



**UNIVERSIDAD NACIONAL DE CHIMBORAZO
FACULTAD INGENIERÍA
CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN**

Título:

Desarrollo del Sistema de bienes e inventarios de la Universidad Nacional de Chimborazo aplicando la plataforma .Net Core

**Trabajo de Titulación para optar al título de Ingeniero en
Tecnologías de la Información**

Autor:

Coro Sayay Danny Francisco

Tutor:

MsC. Pamela Alexandra Buñay Guisñan

Riobamba, Ecuador. 2024

DECLARATORIA DE AUTORÍA

Yo, Danny Francisco Coro Sayay, con cédula de ciudadanía 0604430454, autor del trabajo de investigación titulado: Desarrollo del Sistema de bienes e inventarios de la Universidad Nacional de Chimborazo aplicando la plataforma .Net Core, certifico que la producción, ideas, opiniones, criterios, contenidos y conclusiones expuestas son de mí exclusiva responsabilidad.

Asimismo, cedo a la Universidad Nacional de Chimborazo, en forma no exclusiva, los derechos para su uso, comunicación pública, distribución, divulgación y/o reproducción total o parcial, por medio físico o digital; en esta cesión se entiende que el cesionario no podrá obtener beneficios económicos. La posible reclamación de terceros respecto de los derechos de autor de la obra referida, será de mi entera responsabilidad; librando a la Universidad Nacional de Chimborazo de posibles obligaciones.

En Riobamba, 19/04/2023.



Danny Francisco Coro Sayay

C.I:0604430454

DICTAMEN FAVORABLE DEL PROFESOR TUTOR



Dirección
Académica
VICERRECTORADO ACADÉMICO

en movimiento



UNACH-RGF-01-04-08.11
VERSIÓN 01: 06-09-2021

ACTA FAVORABLE - INFORME FINAL DEL TRABAJO DE INVESTIGACIÓN

En la Ciudad de Riobamba, a los once días del mes de marzo de 2024, luego de haber revisado el Informe Final del Trabajo de Investigación presentado por el estudiante **Danny Francisco Coro Sayay** con CC: **0604430454**, de la carrera de **Ingeniería en Tecnologías de la Información** y dando cumplimiento a los criterios metodológicos exigidos, se emite el **ACTA FAVORABLE DEL INFORME FINAL DEL TRABAJO DE INVESTIGACIÓN** titulado "**Desarrollo del Sistema de bienes e inventarios de la Universidad Nacional de Chimborazo aplicando la plataforma .Net Core**", por lo tanto se autoriza la presentación del mismo para los trámites pertinentes.



PAMELA ALEXANDRA
BUÑAY GUISÑAN

MsC. Pamela Alexandra Buñay Guisñan
TUTOR(A)

CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

Quienes suscribimos, catedráticos designados Miembros del Tribunal de Grado para la evaluación del trabajo de investigación “Desarrollo del Sistema de bienes e inventarios de la Universidad Nacional de Chimborazo aplicando la plataforma .Net Core” por Danny Francisco Coro Sayay, con cédula de identidad número 0604430454, bajo la tutoría de MsC. Pamela Alexandra Buñay Guisñan; certificamos que recomendamos la APROBACIÓN de este con fines de titulación. Previamente se ha evaluado el trabajo de investigación y escuchada la sustentación por parte de su autor; no teniendo más nada que observar.

De conformidad a la normativa aplicable firmamos, en Riobamba a los 17 días del mes de abril 2024.

Ing. Danny Velasco.
PRESIDENTE DEL TRIBUNAL DE GRADO

Ing. Diego Reina..
MIEMBRO DEL TRIBUNAL DE GRADO

Dr. Hugo Paz.
MIEMBRO DEL TRIBUNAL DE GRADO

CERTIFICADO ANTI PLAGIO



Dirección
Académica
VICERRECTORADO ACADÉMICO



CERTIFICACIÓN

Que, **CORO SAYAY DANNY FRANCISCO** con CC: **0604430454**, estudiante de la Carrera **INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN**, Facultad de **INGENIERÍA**; ha trabajado bajo mi tutoría el trabajo de investigación titulado **"DESARROLLO DEL SISTEMA DE BIENES E INVENTARIOS DE LA UNIVERSIDAD NACIONAL DE CHIMBORAZO APLICANDO LA PLATAFORMA .NET CORE"**, cumple con el 10 %, de acuerdo al reporte del sistema Anti plagio **.TURNITIN** porcentaje aceptado de acuerdo a la reglamentación institucional, por consiguiente autorizo continuar con el proceso.

Riobamba, 03 de abril de 2024



Mgs. Pamela Buñay
TUTOR(A)

DEDICATORIA

Hoy, al llegar a este importante logro en mi vida, no puedo dejar de mirar hacia atrás y reflexionar sobre el inmenso papel que cada uno de ustedes ha desempeñado en mi camino. Esta tesis, que representa el esfuerzo de muchos años de estudio y dedicación, no sería posible sin su amor, apoyo y aliento constante.

A mis padres desde mis primeros pasos en la educación hasta este emocionante momento, han sido mis guías y modelos a seguir. Cada día, me han inculcado valores de perseverancia, resiliencia y trabajo arduo. Vuestras palabras de aliento y vuestro respaldo incondicional me han dado la confianza para enfrentar cada desafío y superar cada obstáculo.

AGRADECIMIENTO

En este importante momento de mi vida, quiero expresarles mi más profundo agradecimiento y reconocimiento a mis padres ya que gracias a su ayuda y paciencia efímera he podido llegar al final de esta travesía hacia y finalmente la culminación de mi vida universitaria, sin ellos no hubiera sido posible.

A ambos, les agradezco por estar presentes en cada paso de mi camino educativo, por celebrar mis logros y por sostenerme cuando enfrentaba dificultades. Vuestra confianza en mí ha sido mi mayor motivación.

Esta tesis no solo es un reflejo de mis conocimientos y esfuerzos, sino también del amor y el apoyo incondicional que me han brindado. Agradezco a la vida por tener unos padres tan excepcionales como ustedes.

Con todo mi cariño y gratitud,

Danny Francisco Coro Sayay.

ÍNDICE GENERAL

DECLARATORIA DE AUTORÍA

DICTAMEN FAVORABLE DEL PROFESOR TUTOR

CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

CERTIFICADO ANTI PLAGIO

DEDICATORIA

AGRADECIMIENTO

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

RESUMEN

ABSTRACT

CAPÍTULO I. INTRODUCCIÓN.....	14
1.1 Planteamiento del Problema	14
1.2 Justificación	15
1.3 Formulación del Problema.....	15
1.4 Objetivos.....	15
1.4.1 Objetivo General	15
1.4.2 Objetivos Específicos.....	15
CAPÍTULO II. MARCO TEÓRICO.....	16
2.1 .Net.....	16
2.2 .Net Framework.	16
2.3 .Net Core.	16
2.4 .Net 7.....	18
2.5 ASP.NET MVC	18
2.6 Diferencias de .Net Framework vs .Net Core.....	19
2.7 Visual Studio.....	19
2.7.1 C#.....	19
2.8 SQL Server	19
2.9 Entity Framework	20
2.10 Metodología Scrum.....	20
2.10.1 ¿Por qué implementar Scrum?.....	20

2.11	Modelo Furps.....	21
2.12	Jmeter.....	21
CAPÍTULO III. METODOLOGÍA.....		23
3.1	Enfoque de la Investigación.....	23
3.2	Tipo de Investigación.....	23
3.3	Según la fuente de investigación.....	23
3.4	Técnicas de Recolección de Datos.....	23
3.5	Población y Muestra.	23
3.6	Identificación de variables.	23
3.6.1	Variable dependiente.....	23
3.6.2	Variable independiente.....	23
3.7	Operacionalización de variables.	23
3.8	Fases de desarrollo aplicando Scrum.....	26
3.8.1	Fase de análisis.....	26
3.8.2	Planificación.....	28
3.8.3	Desarrollo.....	34
3.8.4	Implementación.....	36
3.8.5	Lanzamiento.....	40
CAPÍTULO IV. RESULTADOS Y DISCUSIÓN.....		41
4.1	Análisis de eficacia.	41
4.2	Análisis de Tiempo de respuesta.....	41
4.3	Análisis en la utilización de recursos.....	42
4.4	Contraste entre los valores establecidos en el modelo Furps y los valores obtenidos. 43	
4.5	Discusión	43
CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES.		44
5.1	Conclusiones.....	44
5.2	Recomendaciones.	44
BIBLIOGRAFÍA.....		45
Anexo A: Pruebas de rendimiento.....		46
Anexo B: Manual de Usuario.		50
Anexo C: Acta de Entrega recepción.		56

ÍNDICE DE TABLAS.

Tabla 1: Versiones de .Net core.	18
Tabla 2: Beneficios e inconvenientes de .Net Framework y .Net Core.	19
Tabla 3: Valores establecidos según Furps.	21
Tabla 4: Ventajas y Desventajas de Jmeter.	22
Tabla 5: Operacionalización de variables.	24
Tabla 6: Tabla de roles.	26
Tabla 7: Requerimientos Funcionales del proyecto.	26
Tabla 8: Requerimientos no funcionales.	27
Tabla 9: Lista de tareas según su escala de esfuerzo.	28
Tabla 10: Historias de usuario.	29
Tabla 11: Lista de tareas del Sprint 1.	32
Tabla 12: Lista de tareas del Sprint 2.	32
Tabla 13: Lista de tareas del Sprint 3.	33
Tabla 14: Lista de tareas del Sprint 4.	33
Tabla 15: Herramientas para el Desarrollo del proyecto.	35
Tabla 16. Análisis de Eficacia.	41
Tabla 17: Utilización de recursos en porcentaje.	42
Tabla 18: Los resultados alcanzados comparados con los establecidos por el modelo Furps.	43

ÍNDICE DE FIGURAS

Figura 1: Servicios compatibles con .net.....	16
Figura 2: Diagrama Entity Framework.	20
Figura 3: Modelo de calidad Furps.	21
Figura 4: Caso de Uso del Usuario.	30
Figura 5: Caso de Uso ingresar información de un bien.....	30
Figura 6: Caso de Uso editar información de bien.....	31
Figura 7: Caso de Uso Eliminar bien.	31
Figura 8: Caso de Uso Constatación bien.	31
Figura 9: Arquitectura del sistema.	34
Figura 10: Diseño de la base de datos.	35
Figura 11: Creación de aplicación.....	36
Figura 12: Paquetes instalados.	36
Figura 13: Tablas de la base de datos.....	37
Figura 14: Controladores necesario.....	37
Figura 15: Vistas generadas.....	38
Figura 16: Configuraciones necesarias.	38
Figura 17: Modulo para visualizar bienes.	39
Figura 18: Modulo de Constatación por lector de código de barras.	39
Figura 19: Prototipo de reportes bienes Constatados.	40
Figura 20: Prototipo de reportes bienes en Mantenimiento.	40
Figura 21: Grafico Eficacia del Sistema Web.....	41
Figura 22: Tiempo de respuesta.	42
Figura 23: Análisis del tiempo de respuesta.....	42
Figura 24: Análisis del tiempo de respuesta.....	43
Figura 25: Pruebas de Rendimiento configuración.	46
Figura 26: Peticiones Http correctas	46
Figura 27: Grafico según su comportamiento del sistema.	47
Figura 28: Evaluación del rendimiento con 0% de error.	47
Figura 29: Procesos del sistema con 0% de error.....	48
Figura 30: Porcentaje de uso del CPU.	48
Figura 31: Porcentaje de uso de memoria.	49
Figura 32: Porcentaje de uso de disco.....	49

RESUMEN

En este trabajo de investigación presenta el desarrollo del sistema de bienes e inventarios para la Universidad Nacional de Chimborazo, utilizando la plataforma .Net Core. El desarrollo de este sistema representa un avance significativo en el control de los activos de la institución, posibilitando la gestión eficiente de bienes tanto en su estado regular como en procesos de mantenimiento, a la vez que facilita una constatación precisa de los mismos.

Fue desarrollado bajo un enfoque cuantitativo, se utilizó la plataforma .Net Core, aprovechando sus ventajas como su capacidad multiplataforma, escalabilidad y rendimiento. Se optó por el lenguaje de programación de alto nivel C# junto con Entity Framework, y se incorporaron bibliotecas de clases como práctica recomendada en programación. Se implementó la metodología Scrum, la cual consta de cinco etapas, para simplificar las tareas y abordar eficientemente los desafíos planteados. Las pruebas de rendimiento del sistema web, se aplicó el software Jmeter para realizar simulaciones con peticiones en tiempo real, facilitando la supervisión de la correcta ejecución de todas las solicitudes.

Se procedió a analizar el rendimiento de la aplicación web utilizando el modelo Furps. Los resultados de las pruebas confirmaron que la aplicación cumple con los criterios establecidos por Furps, logrando un 100% de eficacia en la ejecución de peticiones, un tiempo de respuesta de 0,114 segundos y un uso de recursos del 21%.

Palabras claves: Plataforma .Net Core, Scrum, bienes, Entity, applications.

ABSTRACT

ABSTRACT

The main objective of this research study focus on the development of an asset and inventory system for the National University of Chimborazo, using the .Net Core platform. The development of this system represents a significant advance in the control of the assets of the institution, enabling the efficient management of assets both in their regular state and in maintenance processes, while facilitating an accurate verification of them. It was developed under a quantitative approach, using the .Net Core platform, taking advantage of its multiplatform capacity, scalability, and performance. The high-level programming language C# was chosen together with the Entity Framework, and class libraries were incorporated as a recommended programming practice. The Scrum methodology, which consists of five stages, was implemented to simplify tasks and efficiently address the challenges posed. The performance tests of the web system, the Jmeter software were applied to perform simulations with real-time requests, facilitating the supervision of the correct execution of all requests. The performance of the web application was analyzed using the Furps model. The results of the tests confirmed that the application meets the criteria established by Furps, achieving 100% efficiency in the execution of requests, a response time of 0.114 seconds, and a resource usage of 21%.

Keywords: .Net Core Platform, Scrum, assets, Entity, applications.



Reviewed by:
Marco Antonio Aquino
ENGLISH PROFESSOR
C.C. 1753456134

1. CAPÍTULO I. INTRODUCCIÓN.

Las Tecnologías de la Información han obtenido gran repercusión a medida que las nuevas herramientas surgen con versiones mejoradas y funciones más avanzadas. En este sentido, la Universidad Nacional de Chimborazo (Unach), ha enfrentado la importancia de optimizar su gestión de bienes e inventarios mediante mejoras en el área tecnológica como lo es la tecnología Net Core, por ser moderno y eficiente.

Los sistemas de gestión de inventario permiten controlar los bienes, registrar si hay nuevos activos y realizar un seguimiento, garantizando así una buena administración de los activos institucionales. Por lo tanto, resulta indispensable que la Universidad Nacional de Chimborazo cuente con un sistema de bienes e inventarios que facilite la constatación y control. [1]

Observando esta necesidad, la (Unach), propuso desarrollar el sistema de bienes e inventarios, usando la plataforma .Net Core en conjunto con el Framework Entity que utiliza tecnologías modernas para mejorar el rendimiento del sistema Web, fue desarrollado con los parámetros de la metodología ágil Scrum tiene 5 fases en las cuales se procedió a ejecutar tareas fundamentales bajo un tiempo de ejecución óptimo.

La calidad del software es un factor fundamental en la Ingeniería del Software [2]. De esta manera, se procedió a evaluar el sistema Web utilizando la herramienta Jmeter, conforme al modelo de calidad Furps.

El sistema ha contribuido a mejorar los procesos de inventario y ha proporcionado a la universidad una herramienta confiable para la gestión, cabe mencionar que los principales beneficiarios de este proyecto de investigación son los miembros del personal de la (Unach). [3]

1.1 Planteamiento del Problema

El problema residía en la falta de eficiencia en el registro y validación de bienes del sistema de bienes e inventarios existentes. Esto resultaba difícil para ingresar nuevos datos sobre bienes adquiridos por la (Unach) y la imposibilidad de generar informes. A raíz de esto, el (Codesi) de la Universidad Nacional de Chimborazo, ante el rendimiento insatisfactorio del sistema de bienes, que había causado inconvenientes en diversas actividades de los usuarios, reconoció la necesidad de incorporar un nuevo sistema de bienes e inventarios a sus servicios. Con el desarrollo del nuevo sistema, se busca actualizar y generar informes técnicos de los bienes de manera efectiva, asegurando un rendimiento sobresaliente en aspectos como la eficacia, el tiempo de respuesta y el uso de recursos. Además, se tiene la intención de emplear tecnologías más actuales, como .Net Core.14

En base a lo expuesto anteriormente, la presente investigación se enfocó en el desarrollo del sistema de gestión de bienes e inventarios para la Universidad Nacional de Chimborazo (Unach).

1.2 Justificación

Gracias al sistema de bienes e inventarios desarrollado en la plataforma .Net Core, con dos módulos, el primero que realiza el registro de bienes y el segundo que mediante un lector de código de barras permite la constatación del bien, hemos conseguido mejorar las tareas y actividades reduciendo el tiempo de ejecución de procesos. La optimización de los procesos ha permitido agilizar su ejecución, mejorando el rendimiento, reduciendo el consumo de recursos y aumentando el tiempo de respuesta.

El aumento en el rendimiento del sistema de bienes e inventarios garantiza un acceso más eficiente a la información proporcionada, agilizando los procesos y ejecutando tareas con mayor calidad. Desarrollar un sistema más ágil que pretende agilizar las actividades de los usuarios, mejorando así el servicio ofrecido en la Universidad Nacional de Chimborazo.

1.3 Formulación del Problema

¿El uso de la plataforma .Net Core incidirá en el rendimiento del sistema de bienes e inventario de la Universidad Nacional de Chimborazo?

1.4 Objetivos

1.4.1 Objetivo General

- Desarrollar el Sistema de bienes e inventarios aplicando la plataforma .Net Core para optimizar los procesos de registro de bienes de la Universidad Nacional de Chimborazo.

1.4.2 Objetivos Específicos

- Analizar la tecnología .Net Core enfocado al desarrollo de aplicaciones web.
- Desarrollar el sistema de bienes e inventarios aplicando la plataforma .Net Core. para la Universidad Nacional de Chimborazo.
- Evaluar el rendimiento del sistema utilizando el modelo de calidad Furps.

2. CAPÍTULO II. MARCO TEÓRICO.

2.1 .Net.

Permite realizar aplicaciones altamente portátiles además de virtualizables e independiente de la nube, como resultado los costos asociados con la implementación de aplicaciones lo hace más fácil para los clientes que no poseen la capacidad para comprar y administrar la arquitectura en la nube y así usar un marco definido, como .Net Core puede ser fácil, pero sobre todo proporciona tecnologías necesarias para la implementación de elementos horizontales en arquitectura [2].

A continuación, en la figura 1 se observa los todos los servicios disponibles en .Net.

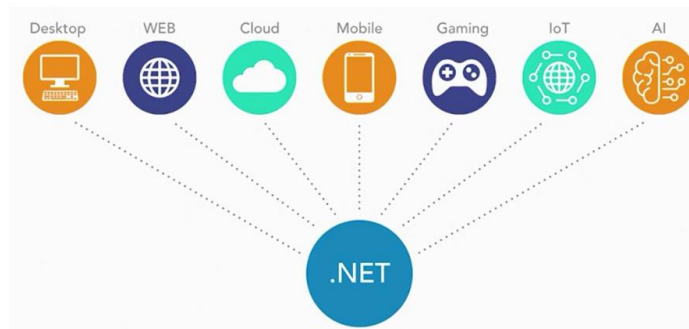


Figura 1: Servicios compatibles con .net.

Fuente: [5].

2.2 .Net Framework.

Es el entorno de desarrollo que permite a los programadores aprovechar las tecnologías para brinda el uso bibliotecas de código reutilizable, de este modo los usuarios pueden llamar todas estas bibliotecas para los proyectos que deseen usar. Algunas de las ventajas que .Net framework ofrece son las siguientes [3]:

- Sobre reutilización de código.
- Especialización de código.
- Gestión de recursos, desarrollo multilinguaje.
- Seguridad, implementación y administración.
- Biblioteca de clases extensa.

2.3 .Net Core.

Se trata de un entorno laboral con una estructura modular, adaptable a diferentes plataformas y con el código accesible al público. Su diseño tiene como objetivo facilitar la portabilidad entre distintos sistemas, fomentando la máxima reutilización del código. Además, proporciona un entorno compartido. En comparación, .NET Core posee algunas apis de .Net Framework, pero la diferencia es que en .Net core es más compacto e incorpora componentes esenciales como tiempo de ejecución, diferentes frameworks y diversas

herramientas que son compatibles con una amplia gama de sistemas operativos y procesadores [4].

Algunas cualidades que habilitan a los programadores para redactar código de manera confiable y eficiente incluyen:

- Expresiones lambda.
- Excepciones.
- Linq.

Utilidad de .Net Core

La utilización de .Net Core abre un amplio abanico de posibilidades para construir aplicaciones Web. Por ejemplo, facilita la creación de aplicaciones web que pueden ser desplegadas en diversas plataformas como Windows, Linux y Mac OS. [8]. Además, ofrece la capacidad de desplegar estas aplicaciones utilizando diferentes plataformas de nube.

.NET Core también es adecuado para desarrollar aplicaciones de escritorio UWP, permite ejecutarlas en múltiples dispositivos, aprovechando el código existente sin la necesidad de reescribir las bibliotecas. [8]. Adicionalmente, .NET Core es una opción viable para proyectos en Internet de las Cosas, Ia, videojuegos y otros campos más.

Beneficios de .Net Core

Net Core tiene varios beneficios de los cuales se procede a detallar los más importantes para el desarrollo de proyectos ya sean sistemas web, apis, microservicios, etc.

- **Rendimiento**

ASP.Net Core se centra en la optimización del rendimiento, siendo una de sus principales prioridades, su diseño y arquitectura están orientados a la eficiencia, permitiendo una ejecución veloz [7].

- **Evolución constante**

ASP.NET Core está concebido para posibilitar una rápida evolución de los componentes en tiempo de ejecución. Esta flexibilidad facilita la adaptación a las nuevas tecnologías y demandas del mercado.

- **Estabilidad y compatibilidad**

A pesar de su capacidad de evolución, ASP.Net Core ofrece compatibilidad para que las aplicaciones se ejecuten en los diferentes entornos de desarrollo. Las actualizaciones se implementan de forma gradual, asegurando la continuidad del funcionamiento de las aplicaciones [7].

- **Beneficios para los desarrolladores**

La combinación de evolución y estabilidad en ASP.Net Core facilita el trabajo de los desarrolladores. Permite crear aplicaciones web modernas y adaptables, sin sacrificar la estabilidad y el rendimiento [7].

- **Soporte multiplataforma**

.NET cuenta con el apoyo de diversas organizaciones que garantizan su funcionamiento en diferentes sistemas operativos. Es compatible con arquitecturas ARM64, x64 y x86, ampliando su alcance a una amplia gama de dispositivos [8].

- **Actualizaciones constantes**

Gracias al trabajo de estas organizaciones, .NET se mantiene actualizado de forma constante. Esto permite a la comunidad de .Net disfruten de las más recientes innovaciones y mejoras en la plataforma.

Todas las versiones de .Net Core.

A continuación, en la tabla 1 se puede observar las todas las versiones de .Net Core.

Tabla 1: Versiones de .Net core.

Versión	Fecha según su versión.
.Net Core 1.0	06/2016
.Net Core 1.1	11/2016
.Net Core 2.0	08/2017
.Net Core 2.1	05/2018
.Net Core 2.2	12/2018
.Net Core 3.0	09/2019
.Net Core 5.0	11/2020
.Net Core 6.0	11/2021
.Net Core 7.0	11/2022

2.4 .Net 7

Según [3].NET 7 se basa en los cimientos de su último predecesor, .NET 6, e incluye mejoras para admitir escenarios de computación en la nube, más herramientas, actualizar proyectos heredados y simplificar la experiencia del desarrollador con contenedores además posee las características:

- La mejora de la experiencia para los desarrolladores.
- Mayor soporte para las herramientas de kit de este software.
- Aumento del rendimiento de las aplicaciones.
- Más opciones para compartir código.
- Una mayor interoperabilidad.

2.5 ASP.NET MVC

Hace referencia a un modelo vista controlador, donde .NET MVC ofrece una opción al patrón de .NET Web Forms para desarrollar aplicaciones web basadas en el patrón Modelo-Vista-Controlador (MVC). MVC es un entorno de visualización de sencilla implementación y comprobación que se adapta a las características ya existentes de ASP.NET, como la implementación de páginas maestras y la autenticación basada en pertenencia [5].

2.6 Diferencias de .Net Framework vs .Net Core

A continuación, en la tabla 2 se puede observar los beneficios e inconvenientes de .Net Core frente a .Net Framework.

Tabla 2: Beneficios e inconvenientes de .Net Framework y .Net Core.

	Beneficios	Inconvenientes
.Net Framework	<ul style="list-style-type: none">• Soporte completo para aplicaciones Windows Forms y WPF.• Gran variedad de bibliotecas para aplicaciones.• Facilidad para trabajar con tecnologías de Microsoft, como SQL Server y ASP.NET.	<ul style="list-style-type: none">• Solo se puede ejecutar en sistemas operativos Windows.• La dependencia hace que las aplicaciones sean menos portables.• Las actualizaciones y mejoras de la plataforma no son tan frecuentes como en .NET Core.
.Net Core	<ul style="list-style-type: none">• Soporte multiplataforma, lo que permite ejecutar aplicaciones en ios, linux y windows.• Mejor rendimiento que .NET Framework.• Las actualizaciones y mejoras de la plataforma son más frecuentes.	<ul style="list-style-type: none">• No tiene soporte completo para algunas tecnologías de Microsoft, como Windows Forms y WPF.• Es posible que algunas bibliotecas y herramientas que funcionan en .NET Framework no sean compatibles con .Net Core.• Algunas características avanzadas de la plataforma solo están disponibles en sistemas operativos Windows.

2.7 Visual Studio

2.7.1 C#

Es un lenguaje de codificación que ofrece a los desarrolladores la capacidad de crear una amplia gama de aplicaciones seguras y confiables que se ejecutan en el entorno .NET. Con raíces en la familia de lenguajes C, C# resultará inmediatamente conocido para desarrolladores con experiencia en C++ y Java. [6].

2.8 SQL Server

Según [7] SQL Server, un producto de Microsoft, es un sistema de gestión de bases de datos relacionales. Utiliza Transact-SQL como el lenguaje principal para realizar consultas, además posee ciertas ventajas como:

- Capacidad de adaptarse al crecimiento, fiabilidad y protección.
- Admite el uso de procedimientos almacenados.

- Incluye una interfaz de administración gráfica robusta.
- Facilita la operación en un entorno cliente-servidor, donde los datos residen en el servidor y los clientes acceden a la información a través de la red.
- Facilita la administración de información de otros servidores de bases de datos.

2.9 Entity Framework

Entity Framework es una herramienta principal para el mapeo objeto-relacional (ORM). Su función principal es facilitar a los desarrolladores el trabajo con datos relacionales al tratarlos como objetos dentro de sus dominios específicos. Esto disminuye la cantidad de código necesario para acceder a los datos, lo que generalmente requeriría una escritura extensa. [8]. A continuación, en la figura 2 se puede observar la estructura de trabajo de Entity Framework.

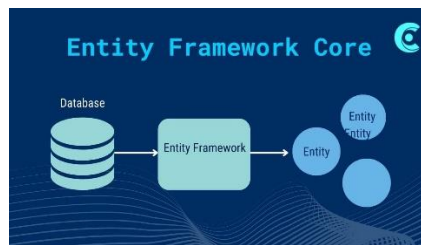


Figura 2: Diagrama Entity Framework.

Fuente: [9].

2.10 Metodología Scrum.

Es una de las metodologías ágiles más reconocidas a nivel global para el desarrollo de software. Sus raíces se remontan a los años 1980, cuando se destacó la importancia del trabajo en conjunto del equipo de desarrollo. En la década de los 1990, Jeff Sutherland y Ken Schwaber revitalizaron este enfoque, definiendo un conjunto de directrices y reglas particulares aplicables al desarrollo de software [13].

2.10.1 ¿Por qué implementar Scrum?

Las metodologías Scrum poseen ciertas características fundamentales como son:

- **Simplicidad:** En esencia esta metodología agiliza a los integrantes del equipo la utilización de la misma [10].
- **Inspección:** Esto facilita la realización de reuniones diarias, revisiones del sprint. Dicha revisión posibilita que la organización fortalezca la metodología e identifique áreas de mejora en cada parte del proceso [10].
- **Adaptación:** En esta característica se puede obtener cierta retroalimentación e implementar los cambios requeridos en el proyecto, en el momento que sean necesarios [10].

2.11 Modelo Furps.

El Modelo de calidad de software, concebido por Hewlett-Packard en 1987, engloba una gama de causas que definen la calidad del software. Los criterios para la evaluación incluyen eficacia, tiempo de respuesta y utilización de recursos. Este modelo implica una serie de pruebas en las que los usuarios realizan pruebas al producto antes de su lanzamiento al mercado y ofrecen retroalimentación. En la figura 3, se puede observar los criterios del modelo de calidad Furps.



Figura 3: Representación gráfica de Furps.

Fuente: [9]

Estos son los valores establecidos para que Furps establezca para un buen rendimiento de un sistema Web. En la tabla 3, se puede observar los criterios establecidos por el modelo de calidad Furps, según su porcentaje aceptado al hacer una evaluación en el rendimiento de Software.

Tabla 3: Valores establecidos según Furps.

Criterios Para Evaluar.	Valores según Furps.
Eficacia	95 %
Tiempo de respuesta	5sg
Utilización de recursos	25 %

2.12 Jmeter

Esta herramienta sumamente beneficiosa se ha desarrollado con la capacidad de simular cargas en diversos tipos de software. Aunque inicialmente se diseñó para pruebas de estrés en aplicaciones web, su arquitectura al paso de los años ha evolucionado para permitir pruebas en componentes habilitados en Internet (HTTP), además de solicitudes FTP [11]. En la tabla 4, se puede observar las ventajas y desventajas de Jmeter.

Ventajas	Desventajas
-----------------	--------------------

Jmeter	<ul style="list-style-type: none"> • Su código está disponible para el público en general. • De uso fácil y sencillo. • Soporte de múltiples protocolos. • Gran variedad de funciones al momento de realizar pruebas. <ul style="list-style-type: none"> • No tiene costo. 	<ul style="list-style-type: none"> • Requerir tiempo y esfuerzo para aprender y dominar la herramienta. • No cuenta con soporte oficial. • Requiere un conocimiento técnico sólido.
---------------	--	--

Tabla 4: Ventajas y Desventajas de Jmeter.

3. CAPÍTULO III. METODOLOGÍA.

El propósito de esta investigación fue crear una aplicación web para la gestionar y constatar los bienes de la Universidad Nacional de Chimborazo. Para lograrlo, se aplicó la metodología de desarrollo Scrum, utilizando la plataforma .Net Core.

3.1 Enfoque de la Investigación.

El proyecto de investigación se sustentó en un enfoque cuantitativo, permitió analizar la tecnología .Net Core para desarrollar el sistema de bienes e inventarios, se evaluó el rendimiento según el modelo de calidad Furps.

3.2 Tipo de Investigación.

3.3 Según la fuente de investigación.

- **Investigación bibliográfica:** Se exploró la información necesaria mediante la consulta de diversas fuentes, como libros, bases de datos científicas, revistas y tesis, entre otras
- **Investigación Aplicada:** El tipo de investigación se puede clasificar como aplicada, ya que se centró en desarrollar un sistema web utilizando la plataforma .Net Core para los procesos relacionados con el control de inventario en la Universidad Nacional de Chimborazo.
- **Según el tipo de variable:** Esta investigación adoptó un enfoque cuantitativo, ya que se centró en la evaluación del rendimiento del sistema web conforme al modelo Furps

3.4 Técnicas de Recolección de Datos.

Se empleó la herramienta jmeter con el fin de recolectar datos relacionados con la evaluación de los estándares de rendimiento definidos por el modelo Furps. Estos datos recopilados con Jmeter proporcionaron una visión detallada y cuantitativa del rendimiento del sistema.

3.5 Población y Muestra.

En esta investigación la población son todos los bienes de la UNACH que están distribuidos en las facultades, departamentos y áreas en los que se constata el bien la categoría, marcas, etc.

Se utilizaron 150 usuarios en 1 minuto para evaluar el rendimiento con la herramienta Jmeter para obtener datos sobre los indicadores del modelo Furps.

3.6 Identificación de variables.

3.6.1 Variable dependiente.

- El rendimiento del sistema de bienes e inventarios.

3.6.2 Variable independiente.

- Tecnología .Net Core.

3.7 Operacionalización de variables.

A continuación, en la tabla 5 se puede observar la Operacionalización de variables.

Tabla 5: Operacionalización de variables.

PROBLEMA	TEMA	OBJETIVOS	VARIABLES	CONCEPTUALIZACIÓN	DIMENSIÓN	INDICADORES
¿El uso de la plataforma .Net Core incidirá en el rendimiento del sistema de bienes e inventario de la Universidad Nacional de Chimborazo?	Desarrollo del Sistema de bienes e inventarios de la Universidad Nacional de Chimborazo aplicando plataforma .Net Core.	General	Independiente	Plataforma de desarrollo de código abierto diseñada para que sea portátil entre plataformas. [3]	Desarrollo del Sistema de bienes e inventarios de la Universidad Nacional de Chimborazo	- Número de funcionalidades. - Número de módulos desarrollados.
		Específicos	Dependiente	Conjunto de procesos y procedimientos diseñados para gestionar el inventario y los bienes de una organización.	Rendimiento	- % Eficacia. - %Tiempo de respuesta. - %Utilización de recursos.
		<ul style="list-style-type: none"> Analizar la plataforma .Net Core enfocado al desarrollo de aplicaciones web. Aplicar la plataforma .Net Core en el desarrollo del sistema de 	El rendimiento del sistema de bienes e inventarios.			

bienes e
inventarios de
la
Universidad
Nacional de
Chimborazo.

- Evaluar el
rendimiento
del sistema
utilizando el
modelo de
calidad Furps.
-

3.8 Fases de desarrollo aplicando Scrum.

Cada fase se realizó según la planificación y según los requerimientos de usuario. A continuación, se presentan las etapas del marco de trabajo Scrum.

3.8.1 Fase de análisis.

En la fase inicial del proyecto, se analizaron los requisitos del sistema. La entidad receptora, que en este caso es la Universidad Nacional de Chimborazo, presentó una lista de requisitos para el sistema. Estos requisitos se basaron en las necesidades de la universidad, así como en los objetivos del proyecto.

- **Personal involucrado.**

Scrum define los roles y las responsabilidades que deben asumir las personas que participan en un proyecto. En la tabla 6, se puede observar los roles de las personas a cargo del proyecto.

Tabla 6: Tabla de roles.

Rol	Descripción	Personal
Product Owner	Es la persona responsable de transmitir las necesidades de la empresa al equipo de trabajo.	Ing. Henry Paca.
Scrum Master	Asigna las tareas a las personas del equipo según su experiencia.	MsC. Pamela Buñay.
Team	Tiene la responsabilidad de llevar a cabo el desarrollo del sistema.	Sr. Danny Coro.

- **Planificación y estimación.**

Durante esta etapa, se lleva a cabo la creación de historias de usuario, la identificación de tareas en el Product Backlog, y la planificación en el Sprint Backlog.

- **Requerimientos funcionales.**

Se obtuvieron los requisitos y especificaciones para el desarrollo del sistema web al entrevistar al Ing. Henry Paca, coordinador del (Codesi). Los requerimientos funcionales de la aplicación se definieron en juntas con el equipo de trabajo.

En la tabla 7, se puede observar los requerimientos funcionales del proyecto.

Tabla 7: Requerimientos Funcionales del proyecto.

Requerimiento:	RF01
Nombre	Base de datos.

Característica	Los datos que se usarán en el sistema de bienes e inventarios serán guardados en una base de datos.
Descripción	Los datos se almacenan en una base de datos relacional.
Prioridad	Alta.
Requerimiento:	RF02
Nombre	Registro de Bienes.
Característica	El sistema deberá permitir registrar la información del Bien.
Descripción	El usuario registrara bienes.
Prioridad	Alta.
Requerimiento:	RF03
Nombre	Modificar Bienes.
Característica	El sistema deberá permitir modificar la información del Bien.
Descripción	El usuario modificara bienes.
Prioridad	Alta.
Requerimiento:	RF04
Nombre	Generar Informe.
Característica	El módulo de constatación permitirá generar un reporte detallado sobre la información de que bien fue constatado, e incluso permitirá descargarse en archivo PDF.
Descripción	El usuario genera y descarga informe sobre bienes constatados.
Prioridad	Alta.
Requerimiento:	RF05
Nombre	Eliminar Bienes.
Característica	El sistema deberá permitir eliminar la información del Bien.
Descripción	El usuario elimina bienes no deseados.
Prioridad	Alta.
Requerimiento:	RF06
Nombre	Lector de Barras.
Característica	El sistema de bienes e inventarios permitirá al usuario utilizar lector código de barras para constatar bienes.
Descripción	El usuario puede utilizar lector de código de barras.
Prioridad	Alta.

- **Requerimientos no funcionales**

Estos requisitos incluyen características como la seguridad, la disponibilidad, seguridad, usabilidad, seguridad, rendimiento etc.

En la tabla 8, se puede observar los requerimientos no funcionales del proyecto.

Tabla 8: Requerimientos no funcionales.

Requerimiento	Requisitos	Descripción
----------------------	-------------------	--------------------

RNF01	Usabilidad	El sistema deberá atraer al usuario, debe ser fácil de utilizar.
RNF02	Seguridad	El sistema deberá salvaguardar su información, de lo cual se encarga el departamento de Tics. Quien garantiza que solamente usuarios que correspondan puedan acceder a la información.
RNF03	Rendimiento	El sistema estará en la capacidad de resolver las peticiones en un tiempo estimado agilizando los procesos y actividades del usuario.
RNF04	Disponibilidad	El sistema debe estar disponible al usuario los 365 días del año y las 24 horas del día, garantizando la disponibilidad del 99.999%.

3.8.2 Planificación.

- **Product Backlog.**

Enumera todas las tareas que se deben realizar para completar un proyecto. La tabla muestra un listado de tareas establecidas según esta propuesta. Las tareas se clasifican como historias de usuario (HU) u historias técnicas (HT). Cada tarea tiene un número de identificación y un esfuerzo estimado, que es un rango de 1 a 5, donde 1 significa poco esfuerzo y 5 significa mucho esfuerzo.

En la tabla 9, se puede observar la lista de tareas denominadas historias técnicas.

Tabla 9: Lista de tareas según su escala de esfuerzo

HISTORIAS TÉCNICAS		
Detail	Asignación	Empeño
Ht 1	Exploración de los requisitos tanto funcionales como no funcionales del sistema.	5
Ht 2	Crear historias de usuario y técnicas.	5
Ht 3	Definir la estructura del sistema.	5
Ht 4	Implementación y ajuste de herramientas necesarias para el desarrollo del sistema.	5
Ht 5	Diseñar el prototipo de la interfaz de usuario.	5

Ht 6	Diseñar la base datos entidad relación.	5
Ht 7	Desarrollo de la base de datos.	5
Ht 8	Crear los modelos, vistas y controladores (MVC).	4
Ht 9	Configuración de servicios utilizados.	4
Ht 10	Desarrollar cada módulo de la aplicación web.	5
Ht 11	Desarrollo de la navegación en la aplicación web.	4
Ht 12	Crear y descargar reportes.	4
Ht 13	Evaluar la aplicación web utilizando el modelo Furps.	4
Ht 14	Examinar los resultados.	3
Ht 15	Elaborar el manual técnico para el sistema web.	2

En la tabla 10, se puede observar las historias de usuario.

Tabla 10: Historias de usuario.

HISTORIAS USUARIO		
Detail	Asignación	Empeño
Hu 1	Como usuario necesito un buscador dinámico que pueda facilitar encontrar cada bien ya sea por Dependencia o Marca.	5
Hu 2	Como usuario necesito un buscador por código que tiene cada bien.	5
Hu 3	Como usuario necesito saber cuál bien está en mantenimiento y porque motivo.	5
Hu 4	Como usuario necesito agregar un nuevo bien.	5
Hu 5	Como usuario necesito visualizar reporte de bienes que están en mantenimiento o funcionando etc.	5

Hu 6	Como usuario necesito un lector de código de barras.	5
Hu 7	Como usuario necesito visualizar el reporte de bienes constatados.	5
Hu 8	Como usuario necesito generar código de barras correspondiente a cada uno de los bienes.	4

Diagrama de casos de uso: Acceso al Sistema

En la figura 4, se puede observar el diagrama de uso del ingreso al sistema.



Figura 4: Caso de Uso del Usuario.

Diagrama de casos de uso: Registro de Nuevo Bien.

En la figura 5, se puede observar el diagrama de uso para el ingreso de un nuevo bien en el sistema.

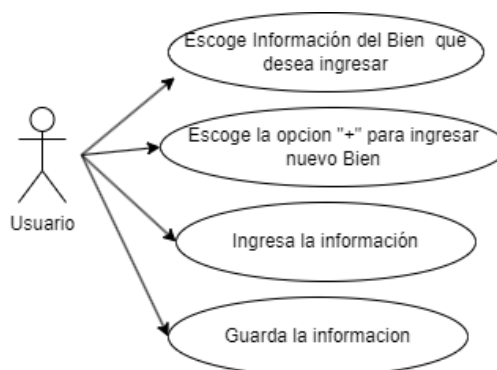


Figura 5: Caso de Uso ingresar información de un bien.

Diagrama de casos de uso: Modificación de la Información del Bien.

En la figura 6, se puede observar el diagrama de uso para la edición de la información de un bien.

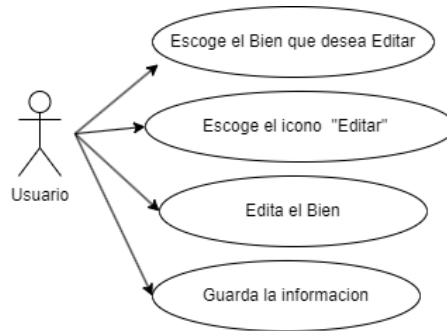


Figura 6: Caso de Uso editar información de bien.

Diagrama de casos de uso: Eliminar Registro de Bien.

En la figura 7, se puede observar el diagrama de uso para la eliminar la información de un bien.

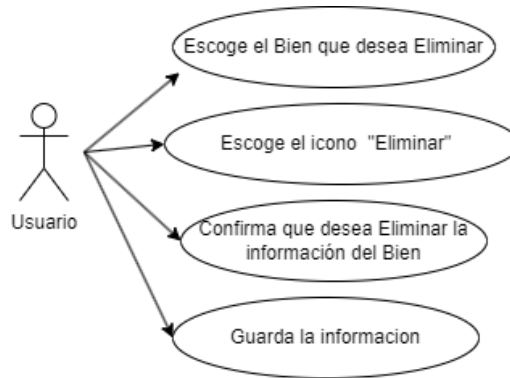


Figura 7: Caso de Uso Eliminar bien.

Diagrama de casos de uso: Constatación de Bien.

En la figura 8, se puede observar el diagrama de uso para la constatación de la información de un bien.

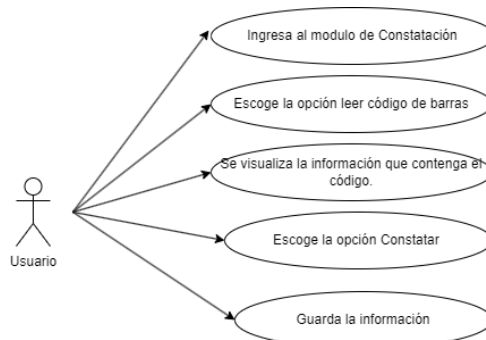


Figura 8: Caso de Uso Constatación bien.

- **Planificación del Sprint.**

El grupo Scrum se congregó con el propósito de organizar las labores correspondientes a los sprints. Durante las reuniones, el Ing. Henry Paca y el grupo de trabajo intercambiaron ideas sobre las actividades pendientes y el plazo disponible para su ejecución. Consideraron las

habilidades y conocimientos del equipo de desarrollo, así como la carga de trabajo asociada a cada ítem del Product backlog. Con esta información como base, el equipo Scrum diseñó la planificación de sprints que contiene un conjunto de actividades que el equipo debe finalizar dentro de un plazo definido.

SPRINT 1.

En el primer Sprint se asignaron 3 semanas en el desarrollo de los siguientes elementos. A continuación en la tabla 11 se aprecia las historias técnicas y de usuario para el Sprint 1.

Tabla 11: Lista de tareas del Sprint 1.

Detail	Asignación	Empeño
Ht 1	Exploración de los requisitos tanto funcionales como no funcionales del sistema.	5
Ht 2	Crear historias de usuario y técnicas.	5
Ht 3	Definir la estructura del sistema.	5
Ht 4	Implementación y ajuste de herramientas necesarias para el desarrollo del sistema.	5

SPRINT 2.

En el segundo Sprint se asignaron 5 semanas en el desarrollo de los siguientes elementos. A continuación, en la tabla 12 se aprecia las historias técnicas para el Sprint 2.

Tabla 12: Lista de tareas del Sprint 2.

Detail	Asignación	Empeño
Ht 5	Diseñar el prototipo de la interfaz de usuario.	5
Ht 6	Diseñar la base datos entidad relación.	5
Ht 7	Desarrollo de la base de datos.	5
Ht 8	Crear los modelos, vistas y controladores (MVC).	5

SPRINT 3.

En el tercer Sprint se asignaron 3 semanas en el desarrollo de los siguientes elementos. A continuación, en la tabla 13 se aprecia las historias técnicas y de usuario para el Sprint 3.

Tabla 13: Lista de tareas del Sprint 3.

Detail	Asignación	Empeño
Ht 9	Configuración de servicios utilizados.	4
Ht 10	Desarrollar cada módulo de la aplicación web.	5
Hu 1	Como usuario necesito un buscador dinámico que pueda facilitar encontrar cada bien ya sea por Dependencia o Marca.	5
Hu 2	Como usuario necesito un buscador por código que tiene cada bien.	5
Hu 3	Como usuario necesito saber cuál bien está en mantenimiento y porque motivo.	5
Ht 11	Desarrollo de la navegación de la aplicación web.	4
Ht 12	Crear y descargar reportes.	5

SPRINT 4.

En el cuarto Sprint se asignaron 3 semanas en el desarrollo de los siguientes elementos. A continuación, en la tabla 14 se aprecia las historias técnicas y de usuario para el Sprint 3.

Tabla 14: Lista de tareas del Sprint 4.

Detail	Asignación	Empeño
Ht 13	Evaluar la aplicación web utilizando el modelo Furps.	4
Ht 14	Examinar los resultados.	3
Ht 15	Elaborar el manual técnico para el sistema web.	2

Incremento del producto.

Al concluir cada sprint, el grupo de trabajo presentó al Product Owner los avances de la aplicación web. Una vez que el grupo de trabajo y el Product Owner verificaron y aprobaron los avances.

3.8.3 Desarrollo.

Durante la fase de desarrollo, se llevaron a cabo las tareas necesarias para crear el producto. A continuación, se detallan las siguientes tareas:

3.8.3.1 Diseño.

Arquitectura de Software.

Se utilizó la arquitectura Cliente/Servidor la cual se basa en la distribución de las tareas entre el servidor encargado de proveer de los recursos y servicios y el cliente quien realiza las peticiones. La arquitectura cliente/servidor se compone de:

- Red: Mediante protocolos de transmisión, un grupo de usuarios, servidores y base de datos se comunican entre sí es física o no.
- Cliente: formula peticiones al servidor mediante una interfaz gráfica.
- Servidor: Procesa las peticiones del cliente y envía la respuesta al cliente.
- Base de datos: Facilitan al servidor y el cliente un conjunto de datos almacenados para su procedimiento.

En la figura 9, se puede observar el diagrama de la arquitectura cliente servidor utilizada en el sistema.

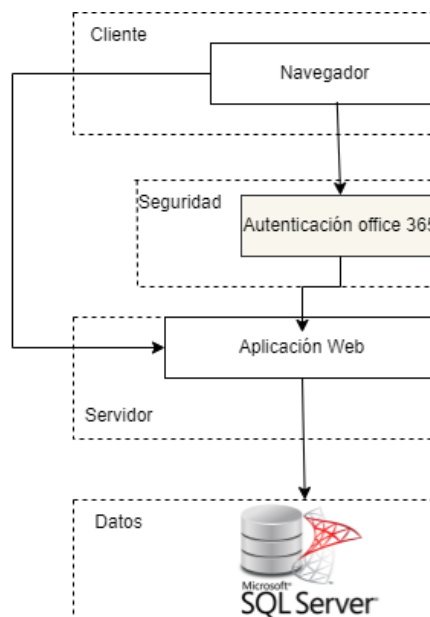


Figura 9: Arquitectura del sistema.

- **Estándar de codificación.**

Pascal Case.

La notación Pascal Case es una forma de escribir nombres de variables, clases y otros elementos de código en la que cada palabra se combina directamente, sin usar ningún símbolo como por ejemplo: HolaSoyDeveloper.

Camel Case.

Es un estilo de nomenclatura que se utiliza para nombrar elementos de código. La primera palabra se escribe en minúscula, y las palabras siguientes se combinan directamente, como por ejemplo: `registorBack`, `holaBien`.

Herramientas utilizadas en el Desarrollo.

La tabla 15 muestra los instrumentos o herramientas utilizadas al desarrollar los módulos correspondientes. Cada herramienta tiene una breve descripción que explica su función.

Tabla 15: Herramientas para el Desarrollo del proyecto.

Herramienta	Descripción
Visual Studio 2022	Entorno de desarrollo
Net Core, Entity Framework	Framework
Net Core, Identity Framework	Framework
Sql Server	Entorno de desarrollo para la base de datos.
C#	Idioma de codificación
Jmeter	Herramienta de evaluación del rendimiento

Elaboración del diseño de la base de datos.

La (BD) está conformada diferentes tablas donde cada una de ellas posee su respectiva Id y claves foráneas si fuese el caso. Se aplicó un proceso de normalización hasta alcanzar la tercera forma, con el propósito de mejorar la calidad de la información almacenada. A continuación en la figura 10, se puede observar el diseño de la base de datos.

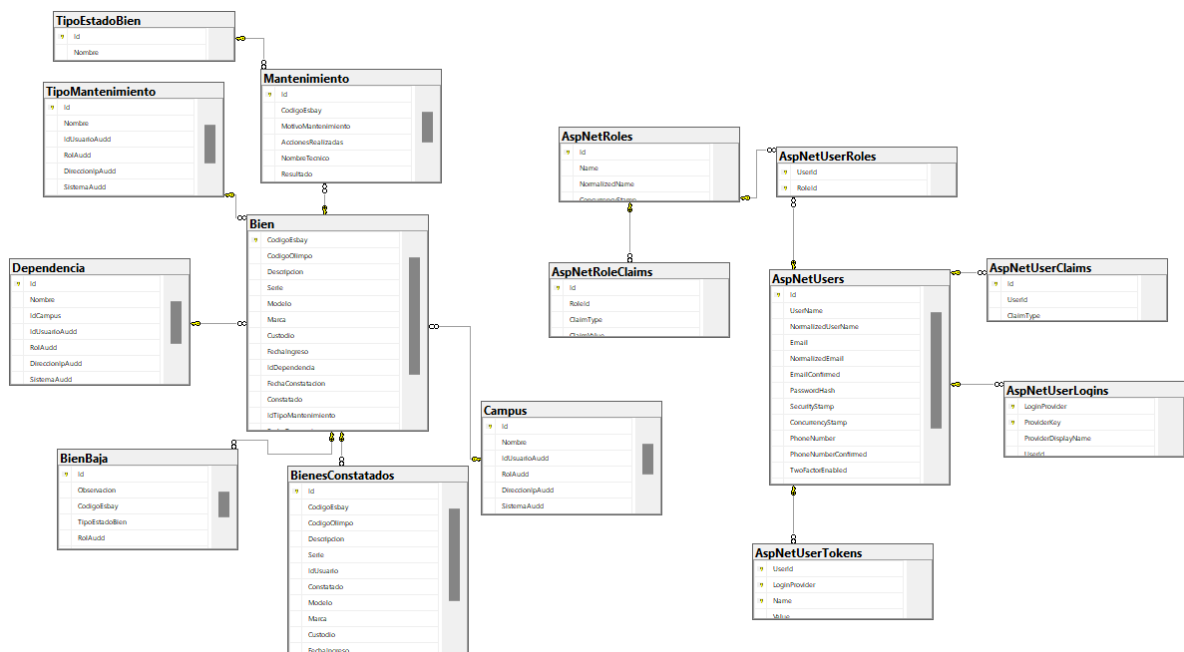


Figura 10: Diseño de la base de datos.

3.8.4 Implementación.

Se creó un proyecto ASP.NET Core MVC .Net en Visual Studio, utilizando la versión 7 de .NET. A continuación, se muestra en la Figura 11 el inicio del proyecto.

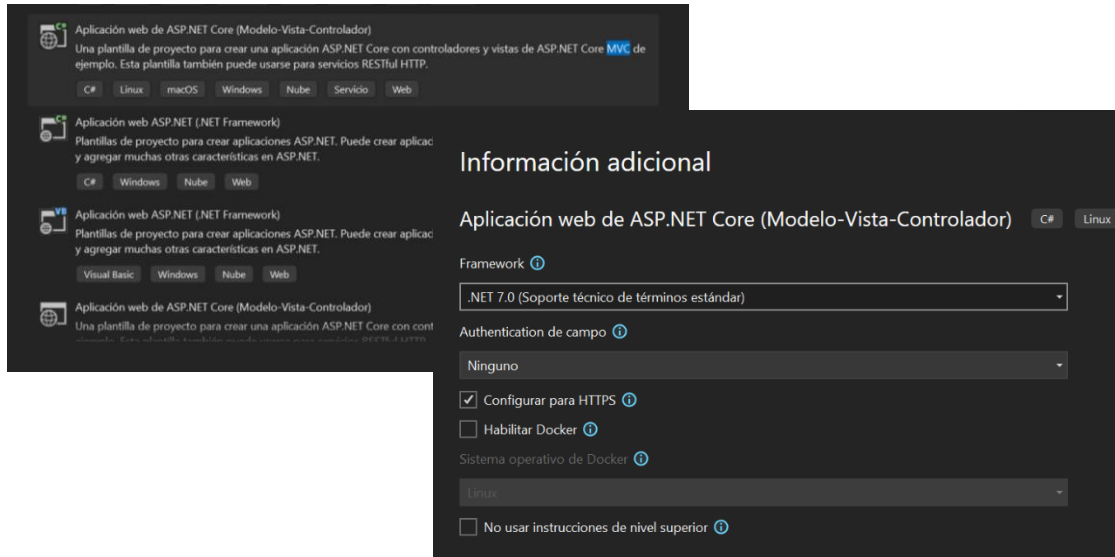


Figura 11: Constitución de la aplicación.

Al crear la aplicación Mvc .net 7 se instalan todos los paquetes nuget necesarios para la aplicación como Entity framework, Sql server Tools etc. A continuación en la figura 12 se observan todos los paquetes instalados.

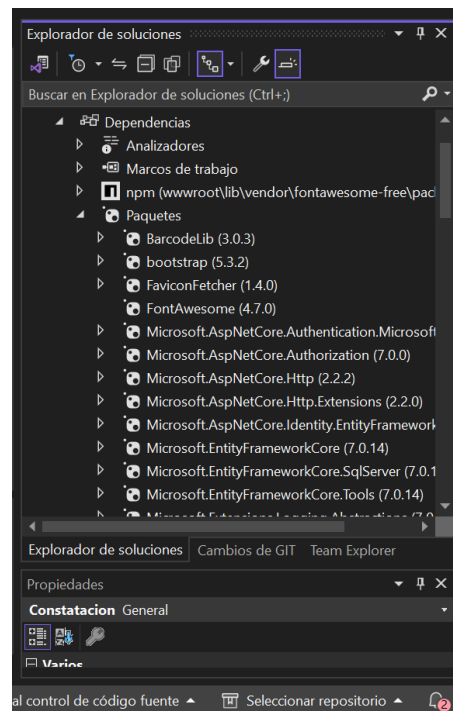


Figura 12: Paquetes instalados.

Se mapean la base de datos gracias a Entity Framework y la función Ef power tools. El manejo de los datos ocurre cuando se utiliza un modelo llamado dominio y un contexto llamado (DbContext) estos permite consultar y guardar los datos. A continuación en la figura 13 se muestran los modelos mapeados.

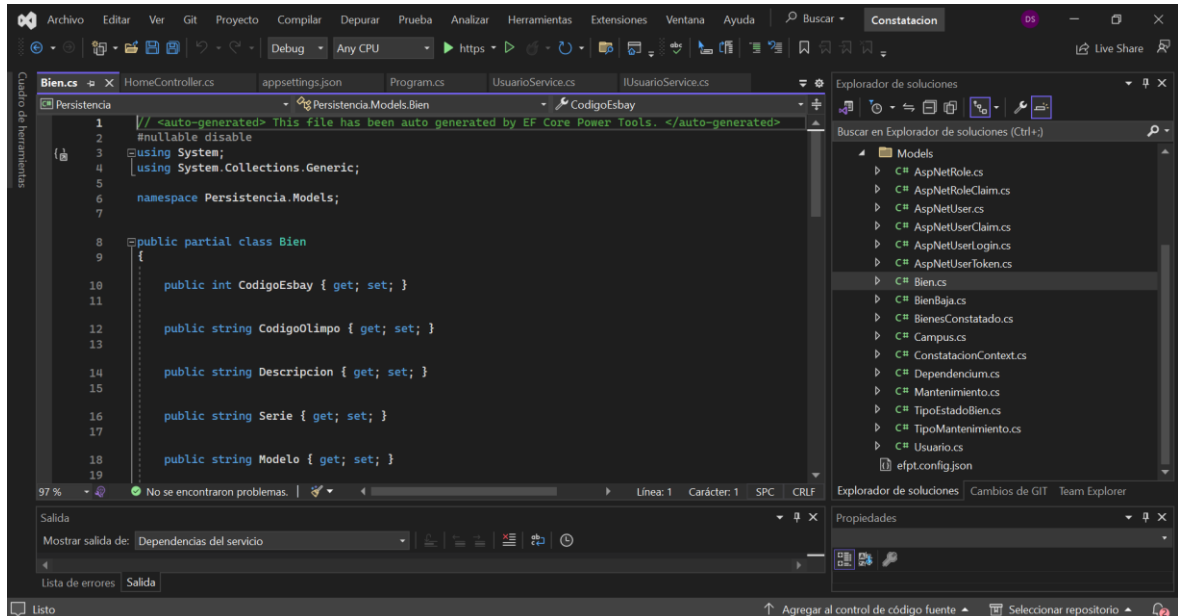


Figura 13: Modelos en la base de datos.

Luego de crear el proyecto, inició la creación de todos los controladores necesarios para la petición de datos y sus respectivos métodos como Post y Get utilizando las bibliotecas de datos justo con los repositorios que ya contenían métodos genéricos para ser utilizados en consultas de información. A continuación en la figura 14 se muestran los *controllers* creados.

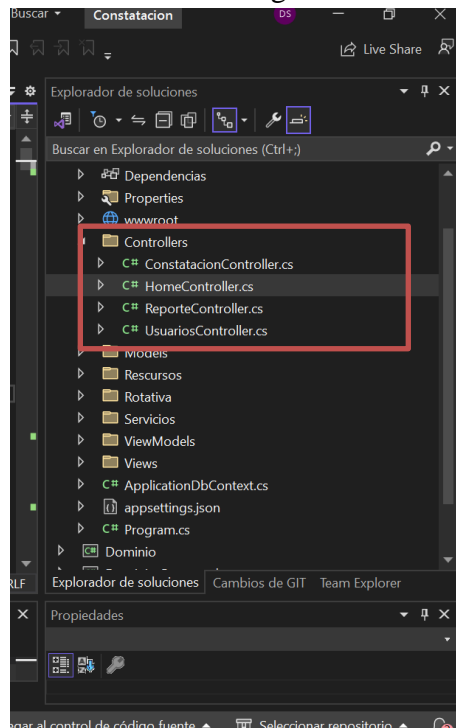


Figura 14: Controladores necesario.

Se creó todas las vistas necesarias para generar la interfaz de usuario. A continuación, en la figura 15 se muestra las vistas utilizadas.

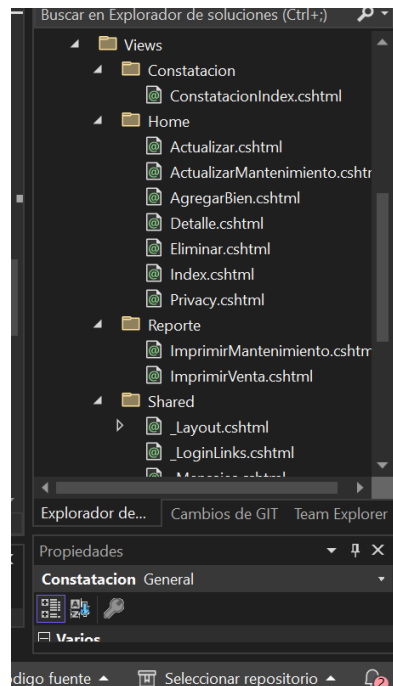


Figura 15: Vistas generadas.

En el proyecto se realizó la configuración de `builder.Services.AddAuthentication`, para que el sistema permita iniciar sesión con la cuenta institucional office 365. A continuación, en la figura 16 se muestra las configuraciones necesarias para el proyecto.

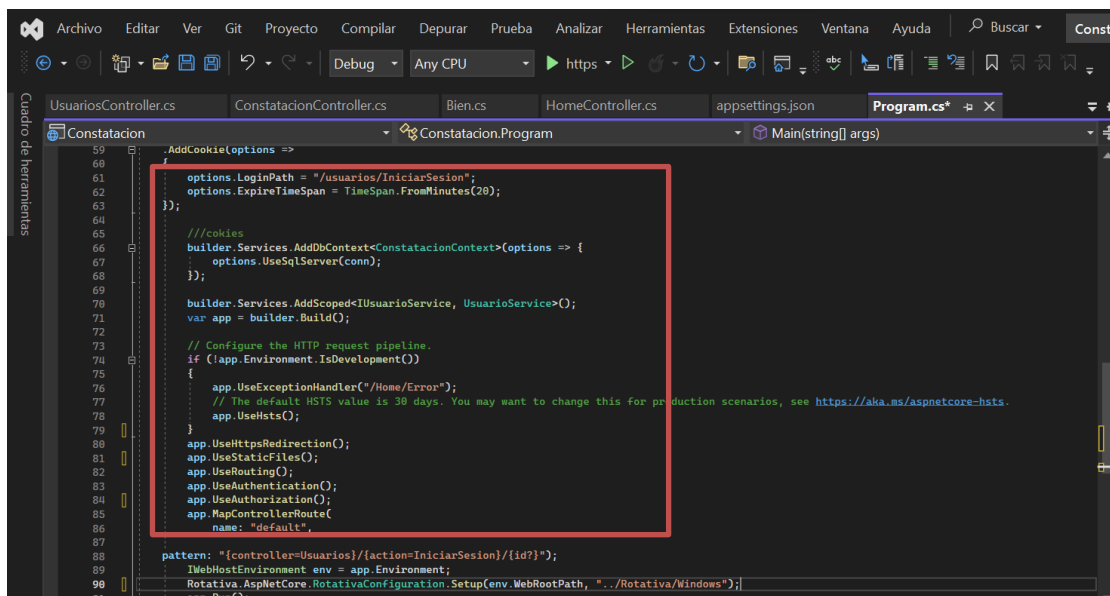


Figura 16: Configuraciones necesarias.

Interfaz gráfica del sistema (prototipo).

El usuario puede ingresar con la autenticación Microsoft 365, luego se tiene la página de inicio donde se muestra el buscador por marca y modelo, se puede filtrar por dependencia y campus, de ese modo se puede interactuar con los datos registrados dinámicamente. A continuación, en la figura 17 se muestra el módulo de bienes.

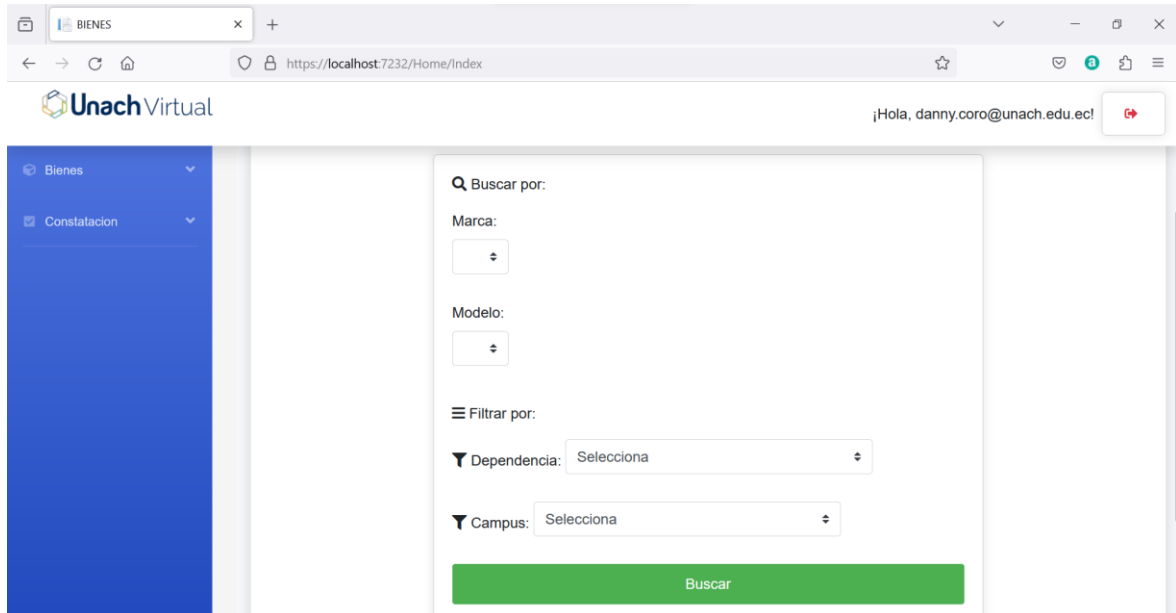


Figura 17: Módulo para visualizar bienes.

El usuario puede ingresar al Módulo de Constatación, donde por medio de una app externa puede leer código correspondiente a cada bien además de poder generar y descargar el código si es necesario. A continuación, en la figura 18 se puede observar Módulo de Constatación por lector de código de barras.

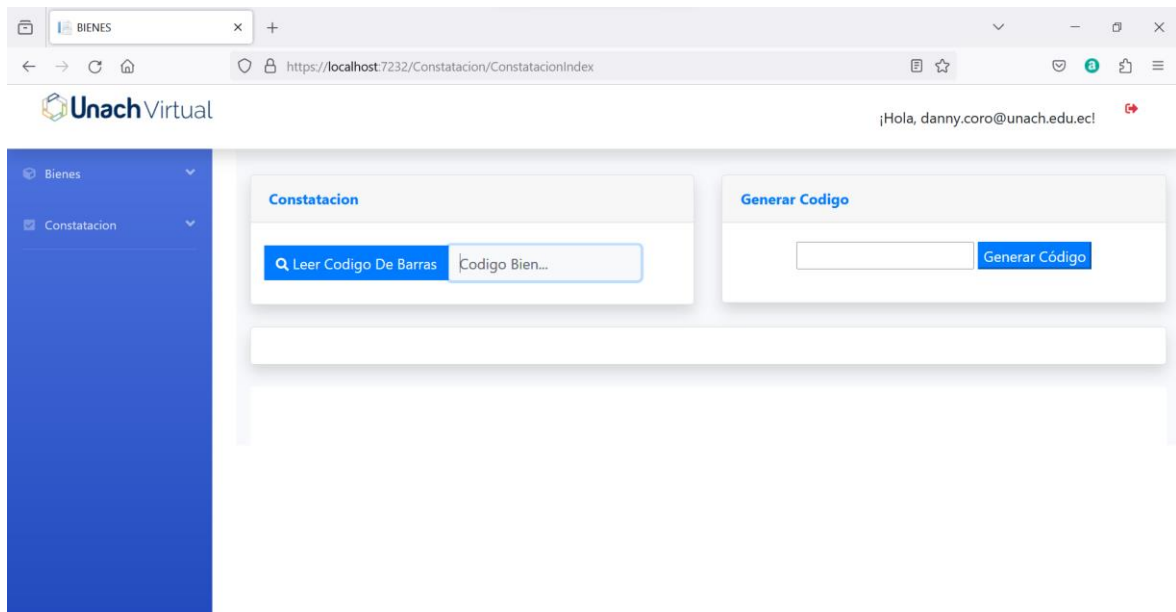


Figura 18: Módulo de Constatación por lector de código de barras.

El usuario puede Generar varios reportes como lo son de Bienes Constatados o en mantenimiento. A continuación, en la figura 19 se puede observar el prototipo de reportes de los bienes constatados

CodigoEsbay	MotivoMantenimiento	AccionesRealizadas	Resultado	Marca	Descripcion	Serie	Modelo	Custodio	Nombre	
1	dsada	lalll bbb	xx	dasdadasda	xxx	SERVDOR	13	samsung	OROZCO PILCO JONATHAN JAVIER	Funcionando
2	xx	xx	xx	xx	xx	SERVDOR	12	samsung	OROZCO PILCO JONATHAN JAVIER	Inserible
3	no se	ninguna	juan pere	333cec	EPSON	SERVDOR	12	samsung	OROZCO PILCO JONATHAN JAVIER	Obsoleto

Figura 19: Prototipo de reportes bienes Constatados.

A continuación, en la figura 20 se puede observar el prototipo de reportes de los bienes en mantenimiento.

Codigo	Serie	Marca	Modelo	Custodio	MotivoMantenimiento	AccionesRealizadas	Tecnico	Estado
1	13	xxx	samsung	OROZCO PILCO JONATHAN JAVIER	ninguno	ninguno	xx	ninguno
2	12	xx	samsung	OROZCO PILCO JONATHAN JAVIER	xx	xx	xx	xx
3	12	EPSON	samsung	OROZCO PILCO JONATHAN JAVIER	no se	ninguna	juan pere	333cec
4	12	espen	samsung	OROZCO PILCO JONATHAN JAVIER				
35902444	QMMS	ARGFIT JUMP	QM	LORENZO BERTHEAU EDDA	ninguno	ninguno	ninguno	ninguno
0	21313	epson	epson	edison	ninguno	ninguno	ninguno	ninguno
35495064	ARDEK1	ARTDEKCOR	ARTDK	MUY CABRERA NELSON ISMEL	ninguno	ninguno	ninguno	ninguno
35780046	N8NRKD01757334	ASUS	LAPTOP GAMER ASUS ROG	FLOR MORA CNRR	ninguno	ninguno	ninguno	ninguno

Figura 20: Prototipo de reportes bienes en Mantenimiento.

3.8.5 Lanzamiento.

Una vez que se completaron los módulos de bienes y constatación, se creó un manual de usuario para facilitar el acceso al sistema. Además, se realizaron pruebas de rendimiento para garantizar que los módulos funcionaran correctamente y para identificar cualquier problema de rendimiento.

4. CAPÍTULO IV. RESULTADOS Y DISCUSIÓN.

Después de completar el desarrollo del sistema web de bienes e inventarios para la Universidad Nacional de Chimborazo aplicando la plataforma .Net core, se llevó a cabo pruebas de rendimiento para evaluar la calidad del sistema. Las pruebas se realizaron con la herramienta Jmeter y se contrastaron con el modelo de calidad FURPS.

4.1 Análisis de eficacia.

Según los datos presentados en la tabla 16, se visualiza la evaluación de la eficacia del sistema de bienes mediante la realización de pruebas. En cuanto a peticiones exitosas, se logró un rendimiento del 100%, realizando 7650 solicitudes HTTP con la participación de 150 usuarios por segundo. El anexo A figura 26 proporciona una descripción detallada de las respuestas satisfactorias generado por el sistema.

Tabla 16. Análisis de Eficacia.

Dimensión	Indicador	Número de Peticiones
Eficacia	Numero de procesos realizados exitosamente.	7650
Promedio %		100%

A continuación, en la figura 21 se puede observar la evaluación de la eficacia del sistema de bienes según su porcentaje de solicitudes fallidas o exitosas.

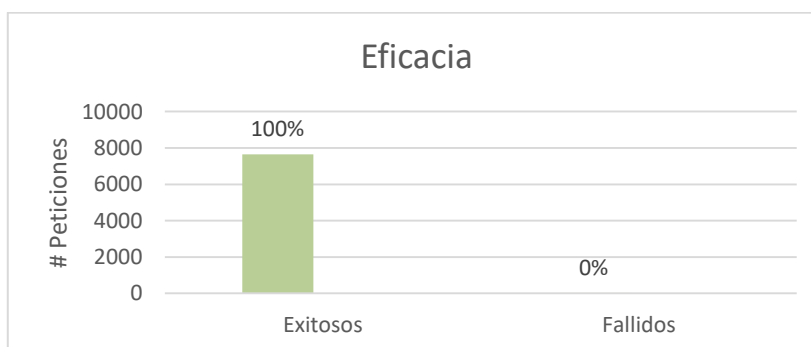


Figura 21: Grafico eficacia de la aplicación Web.

4.2 Análisis de Tiempo de respuesta.

Se evaluó el comportamiento temporal del sistema web de bienes, considerando todas sus funciones en conformidad con los requisitos establecidos. La figura 22 y 23 muestra los datos obtenidos de las mediciones según el tiempo de respuesta, realizadas con la participación de 150 usuarios por segundo. Como resultado, se obtuvo un promedio de 114 ms.

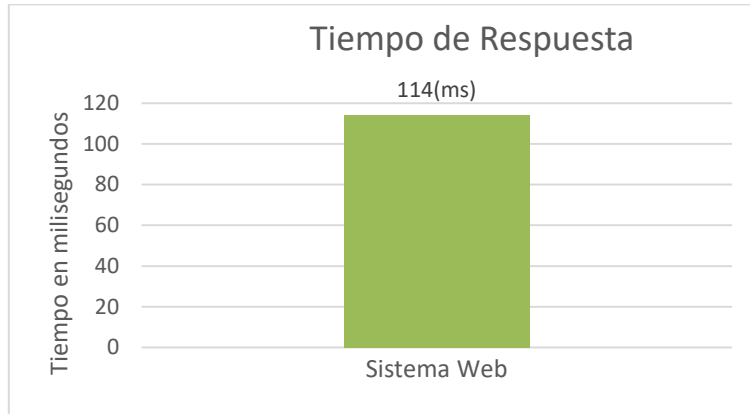


Figura 22: Tiempo de respuesta de la aplicación.

Muestra #	Tiempo de co...	Nombre del h...	Etiqueta	Tiempo de M...	Estado	Bytes	Sent Bytes	Latency	Connect Tim...
7631	23:47:41.171	PruebaRendi...	ReporteCons...	106	✓	216	305	106	0
7632	23:47:41.182	PruebaRendi...	ReporteCons...	95	✓	216	305	95	0
7633	23:47:41.180	PruebaRendi...	ReporteCons...	99	✓	298	303	99	0
7634	23:47:41.171	PruebaRendi...	ReporteCons...	108	✓	216	305	108	0
7635	23:47:41.278	PruebaRendi...	ReporteCons...	100	✓	216	305	100	0
7636	23:47:41.280	PruebaRendi...	ReporteCons...	98	✓	216	305	98	0
7637	23:47:41.279	PruebaRendi...	ReporteCons...	91	✓	216	305	91	0
7638	23:47:41.271	PruebaRendi...	ReporteCons...	93	✓	216	305	93	0
7639	23:47:41.267	PruebaRendi...	ReporteCons...	94	✓	216	305	94	0
7640	23:47:41.267	PruebaRendi...	ReporteCons...	91	✓	216	305	91	0
7641	23:47:41.263	PruebaRendi...	ReporteCons...	94	✓	298	303	94	0
7642	23:47:41.263	PruebaRendi...	ReporteCons...	93	✓	216	305	93	0
7643	23:47:41.262	PruebaRendi...	ReporteCons...	94	✓	216	305	94	0
7644	23:47:41.257	PruebaRendi...	ReporteCons...	94	✓	216	305	93	0
7645	23:47:41.257	PruebaRendi...	ReporteCons...	93	✓	216	305	93	0
7646	23:47:41.250	PruebaRendi...	ReporteCons...	89	✓	216	305	89	0
7647	23:47:41.610	PruebaRendi...	ReporteCons...	92	✓	216	305	92	0
7648	23:47:41.610	PruebaRendi...	ReporteCons...	93	✓	216	305	93	0
7649	23:47:41.610	PruebaRendi...	ReporteCons...	94	✓	216	305	94	0
7650	23:47:41.703	PruebaRendi...	ReporteCons...	103	✓	216	305	103	0

Figura 23: Análisis de peticiones y tiempo de respuesta de la aplicación.

Se logró un tiempo de reacción de 0.114 segundos por cada solicitud, y se muestra que cada registro refleja un estado de finalización exitoso.

4.3 Análisis en la utilización de recursos.

Se procedió a verificar el tiempo de consumo de recursos durante la ejecución del sistema. A continuación en la tabla 17, se muestra el análisis del uso del CPU, utilización de la memoria RAM y el promedio de la utilización del disco duro.

Tabla 17: Utilización de recursos en porcentaje.

Dimensión	Indicador	Porcentaje
Consumo de recursos	Utilización CPU.	10%
	Consumo de memoria RAM	53%
	Promedio de utilización del disco.	0%

En la figura 24 se mostró que se había alcanzado un promedio de utilización de la CPU del 10%, con un promedio del 53% para la memoria RAM. Por último, se registró un uso del disco del 0%.

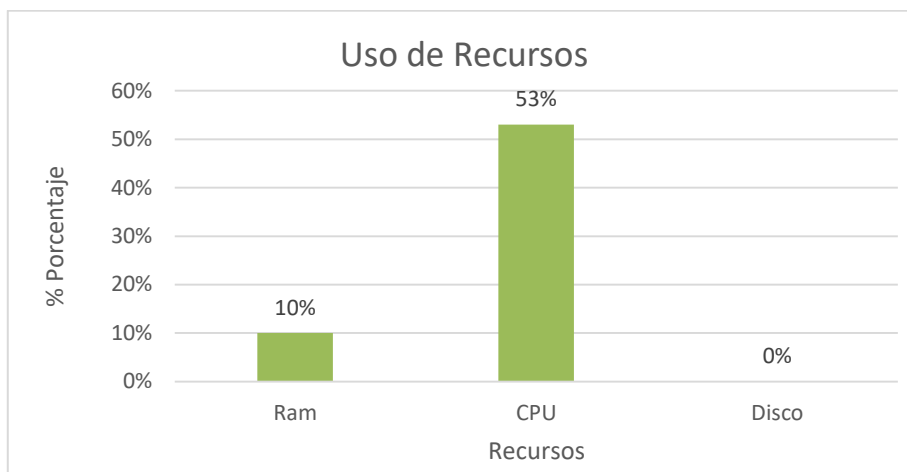


Figura 24: Análisis del tiempo de respuesta.

4.4 Contraste entre los valores establecidos en el modelo Furps y los valores obtenidos.

Tras analizar el rendimiento del Sistema de Bienes e inventarios, se compararon los resultados con los parámetros establecidos por Furps. A continuación en la tabla 18 se observa los valores que obtuvieron en la evolución junto con los parámetros establecidos por el modelo Furps:

Tabla 18: Los resultados alcanzados comparados con los establecidos por el modelo Furps.

Dimensión	Indicador	Peticiones
Eficacia	95 %.	100 %.
Tiempo de respuesta	5 s.	0,114 s.
Uso de Recursos	25 %.	21 %.

4.5 Discusión

Para evaluar el rendimiento del sistema de bienes e inventarios de la (Unach), se utilizó la herramienta Jmeter. Los resultados obtenidos se compararon con los parámetros establecidos por el modelo furps. Los resultados mostraron que el sistema es altamente eficaz, con un éxito del 100 %. Esto supera el requisito establecido por furps de un 95 % de éxito. El tiempo de respuesta promedio del sistema es de 0,114 segundos, lo que es mucho más rápido que el límite de 5 segundos establecido por furps. La utilización de recursos promedio del sistema es del 21 %, lo que es significativamente menor que el límite del 25 % establecido por furps. Estos resultados indican que el sistema de bienes e inventarios de la (Unach) es de alta calidad y además cumple con los estándares de rendimiento establecidos.

5. CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES.

5.1 Conclusiones.

- Se aplicó .Net Core ya que ofrece un entorno de desarrollo robusto que permite la creación de aplicaciones web modernas y eficientes. Al analizar la plataforma .Net Core desde el punto de vista del desarrollo de aplicaciones web, se destaca su relevancia y versatilidad.
- El sistema de bienes e inventarios se desarrolló mediante la metodología ágil de desarrollo de software Scrum, bajo el entorno de desarrollo Visual Studio 2022 y el lenguaje de programación de alto nivel C#. Además, se incorporaron CSS y JavaScript para simplificar el diseño de la interfaz, y se empleó Sql server para la base de datos.
- Resultó altamente beneficioso el uso de Entity Framework para mapear las bases de datos de manera simple y eficiente, incluyendo una biblioteca de clases con métodos de consultas genéricos en Linq, favorece la modularidad y la integración de funciones adicionales en relación con la consulta de datos.
- Jmeter fue utilizado para obtener los valores del rendimiento del sistema evaluado, se realizó la ejecución de 150 peticiones simultáneas. Cada una de las peticiones fueron exitosas, superando el 95 % establecido. Además, el tiempo de respuesta promedio del sistema fue de 0,114 segundos, es muy rápido y cumple con el límite de 5 segundos establecido por Furps. En cuanto al uso de recursos, el consumo promedio de recursos fue del 21 %, es inferior al límite del 25 % establecido por Furps. Estos resultados indican que el sistema satisface los requisitos de rendimiento establecidos por el modelo Furps.

5.2 Recomendaciones.

- Para el desarrollo de sistemas Web se sugiere el uso de la plataforma .Net Core incluyendo bibliotecas de clase, además, Entity Framework facilitará el desarrollo de cualquier proyecto a futuro, mejorará el rendimiento con respecto a grandes volúmenes de datos procesados e incluirá buenas prácticas al momento de programar.
- Emplear metodologías de desarrollo de software ágiles que incluyan etapas en sus procesos, asegurando la agilidad en la gestión de requisitos y la facilidad para modificar los procesos.
- Al desarrollar un producto (Software) y luego evaluarlo, se sugiere definir parámetros tomando como referencia modelos de calidad, como el modelo Furps, además, la utilización del software Jmeter por su interfaz intuitiva y es gratuito, puede monitorear el rendimiento en tiempo real, esto hace que la comunidad de Jmeter sea extensa y por lo tanto, se puedan encontrar una amplia gama de documentos para facilitar el proceso de aprendizaje.

6. BIBLIOGRAFÍA.

- [1] V. Garcia, "kizeo," 21 03 2022. [Online]. Available: <https://www.kizeo-forms.com/es-lat/que-es-un-sistema-de-control-de-inventarios/>.
- [2] J. J. Moreno, "revistas," [Online]. Available: <https://revistas.uis.edu.co/index.php/revistausingenierias/article/view/1055>. [Accessed 19 01 2024].
- [3] D. A. Caillamara Encalada, Artist, *Sistema de desvinculación laboral y permisos del departamento de*. [Art]. UNIVERSIDAD NACIONAL DE CHIMBORAZO , 2023.
- [4] J. A. C. LÁZARO, "DESARROLLO DEL MÓDULO DE CONTABILIDAD EN LENGUAJE," pamplona, 2019.
- [5] J. Richter, Artist, *Applied Microsoft .NET Framework*. [Art]. Microsoft Press, 2002.
- [6] m. yazmin, Artist, *Introducción a .net*. [Art]. TI Capacitación, 2008.
- [7] microsoft, "microsoft," 2 05 2022. [Online]. Available: <https://learn.microsoft.com/es-es/aspnet/mvc/overview/older-versions-1/overview/asp-net-mvc-overview>.
- [8] microsoft, "Paseo por el lenguaje C#," 2022.
- [9] J. S. y J. Hernández, "Microsoft SQL Server," pp. 1-16.
- [10] e. framework, "entity framework," *ENTITY FRAMEWORK*, p. 1, 2021.
- [11] coreview, "Using transactions in entity framework core," coreviewsystems, 23 1 2020. [Online]. Available: <https://coreviewsystems.com/using-transactions-in-entity-framework-core/>.
- [12] R. D. V. César Rodríguez, "¿Por qué implementar Scrum?," *Revista ONTARE*, vol. 3, pp. 125-144, 2015.
- [13] madeja, "juntadeandalucia," 2 1 2020. [Online]. Available: <https://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/388>.
- [14] M. M. Yazmin Nabor, *Introducción a .net*, 2008.
- [15] N. Rojas, "techsbcn," techsbcn, 16 5 2022. [Online]. Available: <https://techsbcn.com/es/descubre-las-principales-novedades-net7/>.
- [16] netapp, "netapp," *Qué es DevOps*, 23 3 2022. [Online]. Available: <https://www.netapp.com/es/devops-solutions/what-is-devops/#:~:text=DevOps%20es%20un%20marco%20de,o%20productos%20para%20los%20clientes..>

7. Anexo A: Pruebas de rendimiento.

Se tomaron en cuenta todos los procesos del sistema web según los requerimientos los cuales se guardaron en un controlador de grabación y se configuró Jmeter para que existan 150 usuarios en 1 segundo. A continuación, en la figura 25 se puede observar la configuración de Jmeter

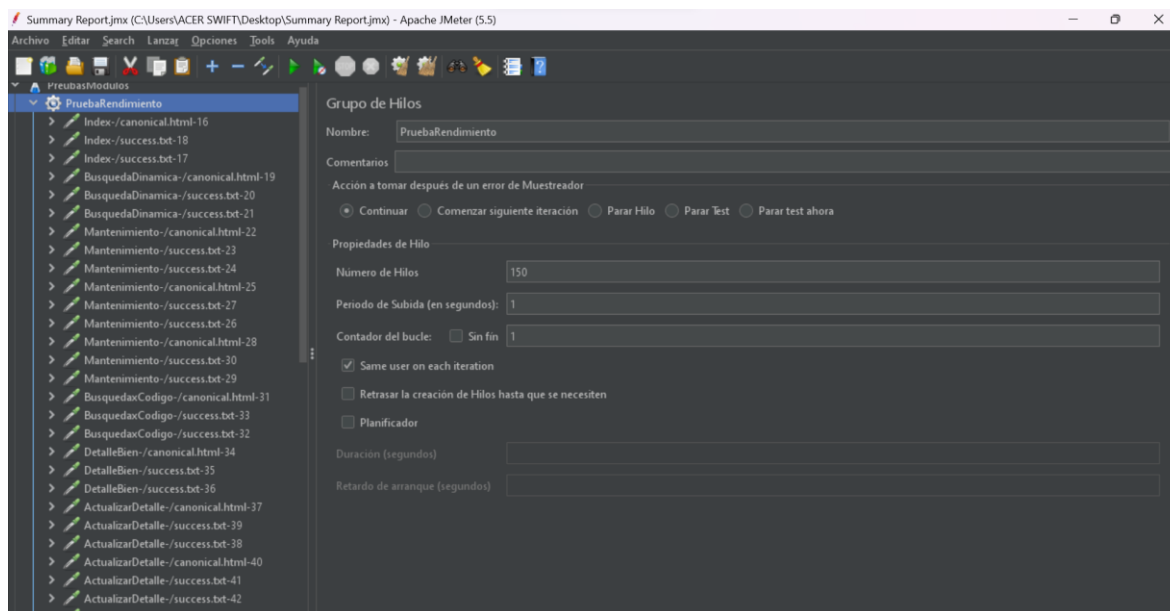


Figura 25: Pruebas de Rendimiento configuración.

Una vez hechas todas las configuraciones necesarias se procedió a iniciar la simulación donde apareció el cuadro de todas las peticiones con el estatus de 100% completadas. A continuación, en la figura 26 se puede observar las peticiones realizadas son correctas.

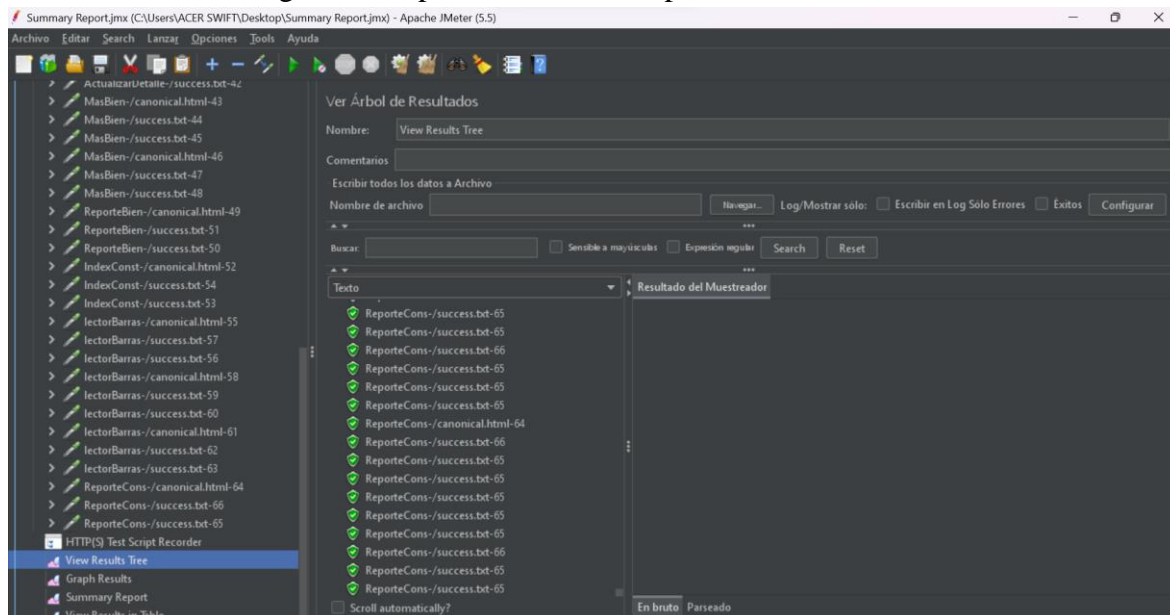


Figura 26: Peticiones Http correctas

En la figura 27, se puede observar las peticiones realizadas son correctas según el comportamiento del sistema Web.

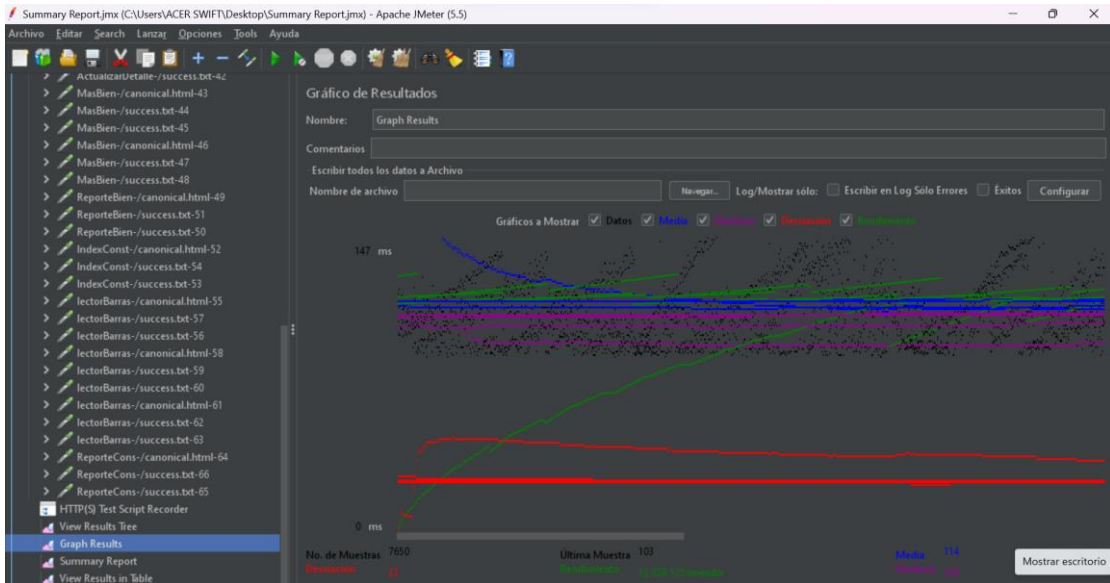


Figura 27: Grafico según su comportamiento del sistema.

En la figura 28, se puede observar las peticiones realizadas no tuvieron ningún fallo en el módulo de registro.

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estánd...	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de By...
ReporteBien...	50	97	87	120	6.17	0.00%	16.5/sec	3.49	4.92	216.0
ReporteBien...	50	95	89	112	4.37	0.00%	18.4/sec	3.89	5.49	216.0
IndexConst-...	50	99	91	113	5.10	0.00%	18.4/sec	5.36	5.45	298.0
IndexConst-...	50	95	89	110	4.59	0.00%	17.1/sec	3.61	5.10	216.0
IndexConst-...	50	95	90	158	9.44	0.00%	18.5/sec	3.91	5.52	216.0
lectorBarras...	50	99	90	115	5.41	0.00%	18.5/sec	5.38	5.47	298.0
lectorBarras...	50	95	88	109	4.55	0.00%	16.9/sec	3.56	5.03	216.0
lectorBarras...	50	94	89	113	4.52	0.00%	18.2/sec	3.84	5.42	216.0
lectorBarras...	50	99	89	118	6.52	0.00%	16.3/sec	4.75	4.83	298.0
lectorBarras...	50	94	89	117	6.11	0.00%	16.3/sec	3.44	4.86	216.0
lectorBarras...	50	95	89	113	4.94	0.00%	15.6/sec	3.30	4.65	216.0
lectorBarras...	50	98	89	126	7.46	0.00%	15.6/sec	4.54	4.62	298.0
lectorBarras...	50	96	89	111	5.76	0.00%	14.1/sec	2.97	4.19	216.0
lectorBarras...	50	96	89	114	6.33	0.00%	15.0/sec	3.16	4.47	216.0
ReporteCon...	50	96	89	112	5.33	0.00%	13.9/sec	4.04	4.11	298.0

Figura 28: Evaluación del rendimiento con 0% de error.

En la figura 29, se puede observar las peticiones realizadas no tuvieron ningún fallo.

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estánd...	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de By...
ReporteBien...	150	114	90	191	20.97	0.00%	42.3/sec	8.91	12.59	216.0
ReporteBien...	150	109	88	154	17.33	0.00%	45.9/sec	9.68	13.67	216.0
IndexConst...	150	115	89	170	19.81	0.00%	41.4/sec	12.04	12.24	298.0
IndexConst...	150	111	87	162	17.96	0.00%	41.2/sec	8.68	12.26	216.0
IndexConst...	150	113	88	187	19.90	0.00%	40.8/sec	8.61	12.16	216.0
lectorBarras...	150	115	90	185	21.20	0.00%	40.8/sec	11.87	12.07	298.0
lectorBarras...	150	115	88	178	18.21	0.00%	41.0/sec	8.65	12.21	216.0
lectorBarras...	150	110	88	175	17.17	0.00%	41.5/sec	8.76	12.37	216.0
lectorBarras...	150	113	90	180	19.09	0.00%	41.3/sec	12.01	12.21	298.0
lectorBarras...	150	109	88	179	17.99	0.00%	41.1/sec	8.67	12.24	216.0
lectorBarras...	150	111	88	182	20.17	0.00%	37.9/sec	7.99	11.28	216.0
lectorBarras...	150	112	89	177	18.73	0.00%	40.1/sec	11.66	11.85	298.0
lectorBarras...	150	110	88	183	19.27	0.00%	37.1/sec	7.83	11.06	216.0
lectorBarras...	150	107	88	172	18.27	0.00%	37.1/sec	7.82	11.04	216.0
ReporteCon...	150	109	89	178	16.88	0.00%	39.2/sec	11.42	11.61	298.0
ReporteCon...	150	102	88	153	13.80	0.00%	37.4/sec	7.88	11.13	216.0
ReporteCon...	150	103	87	149	13.24	0.00%	39.9/sec	8.42	11.90	216.0
Total	7650	114	87	208	23.61	0.00%	539.3/sec	128.15	160.28	243.3

¿Incluir el nombre del grupo en la etiqueta?

 Guardar la cabecera de la tabla

Figura 29: Procesos del sistema con 0% de error.

En la figura 30, se puede observar el uso del CPU.

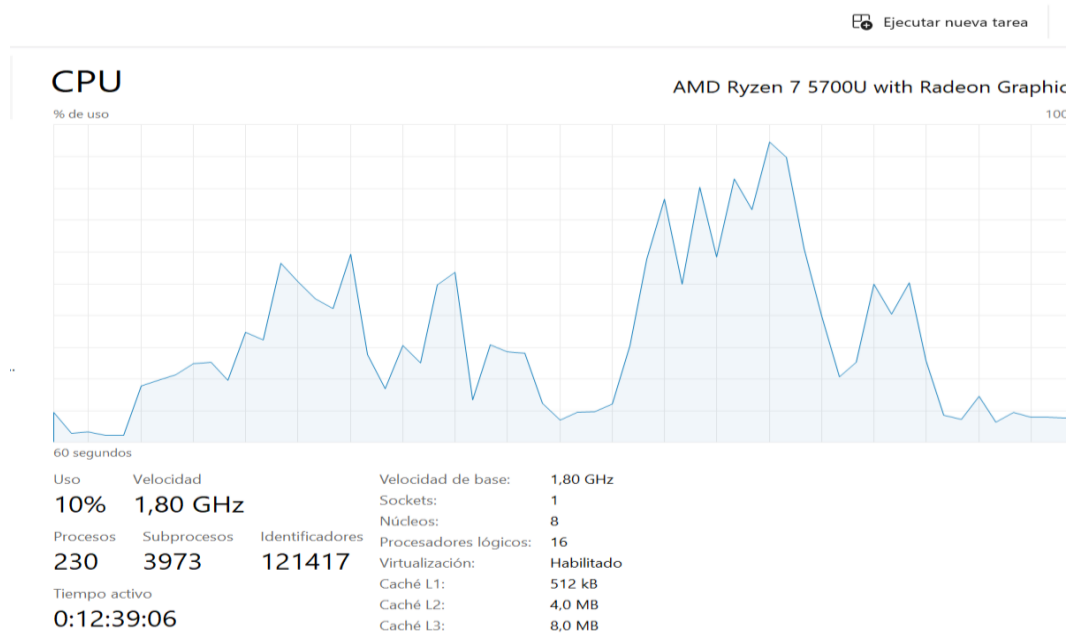


Figura 30: Porcentaje de uso del CPU.

En la figura 31, se puede observar el uso de la memoria.

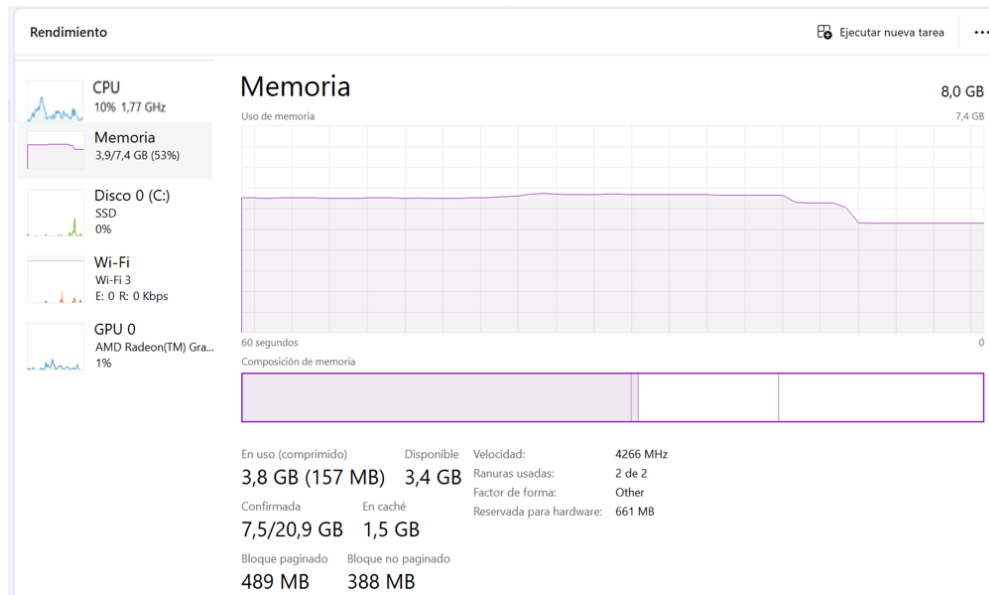


Figura 31: Porcentaje de uso de memoria.

En la figura 32, se puede observar el porcentaje del uso del Disco.

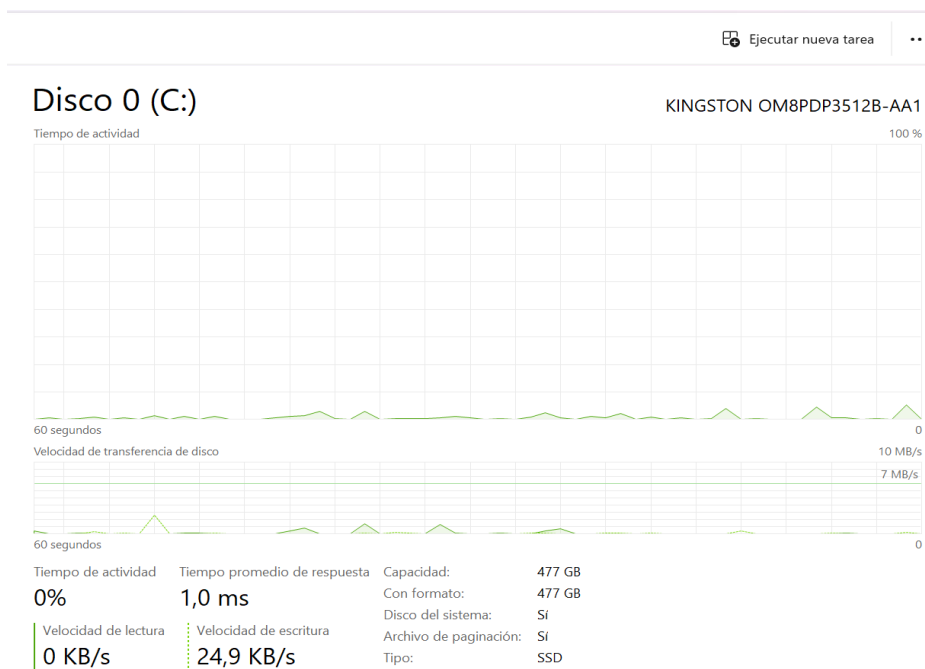


Figura 32: Porcentaje de uso de disco.

8. Anexo B: Manual de Usuario.



INTRODUCCIÓN.

El propósito del manual es ofrecer instrucciones acerca de cómo funciona el sistema de bienes e inventarios de la Universidad Nacional de Chimborazo. Permite registrar cada bien y constatarlo mediante el lector de código de barras, además poder ingresar nuevos activos y obtener un informe sobre ellos. A continuación, en la figura 1 se muestra el ingreso al Sistema donde está el Index de bienvenida.

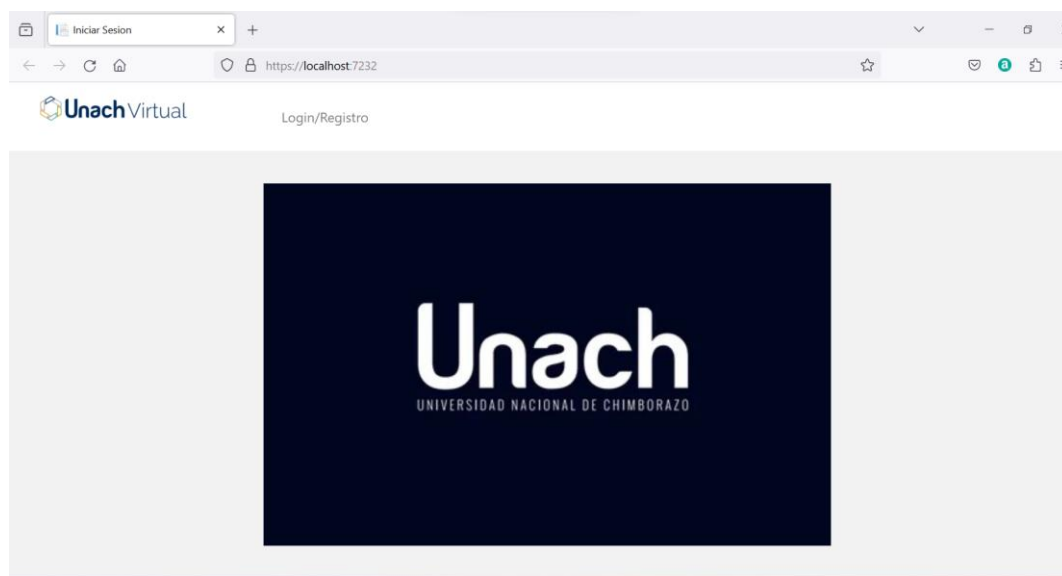


Figura 1: Página de inicio.

En la figura 2 se muestra el Inicio de sesión a través de la cuenta institucional de la Universidad Nacional de Chimborazo.

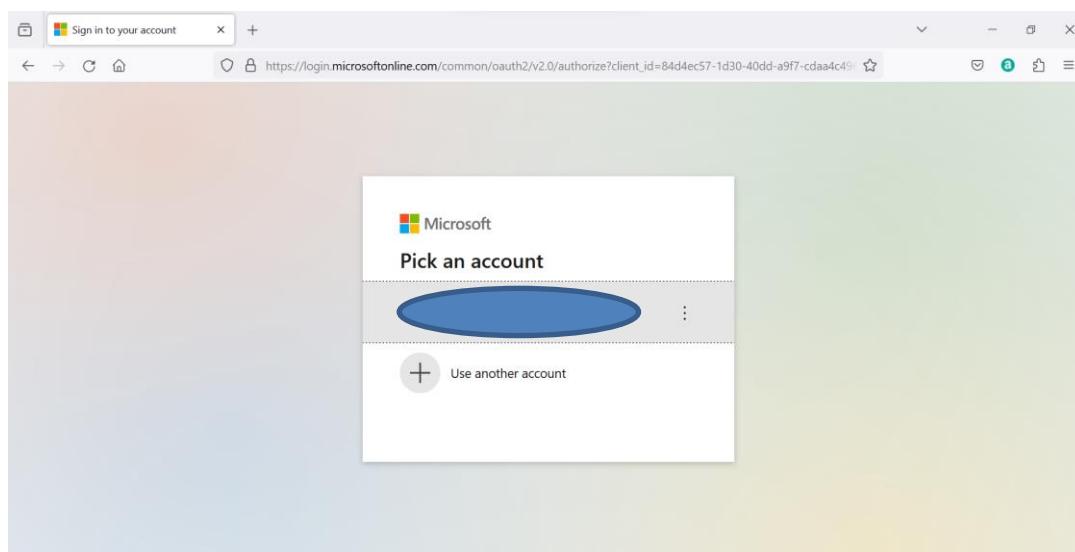


Figura 2: Login Office 365.

Una vez dentro del sistema se puede filtrar cada bien, es decir, se escoge la lista de marcas donde la siguiente será filtrada, es decir solo, existirán bienes que existe por la marca que se escogió. En la figura 3 se muestra el menú de búsqueda dinámica.

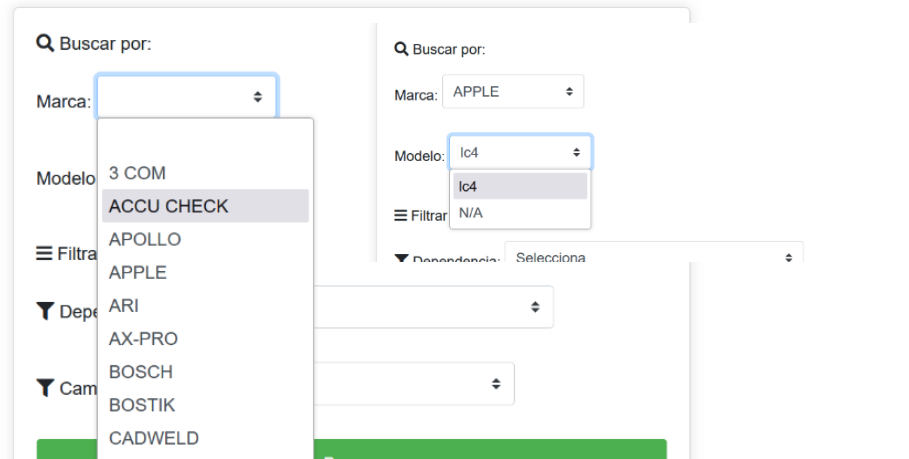


Figura 3: Menú de Búsqueda dinámica.

Al seleccionar un bien, se muestra una lista de opciones que permiten al usuario ver detalles como la descripción, características y estado del bien; gestionar el mantenimiento registrando nuevas tareas. A continuación, en la figura 4 se muestra la interfaz de búsqueda.

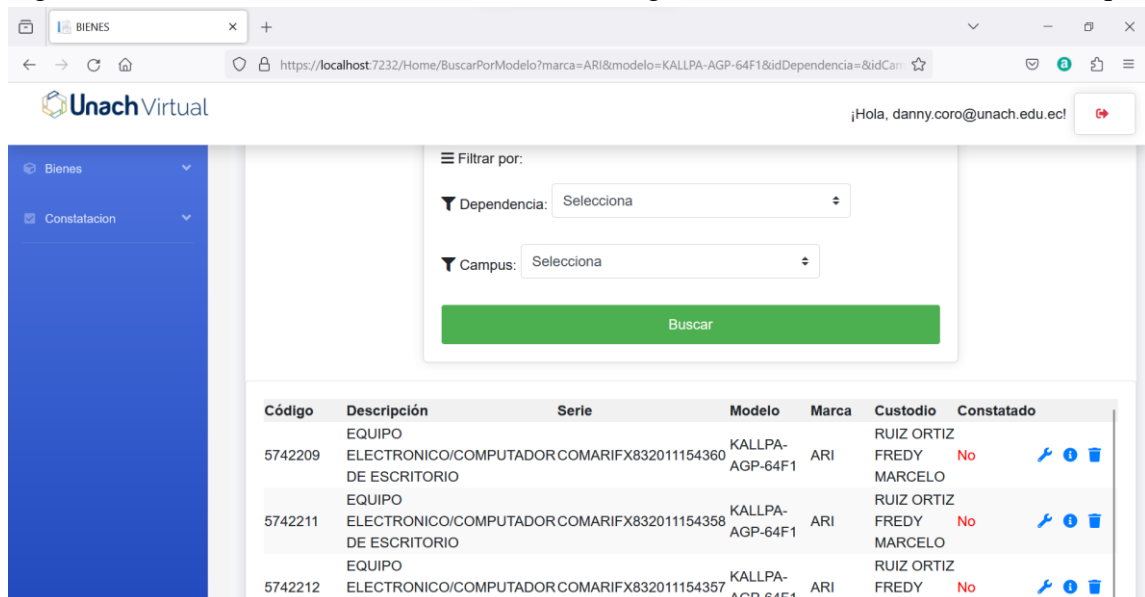


Figura 4: Interfaz de búsqueda y resultado.

Dentro del sistema, además, se cuenta con:

A: botón para agregar nuevo bien.

B: Buscador por código.

C: Pestañas desplegable para descargar los registros realizados.



Figura 5: Diferentes funcionalidades.

En el siguiente módulo se tiene el escáner de códigos de barras, el cual se puede utilizar por una aplicación externa instalada en el celular (BarcodePc), además, se puede generar los códigos de barras y poder descargarlos. A continuación, en la figura 6 se muestra el Panel de funcionalidades.

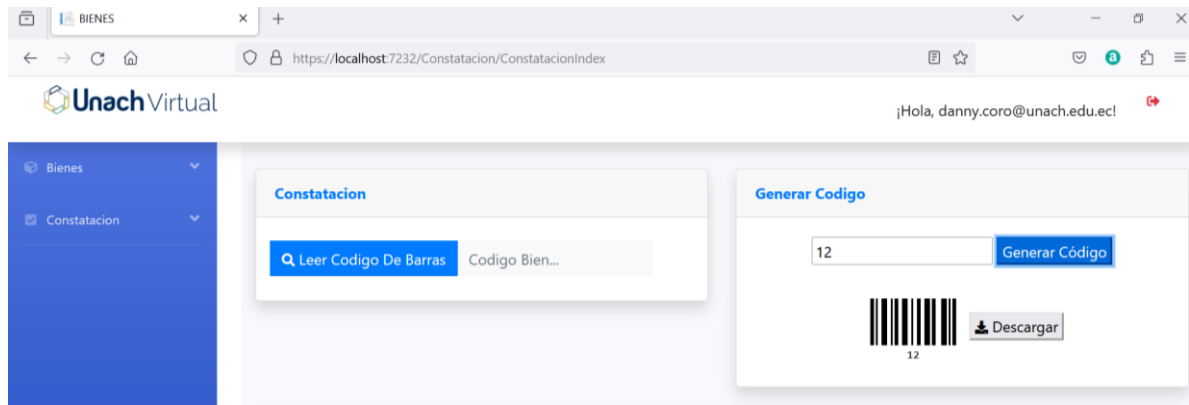


Figura 6: Panel de funcionalidades.

Generar Reportes.

Para obtener el reporte de Bienes que estén en el estado de Mantenimiento simplemente podemos dar clic en Reporte mantenimiento y se descargara en formato pdf. A continuación, en la figura 7 se muestra como descargar el reporte de mantenimiento.

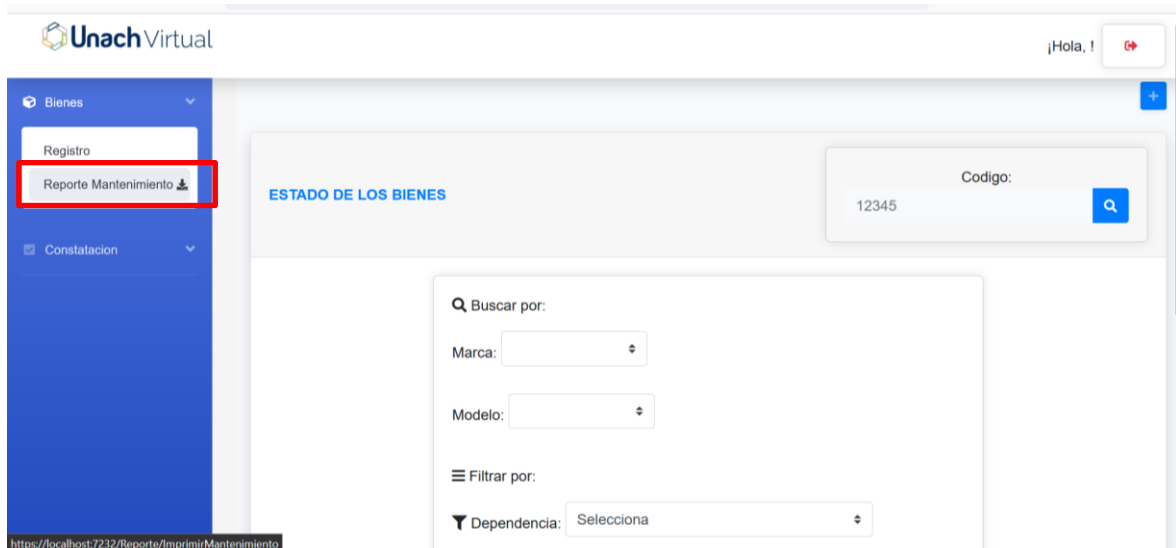


Figura 7: Obtener reportes.

Ejemplo del reporte de bienes en mantenimiento con los campos que se pueden visualizar en la figura siguiente. A continuación, en la figura 8 se muestra el reporte de mantenimiento.

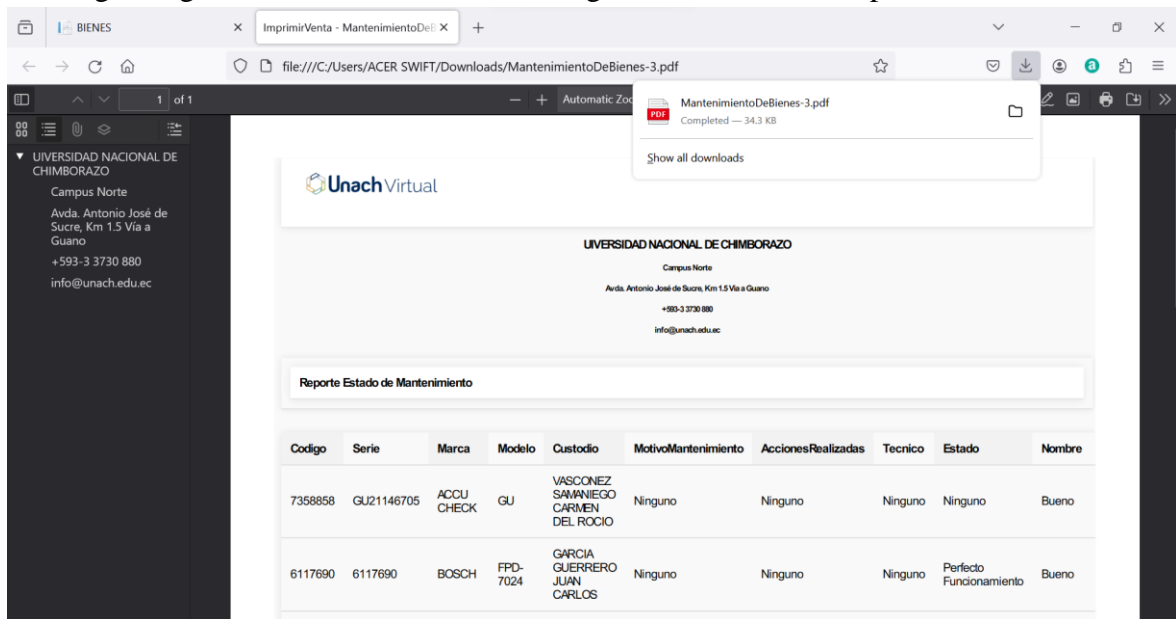


Figura 8: Ejemplo de Reporte.

Para obtener el reporte de bienes constatados se procede a dar clic en “reporte Constatados” donde se descargar en formato pdf. A continuación, en la figura 9 se muestra como descargar el reporte de constatación.

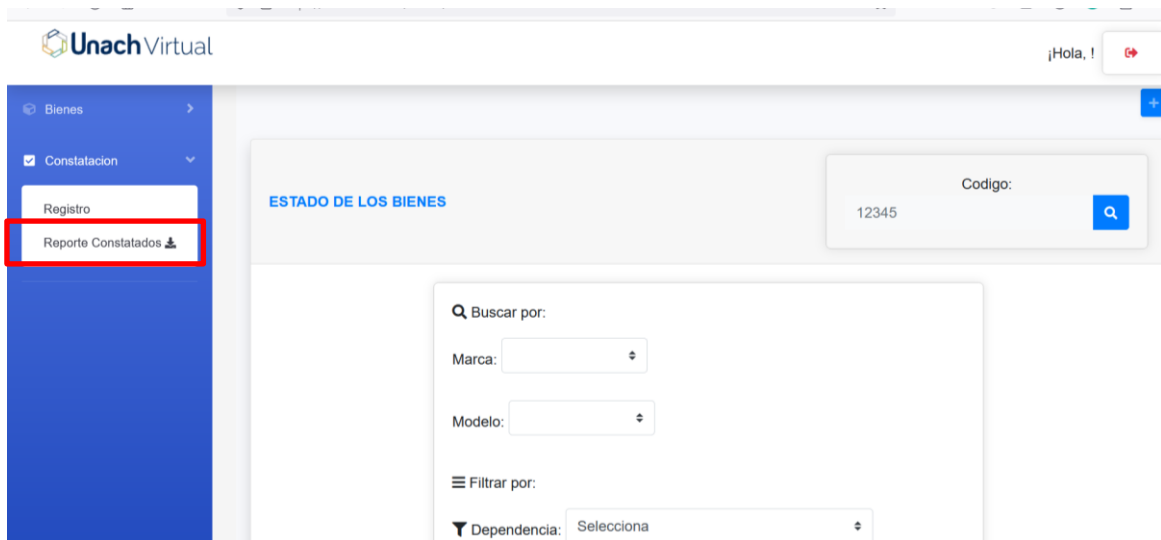


Figura 9: Obtener reporte de constatación.

A continuación, en la figura 10 se muestra el reporte de constatación.

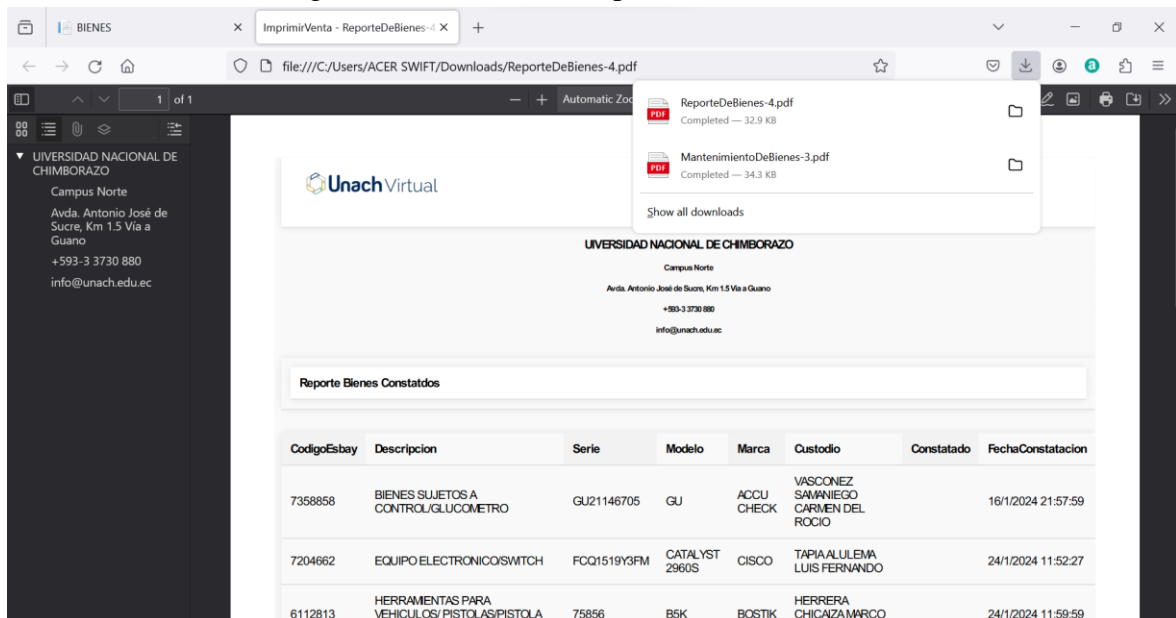


Figura 10: Ejemplo de reporte de constatación.

9. Anexo C: Acta de Entrega recepción.



ACTA DE ENTREGA, SATISFACCIÓN Y RECEPCIÓN DEFINITIVA

El día 01 de marzo del 2024 comparece: por parte de Coordinación de Desarrollo de Sistemas Informáticos Dirección de la Universidad Nacional de Chimborazo el Ing. Henry Javier Paca Quinaluiza, como Coordinador de Desarrollo de Sistemas Informáticos (CODESI) de la Universidad Nacional de Chimborazo y por parte de la carrera de Tecnologías de la Información la Ing. Pamela Alexandra Buñay Guisñan DOCENTE TUTOR DEL TEMA DE TESIS “Desarrollo del Sistema de bienes e inventarios de la Universidad Nacional de Chimborazo aplicando la plataforma .Net Core”, con la finalidad de realizar el Acta de entrega, satisfacción y recepción definitiva del sistema realizado por el estudiante Danny Francisco Coro Sayay. Según los detalles:

PRIMERA: ANTECEDENTES

Entre la Coordinación de Desarrollo de Sistemas Informáticos y la carrera de Tecnologías de la Información, las partes acuerdan desarrollar un sistema de bienes e inventarios para la Universidad Nacional de Chimborazo para el registro, constatación y gestión de bienes. Para el desarrollo del proyecto participan: Ing. Henry Javier Paca Quinaluiza Coordinador de Desarrollo de Sistemas Informáticos y en calidad de docente tutor la Ing. Pamela Alexandra Buñay Guisñan junto a el estudiante Danny Francisco Coro Sayay.

Mediante la resolución N.º 370-CITI-2023 de parte de la Universidad Nacional de Chimborazo se aprueba el tema de tesis “Desarrollo del Sistema de bienes e inventarios de la Universidad Nacional de Chimborazo aplicando la plataforma .Net Core”.

SEGUNDA: PRODUCTOS ENTREGADOS

La carrera de Tecnologías de la Información con relación al proyecto de tesis “Desarrollo del Sistema de bienes e inventarios de la Universidad Nacional de Chimborazo aplicando la plataforma .Net Core” entrega lo siguiente:

ITEMS	CANTIDAD	DETALLE	CONTENIDO
1	1	Aplicación Web	Contiene: <ul style="list-style-type: none">• Aplicación Web desarrollada en .Net Core 7
2	1	Manual de usuario	Contiene un manual técnico



TERCERA: RECEPCIÓN Y CONFORMIDAD

Previa a la suscripción de la presente Acta, el representante de la Coordinación de Desarrollo de Sistemas Informáticos de la Universidad Nacional de Chimborazo recibe a entera satisfacción los productos detallados en la segunda cláusula, sobre todo teniendo en cuenta la conformidad con los contenidos entregados dado el funcionamiento de cada uno de ellos y también siguiendo los parámetros de evaluación del producto principal (sistema de bienes e inventarios para la Universidad Nacional de Chimborazo).

CUARTA: CAPACITACION


La capacitación esta desarrollada en base a los temas, periodos de tiempo y participantes detallados a continuación:

N.º	TEMA	DETALLE	PARTICIPANTES	HORARIO
1	Socialización del funcionamiento del sistema de bienes e inventarios.	Se trata aspectos como: <ul style="list-style-type: none">• Introducción• Funcionamiento	Participan: <ul style="list-style-type: none">• Coordinador CODESI	Fecha: 26/01/2024 Hora: 15:00 pm

En constancia a lo anterior mencionado firman los presentes:



Ing. Henry Javier Paca Quinaluiza
Coordinador CODESI



Ing. Pamela Alexandra Borja Guisán
Docente Tutor



Danny Francisco Coro Sayay
Estudiante