |
|  |
|  |
|  |
| **Excepciones** | 1. Si la validación de datos es incorrecta el sistema arroja un mensaje de error. |
|  |
|  |
| **Pos condición** | Si el usuario y la contraseña del administrador son correctos accede al sistema. |
| **Importancia** | Importancia del requisito para el usuario **Vital** |
| **Prioridad/urgencia** | Prioridad del requisito para la dirección del sistema **Inmediata** |
| **Comentarios** |  |

Tabla Acceso al docente, Fuente: Original.

|  |  |
| --- | --- |
| **RF-02** | **Acceso al Estudiante** |
| **Versión** | 1.0 |
| **Autor(es)** | Anacleto Angeles Aritzi Denalyh Estrada Mejía Raúl y Herrera Santiago Ulises |
| **Objetivos** | Controlar el acceso de usuarios al sistema. |
| **Requisitos** | Registrar información del administrador: Correo y/o Matricula, Contraseña. |
| **Descripción** | El estudiante deberá seguir los pasos descritos aquí para poder realizar el acceso al sistema. |
| **Precondición** | El estudiante deberá contar con su número de matrícula y una contraseña para poder ingresar al sistema. |
| **Secuencia normal** | 1. El estudiante realiza formulario de registro. |
| 1. El estudiante solicita el ingreso al sistema contando con su número de matrícula y su contraseña. |
| 1. El sistema valida los datos (autenticar usuario y contraseña). |
| 1. Si la validación es correcta el usuario podrá ingresar al sistema. |
|  |
|  |
|  |
| **Excepciones** | 1. Si la validación de datos es incorrecta el sistema arroja un mensaje de error. |
|  |
|  |
| **Pos condición** | Si el usuario y la contraseña del estudiante son correctos accede al sistema. |
| **Importancia** | Importancia del requisito para el sistema **Vital** |
| **Prioridad/urgencia** | Prioridad del requisito para la dirección del proyecto **Inmediata** |
| **Comentarios** |  |

Tabla Acceso al estudiante, Fuente: original.

|  |  |
| --- | --- |
| **RF-03** | **Llenado de datos para el registro a ED** |
| **Versión** | 1.0 |
| **Autor(es)** | Anacleto Angeles Aritzi Denalyh Estrada Mejía Raúl y Herrera Santiago Ulises |
| **Objetivos** | Complementar los datos para el seguimiento de este proceso. |
| **Requisitos** | Datos del estudiante: matricula, nombre completo, grupo, cuatrimestre, ciclo, etc.  Datos del docente (UE): correo y/o matricula, contraseña. |
| **Descripción** | El estudiante deberá llenar el formulario establecido en el sistema para llenar un nuevo registro de alumno dual, la cual se almacenará en la base de datos. |
| **Precondición** | El estudiante deberá contar con los datos de acceso ya al sistema. |
| **Secuencia normal** | 1. El estudiante solicita el ingreso al sistema contando con su número de matrícula y/o correo y su contraseña. 2. El estudiante realiza formulario de registro. Los datos son: Apellido paterno, Apellido materno, Nombre, Matricula, Correo electrónico, teléfono celular, Cup, Numero de seguro social, Sexo y contraseña. |
| 1. El docente solicita el ingreso al sistema contando con su número de matrícula y/o correo y su contraseña. 2. El docente realiza formulario de registro. Los datos son: Apellido paterno, Apellido materno, Nombre, Matricula, Correo electrónico, teléfono celular, Cup, Numero de seguro social, Sexo y contraseña. |
| 1. El estudiante captura sus datos solicitados para continuar con el proceso de ED los cuales son: Apellido paterno, Apellido materno, Nombre, teléfono, Correo, Cuatrimestre, Promedio, Fecha de registro, Programa educativo, Domicilio y estatus. |
| 1. El docente guarda los datos capturados para así tener un registro. |
|  |
| **Excepciones** | 1. Si el estudiante ya se encuentra registrado, el sistema arrojara una alerta de estudiante ya registrado anteriormente. |
| 1. Si el docente ya se encuentra registrado, el sistema arrojara una alerta de docente ya registrado anteriormente. |
|  |
| **Pos condición** | 1. Se deben de llenar todos los campos del formulario de lo contrario el sistema no podrá guardar un registro y mandará una alerta de error. |
| **Importancia** | Importancia del requisito para la usuaria alta |
| **Prioridad/urgencia** | Prioridad del requisito para la dirección del sistema alta |
| **Comentarios** |  |

Tabla Llenado de datos, Fuente: original

|  |  |
| --- | --- |
| **RF-04** | **Llenado de datos Asesor Institucional** |
| **Versión** | 1.0 |
| **Autor(es)** | Anacleto Angeles Aritzi Denalyh Estrada Mejía Raúl y Herrera Santiago Ulises |
| **Objetivos** | Complementar los datos para el seguimiento de este proceso. |
| **Requisitos** | Datos del estudiante: matricula, nombre completo, grupo, cuatrimestre, ciclo, etc.  Datos del docente (UE): correo y/o matricula, contraseña. |
| **Descripción** | El estudiante deberá llenar el formulario establecido en el sistema para llenar un nuevo registro de alumno dual, la cual se almacenará en la base de datos. |
| **Precondición** | El estudiante deberá contar con los datos de acceso ya al sistema. |
| **Secuencia normal** | 1. El estudiante solicita el ingreso al sistema contando con su número de matrícula y/o correo y su contraseña. 2. El estudiante realiza formulario de registro. Los datos son: Apellido paterno, Apellido materno, Nombre, Matricula, Correo electrónico, teléfono celular, Cup, Numero de seguro social, Sexo y contraseña. |
| 1. El docente solicita el ingreso al sistema contando con su número de matrícula y/o correo y su contraseña. 2. El docente realiza formulario de registro. Los datos son: Apellido paterno, Apellido materno, Nombre, Matricula, Correo electrónico, teléfono celular, Cup, Numero de seguro social, Sexo y contraseña. |
| 1. El estudiante captura sus datos solicitados para continuar con el proceso de ED los cuales son: Apellido paterno, Apellido materno, Nombre, teléfono, Correo, Fecha de registro y Domicilio. |
| 1. El docente guarda los datos capturados para así tener un registro. |
|  |
| **Excepciones** | 1. Si el estudiante ya se encuentra registrado, el sistema arrojara una alerta de estudiante ya registrado anteriormente. |
| 1. Si el docente ya se encuentra registrado, el sistema arrojara una alerta de docente ya registrado anteriormente. |
|  |
| **Pos condición** | 1. Se deben de llenar todos los campos del formulario de lo contrario el sistema no podrá guardar un registro y mandará una alerta de error. |
| **Importancia** | Importancia del requisito para la usuaria alta |
| **Prioridad/urgencia** | Prioridad del requisito para la dirección del sistema alta |
| **Comentarios** |  |

Tabla 5 Llenado de datos, Fuente: original

|  |  |
| --- | --- |
| **RF-05** | **Llenado de datos Asesor Empresarial** |
| **Versión** | 1.0 |
| **Autor(es)** | Anacleto Angeles Aritzi Denalyh Estrada Mejía Raúl y Herrera Santiago Ulises |
| **Objetivos** | Complementar los datos para el seguimiento de este proceso. |
| **Requisitos** | Datos del estudiante: matricula, nombre completo, grupo, cuatrimestre, ciclo, etc.  Datos del docente (UE): correo y/o matricula, contraseña. |
| **Descripción** | El estudiante deberá llenar el formulario establecido en el sistema para llenar un nuevo registro de alumno dual, la cual se almacenará en la base de datos. |
| **Precondición** | El estudiante deberá contar con los datos de acceso ya al sistema. |
| **Secuencia normal** | 1. El estudiante solicita el ingreso al sistema contando con su número de matrícula y/o correo y su contraseña. 2. El estudiante realiza formulario de registro. Los datos son: Apellido paterno, Apellido materno, Nombre, Matricula, Correo electrónico, teléfono celular, Cup, Numero de seguro social, Sexo y contraseña. |
| 1. El docente solicita el ingreso al sistema contando con su número de matrícula y/o correo y su contraseña. 2. El docente realiza formulario de registro. Los datos son: Apellido paterno, Apellido materno, Nombre, Matricula, Correo electrónico, teléfono celular, Cup, Numero de seguro social, Sexo y contraseña. |
| 1. El estudiante captura sus datos solicitados para continuar con el proceso de ED los cuales son: Apellido paterno, Apellido materno, Nombre, teléfono, Correo, Fecha de registro y Domicilio. |
| 1. El docente guarda los datos capturados para así tener un registro. |
|  |
| **Excepciones** | 1. Si el estudiante ya se encuentra registrado, el sistema arrojara una alerta de estudiante ya registrado anteriormente. |
| 1. Si el docente ya se encuentra registrado, el sistema arrojara una alerta de docente ya registrado anteriormente. |
|  |
| **Pos condición** | 1. Se deben de llenar todos los campos del formulario de lo contrario el sistema no podrá guardar un registro y mandará una alerta de error. |
| **Importancia** | Importancia del requisito para la usuaria alta |
| **Prioridad/urgencia** | Prioridad del requisito para la dirección del sistema alta |
| **Comentarios** |  |

Tabla 6 Llenado de datos AE, Fuente: original

|  |  |
| --- | --- |
| **RF-06** | **Llenado de datos de Empresa** |
| **Versión** | 1.0 |
| **Autor(es)** | Anacleto Angeles Aritzi Denalyh Estrada Mejía Raúl y Herrera Santiago Ulises |
| **Objetivos** | Complementar los datos para el seguimiento de este proceso. |
| **Requisitos** | Datos del estudiante: matricula, nombre completo, grupo, cuatrimestre, ciclo, etc.  Datos del docente (UE): correo y/o matricula, contraseña. |
| **Descripción** | El estudiante deberá llenar el formulario establecido en el sistema para llenar un nuevo registro de alumno dual, la cual se almacenará en la base de datos. |
| **Precondición** | El estudiante deberá contar con los datos de acceso ya al sistema. |
| **Secuencia normal** | 1. El estudiante solicita el ingreso al sistema contando con su número de matrícula y/o correo y su contraseña. 2. El estudiante realiza formulario de registro. Los datos son: Apellido paterno, Apellido materno, Nombre, Matricula, Correo electrónico, teléfono celular, Cup, Numero de seguro social, Sexo y contraseña. |
| 1. El docente solicita el ingreso al sistema contando con su número de matrícula y/o correo y su contraseña. 2. El docente realiza formulario de registro. Los datos son: Apellido paterno, Apellido materno, Nombre, Matricula, Correo electrónico, teléfono celular, Cup, Numero de seguro social, Sexo y contraseña. |
| 1. El estudiante captura sus datos solicitados para continuar con el proceso de ED los cuales son: Nombre, Razón social, Sector social, Representante legal, teléfono, Correo, Fecha de registro y Domicilio. |
| 1. El docente guarda los datos capturados para así tener un registro. |
|  |
| **Excepciones** | 1. Si el estudiante ya se encuentra registrado, el sistema arrojara una alerta de estudiante ya registrado anteriormente. |
| 1. Si el docente ya se encuentra registrado, el sistema arrojara una alerta de docente ya registrado anteriormente. |
|  |
| **Pos condición** | 1. Se deben de llenar todos los campos del formulario de lo contrario el sistema no podrá guardar un registro y mandará una alerta de error. |
| **Importancia** | Importancia del requisito para la usuaria alta |
| **Prioridad/urgencia** | Prioridad del requisito para la dirección del sistema alta |
| **Comentarios** |  |

Tabla 7 Llenado de datos Empresa, Fuente: original

|  |  |
| --- | --- |
| **RF-07** | **Descargar formatos en Excel** |
| **Versión** | 1.0 |
| **Autor(es)** | Anacleto Angeles Aritzi Denalyh Estrada Mejía Raúl y Herrera Santiago Ulises |
| **Objetivos** | descargar el formato generado para obtención de todos los datos que la UE necesita |
| **Requisitos** | Contar con las credenciales del docente para poder descargarlo. |
| **Descripción** | El docente deberá descargar el formato de Excel para poder seguir con el proceso. |
| **Precondición** | El docente deberá contar con los datos de acceso ya al sistema. |
| **Secuencia normal** | 1. El docente solicita el ingreso al sistema contando con su número de matrícula y/o correo y su contraseña. 2. El docente realiza formulario de registro. Los datos son: Apellido paterno, Apellido materno, Nombre, Matricula, Correo electrónico, teléfono celular, Cup, Numero de seguro social, Sexo y contraseña |
| 1. El docente deberá realizar un estatus de la información que necesita y verificar que esta sea correcta y precisa. |
| 1. El docente se dirige al botón de descargar formato en .XML. |
| 1. El docente podrá abrir el formato descargado en Excel. Con un esquema ya establecido |
|  |
|  |
|  |
| **Excepciones** | 1. Si el formato que se desea descargar en formato .XML no se ha guardado en la base de datos no se podrá descargar. |
|  |
|  |
| **Pos condición** | 1. Deberá encontrarse registrado el dato que se desea encontrar de lo contrario no arrojará ningún resultado el sistema. |
| **Importancia** | Importancia del requisito para usuario alta |
| **Prioridad/urgencia** | Prioridad del requisito para la dirección del sistema media |
| **Comentarios** |  |

Tabla 8 Descargar Formato en Excel, Fuente: original

LEVANTAMIENTO DE REQUERIMIENTOS NO FUNCIONALES

|  |  |
| --- | --- |
| **RNF-01** | Base de datos optimizada |
| **Versión** | 1.0 |
| **Autor(es)** | Anacleto Angeles Aritzi Denalyh Estrada Mejía Raúl y Herrera Santiago Ulises |
| **Objetivos** | Guardas los registros de usuario y así mismo poder interactuar con esos datos |
| **Requisitos** | Contar con una base de datos en cualquier gestor de base de datos en este caso será Microsoft SQL |
| **Descripción** | El desarrollador deberá crear una base de datos en la cual se deberán crear tablas para almacenar  Los datos de los registros de clientes y registros de administradores. |
| **Precondición** | El desarrollador deberá contemplar los atributos de cada tabla para poder crearla y no falte ningún dato importante por capturar. |
| **Secuencia normal** | 1. El usuario deberá rellenar el formulario del sistema con sus datos |
| 1. El usuario deberá rellenar sus datos en el sistema. |
| **Excepciones** |  |
|  |
|  |
| **Pos condición** | 1. Deberá encontrarse registrado el dato que se desea encontrar de lo contrario no arrojará ningún resultado el sistema. |
| **Importancia** | Importancia del requisito para el usuario alta |
| **Prioridad/urgencia** | Prioridad del requisito para la dirección del sistema media |

Tabla 9............, Fuente: original.

|  |  |
| --- | --- |
| **RNF-02** | **Seguridad en el proceso de desarrollo del sistema y protección de datos** |
| **Versión** | 1.0 |
| **Autor(es)** | Anacleto Angeles Aritzi Denalyh Estrada Mejía Raúl y Herrera Santiago Ulises |
| **Objetivos** | Contar con un alto nivel de seguridad en el proyecto para que la información de los usuarios este bien resguardada |
| **Requisitos** | Contar con un buen servidor o dominio, así como de seguir buenas practicas al momento de desarrollar el sistema, de igual manera seguir estándares correctos para la protección de los datos de los usuarios que se registren |
| **Descripción** | Los desarrolladores serán los encargados de seguir buenas prácticas de programación, así como de implementación para que los usuarios finales puedan tener un sistema que pueda satisfacer las necesidades de estos, de igual manera el tener una buena protección de datos por medio de certificados SSL o encriptación de datos importantes |
| **Precondición** | Los desarrolladores deberán seguir buenas prácticas de programación para que el sistema funcione de la manera más optima posible |
| **Secuencia normal** | Los desarrolladores deberán crear un sistema útil |
|  |
| **Excepciones** |  |
| **Postcondición** |  |
| **Importancia** | Importancia para los desarrolladores alta |
| **Prioridad/urgencia** | Prioridad del requisito para la dirección del sistema alta |

Tabla 10 , Fuente: Original

|  |  |
| --- | --- |
| **RNF-03** | **Flexibilidad para poder ser utilizado en cualquier dispositivo cuando sea presentado al usuario final** |
| **Versión** | 1.0 |
| **Autor(es)** | Anacleto Angeles Aritzi Denalyh Estrada Mejía Raúl y Herrera Santiago Ulises |
| **Objetivos** | Contar con un sistema que pueda ser utilizado en cualquier dispositivo esto para que pueda ser un sistema útil para cualquier usuario |
| **Requisitos** | Desarrollar un sistema que sea flexible y útil para cualquier usuario y que este mismo no se limite a ser utilizado en un solo ambiente de trabajo |
| **Descripción** | Los desarrolladores serán los encargados de crear un sistema capaz de satisfacer las necesidades de los usuarios que quieran acceder a el |
| **Precondición** | Los desarrolladores deberán crear el sistema con base a los estándares que se establezca con base al diseño y facilidad de uso que el cliente requiera |
| **Secuencia normal** | Los desarrolladores deberán crear un sistema útil |
|  |
| **Excepciones** |  |
| **Postcondición** |  |
| **Importancia** | Importancia para los desarrolladores alta |
| **Prioridad/urgencia** | Prioridad del requisito para la dirección del sistema alta |

Tabla 10 , Fuente: Original

|  |  |
| --- | --- |
| **RNF-04** | **Servidor o dominio para la implementación de la base de datos.** |
| **Versión** | 1.0 |
| **Autor(es)** | Anacleto Angeles Aritzi Denalyh Estrada Mejía Raúl y Herrera Santiago Ulises |
| **Objetivos** | Contar con un buen servidor para poder desplegar la base de datos y así poder trabajar de manera más optima |
| **Requisitos** | Contar con un buen servidor o dominio, en el que se pueda hacer un buen despliegue de la base de datos |
| **Descripción** | Los desarrolladores serán los encargados de configurar el servidor en el que se desplegara la base de datos esto para que se pueda trabajar de una manera más optima al momento de hacer consultas a los diversos datos que se quieran solicitar |
| **Precondición** | Los desarrolladores se encargarán de desplegar la base de datos en el servidor |
| **Secuencia normal** | Los desarrolladores deberán hacer una buena configuración del servidor |
|  |
| **Excepciones** |  |
| **Postcondición** |  |
| **Importancia** | Importancia para los desarrolladores alta |
| **Prioridad/urgencia** | Prioridad del requisito para la dirección del sistema alta |

Tabla 11 , Fuente: Original

Para la implementación del guardado de datos se necesita el diseño y el desarrollo de una base de datos, en la cual se encuentra la información y datos que se implementen en el sistema ED, como los datos personales de los usuarios y los datos sobre las empresas que estén dadas de alta en la Universidad.

El uso de esta herramienta SQL server management studio 18 brinda poder hacer la construcción de las tablas de los usuarios, empresas y docentes en donde se encuentran los datos personales y necesarios para el funcionamiento adecuado del guardado y respaldo de datos, con el fin de que sea más fácil identificar una búsqueda y extracción de información cuando sea necesaria. Para la creación de la base de datos se van a establecer los tipos de datos y los tamaños que se utilizan.

El tipo y tamaño de datos que se ocupan son los siguientes:

* **Int:** son datos numéricos en donde se puede almacenar de 1 hasta 16 bytes.
* **String**: es una secuencia ordenada de letras dentro de una cadena.
* **Varchar:** almacena una serie de caracteres que puede almacenar hasta 255 bytes en un campo de longitud.
* **Datetime:** Almacena una fecha (año-mes-día) con una hora (horas-minutos-segundos).
* **Float**: son datos numéricos que admite una parte en decimal.

Para la realización de las tablas de estudiante, empresa y universidad se utiliza previamente el guardado de sus datos para poder realizar el llenado de datos correcto, y los campos contienen información que es muy importante para su conclusión de la petición ya sea para nuevo registro o cambio de información en el sistema, cabe señalar que estas tablas se relacionan con otras dos tablas, las cuales son importantes que ayudan en el almacenamiento de los id y así poder tener una relación correcta entre ellas.

En el modelo del estudiante se hace uso de un id (al igual que en todos los modelos) el cual sirve como identificador de esa entidad, también sirve para indicar al software cual elemento es el id, así como relacionar los datos con otros modelos que necesiten heredar o recibir atributos de otros.

Las tablas que se desarrollan dentro del proyecto de Educación dual son nombradas como:

* Tabla Estudiantes.
* Tabla Estudiante-Asignaturas.
* Tabla Asesor Institucional.
* Tabla de Asignaturas.
* Tabla de Catálogo Proyecto.
* Tabla Domicilios.
* Tabla Responsable Institucional.
* Tabla Universidad.
* Tabla Mentor académico.
* Tabla Empresas.
* Tabla Mentor empresarial.

Para el desarrollo con Back-End, se desarrollaron funciones donde se especifican las rutas que se necesitan para el consumo de los endpoints para el correcto desarrollo y funcionamiento del proyecto, cabe mencionar que dentro del desarrollo del Back-End se tiene un apartado para la conectividad con la base de datos por otro lado, en la parte del Front-End se ocupa el consumo de los endpoints para la creación de métodos o funciones y así poder conectar el Back- End con Front-End y tener un sistema funcional. También cabe señalar que el lenguaje de programación C# utilizado a lo largo de este proyecto es uno de los más utilizados en la actualidad y se considera que es estándar dentro del funcionamiento con su sistema de ED que solicito este aplicativo.

Junto con el framework .NET Core, existe Entity Framework, siendo Microsoft.EntityFrameworkCore el framework, Microsoft.EntityFrameworkCore.SqlServer para administrar bases de datos y Microsoft.EntityFrameworkCore.Tools las herramientas para desarrollar.

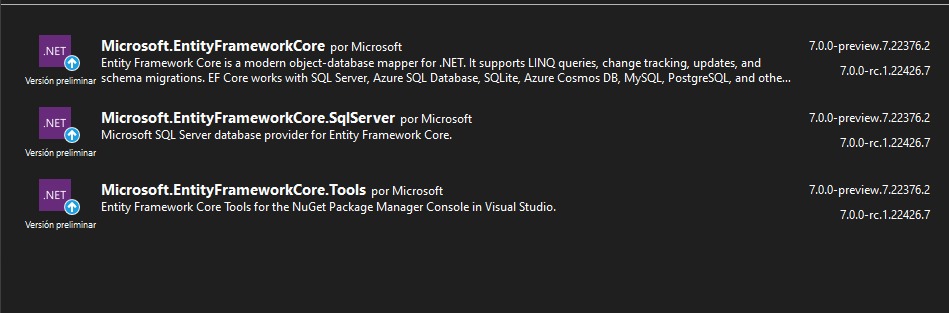


Figura Paquetería NuGet, Fuente: Servidor localhost del proyecto.

Con esta herramienta de base de datos, encargada de correr la API[[1]](#footnote-1) es de gran ayuda porque con ella se realiza el correcto funcionamiento del sistema, se pueden consumir todos los endpoints que se generaron por medio de las funciones, Marca errores en la terminal de visual c#.

Para la prueba de funcionamiento con la API, esta se corre mediante un botón que se ejecuta por medio de este c#, esto permite la conectividad con la base de datos y con el desarrollo del Front-end, y así todos los datos puedan guardarse de forma segura y sin mandar errores a la hora del almacenamiento de datos, si el sistema encuentra alguna falla esta se muestra en consola del navegador en donde se esté abriendo el proyecto.

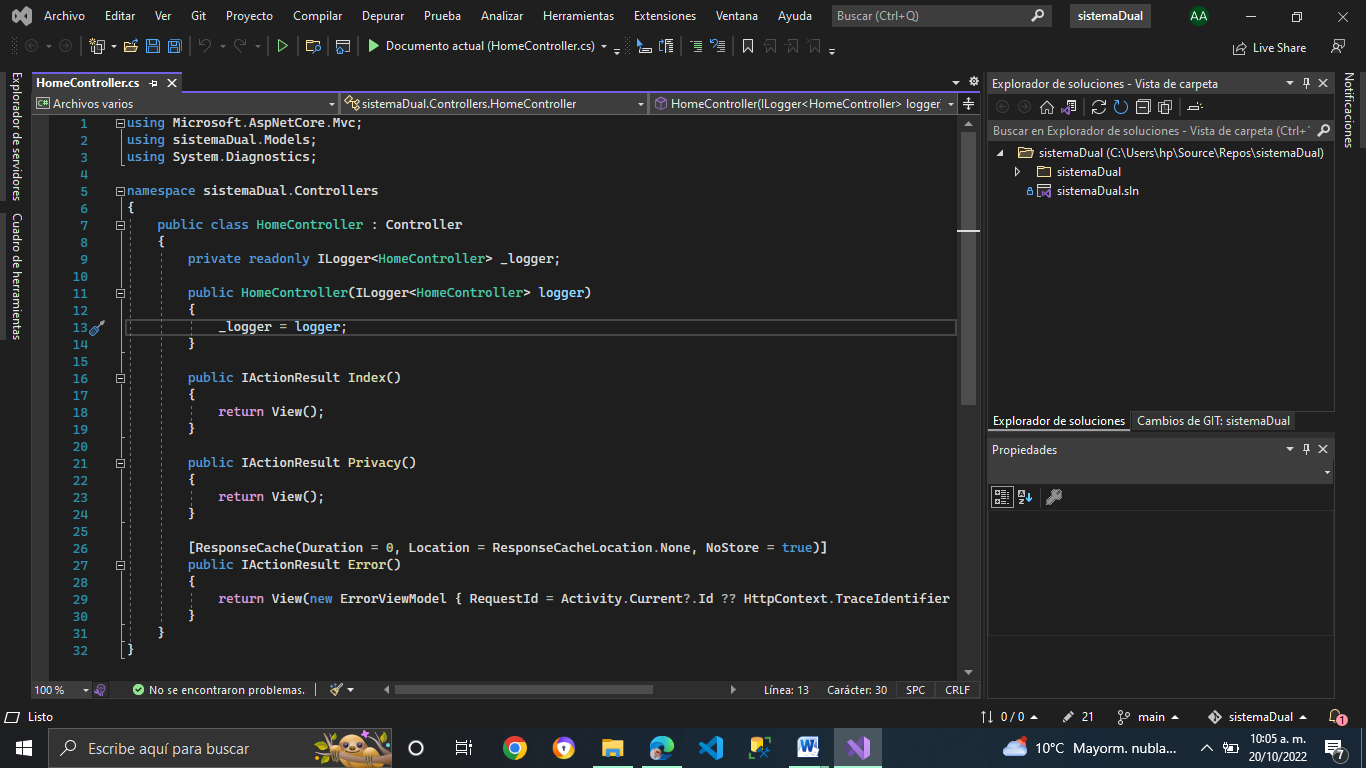


Figura Botón para correr la API, Fuente: Servidor localhost del proyecto.

Para la realización del modelado de base de datos, que para su implementación se requiere tener ya instalado visual studio y los paquetes Nugets como ya se mencionó anteriormente, como herramienta se utiliza Code Firts en donde se comienza realizando todas nuestras clases en el leguaje que se desea, es decir en C# para posteriormente crear los modelos y así automáticamente generar la base de datos.

Se muestra la herramienta llamada CSS y Booststrap que se ocupa para darle un diseño agradable y sencillo para el usuario; esta herramienta es utilizada a lo largo de todo el proceso de desarrollo, la cual ayuda a dar tamaño y tipo de letra, colores, etc. Para la elección de colores del proyecto se ocupan los colores en código Hexadecimal proporcionado por el código de cada color.

Para dar formato a las vistas del proyecto de ED, además se agregó algunos componentes del Framework Booststrap en donde se muestra documentación de su implementación con HTML como son:

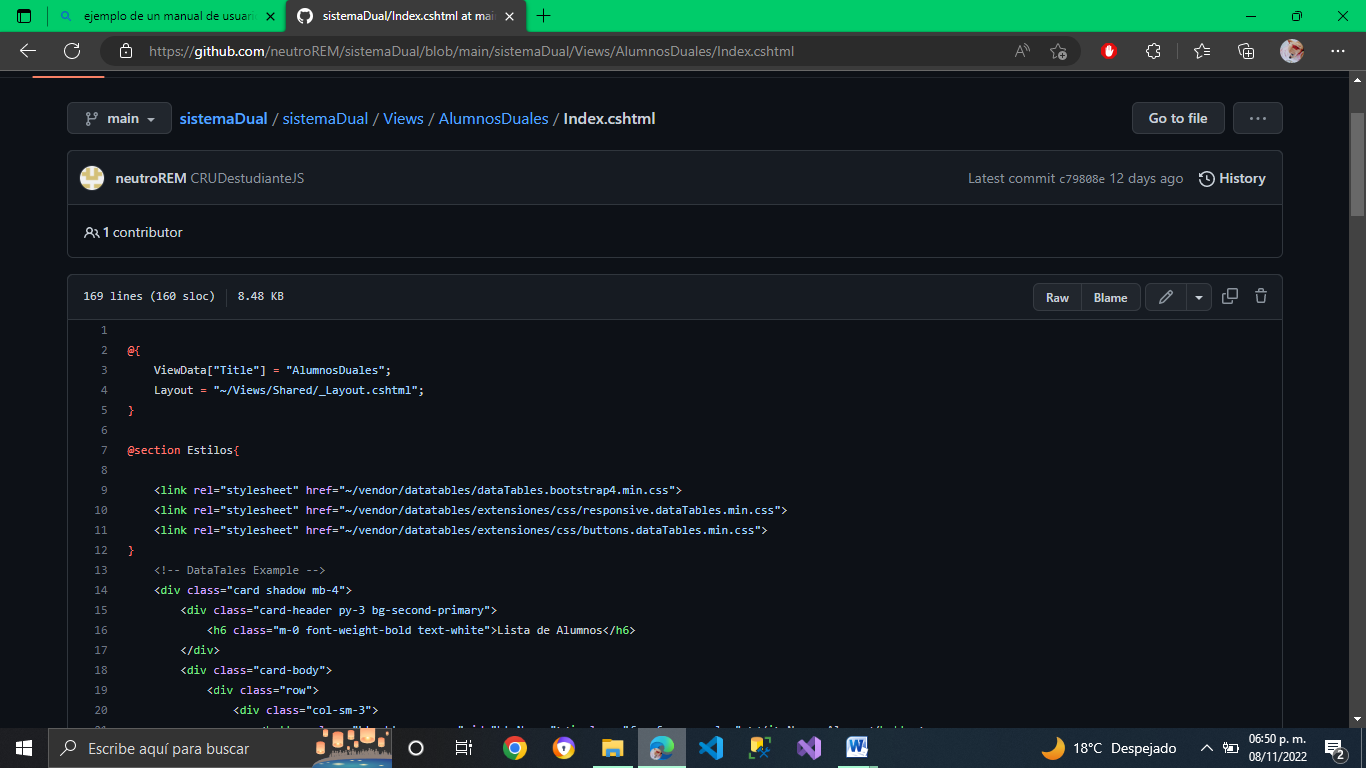
* Tablas.
* Iconos.
* Botones.

Figura Estilos y colores, Fuente: Servidor localhost del proyecto.

Para la codificación de las visas se utiliza HTML y se le agrega interactividad por medio de los modelos y controladores que fueron codificados con el lenguaje c# de igual forma se utiliza JavaScript para perder consumir los webServices con los que se puede obtener los datos y así poder manejar los datos de una mejor manera.

En la siguiente figura 24 se muestra la vista principal del sistema el cual cuenta con otras vistas que se están utilizando para poder hacer pruebas al momento de hacer consultas o insertar datos.

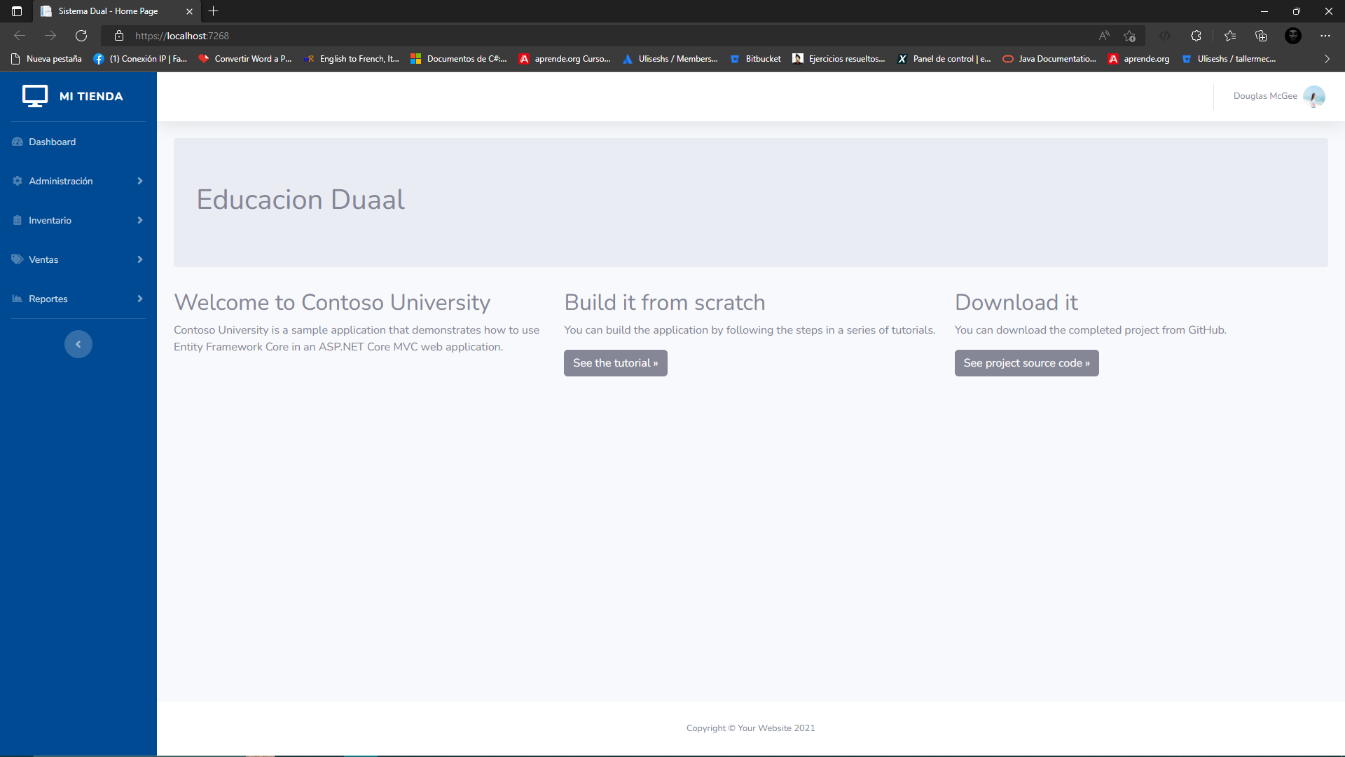
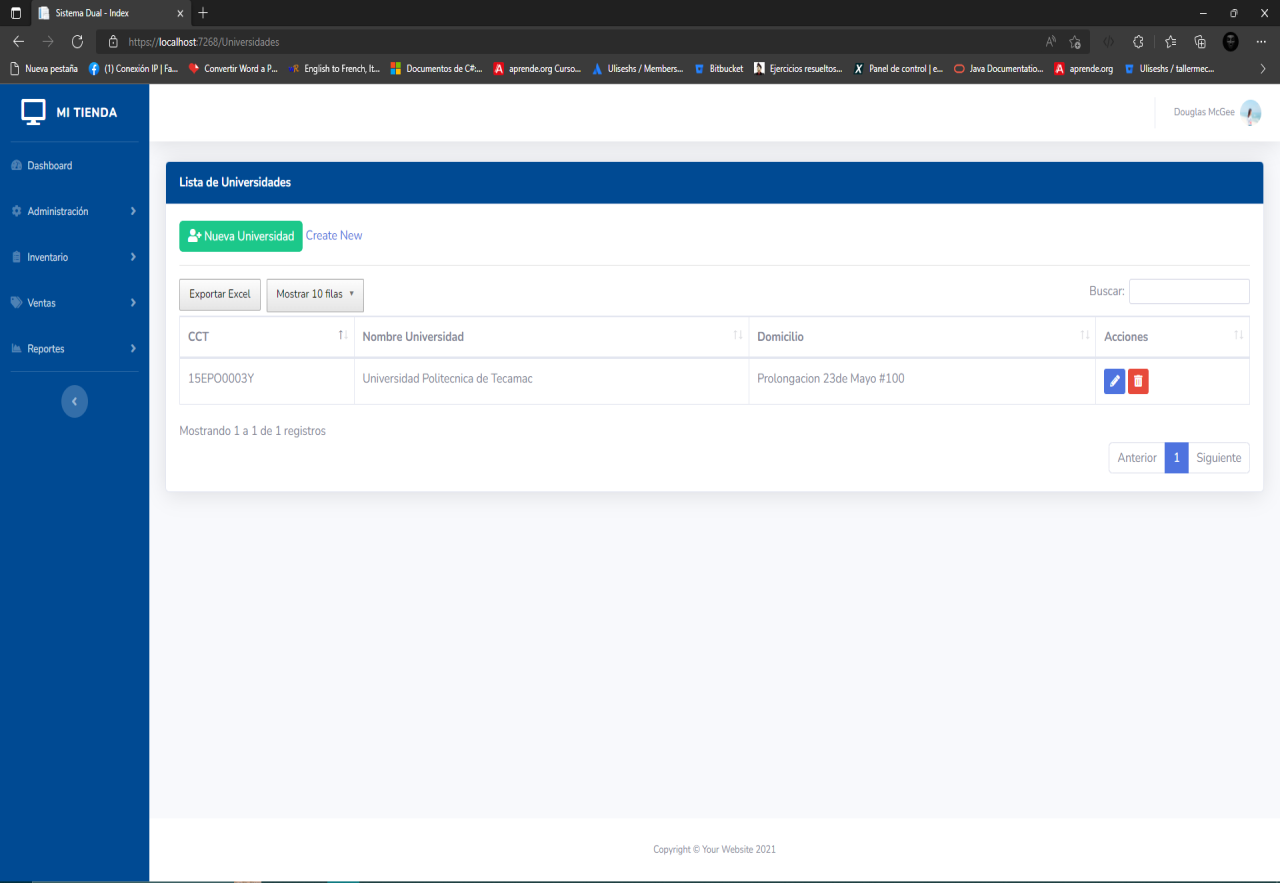


Figura Ventana principal del sistema, Fuente: Servidor localhost del proyecto.

En las siguientes imágenes se muestran las vistas del sistema en ellas es posible hacer nuevos registros, así como poder eliminarlos, las vistas cuentan con una tabla para poder ver los datos que se van ingresando de igual forma cuentan con un apartado para poder insertar más datos.

Figura Vista registro Universidad, Fuente: Servidor localhost del proyecto.



# **RESULTADOS**

Los objetivos que fueron propuestos al inicio del proyecto se llevaron a cabo de manera correcta y sencilla para el usuario, el sistema funciona correctamente de acuerdo con lo solicitado con el cliente, la información que se maneja en el sistema esta ordenada de acuerdo con el área que le corresponde y cada parte de la API está en correcto funcionamiento junto con la base de datos específicamente a lo que se solicitó con anterioridad. Basándose en desarrollo del proyecto se cumplieron con los objetivos a largo plazo y los estándares de seguridad que se plantearon con API explicando todos los funcionamientos del sistema de ED, se da mantenimiento cada cierto tiempo al sistema siempre y cuando el cliente lo solicite y con ello se obtenga la actualización de datos.

|  |  |  |
| --- | --- | --- |
| OBJETIVOS | CUMPLEN | OBSERVACIONES |
| Identificar los elementos que deberan contener de las vistas para integrar un levantamiento de requerimientos completo y exitoso. | SI | Se muestran los elementos solicitados por el cliente en la vista. |
| Definir los requerimientos funcionales del sistema para desarrollar las vistas adecuadas y que brinden información efectiva. | SI | Su cumplió con los estándares acordes a lo solicitado. |
| Diseñar la maquetación de la vista y base de datos, para el flujo del manejo del sistema con el fin de crear una interfaz amigable para el usuario. | NO | Se omite la parte de maquetación debió a motivos de confidencialidad de información por parte de la unidad Educativa. |

# **CONCLUSIONES Y RECOMENDACIONES**

Se generó un sistema de ED para la Universidad Politécnica de Tecámac en donde se desarrolló las vistas las cuales son: lista de Universidades, lista de empresas y lista de estudiantes que coexistieron para el adecuado funcionamiento del sistema y conveniente para el usuario y así evitar errores.

Es importante conocer acerca de la programación para poder realizar este proyecto, pero sobre todo realizar las actividades asignadas por la Universidad, para la conclusión de este trabajo las bases de datos fueron de suma importancia porque la mayor parte de los datos que se importan a una base de datos es a través de una vista generada con anterioridad las cuales se conforman con acciones que se encargan de ejecutar diferentes tareas por ejemplo el botón de guardar datos en la base de datos, el botón en donde se puede descargar un documento en formato XML y este se encuentra almacenado en la nube en donde solamente la unidad educativa y empresarial tiene acceso a él por si hay algún mal entendido con algún cliente y en el formato se cuenta con una sección en la cual el usuario elige los datos solicitados proporcionado con anterioridad. Es significativo conocer el correcto llenado de datos en el sistema para así evitar errores y problemas en él, y así poder realizar mantenimiento, correcciones y edición en el sistema si es que las hay o se solicitan.

Para la Universidad Politécnica de Tecámac es muy importante contar con este sistema, debido a que satisface una de sus necesidades, pero sobre todo la más importante que es el respaldo de información en una base de datos en donde se pueda consultar lo que se solicita, para evitar la pérdida de información.

Y como recomendaciones se propone dar los datos correctos a la unidad educativa los cuales son colocados en la base de datos del sistema para que el usuario efectúa de manera más rápido su petición en el su archivo ya comentado anteriormente, todo esto para evitar la falsa información y que ocurra algún tipo de error en sistema (ED). El administrador será capacitado para realizar pruebas del correcto llenado de datos y sobre todo el correcto funcionamiento de la API y con ello pueda visualizar algunas sugerencias que se le hacen. Cómo tener un Login de acceso solamente con credenciales que ellos indiquen para evitar la fuga de información por otros usuarios que no sean de la empresa o autorizados.

Como recomendación final se encomienda el uso correcto de la API para evitar error al momento de llamar al sistema el navegador web y darle mantenimiento cada que requiera alguna actualización en los datos para así tener un mejor acceso rapidez como para usuarios y administradores.

# **REFERENCIAS BIBLIOGRÁFICAS**

Bascón Pantoja, E. (Diciembre de 2004). El patrón de diseño Modelo-Vista-Controlador (MVC). *ScieELO*, 15. Obtenido de http://www.scielo.org.bo/pdf/ran/v2n4/v2n4\_a05.pdf

Begoña, G. (2000). *DEL SOFTWARE EDUCATIVO A EDUCAR CON SOFTWARE.* Obtenido de http://www.quadernsdigitals.net/articuloquaderns.asp?IdArticle=3743

Corporation, M. (May de 2001). *Collaboration Evaluation Guide Whitepaper.* Microsoft Corporation.

Dykstra, T., & Seders, D. (25 de Mayo de 2021). *Microsoft Corportation*. Obtenido de https://learn.microsoft.com/en-us/ef/core/

Equipo editorial, E. (5 de Agosto de 2021). *Sistema de inforamción*. Obtenido de Concepto.de: https://concepto.de/sistema-de-informacion/

Fuentes, H. J., Aramburu, V., & Cervantes, J. A. (2 de Julio de 2021). *La adopción del modelo alemán de formación dual en México.* Obtenido de Transferencia Tec: https://transferencia.tec.mx/2021/07/02/la-adopcion-del-modelo-aleman-de-formacion-dual-en-mexico/#:~:text=Si%20bien%20la%20modalidad%20de,durante%20la%20actual%20administraci%C3%B3n%20federal.

Hernández Trasobares, A. (2016). Los sistemas de información: evolución y desarrollo. *Departamento de Economía y Dirección de Empresas, Universidad de Zaragoza* , 15.

Mishra, R. (Junio de 16 de 2011). *SQL Server Development History.* Obtenido de SQL Server geeks: https://www.sqlservergeeks.com/sql-server-development-history/

Randolph, R., Rabeler, C., & Leavitt, S. (08 de Agosto de 2022). *Microsoft Corporation*. Obtenido de https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16

Warren, G., Coulter, D., & Dykstra, T. (22 de 09 de 2022). Obtenido de Microsoft Corporation Web Site: https://learn.microsoft.com/es-es/dotnet/framework/get-started/

Williams, S., & Kindel, C. (Octubre de 1994). *The Component Object Model: A Technical Overview*. Obtenido de Microsoft Corporation: http://msdn.microsoft.com/library/

Tablado, F. (2020, septiembre 7). Bases de datos dinámicas. ¿Qué son? Ejemplos. Ayudaleyprotecciondatos.es; AyudaLeyProteccionDatos.

*Lenguajes de programación visual*. (2021, enero 14). Cipsa.net.

*Conceptos básicos de HTML*. (s/f). Mozilla.org. Recuperado el 4 de octubre de 2021.

Gustavo, B., Por, B. V., & Por, D. A. (2019, enero 24). ¿Qué Es CSS? Hostinger.mx.

¿Qué es C#? Introducción. (2017, marzo 29). Openwebinars.net.

Pellicer, P. (2021, octubre 21). ¿Qué es el .NET? ¿Para qué sirve? Blog Emagister;

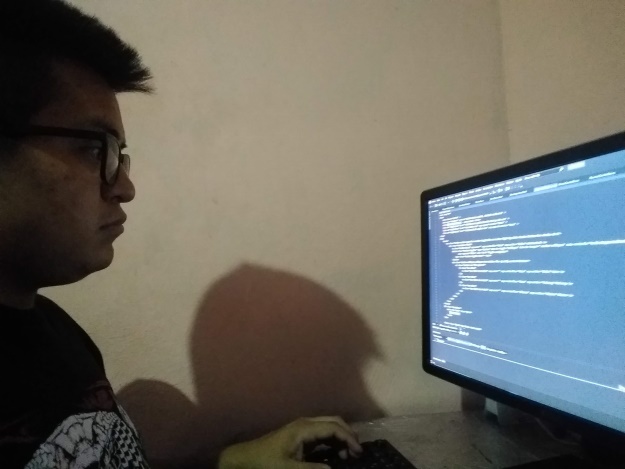
Qué es la nube: definición. (s/f). Microsoft.com. Recuperado el 4 de octubre de 2021.

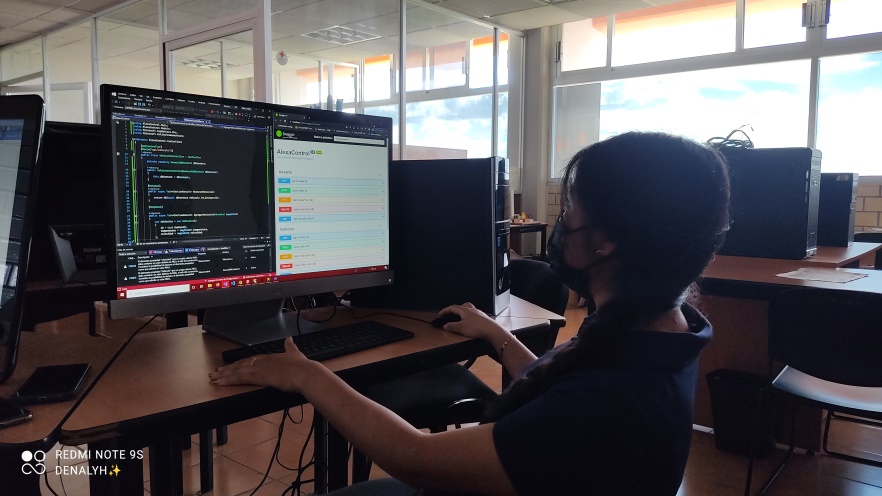
¿Qué es GitHub? Una Guía para Principiantes sobre GitHub. (2018, May 8).

Giraldo, F. E. J. (n.d.). Funcionalidad · Atributos de calidad y patrones de diseño. Gitbooks.io. Retrieved October 20, 2022, from.

Calidad de Software. (n.d.). Ecured.cu. Retrieved October 20, 2022, from.

# **ANEXOS**





1. o interfaz de programación de aplicaciones es un conjunto de definiciones y protocolos que se usa para diseñar e integrar el software de las aplicaciones [↑](#footnote-ref-1)