



**FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE COMPUTACIÓN Y SISTEMAS**

**FRAMEWORK PARA LA CREACIÓN DE VIDEOJUEGOS
FORMATIVOS EN UNITY**

**PRESENTADA POR
CLAUDIO JEAN PAUL EUSEBIO GARAY**

**ASESOR
GENER VICTOR ZAMBRANO LOLI**

**TESIS
PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE
COMPUTACIÓN Y SISTEMAS**

**LIMA – PERÚ
2022**



CC BY-NC-ND

Reconocimiento – No comercial – Sin obra derivada

El autor sólo permite que se pueda descargar esta obra y compartirla con otras personas, siempre que se reconozca su autoría, pero no se puede cambiar de ninguna manera ni se puede utilizar comercialmente.

<http://creativecommons.org/licenses/by-nc-nd/4.0/>



USMP
UNIVERSIDAD DE
SAN MARTÍN DE PORRES

Facultad de
Ingeniería y
Arquitectura

**ESCUELA PROFESIONAL DE INGENIERÍA DE COMPUTACIÓN Y
SISTEMAS**

**FRAMEWORK PARA LA CREACIÓN DE VIDEOJUEGOS
FORMATIVOS EN UNITY**

TESIS

**PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE
COMPUTACIÓN Y SISTEMAS**

PRESENTADA POR

EUSEBIO GARAY, CLAUDIO JEAN PAUL

ASESOR:

MAG. GENER VICTOR ZAMBRANO LOLI

LIMA – PERÚ

2022

Este trabajo está dedicado a mis padres y hermana, gracias a ellos logro alcanzar mis objetivos y metas en la vida.

ÍNDICE

	Pág.
RESUMEN	xii
ABSTRACT	xiii
INTRODUCCIÓN	xv
CAPÍTULO I. PLANTEAMIENTO DEL PROBLEMA	
1.1. Situación Problemática	1
1.2. Definición del Problema	4
1.3. Formulación del Problema	4
1.4. Objetivo General y Específicos	5
1.5. Importancia y Aportes de la Investigación	5
1.6. Viabilidad de la Investigación	7
1.7. Alcances y Limitaciones del Estudio	8
CAPÍTULO II. MARCO TEÓRICO	
2.1. Antecedentes de la Investigación	10
2.2. Bases Teóricas	16
2.3. Definición de Términos Básicos	24
CAPÍTULO III. METODOLOGÍA	
3.1. Método	26

CAPÍTULO IV. DESARROLLO DE LA SOLUCIÓN	
4.1. Requerimientos	35
4.2. Análisis	44
4.3. Diseño	49
4.4. Desarrollo	64
4.5. Plan de Pruebas	70
4.6. Producto Mínimo Viable	79
CAPÍTULO V. RESULTADOS	
5.1. Respecto al Primer Objetivo Específico	88
5.2. Respecto al Segundo Objetivo Específico	90
5.3. Respecto al Tercer Objetivo Específico	91
5.4. Respecto al Cuarto Objetivo Específico	93
CAPÍTULO VI DISCUSIÓN	95
CONCLUSIONES	98
RECOMENDACIONES	100
FUENTES DE INFORMACIÓN	101

ÍNDICE DE FIGURAS

	Pág.
Figura 1 Resumen de la Técnica	11
Figura 2 Suggested Video Games Developers Framework, Ahmad et al.	13
Figura 3 EDT - Framework para la Creación de Videojuegos Formativos en Unity	29
Figura 4 Diagrama de Paquetes del Framework SGOP	45
Figura 5 Diagrama del Paquete SGOP	45
Figura 6 Diagrama del Paquete UI	46
Figura 7 Diagrama del Paquete Base	46
Figura 8 Diagrama del Paquete Manager	47
Figura 9 Diagrama del Paquete Model	48
Figura 10 Diagrama del Paquete Util	49
Figura 11 Diagrama de Clases del Paquete SGOP	50
Figura 12 Diagrama de Clases del Paquete Manager	51
Figura 13 Diagrama de Clases del Paquete Base	52

Figura 14 Diagrama de Clases del Paquete Model	53
Figura 15 Diagrama de Clases del Paquete Util	54
Figura 16 Diagrama de Inicio de Sesión	55
Figura 17 Diagrama de Visualización de Videos de Lección por Franquicia	55
Figura 18 Diagrama de Juego de Carta Memoria	56
Figura 19 Diagrama de Stickerbook RA	56
Figura 20 Diagrama desde Inicia la Aplicación hasta Abrir un Juego	57
Figura 21 Diagrama del Proceso “BuildView” del Framework	58
Figura 22 Diagrama del Proceso “Downloadcontent” del Framework	58
Figura 23 Propuesta de Modelo de Arquitectura para Videojuegos Formativos	59
Figura 24 Propuesta de Modelo de Desarrollo de Software de Videojuegos Formativos Utilizando el Framework SGOP	60
Figura 25 Propuesta de Modelo de Desarrollo de Proyectos de Videojuegos Formativos Utilizando el Framework SGOP	61
Figura 26 Arquitectura de la Aplicación de la Empresa	64
Figura 27 Captura del Método “DownloadTexture” de la Clase “DownloadManager”	65
Figura 28 Captura del Método “OpenView” de la Clase “ViewManager”	66
Figura 29 Captura de los Métodos “SetTextItem” y “SetImageItem” de la Clase “UserInterface”	67
Figura 30 Página de Login en la Herramienta Unity	68
Figura 31 Página de Minijuego Carta Memoria en la Herramienta Unity	69

Figura 32 Desarrollo de Stickerbook RA en la Herramienta Unity	70
Figura 33 Captura de Pantalla de la Vista Login en el Celular	80
Figura 34 Captura de Pantalla de la Vista TyC en el Celular	80
Figura 35 Captura de Pantalla de la Vista “Confirmación de Edad” en el Celular	81
Figura 36 Captura de Pantalla de la Vista Home en el Celular	82
Figura 37 Captura de Pantalla de la Vista Franquicia - Videos en el Celular	82
Figura 38 Captura de Pantalla de la Vista Reproductor de Video en el Celular	83
Figura 39 Captura de Pantalla de la Vista Franquicia - Juegos en el Celular	84
Figura 40 Captura de Pantalla de la Vista Minijuego Carta Memoria en el Celular	85
Figura 41 Captura de Pantalla de la Obtención de un Nuevo Sticker en el Celular	85
Figura 42 Captura de Pantalla de la Vista Stickerbook en el Celular	86
Figura 43 Captura de Pantalla de un Sticker Maximizado	87
Figura 44 Captura de Pantalla de la Vista Sticker RA en el Celular	87
Figura 45 Diagrama de Clases del Framework SGOP	90
Figura 46 Captura de Pantalla de la Página de GitHub del Proyecto SGOP-Framework	92
Figura 47 Captura de Pantalla de Google Play - Magical English	94

ÍNDICE DE TABLAS

	Pág.
Tabla 1 <i>Proyectos en Desarrollo de la Empresa</i>	2
Tabla 2 <i>Tiempos de Desarrollo de Cada Producto de la Empresa</i>	3
Tabla 3 <i>Recursos Técnicos</i>	7
Tabla 4 <i>Presupuesto del Proyecto</i>	8
Tabla 5 <i>Evaluación Comparativa de Artículos</i>	15
Tabla 6 <i>Criterios de Evaluación de Metodología</i>	28
Tabla 7 <i>Cronograma de Actividades</i>	30
Tabla 8 <i>Asignación de Roles Scrum</i>	31
Tabla 9 <i>Listado de Épicas</i>	31
Tabla 10 <i>Lista Priorizada de Pendientes del Producto</i>	32
Tabla 11 <i>Lista de Pendientes del Sprint 1</i>	33
Tabla 12 <i>Lista de Pendientes del Sprint 2</i>	33
Tabla 13 <i>Lista de Pendientes del Sprint 3</i>	34

Tabla 14 <i>Lista de Pendientes del Sprint 4</i>	34
Tabla 15 <i>Listado de Requerimientos Funcionales</i>	36
Tabla 16 <i>Listado de Puntos Funcionales</i>	37
Tabla 17 <i>Listado de Requerimientos No Funcionales</i>	38
Tabla 18 <i>HU-01 Controlador de Descarga de Contenido</i>	38
Tabla 19 <i>HU-02 Controlador de Corrutina</i>	39
Tabla 20 <i>HU-03 Controlador de Vista</i>	40
Tabla 21 <i>HU-04 Controlador de Audio</i>	40
Tabla 22 <i>HU-05 Controlador de Interfaz de Usuario</i>	41
Tabla 23 <i>HU-06 Ingreso al Sistema</i>	41
Tabla 24 <i>HU-07 Reproducir Videos de Franquicias</i>	42
Tabla 25 <i>HU-08 Ejecutar el Minijuego de Carta Memoria</i>	42
Tabla 26 <i>HU-09 Obtener Estrellas</i>	43
Tabla 27 <i>HU-09 Obtener Sticker</i>	43
Tabla 28 <i>HU-11 Visualización de Sticker RA</i>	44
Tabla 29 <i>Trazabilidad Puntos Funcionales/Historias de Usuarios</i>	44
Tabla 30 <i>CP-01 Realizar Ingreso al Sistema</i>	71
Tabla 31 <i>CP-02 Ver Video de Lección</i>	72
Tabla 32 <i>CP-03 Realizar Minijuego Carta Memoria</i>	73
Tabla 33 <i>Cp-04 Ver Stickerbook RA</i>	74
Tabla 34 <i>Ejecución de CP-01 Realizar Ingreso al Sistema</i>	75

Tabla 35 <i>Ejecución de CP-02 Ver Video de Lección</i>	76
Tabla 36 <i>Ejecución de CP-03 Realizar Minijuego Carta Memoria</i>	77
Tabla 37 <i>Ejecución de CP-04 Ver Stickerbook RA</i>	78

RESUMEN

Ante la búsqueda de una solución para optimizar el método y las herramientas para la creación de videojuegos formativos con el fin de disminuir el tiempo de desarrollo, se planteó el objetivo de desarrollar un framework para la construcción de videojuegos formativos basados en páginas en Unity,

Para lograr el objetivo se identificó el código fuente reutilizable de proyectos anteriores, se diseñó la metodología que implementó el framework, se desarrolló el código fuente reutilizable optimizando su funcionamiento y se implementó en un proyecto de desarrollo de producto mínimo viable.

El producto de la ejecución de estas actividades fue: 1) un modelo de marco de trabajo con el enfoque que requieren los videojuegos formativos, denominado, “Modelo de Desarrollo de Proyectos de Videojuegos Formativos Utilizando el Framework SGOP”; y, 2) la publicación del código fuente del framework en un repositorio público de GitHub.

Como principal resultado de la implementación del framework se tuvo que es capaz de reducir hasta en un 75.00 % el tiempo de desarrollo previsto del front end para un producto mínimo viable.

Palabras clave: framework, unity, videojuegos formativos, metodología.

ABSTRACT

In the search for a solution to optimize the method and tools for the creation of training games to reduce development time, the objective was to develop a framework for the construction of training video games based on pages in Unity, identifying the needs and optimizing the work of developers and the project.

To achieve the objective, the reusable source code from previous projects was identified; the methodology that implemented the framework was designed; the reusable source code was developed, optimizing its operation, and it was implemented in a minimum viable product development project.

The product of carrying out these activities was: 1) a framework model with the approach required by training videogames, called "Training Video Game Project Development Model Using the SGOP Framework", and 2) the publication of the source code of the framework in a public repository of GitHub.

The main result of the framework implementation was that it can reduce up to 75.00% of the planned development time of the front end for a minimum viable product.

Keywords: framework, unity, serious video games, methodology.

NOMBRE DEL TRABAJO

FRAMEWORK PARA LA CREACIÓN DE VIDEOJUEGOS FORMATIVOS EN UNITY

AUTOR

CLAUDIO JEAN PAUL EUSEBIO GARAY

RECuento DE PALABRAS

16068 Words

RECuento DE CARACTERES

85712 Characters

RECuento DE PÁGINAS

119 Pages

TAMAÑO DEL ARCHIVO

6.3MB

FECHA DE ENTREGA

Aug 10, 2023 1:36 PM GMT-5

FECHA DEL INFORME

Aug 10, 2023 1:37 PM GMT-5**● 7% de similitud general**

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para cada base de datos:

- 6% Base de datos de Internet
- Base de datos de Crossref
- 5% Base de datos de trabajos entregados
- 1% Base de datos de publicaciones
- Base de datos de contenido publicado de Crossref

● Excluir del Reporte de Similitud

- Material bibliográfico
- Material citado
- Coincidencia baja (menos de 10 palabras)



INTRODUCCIÓN

Los frameworks son conjuntos de herramientas que facilitan la creación de nuevas aplicaciones. Esta investigación tiene como propósito optimizar los métodos y mejorar las herramientas para la creación de productos mínimos viables de nuevas aplicaciones a fin de disminuir el tiempo de elaboración.

Del análisis de la situación de desarrollo de nuevos proyectos en la empresa se obtuvo como problemática la necesidad de optimizar los métodos o herramientas para la creación de videojuegos formativos, ya que se requerían realizar múltiples productos mínimos viables.

El objetivo planteado fue desarrollar un framework para el desarrollo de videojuegos formativos en Unity, el cual incluye la metodología y el código fuente. Esto permite disminuir el tiempo de desarrollo, reducir los costes de producción y los problemas relacionados con errores de programación.

A fin de cumplir este objetivo se procedió a identificar el código fuente reutilizable de proyectos anteriores a fin de optimizarlos y expandirlos para reducir la carga de trabajo, también se diseñó la metodología para implementarse con el framework y el código fuente que ayude a mejorar el desarrollo. Por último, se validó su rendimiento implementándolo en una aplicación de producto mínimo viable.

El producto de la investigación ha sido el Framework SGOP, el cual disminuye el tiempo de desarrollo al generalizar herramientas de uso

constante durante el desarrollo de aplicaciones en su fase de producto mínimo viable. También se modificó la metodología planteada por Ahmad et al., creando un método en el cual se implementa de manera eficiente el Framework SGOP.

La metodología elegida para el desarrollo de la investigación fue SCRUM, el método ágil fue muy importante debido a que durante el proceso de desarrollo del código fuente del framework se encontraron mejoras que no fueron detectadas en las fases de análisis y diseño, estas significaron un aporte significativo al Framework SGOP.

La principal limitante encontrada en la investigación fue el tiempo, por lo cual solo se fue capaz de desarrollar un framework para la parte del front end de las aplicaciones. Adicionalmente, con respecto a la validación de las funcionalidades del framework, se desarrolló un producto mínimo viable de una aplicación, el cual no contempló la creación de contenido multimedia, todo este contenido fue proporcionado por la empresa.

El desarrollo de este proyecto buscó incentivar el campo de desarrollo de videojuegos educativos, al reducir la necesidad de comprometer horas de desarrollo a componentes que puedan haberse elaborado con anterioridad, para que así, se pueda centrar todo el trabajo en el desarrollo del núcleo de las aplicaciones.

Esta investigación comprende seis capítulos, el primer capítulo consta de la definición del problema, la definición de los objetivos y la importancia de la investigación. En el segundo capítulo se hace una revisión de los antecedentes de la investigación y las bases teóricas para el desarrollo de esta tesis. En el tercer capítulo se determina la metodología y el cómo se desarrollará la metodología. En el cuarto capítulo se presenta el desarrollo de los objetivos planteados. En el quinto capítulo se detallan los resultados obtenidos del desarrollo del capítulo previo. Por último, en el sexto capítulo se presenta la discusión sobre los resultados obtenidos.

CAPÍTULO I

PLANTEAMIENTO DEL PROBLEMA

1.1. Situación Problemática

En la actualidad la educación es un tema muy importante alrededor del mundo, es necesario cultivar nuevos conocimientos en los niños a fin de ofrecerles mejores oportunidades hacia su futuro. Se tiene como institución principal al sistema educativo, pero este ha demostrado tener problemas con la capacidad para satisfacer la demanda de educación, una no equitativa distribución de las oportunidades para acceder a la educación y los problemas conllevados de la administración por parte del Estado (Muñoz Izquierdo, 2012, pág. 154).

En la misma línea, Jopen, Gómez, & Olivera (2014, pág. 41) mencionan que dentro de las prácticas pedagógicas se debería aprovechar el uso de materiales educativos, no obstante, aunque hoy hay mejor proyección con respecto al acceso a las TIC, esto aún es un problema en el ambiente educativo.

Por otra parte, desde el inicio de la pandemia de COVID-19, se ha visto un incremento del 54 % en el uso de aplicaciones para la enseñanza o aplicaciones educativas, esto debido al incremento en

gran medida de la actividad digital como efecto de la educación desde casa, esto es un gran indicativo de que hay un sector de la población que está interesada en aprender mediante aplicaciones móviles o web (Qustodio, 2021, pág. 30).

Cuervo Gómez & Ballesteros-Ricaurte (2017, pág. 151), en sus conclusiones mencionan que la creación de herramientas digitales para el aprendizaje es un trabajo complejo y de ardua labor que conlleva la necesidad de herramientas de fácil desarrollo.

En este sentido, la empresa La Mark Vision Perú S.A.C., con el objetivo de hacer que el aprendizaje sea accesible para la mayor cantidad de personas, viene trabajando con diferentes tecnologías para el desarrollo de aplicaciones web y móviles que contengan juegos.

Los proyectos de la empresa son los siguientes:

Tabla 1

Proyectos en Desarrollo de la Empresa

Proyecto	Tecnología	Lenguaje de programación
<i>Magical English</i>	Unity	C#
<i>Magical English Web</i>	Angular	TypeScript
<i>Adventure</i>	Flutter	Dart
<i>Hollywood</i>	Flutter	Dart

Elaborado por: el autor

Al tipo de proyectos en los que trabaja la empresa se les puede denominar aplicaciones educativas o videojuegos formativos, ambos en la categoría de aplicaciones basadas en páginas.

El tiempo de desarrollo de cada uno de los proyectos mencionados se conforman de tres partes: primero, el tiempo en la generación del producto mínimo viable; segundo, el tiempo de desarrollo de producto completo; tercero, el tiempo de mantenimiento. La primera duración de tiempo hace referencia al desarrollo de los módulos principales de la

aplicación de tal manera que sea posible su visualización por la parte administrativa a fin de dar la aprobación para continuar con el proyecto y/o añadir algún cambio de ser necesario. La segunda duración de tiempo se refiere al desarrollo de todos los componentes que contiene el proyecto, de tal manera que pueda ser utilizado por el usuario final. El tercero es el tiempo desde que se terminó el desarrollo del producto y hasta la actualidad, tiempo en el cual se han corregido problemas y/o añadido cambios solicitados por los clientes, así como también la adición de nuevo contenido multimedia. Estos tiempos se detallan a continuación para cada producto:

Tabla 2

Tiempos de Desarrollo de Cada Producto de la Empresa

Proyecto	Tiempo de PMV	Tiempo de Desarrollo	Tiempo de Mantenimiento
<i>Magical English</i>	4 meses	8 meses	6 meses
<i>Magical English Web</i>	6 meses	12 meses	6 meses
<i>Adventure</i>	6 meses	12 meses	12 meses
<i>Hollywood</i>	4 meses	10 meses	0 meses

Elaborado por: el autor

Al momento de desarrollar una nueva aplicación con la característica multiplataforma la empresa ha decidido trabajar con dos tecnologías, Flutter y Unity. Estas tecnologías tienen diferentes propósitos, el primero está enfocado en el desarrollo de interfaces de usuario para aplicaciones Android, iOS y web, mientras que el segundo es principalmente una herramienta para desarrollar videojuegos. Los motivos para desarrollar aplicaciones con tecnologías con enfoques diferentes han sido por las características de los proyectos, el proyecto de Adventure está enfocado en mostrar información o contenido multimedia en una interfaz sencilla, por otro lado, Magical English es un proyecto enfocado en los minijuegos.

Los problemas que se han presentado en el desarrollo de los proyectos en Flutter son principalmente el manejo de contenido localizado, el alto peso del contenido a descargar y que no está diseñado para crear videojuegos. Por su parte, Magical English en Unity, los

problemas son principalmente que la herramienta no está diseñada para la creación de aplicaciones basadas en páginas, si bien se puede desarrollar interfaces de usuario, el manejo de páginas y contenido multimedia bajo demanda son componentes que no están incluidos dentro de las herramientas que proporciona, así como también el manejo de contenido localizado y el alto peso del contenido a descargar. Lo que si proporciona la Herramienta Unity es la facilidad con la que se pueden crear videojuegos.

Además de estos problemas, también se han presentado en todos los proyectos el desarrollo reiterativo de código fuente que proporcionen de manera sencilla métodos para la creación de los componentes más complejos o que contienen lógicas que no son proporcionadas por el framework o herramienta de desarrollo.

Por otro lado, la empresa se encuentra en la proyección de empezar una mayor cantidad de productos que contengan partes de videojuegos, a su vez también requieren simplificar y disminuir el tiempo del desarrollo principalmente de los productos mínimos viables.

1.2. Definición del Problema

Luego de realizar un análisis de la situación del desarrollo de proyectos de la empresa y con el requerimiento de la empresa por disminuir el tiempo de desarrollo de los productos mínimos viables, se ha determinado que existe la necesidad de optimizar el desarrollo de videojuegos formativos. Para lo cual se ha decidido por continuar con la herramienta Unity, la cual proporciona las herramientas necesarias para videojuegos.

1.3. Formulación del Problema

1.3.1. Pregunta Principal

¿Cómo optimizar el método y las herramientas para la creación de videojuegos formativos con el fin de disminuir el tiempo de desarrollo?

1.3.2. Preguntas Secundarias

- a) ¿Qué código fuente requieren de mayor tiempo y son reiterativos en el desarrollo de videojuegos formativos?
- b) ¿Cómo mejorar el método de desarrollo de videojuegos formativos?
- c) ¿De qué manera el código fuente reiterativo puede formar parte de un mismo proyecto de manera que sean fáciles de utilizar y reutilizables en múltiples proyectos?
- d) ¿Cómo corroborar si las herramientas creadas permiten optimizar el desarrollo de videojuegos formativos?

1.4. Objetivo General y Específicos

1.4.1. Objetivo General

Desarrollar un framework para la construcción de videojuegos formativos en Unity, identificando las necesidades y optimizando el trabajo de los desarrolladores y del proyecto.

1.4.2. Objetivos Específicos

- a) Identificar el código fuente reutilizable con el fin de optimizarlos e implementarlos en futuros proyectos de videojuegos formativos.
- b) Diseñar la metodología que implementará el framework para el desarrollo para videojuegos formativos.
- c) Desarrollar el código fuente reutilizable optimizando su funcionamiento y encapsulándolos en un mismo proyecto.
- d) Validar que el framework creado cumple con las necesidades y optimiza el tiempo de desarrollo de un videojuego formativo.

1.5. Importancia y Aportes de la Investigación

1.5.1. Importancia

La presente investigación está enfocada en ayudar a optimizar el desarrollo de nuevos videojuegos formativos para la empresa La Mark Vision Perú SAC. También podrá ser usado como base para

proyectos o investigaciones con respecto al desarrollo de videojuegos con capacidades educativas.

Por otro lado, el framework que se plantea desarrollar permitirá disminuir el tiempo de desarrollo al generalizar herramientas de uso constante durante el desarrollo de una aplicación, con esta reducción de tiempo también se traducirá a una reducción de costos de nuevos desarrollos.

Finalmente, se creará un framework que podrá ser mantenido y actualizado para cumplir con nuevas necesidades del desarrollo de aplicaciones, el cual incluirá un modelo metodológico que servirá de guía para desarrollar de manera eficiente nuevos videojuegos formativos.

1.5.2. Aportes

Se creó un framework que permitirá aumentar la eficiencia y simplificar el desarrollo de videojuegos formativos, lo cual permitirá reducir el tiempo y costos de desarrollo de nuevos proyectos para la empresa La Mark Vision Perú SAC.

El proyecto plantea el uso de componentes simplificados y reutilizables en el desarrollo de videojuegos formativos, con el fin de llevar a cabo un análisis de las partes más importantes de los proyectos de manera que se cubra horizontalmente el desarrollo de nuevas aplicaciones, así como también mejorando el rendimiento de las aplicaciones al especificar y contener de manera eficientes el código fuente que cumplan con funciones complejas y reiterativas.

Por último, el desarrollo de este proyecto pretende incentivar aún más el campo de desarrollo de videojuegos y de videojuegos educativos, al reducir la necesidad de comprometer horas al desarrollo de componentes reutilizables para que se pueda centrar en el desarrollo del núcleo de las aplicaciones.

1.6. Viabilidad de la Investigación

Para el desarrollo del proyecto se cuenta con la aprobación de la empresa La Mark Vision Perú SAC la cual está interesada en la implementación del framework, para lo cual proporcionará las licencias de softwares, arquitectura y contenido necesarias para el funcionamiento de la aplicación final.

Tomando esto en cuenta estas consideraciones se proceden a mencionar los siguientes detalles de viabilidad para el proyecto.

1.6.1. Viabilidad Técnica

Para el desarrollo del modelo estructural se contempla un especialista en desarrollo de aplicaciones en Unity (el cual será llamado “desarrollador” en adelante), en los aspectos tecnológicos se utilizarán equipos, servicios y licencias de software con lo que ya se cuenta, los cuales se describen en la **Tabla 3**.

Tabla 3

Recursos Técnicos

Tipo	Recurso	Descripción
Hardware	PC	Intel Core i7, 32GB Ram, 1TB almacenamiento
	Smartphone	Samsung S22
Software	Windows 10	Sistema Operativo
	Unity	Herramienta de desarrollo de videojuegos
	Visual Studio Code	Editor de código fuente
	AWS	Servicio Nube
	Azure	Servicio Nube

Elaborado por: el autor

1.6.2. Viabilidad Operativa

Para la operación de la aplicación final entregada a la empresa se dispondrá del compilado que será desplegado en

la tienda de aplicaciones Play Store para Android mediante la cual llegará a los usuarios finales.

1.6.3. Viabilidad Ética

A nivel ético con respecto a la investigación se citará todas las fuentes consultadas para el desarrollo de esta.

1.6.4. Viabilidad Legal

En los aspectos legales, se mantendrá en reserva toda información de los usuarios finales de la aplicación de acuerdo con la Ley N.º 29733 Ley de Protección de Datos.

1.6.5. Viabilidad Económica

Los servicios de nube, licencias, contenido multimedia y hardware para el desarrollo del proyecto serán provistos por la empresa La Mark Vision Perú S.A.C. En lo que respecta al presupuesto previsto para el desarrollo del proyecto solo se tomará en consideración el pago para el desarrollador, el cual se detalla en la Tabla 4. La financiación del proyecto será por parte de la empresa.

Tabla 4

Presupuesto del Proyecto

Ítem	Unidad	Cantidad	Costo Para El Negocio	Total Para el Negocio	Costo Unitario	Costo Total
Desarrollador	Mes	5	S/ 3 500.00	S/ 17 500.00	S/ 3 500.00	S/ 17 500.00
Unity Pro	Licencia	1	S/ -	S/ -	S/ 6 930.00	S/ 6 930.00
Google Developer account	Licencia	1	S/ -	S/ -	S/ 96.25	S/ 96.25
Total				S/ 17 500.00		S/ 24 526.25

Elaborado por: el autor

1.7. Alcances y Limitaciones del Estudio

1.7.1. Alcance

- a) Desarrollo de las siguientes herramientas que incluye el framework:
 - Controlador de Descarga de Contenidos,

- Controlador de Corrutinas,
 - Controlador de Vistas,
 - Controlador de Audios, y
 - Controlador de Interfaces de Usuario.
- b) Implementación del framework en el desarrollo de la aplicación Magical English (en adelante “aplicación de la empresa”) de la empresa La Mark Vision Perú S.A.C.
- c) El desarrollo de la aplicación para la empresa solo contempla las siguientes secciones: Login, Home, Franquicias, Reproductor de videos, Juegos y Stickerbook.
- d) El desarrollo de la aplicación de la empresa comprende solo el desarrollo de un producto mínimo viable.
- e) El framework esta creado solo para el desarrollo de videojuegos formativos basados en páginas.

1.7.2. Limitaciones

- Debido a la limitante del tiempo solo se contempla el desarrollo del framework para la creación del front end.
- El proyecto no contempla la creación de nuevo contenido multimedia para el desarrollo de la aplicación de la empresa y se utilizará el contenido multimedia proporcionado por la empresa.
- Todos los diseños de interfaces de usuario serán proporcionados por la empresa.

CAPÍTULO II

MARCO TEÓRICO

2.1. Antecedentes de la Investigación

2.1.1. Artículos Nacionales

Sisa Yataco (2020) en su tesis de pregrado: “Técnica para validar el diseño de software durante la elicitación de requisitos empleando especificaciones de usabilidad”, ante la problemática presentada:

Los desarrolladores de software, para validar el resultado de su codificación muchas veces tienen que esperar a la construcción total del producto de software para saber si lograron entender el modelo de negocio” se plantea en la tesis el objetivo de “Proponer una técnica que permita validar el diseño del software cuyas necesidades se encuentran definidas en el catálogo de requisitos y cuya estructura acepta las especificaciones de usabilidad (pág. 8)

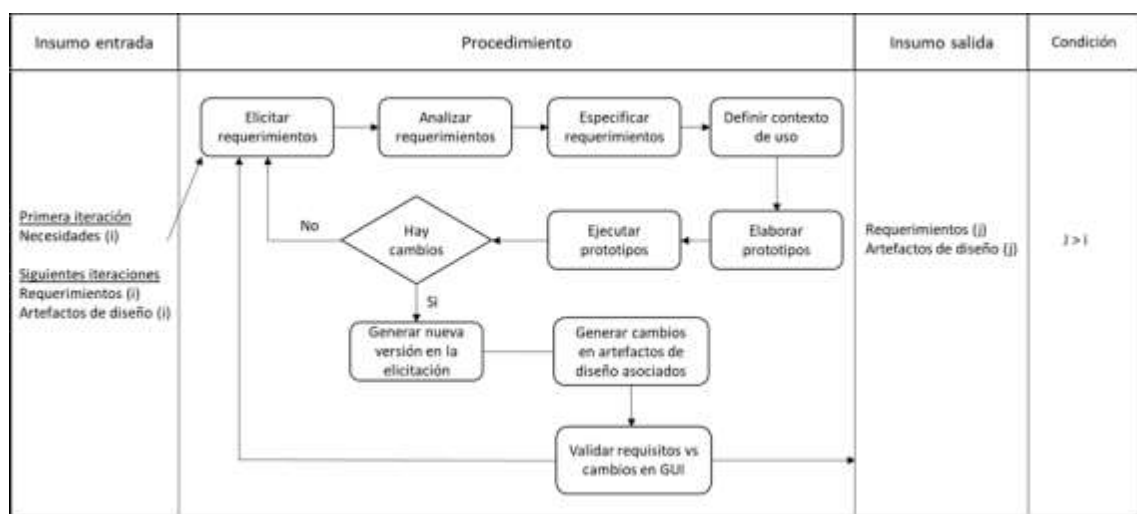
Para cumplir con este objetivo Sisa Yataco desarrolló una técnica de validación del diseño de software durante la elicitación de requisitos empleando especificaciones de usabilidad e

independiente del proceso de construcción del producto de software, la cual se puede ver en la **Figura 1**.

En sus conclusiones mencionó que la “estructura propuesta para la elicitación de requisitos mejora la toma de requerimientos y la documentación de construcción del producto desde los criterios de usabilidad del usuario final. El catálogo de requisitos mejora su calidad y contempla casos del usuario final” (Sisa Yataco, 2020, pág. 88).

Figura 1

Resumen de la Técnica



Nota. La figura muestra el resumen de la técnica utilizado para validación del diseño de software durante la elicitación de requisitos.

Fuente: Sisa Yataco, 2020, pág. 52

En su tesis Preciado Cossío & Silva Paucar (2018) realizaron una investigación para determinar “Buenas Prácticas Para El Desarrollo De Videojuegos Educativos Aplicados A Historia Y Arqueología Usando Realidad Aumentada Para La Educación Primaria En El Perú”; para lo cual realizan un análisis de casos de desarrollo de videojuegos educativos, con lo cual elaboran un documento de buenas prácticas y una metodología para el desarrollo de videojuegos. Entre sus conclusiones mencionan que se logró reducir el tiempo de desarrollo en un 42.0 % utilizando la metodología desarrollada (2018, pág. 99).

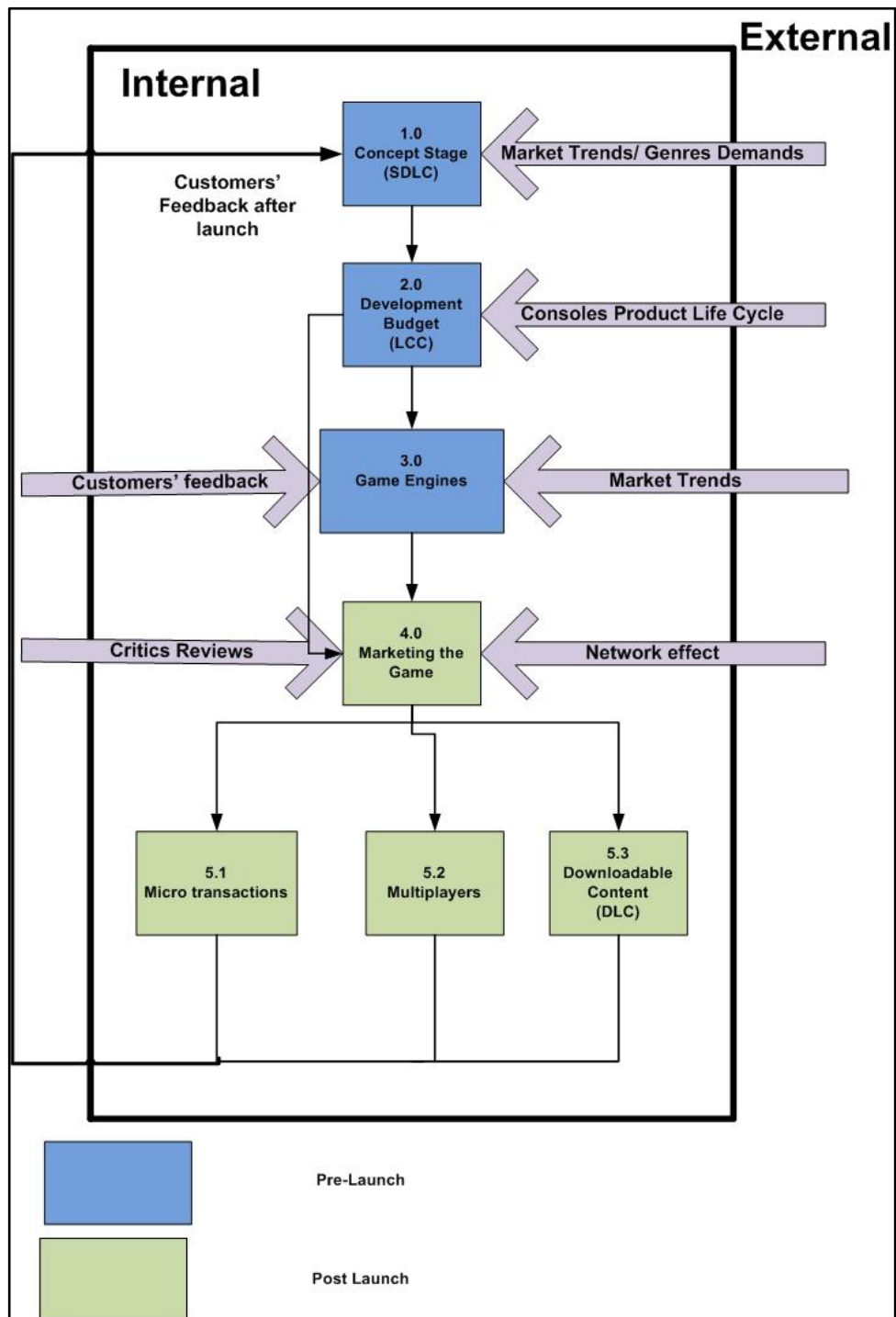
Apaza Yllachura (2019) en su tesis de grado “Framework para diseño de contenidos en aplicaciones móviles de realidad aumentada orientado a personal no-experto” describe la problemática de la “Necesidad de que los creadores de contenidos deban tener ciertos conocimientos de programación” (2019, pág. 3), por ello con el objetivo de “Desarrollar un FDCA para aplicaciones móviles orientado a personal con poca o nula experiencia en programación” (2019, pág. 11) realiza un análisis de otros frameworks de diseño de contenidos y con los datos recabados genera una propuesta de Framework. La propuesta define como una aplicación web en Angular desarrollada en Google Blockly, el cual permite extenderse para utilizar framework como Vuforia para la implementación de RA. Entre sus conclusiones menciona que “El framework generado es adaptable a cualquier framework de programación de RA, en particular Vuforia” (2019, pág. 44).

2.1.2. Artículos Internacionales

Ahmad et al. (2017) realizaron su artículo “How to Launch A Successful Video Game: A Framework” el cual nace a propósito de que los frameworks existentes solo consideraban aspectos limitados del desarrollo de videojuegos, por ello su objetivo es el de “Desarrollar un Framework Holístico que cubra todos los aspectos del desarrollo de videojuegos” (2017, pág. 1). Describe un modelo de desarrollo de videojuegos el cual está compuesto de dos partes, el Pre-Launch que consta de las fases: Concept Stage, Development Budget, Game Engines; y Post-Launch que contiene las fases: Marketing the Game, Micro Transactions, Multiplayer y DLC (ver **Figura 2**).

Figura 2

Suggested Video Games Developers Framework, Ahmad et al.



Nota. La figura muestra el modelo de desarrollo de videojuegos holístico.
Fuente: Ahmad et al., 2017

Por último, Garita et al. (2019) realizan su investigación "Metodologías para el Desarrollo de Videojuegos Serios: Una

Revisión de Literatura”, con el objetivo de “Elaborar una revisión de literatura de las metodologías para el desarrollo de videojuegos serios y así poder determinar su estado” (2019, pág. 103). En este artículo hacen una revisión de múltiples metodologías de desarrollo de videojuegos con características diferentes las cuales llama: Edugame, Advergame, Artgame y Exergame. Las metodologías que presentan son los Modelos Agiles, Modelo de Diseño Centrado en el Usuario, Modelo Holístico, Modelos Basados en Paradigmas Específicas y Modelos Basados en Necesidades Específicas. En su análisis mencionan que las empresas desarrolladoras de videojuegos se han visto en la necesidad de adoptar metodologías existentes que no habían sido diseñadas para el fin de desarrollar videojuegos ya que no querían verse experimentando con una metodología completamente nueva. Mencionan que las metodologías Desarrollo Centrado en el Usuario y Scrum podrían complementarse para cumplir con el objetivo del desarrollo de videojuegos (Garita et al., 2019, págs. 6-10).

2.1.3. Benchmarking

Tabla 5

Evaluación Comparativa de Artículos

Nº	Título	Problema	Objetivo	Proceso	Conclusiones
1	Técnica para validar el diseño de software durante la elicitación de requisitos empleando especificaciones de usabilidad	Los desarrolladores de software, para validar el resultado de su codificación muchas veces tienen que esperar a la construcción total del producto de software para saber si lograron entender el modelo de negocio.	Proponer una técnica que permita validar el diseño del software cuyas necesidades se encuentran definidas en el catálogo de requisitos y cuya estructura acepta las especificaciones de usabilidad.	Desarrollaron una técnica de validación del diseño de software durante la elicitación de requisitos empleando especificaciones de usabilidad e independiente del proceso de construcción del producto de software.	La estructura propuesta para la elicitación de requisitos mejora la toma de requerimientos y la documentación de construcción del producto desde los criterios de usabilidad del usuario final. El catálogo de requisitos mejora su calidad y contempla los casos del usuario final.
2	Buenas Prácticas Para El Desarrollo De Videojuegos Educativos Aplicados A Historia Y Arqueología Usando Realidad Aumentada Para La Educación Primaria En El Perú	Existen pocas referencias acerca de las buenas prácticas de desarrollo de videojuegos educativos aplicados a la enseñanza de arqueología usando realidad aumentada para la educación primaria en el Perú.	Proponer buenas prácticas de desarrollo de videojuegos educativos aplicados a la enseñanza de historia y arqueología integrando realidad aumentada para contribuir con la mejora del desarrollo de videojuegos educativos en el Perú.	Realizaron un análisis de casos de desarrollo de videojuegos educativos, con lo cual elaboran un documento de buenas prácticas y una metodología para el desarrollo de videojuegos.	Se logró reducir el tiempo de desarrollo en un 42.0% utilizando la metodología desarrollada
3	How to Launch A Successful Video Game: A Framework	Los frameworks existentes solo consideran aspectos limitados del desarrollo de videojuegos	Desarrollar un Framework Holístico que cubra todos los aspectos del desarrollo de videojuegos	Propone un framework dividido en dos partes: Pre-Lanzamiento y Post-Lanzamiento del juego. Los cuales contienen fases de Concepto, Desarrollo y Mantenimiento, que a su vez describen actividades a realizar para el desarrollo de un videojuego.	Se propuso un framework con una aplicación completa para fomentar un método alternativo de desarrollo eficaz de videojuegos desde la concepción hasta el lanzamiento.
4	Metodologías para el Desarrollo de Videojuegos Serios: Una Revisión de Literatura	El desarrollo de videojuegos serios es un tema sumamente nuevo, como consecuencia de esto se ha dado pie a la improvisación y a una falta de estandarización a la hora en que estos son desarrollados	Elaborar una revisión de literatura de las metodologías para el desarrollo de videojuegos serios y así poder determinar su estado.	Se realizó una búsqueda de artículos sobre metodología de videojuegos serios. Luego se seleccionaron y organizaron. Con los restantes se procedió a detallar cada uno de ellos.	Se enfocó en el estado de la literatura del tema de las metodologías para desarrollo de videojuegos serios, siendo este poco documentado y relativamente nuevo.
5	Framework para diseño de contenidos en aplicaciones móviles de realidad aumentada orientado a personal no-experto	Necesidad de que los creadores de contenidos deban tener ciertos conocimientos de programación.	Desarrollar un FDCA para aplicaciones móviles orientado a personal con poca o nula experiencia en programación.	Realiza un análisis de otros frameworks de diseño de contenidos y con los datos recabados genera una propuesta de Framework. La propuesta define como una aplicación web en Angular desarrollada en Google Blockly, el cual permite extenderse para utilizar framework como Vuforia para la implementación de RA.	El framework generado es adaptable a cualquier framework de programación de RA, en particular Vuforia.

Elaborado por: el autor

2.2. Bases Teóricas

2.2.1. Videojuego

Según el libro "Half-Real: Video Games between Real Rules and Fictional Worlds" de Jesper Juul (2005), un videojuego es un juego que combina reglas reales y mundos ficticios. El autor argumenta que los videojuegos son una forma de juego en la que los jugadores interactúan con un sistema de reglas y objetivos, y en el que también se representa un mundo ficticio.

De esta forma, para Juul, existen dos tipos principales de juegos: los juegos de reglas y los juegos de ficción.

- Los juegos de reglas son aquellos en los que el jugador se enfrenta a un conjunto de reglas y objetivos definidos. Estos juegos se caracterizan por tener una estructura clara y un objetivo final. Ejemplos de juegos de reglas incluyen juegos de mesa, juegos de estrategia y juegos deportivos (pág. 5).
- Los juegos de ficción son aquellos en los que el jugador se sumerge en un mundo ficticio y experimenta una historia o una narrativa. Estos juegos se caracterizan por tener una estructura menos definida y una mayor libertad para el jugador. Ejemplos de juegos de ficción incluyen juegos de aventura, juegos de rol y juegos de simulación (pág. 6).

Juul argumentó que los videojuegos son un medio único que combina ambos tipos de juegos, ya que utilizan reglas y objetivos definidos para crear una experiencia de juego, pero también utilizan ficción y narrativa para contar historias y crear una sensación de inmersión.

Jesper Juul describió varios modelos de juegos, algunos de ellos son:

- El modelo clásico de juego, que se compone de cinco elementos: reglas, objetivos, un sistema de retroalimentación, esfuerzo del jugador y participación voluntaria.

- El modelo de juego narrativo, que se centra en cómo los juegos utilizan la narrativa para contar historias y crear significado.
- El modelo de juego emergente, en el que los jugadores tienen la capacidad de generar contenido y acciones dentro del juego.
- El modelo de juego de autor, en el que el juego es una obra artística y el diseñador es el autor.
- El modelo de juego de rol, en el que los jugadores asumen el papel de un personaje y actúan en un mundo ficticio.
- El modelo de juego de simulación, en el que los jugadores imitan o replican un sistema real.
- El modelo de juego de deportes, en el que los jugadores compiten en un juego simulando un deporte.
- El modelo de juego de estrategia, en el que los jugadores toman decisiones para lograr un objetivo.

Juul analizó cómo estos modelos se relacionan con la realidad y la ficción en los juegos, y cómo estos modelos afectan la experiencia del jugador.

Adicionalmente Juul, describió una nueva definición del modelo clásico de juego, la cual es una teoría que define las características esenciales del juego y cómo éste se diferencia de otras actividades. A continuación, se describen con más detalle las seis características:

1. El juego tiene reglas establecidas y limitaciones claras: El juego tiene un conjunto de reglas y límites que los jugadores deben seguir. Estas reglas determinan cómo se juega el juego y cómo se gana o pierde.
2. El juego es una actividad voluntaria y autodirigida: Los jugadores deciden jugar por su cuenta y tienen control sobre cuándo empezar y terminar el juego.
3. El juego tiene un objetivo o meta: El juego tiene un propósito o una meta a alcanzar. Esta meta puede ser ganar una partida, obtener un puntaje alto o completar un desafío.

4. El juego implica una cierta cantidad de incertidumbre o azar: El juego tiene un elemento de incertidumbre o azar, lo que significa que el resultado final no puede ser predicho con certeza. Esto añade emoción y tensión al juego.
5. El juego es una actividad temporal, con un comienzo y un final: El juego tiene un comienzo y un final, ya sea en forma de una sesión de juego individual o una serie de partidas.
6. El juego implica un cambio temporal en el estado de ánimo del jugador: El juego puede causar un cambio temporal en el estado emocional del jugador, desde un estado de tensión o concentración al comienzo del juego hasta un estado de relajación o alivio al final del juego.

El modelo clásico de juego de Jesper Juul sostiene que el juego es una actividad voluntaria, regida por reglas, con un objetivo, un componente de azar, un comienzo y un fin, y con un cambio temporal en el estado emocional del jugador (págs. 36 - 43).

2.2.2. Framework

En el libro *Framework Design Guidelines: Conventions, Idioms, and Patterns for Reusable .NET Libraries* de Cwalina & Brad (2009), un framework se define como un conjunto de componentes de software que proporciona una estructura común para la solución de un conjunto específico de problemas. Los componentes del marco se utilizan juntos para resolver problemas comunes y proporcionar una base sólida para el desarrollo de aplicaciones. Los marcos también proporcionan una interfaz pública estandarizada para interactuar con los componentes, lo que facilita la reutilización y la interoperabilidad entre diferentes aplicaciones y componentes (págs. 1-7).

Según los autores, los pasos para crear un framework son los siguientes:

1. Identificar los problemas comunes: El primer paso en la creación de un framework es identificar los problemas comunes que se presentan en

el desarrollo de aplicaciones. Esto ayudará a determinar el alcance del marco y los componentes que se necesitarán.

2. Diseño de la interfaz pública: Una vez que se han identificado los problemas comunes, el siguiente paso es diseñar la interfaz pública del marco. La interfaz pública debe ser fácil de usar, coherente y proporcionar una interfaz estandarizada para interactuar con los componentes del marco.
3. Diseño de la arquitectura: Una vez que se ha diseñado la interfaz pública, el siguiente paso es diseñar la arquitectura del marco. Esto incluye la selección de patrones de diseño, la organización de los componentes y la definición de las relaciones entre ellos.
4. Implementación: Una vez que se ha diseñado la arquitectura, el siguiente paso es implementar el marco. Es importante seguir las convenciones y patrones establecidos en el diseño para garantizar la coherencia y la facilidad de uso del marco.
5. Pruebas y documentación: Antes de lanzar el marco al público, es necesario realizar pruebas exhaustivas y proporcionar una documentación detallada para ayudar a los desarrolladores a usar el marco de manera eficiente.
6. Mantenimiento y mejoras: una vez lanzado, es importante seguir manteniendo y mejorando el framework para adaptarlo a las nuevas necesidades y tecnologías.

Algunas buenas prácticas mencionadas por los autores en el desarrollo de un framework son las siguientes:

- Diseño simple y coherente: Es importante mantener un diseño simple y coherente para garantizar que el marco sea fácil de usar y entender.
- Convenciones estandarizadas: Es importante seguir convenciones estandarizadas para garantizar la coherencia y la facilidad de uso del marco.
- Documentación detallada: Es importante proporcionar una documentación detallada para ayudar a los desarrolladores a usar el marco de manera eficiente.

- Pruebas exhaustivas: Es importante realizar pruebas exhaustivas para garantizar la estabilidad y la confiabilidad del marco.
- Interoperabilidad: Es importante proporcionar una interfaz pública estandarizada para interactuar con los componentes del marco, lo que facilita la reutilización y la interoperabilidad entre diferentes aplicaciones y componentes.
- Flexibilidad y escalabilidad: Es importante diseñar el marco de manera que sea flexible y escalable para adaptarse a las futuras necesidades y cambios tecnológicos.
- Enfoque en resolver problemas comunes: El marco debe estar diseñado para resolver problemas comunes y proporcionar una base sólida para el desarrollo de aplicaciones.
- Evitar depender de una tecnología específica: Es importante evitar depender de una tecnología específica para garantizar la portabilidad y la flexibilidad del marco.

2.2.3. Videojuegos Formativos o Juegos Serios

El libro "Serious Games: Games That Educate, Train, and Inform" de Michael & Chen (2005), los autores definen un videojuego serio como un juego que tiene un propósito más allá de la simple diversión, como la educación, la capacitación o la información. Estos juegos utilizan elementos para enseñar habilidades, transmitir conocimientos o fomentar la reflexión sobre temas importantes. También señalan que estos juegos son diseñados específicamente para tener un impacto positivo en el usuario. Están basados en la pedagogía, la psicología y la tecnología para crear experiencias de juego enriquecedoras que lleven a un aprendizaje más profundo y significativo. Estos videojuegos son utilizados en una amplia variedad de entornos, desde la educación formal hasta el entrenamiento empresarial y la sensibilización social. Los videojuegos serios son una herramienta poderosa para lograr objetivos educativos, de capacitación y de información (págs. 17-28).

Las características principales de los juegos serios incluyen:

- Propósito educativo: tienen un objetivo educativo o de capacitación claro.
- Basados en investigación: están diseñados en base a la investigación en pedagogía, psicología y tecnología para asegurar un impacto positivo en el usuario.
- Experiencia enriquecedora: utilizan elementos de juego para crear una experiencia de juego enriquecedora que lleve a un aprendizaje más profundo y significativo.
- Interacción y motivación: fomentan la interacción y la motivación del usuario a través de desafíos y recompensas.
- Personalización y adaptabilidad: permiten la personalización y adaptabilidad a las necesidades individuales del usuario.
- Evaluación: proporcionan una evaluación continua y precisa del progreso y desempeño del usuario.

Los juegos serios son juegos que tienen un propósito más allá de la diversión y están diseñados para asegurar un impacto positivo en el usuario a través de una experiencia de juego enriquecedora y motivadora.

Algunas de los campos en los que se pueden aplicar los juegos serios son:

- Educación formal: se utilizan en el aula para mejorar el aprendizaje y la comprensión de los conceptos.
- Entrenamiento empresarial: se utilizan en el ámbito empresarial para capacitar a los empleados en habilidades específicas.
- Sensibilización social: se utilizan para sensibilizar sobre temas sociales y ambientales, como la igualdad de género, la sostenibilidad y la salud mental.

- Entrenamiento militar: se utilizan en el entrenamiento militar para preparar a los soldados en situaciones de combate y simulación.
- Terapia: se utilizan en la terapia para tratar problemas de salud mental y físicos.
- Investigación: se utilizan en investigaciones científicas para explorar temas como la cognición, el comportamiento y la percepción.

2.2.4. Metodología Ágil

Según Schwaber & Sutherland es una metodología de trabajo ágil, el cual que permite abordar problemas complejos y de constantes cambios, de manera incremental e iterativa, para que se puedan generar entregas de productos de manera eficiente (2016, pág. 3).

2.2.5. Scrum

Es un marco de trabajo ágil, que permite generar una metodología adaptativa, rápida, eficaz, flexible e iterativa, con el fin de aumentar significativamente el valor de los proyectos gracias a la rapidez con la que son gestionados. Scrum se basa en el trabajo en equipo, la autoorganización y la experiencia, lo cual permite garantizar un ambiente responsabilidad colectiva y de transparencia en la comunicación (SCRUMstudy, 2016, pág. 2).

a) Principios Scrum

Control del proceso empírico. Se base principalmente en la adaptación rápida al cambio, la inspección de las actividades y entregables y la transparencia del equipo del proyecto.

Autoorganización. Basado principalmente en el valor agregado que brindan los equipos con un gran compromiso y responsabilidad, que pueden llegar a organizarse a sí mismos.

Colaboración. Tiene como principal foco el trabajo colaborativo, este debe presentar tres dimensiones esenciales que son la apropiación, articulación y conocimiento.

Priorización basada en valor. Este principio tiene como base el límite del tiempo, para lo cual implementan lo que son los Sprint, reuniones diarias, reuniones de planificación y reuniones de revisión de estas, que ayudan a cumplir con el cronograma del proyecto.

Desarrollo Iterativo. Este principio se enfoca en el control de cambios y la creación de productos que cumplan con los requerimientos establecidos por el cliente, con ello también se delimita las responsabilidades de propietario del producto y el de la organización.

b) Artefactos

La lista priorizada de pendientes del producto. Documento que establece los requisitos del cliente que definirán el proyecto, dado que prioriza las características del producto o servicio que deberá tener el entregable final.

La lista de pendientes del sprint. Lista de actividades a realizar por el equipo de proyecto para el próximo Sprint.

Cronograma de planificación del lanzamiento. Documento donde se establecen las fechas de lanzamiento, los intervalos que se planificaron y los entregables a ser presentados en cada fecha.

c) Roles

Propietario del producto. Persona responsable de interpretar y articular los requisitos del cliente y de mantener siempre presente el alcance del proyecto.

Scrum Máster. Persona facilitadora que asegura el ambiente adecuado para equipo, facilita e imparte las practicas del Scrum. Asimismo, ayuda a eliminar los impedimentos que afronta el equipo y

se asegura de que cumplan con los procesos correspondientes a este marco de trabajo ágil.

Equipo Scrum. Grupo de personas que se encargan de comprender y llevar a cabo los requisitos del cliente para la presentación respectiva del producto o servicio al propietario del producto.

d) Bloques de Tiempo de Scrum

Sprint. Es una iteración que representa un bloque de una a seis semanas. Es en este tiempo donde se genera el entregable requerido y el Scrum Máster, brinda las facilidades y protege al equipo Scrum.

Reunión diaria. Hace referencia a reuniones diarias de tiempos no mayores a 15 minutos, donde los miembros del equipo se reúnen para comunicar sobre sus avances y pendientes.

Reunión de planificación del sprint. Es una reunión realizada antes del inicio de cada Sprint, se debe considerar que para un Sprint valorizado en un mes de duración se debe tener una planificación de un máximo de ocho horas. Asimismo, se debe considerar que en estas reuniones se deben de generar la lista de pendientes a realizarse.

Reunión de revisión del sprint. En estas reuniones el equipo Scrum expone el producto o servicio realizado al Propietario del Producto, es aquí donde se evalúa y aprueba o rechaza las historias de usuario realizadas. Estas suelen tener la duración de 4 horas por un sprint de un mes.

Reunión de retrospectiva del sprint. En esta reunión el equipo Scrum revisa y considera todo lo realizado en el sprint anterior, con el fin de mejorar la agilidad y transparencia en los próximos Sprint. Se considera un tiempo de 4 horas por un sprint de un mes.

2.3. Definición de Términos Básicos

Como parte de la información necesaria para la presente investigación es necesario acotar las siguientes definiciones:

- a) **Asset.** “Un asset es un componente que se puede utilizar en juegos o proyectos que se desarrollen. Un asset puede ser originado de un archivo generado como una imagen, un video, un audio o incluso como un modelo 3D” (Unity3d, s.f.).
- b) **Asset Package.** “Los paquetes de activos son colecciones de archivos y datos de proyectos de Unity, o elementos de proyectos, que Unity comprime y almacena en un archivo con la extensión “unitypackage” (Unity3d, s.f.).
- c) **Corrutina.** “Los métodos de corrutina se ejecutan pieza por pieza a lo largo del tiempo, pero todos los procesos aún los realiza un único subproceso principal” (Unity3d, s.f.).
- d) **GameObject.** Los GameObjects son elementos primordiales en Unity debido a que representa escenarios, propiedades y principalmente personajes (Unity3d, s.f.).
- e) **Prefab.** Es un gameobject que actúa como una plantilla la cual se puede instanciar múltiples veces en la escena. Cualquier modificación realizada a un prefab será reflejada en todas las instancias de esta, pero, también se puede ajustar cada instancia individualmente (Unity3d, s.f.).
- f) **Vuforia.** “Es una herramienta para el desarrollo de aplicaciones de Realidad Aumentada (AR) y Realidad Mixta (MR) multiplataforma” (Vuforia, s.f.).

CAPÍTULO III

METODOLOGÍA

3.1. Método

3.1.1. Enfoque de la Investigación

La investigación aplicada tiene como principal objetivo la generación de conocimiento en base a los descubrimientos tecnológicos que buscan resolver problemas de la sociedad o del sector productivo, mediante el enlace de la teoría y el producto (Lozada, 2014, pág. 34).

De acuerdo con la definición de Lozada, la investigación actual es de tipo Investigación Aplicada, ya que se busca utilizar conocimiento y la tecnología para cubrir un objetivo productivo.

3.1.2. Selección de la Metodología

Para este proyecto se ha determinado que se trabajará bajo una metodología ágil, a continuación, se determinará que metodología se utilizará entre Scrum y XP.

Es natural que tanto Scrum como XP sean metodologías que cuentan con conceptos similares, aunque también con sus respectivas diferencias, dentro de las cuales la principal es la estructura del equipo y roles, siendo así que se requiera una mayor cantidad de personas involucradas en el proyecto, haciendo que se eleven los costos de desarrollo, por lo cual Scrum toma ventaja al contar con 3 roles que pueden ser realizadas por un mismo individuo de ser necesario. Así mismo, no se encuentran artefactos o entregables estandarizados dentro de XP, así como eventos que permitan una visión global del avance del proyecto, aunque si cuenta con una mejora en el desarrollo del proyecto, aun así, Scrum se maneja como una metodología con mayor performance para el desarrollo documentado de un proyecto, ofreciendo una mayor confianza al cliente o inversionista. En aspectos de tiempo y adaptabilidad XP si es un marco que ofrece mejores enfoques al respecto, siendo mejor que Scrum ya que su tiempo de iteración es un poco más largo, así como la adaptabilidad para por un proceso de comunicación entre el Dueño del Producto y el Cliente mientras que, en XP, el cliente está involucrado directamente en el desarrollo del producto.

Con todo lo antes explicado, se ha realizado una tabla de criterios y puntuación (ver **Tabla 6**) para determinar la metodología a utilizar.

Tabla 6*Criterios de Evaluación de Metodología*

Criterios de evaluación	Detalle	SCRUM	XP
Equipo	Debe tener una cantidad minima y flexibilidad en los roles	5	2
Artefactos	Debe contar con entregables tanto de documentación como de desarrollo del producto	5	3
Tiempo	Debe tener iteraciones de corto tiempo	3	5
Adaptabilidad	Debe permitir los cambios continuos	4	5
EVALUACIÓN		17	15

Elaborado por: el autor

Según los criterios de evaluación y lo resultados, la metodología para usarse en la investigación será Scrum.

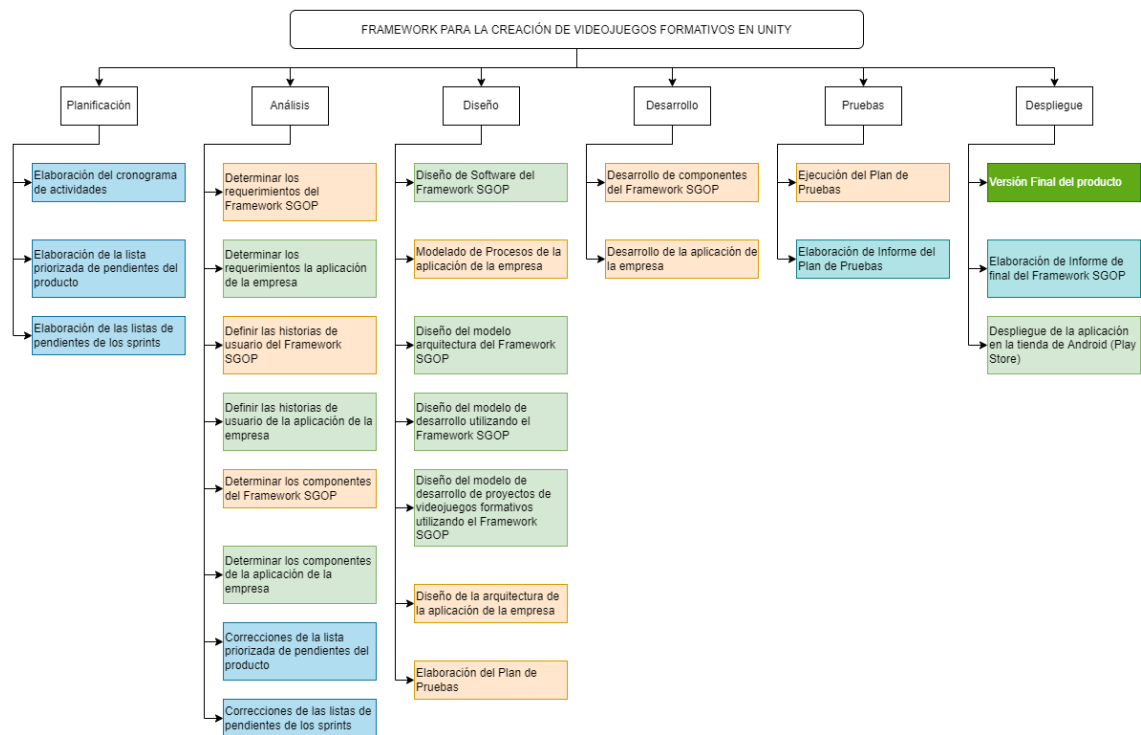
3.1.3. Desarrollo Metodológico

a) Estructura De Desglose De Trabajo

En lo que respecta al desarrollo se ha elaborado una estructura de desglose de trabajo (ver **Figura 3**) para detallar las actividades que se llevarán a cabo en cada etapa del ciclo de vida del software.

Figura 3

EDT - Framework para la Creación de Videojuegos Formativos en Unity



Nota. La figura muestra la Estructura de Desglose de Trabajo con las actividades propuestas para el desarrollo de la presente investigación.
Elaborado por: el autor

Tabla 7

Cronograma de Actividades

Nombre de la actividad	Fecha de inicio	Fecha de finalización	Estado	Cronograma de actividades (Días)																																																																											
				06.08.2022	08.08.2022	09.08.2022	10.08.2022	11.08.2022	12.08.2022	15.08.2022	16.08.2022	17.08.2022	18.08.2022	19.08.2022	22.08.2022	23.08.2022	24.08.2022	25.08.2022	29.08.2022	30.08.2022	31.08.2022	01.09.2022	02.09.2022	05.09.2022	06.09.2022	07.09.2022	08.09.2022	09.09.2022	12.09.2022	13.09.2022	14.09.2022	15.09.2022	16.09.2022	19.09.2022	20.09.2022	21.09.2022	22.09.2022	23.09.2022	26.09.2022	27.09.2022	28.09.2022	29.09.2022	30.09.2022	03.10.2022	04.10.2022	05.10.2022	06.10.2022	07.10.2022	10.10.2022	11.10.2022	12.10.2022	13.10.2022	14.10.2022	17.10.2022	18.10.2022	19.10.2022	20.10.2022	21.10.2022	24.10.2022	25.10.2022	26.10.2022	27.10.2022	28.10.2022	31.10.2022	01.11.2022	02.11.2022	03.11.2022	04.11.2022	07.11.2022	08.11.2022	09.11.2022	10.11.2022	11.11.2022	14.11.2022	15.11.2022	16.11.2022	17.11.2022	18.11.2022	21.11.2022
Framework Para La Creación De Videojuegos Formativos En Unity	06.08.2022	25.11.2022	Terminado	[Barra verde continua]																																																																											
1 Planificación del Proyecto	06.08.2022	06.08.2022	Terminado	[Barra verde]																																																																											
1 Elaboración del cronograma de actividades	06.08.2022	06.08.2022	Terminado	[Barra verde]																																																																											
1 Elaboración de la lista priorizada de pendientes del producto	06.08.2022	06.08.2022	Terminado	[Barra verde]																																																																											
1 Elaboración de las listas de pendientes de los sprints	06.08.2022	06.08.2022	Terminado	[Barra verde]																																																																											
2 Análisis	08.08.2022	02.09.2022	Terminado	[Barra verde]																																																																											
2 Determinar los requerimientos del Framework SGOP	08.08.2022	11.08.2022	Terminado	[Barra verde]																																																																											
2 Determinar los requerimientos la aplicación de la empresa	11.08.2022	16.08.2022	Terminado	[Barra verde]																																																																											
2 Definir las historias de usuario del Framework SGOP	16.08.2022	18.08.2022	Terminado	[Barra verde]																																																																											
2 Definir las historias de usuario de la aplicación de la empresa	18.08.2022	19.08.2022	Terminado	[Barra verde]																																																																											
3 Determinar los componentes del Framework SGOP	22.08.2022	30.08.2022	Terminado	[Barra verde]																																																																											
3 Determinar los componentes de la aplicación de la empresa	30.08.2022	02.09.2022	Terminado	[Barra verde]																																																																											
3 Correcciones de la lista priorizada de pendientes del producto	02.09.2022	02.09.2022	Terminado	[Barra verde]																																																																											
3 Correcciones de las listas de pendientes de los sprints	02.09.2022	02.09.2022	Terminado	[Barra verde]																																																																											
3 Diseño	05.09.2022	28.09.2022	Terminado	[Barra verde]																																																																											
3 Diseño de Software del Framework SGOP	05.09.2022	13.09.2022	Terminado	[Barra verde]																																																																											
3 Modelado de Procesos de la aplicación de la empresa	13.09.2022	16.09.2022	Terminado	[Barra verde]																																																																											
3 Diseño del modelo arquitectura del Framework SGOP	19.09.2022	22.09.2022	Terminado	[Barra verde]																																																																											
3 Diseño de la arquitectura de la aplicación de la empresa	22.09.2022	23.09.2022	Terminado	[Barra verde]																																																																											
4 Elaboración del Plan de pruebas	23.09.2022	28.09.2022	Terminado	[Barra verde]																																																																											
4 Desarrollo	28.09.2022	22.11.2022	Terminado	[Barra verde]																																																																											
4 Desarrollo de componentes del Framework SGOP	28.09.2022	31.10.2022	Terminado	[Barra verde]																																																																											
4 Desarrollo de la aplicación de la empresa	31.10.2022	22.11.2022	Terminado	[Barra verde]																																																																											
5 Pruebas	22.11.2022	10.11.2022	Terminado	[Barra verde]																																																																											
5 Ejecución del Plan de Pruebas	22.11.2022	25.11.2022	Terminado	[Barra verde]																																																																											
5 Elaboración de Informe del Plan de Pruebas	25.11.2022	25.11.2022	Terminado	[Barra verde]																																																																											
6 Despliegue	25.11.2022	25.11.2022	Terminado	[Barra verde]																																																																											
6 Versión Final del producto	25.11.2022	25.11.2022	Terminado	[Barra verde]																																																																											
6 Elaboración de Informe de final del Framework SGOP	25.11.2022	25.11.2022	Terminado	[Barra verde]																																																																											
6 Despliegue de la aplicación en la tienda de Android (Play Store)	25.11.2022	25.11.2022	Terminado	[Barra verde]																																																																											

Elaborado por: el autor

En la **Tabla 7** se detallan las actividades y tareas que se realizaron para llevar a cabo el proyecto actual. El método que se ha visto más adecuado para cumplir con el objetivo del proyecto se basa en analizar y definir los requerimientos del producto que la empresa espera obtener como resultado, y en función de esto proceder a analizar y definir los requerimientos para el diseño del framework de desarrollo de videojuegos formativos. Con el análisis concluido se procederá a realizar el diseño y desarrollo del framework y una vez concluido se desarrollará el diseño y desarrollo de la aplicación para la empresa.

b) Desarrollo Scrum

La asignación de roles de Scrum se detalla en la siguiente tabla:

Tabla 8

Asignación de Roles Scrum

Rol	Responsable
Product Owner	Eusebio Garay, Claudio
Scrum Master	
Equipo de trabajo	

Elaborado por: el autor

Para la proyección de actividades el proyecto se ha dividido en dos épicas el primero que hace referencia al tema principal de la investigación que es el Framework SGOP (Serious Game On Page) y el segundo que es el producto que se ha realizado para la empresa.

Tabla 9

Listado de Épicas

N.º	Épicas	Pts. Estimados	h
1	SGOP Framework	93	93h
2	Aplicación de la empresa	75	75h
Total		168	168h

Elaborado por: el autor

A continuación, en la **Tabla 10** se detalla la lista priorizada de pendientes del producto el cual incluye la épica a la que corresponde

cada actividad, los puntos estimados de complejidad de la actividad y las horas de desarrollo.

Tabla 10

Lista Priorizada de Pendientes del Producto

N.º Épica	N.º	Actividad	Pts. Estimados	h
1	1	Determinar los requerimientos del Framework SGOP	8	8h
2	2	Determinar los requerimientos la aplicación de la empresa	5	5h
1	3	Definir las historias de usuario del Framework SGOP	5	5h
2	4	Definir las historias de usuario de la aplicación de la empresa	3	3h
1	5	Determinar los componentes del Framework SGOP	13	13h
2	6	Determinar los componentes de la aplicación de la empresa	8	8h
1	7	Diseño de Software del Framework SGOP	13	13h
2	8	Modelado de Procesos de la aplicación de la empresa	8	8h
1	9	Diseño del modelo arquitectura del Framework SGOP	2	2h
1	10	Diseño del modelo de desarrollo utilizando el Framework SGOP	2	2h
1	11	Diseño del modelo de desarrollo de proyectos de videojuegos formativos utilizando el Framework SGOP	4	4h
2	12	Diseño de la arquitectura de la aplicación de la empresa	2	2h
2	13	Elaboración del Plan de Pruebas	5	5h
1	14	Desarrollo de componentes del Framework SGOP	51	51h
2	15	Desarrollo de la aplicación de la empresa	30	30h
2	16	Ejecución del Plan de Pruebas	8	8h
2	17	Versión Final del producto	1	1h
Total			168	168h

Elaborado por: el autor

Las actividades se dividieron en tareas las cuales se agruparon y desarrollaron en 4 sprints, en cada sprint se completaron un total de 42 horas de trabajo aproximadamente.

A continuación, se detallan las listas de pendientes de los sprints realizados, cada tarea está relacionada con una actividad, tiene sus puntos estimados de dificultad y sus horas de duración.

En el sprint 1 se realizaron las tareas relacionadas con el análisis requerimientos y componentes del tanto del Framework SGOP como de la aplicación.

Tabla 11*Lista de Pendientes del Sprint 1*

N.º act.	N.º	Tarea	Pts. Estimados	h
1	1	Definir los requerimientos funcionales del Framework SGOP	6	6h
1	2	Definir los requerimientos no funcionales del Framework SGOP	2	2h
2	3	Definir los requerimientos funcionales de la aplicación de la empresa	3	3h
2	4	Definir los requerimientos no funcionales de la aplicación de la empresa	2	2h
3	5	Definir las historias de usuario del Framework SGOP	5	5h
4	6	Definir las historias de usuario de la aplicación de la empresa	3	3h
5	7	Análisis de proyectos anteriores para el Framework SGOP	8	8h
5	8	Determinar componentes del Framework SGOP	5	5h
6	9	Determinar los componentes de la aplicación de la empresa	8	8h
Total			42	42h

Elaborado por: el autor

En el sprint 2 se diseñó los componentes del framework y la aplicación, también el modelo de arquitectura del Framework SGOP y la arquitectura de la aplicación. Por último, se empezó con el desarrollo de los componentes del Framework SGOP.

Tabla 12*Lista de Pendientes del Sprint 2*

N.º act.	N.º	Tarea	Pts. Estimados	h
7	1	Diseño de diagramas de clases del Framework SGOP	13	13h
8	2	Modelado de Procesos de la aplicación de la empresa	8	8h
9	3	Diseño del modelo arquitectura del Framework SGOP	2	2h
10	4	Diseño del modelo de desarrollo utilizando el Framework SGOP	2	2h
11	5	Diseño del modelo de desarrollo de proyectos de videojuegos formativos utilizando el Framework SGOP	4	4h
12	6	Diseño de la arquitectura de la aplicación de la empresa	2	2h
13	7	Diseño de Casos de pruebas	5	5h
14	8	Desarrollo de componentes del paquete SGOP	1	1h
14	9	Desarrollo de componentes del paquete SGOP.Model	5	5h
Total			42	42h

Elaborado por: el autor

En el sprint 3 se finalizó el desarrollo de los componentes del Framework SGOP

Tabla 13*Lista de Pendientes del Sprint 3*

N.º act.	N.º	Tarea	Pts. Estimados	h
14	1	Desarrollo de componentes del paquete SGOP.Base	8	8h
14	2	Desarrollo de componentes del paquete SGOP.UI	8	8h
14	3	Desarrollo de componentes del paquete SGOP.Util	8	8h
14	4	Desarrollo de componentes del paquete SGOP.Manager	18	18h
Total			42	42h

Elaborado por: el autor

En el último sprint se desarrolló la aplicación, se ejecutaron los casos de pruebas y se generó la versión final del producto.

Tabla 14*Lista de Pendientes del Sprint 4*

N.º act.	N.º	Tarea	Pts. Estimados	h
14	1	Desarrollo de componentes del paquete SGOP.Manager	3	3h
15	2	Desarrollo de la vista Login	4	4h
15	3	Desarrollo de la vista Home	3	3h
15	4	Desarrollo de la vista Franchise	2	2h
15	5	Desarrollo de la vista Video Player	2	2h
15	6	Desarrollo del juego Memory Card	8	8h
15	7	Desarrollo de la vista Stickerbook	3	3h
15	8	Desarrollo de la vista Stickerbook RA	8	8h
16	9	Ejecución de Casos de pruebas	8	8h
17	10	Versión Final del producto	1	1h
Total			42	42h

Elaborado por: el autor

3.1.4. Entregables

Se han identificado los siguientes entregables:

- Framework para la creación de videojuegos formativos.
- Aplicación con implementación del framework para la empresa.
- Manual de usuario del Framework.

CAPÍTULO IV

DESARROLLO DE LA SOLUCIÓN

4.1. Requerimientos

El desarrollo del proyecto consta de dos partes: la primera es el desarrollo de un framework, la segunda es la validación del framework en la creación de un producto. Ambas partes del desarrollo del proyecto se dividieron en Requerimientos Funcionales y Requerimientos No Funcionales.

4.1.1. Requerimientos Funcionales

Se consideraron nueve requerimientos funcionales, los cuales se describen a continuación:

Tabla 15*Listado de Requerimientos Funcionales*

Obj.	Cód. Req.	Nombre	Descripción
03	RF-01	Controlador de Descarga de Contenido	Desarrollar los módulos necesarios para controlar la descarga paralela de contenido multimedia, su manipulación y almacenamiento para su uso dentro de la aplicación.
03	RF-02	Controlador de Coroutine	Desarrollar los módulos necesarios para controlar código asíncrono, verificación de sus estados y mantener el rendimiento de la aplicación.
03	RF-03	Controlador de Vista	Desarrollar los módulos necesarios para controlar las vistas de la aplicación, páginas, componentes y sus interacciones.
03	RF-04	Controlador de Audio	Desarrollar los módulos necesarios para controlar los sonidos, efectos y grabación del micrófono.
03	RF-05	Controlador de Interfaz de Usuario	Desarrollar los módulos necesarios para controlar las interfaces de usuario, sus interacciones y manipular el contenido multimedia para su establecimiento estas.
04	RF-06	Ingresar al sistema	Desarrollar la vista con las características y funcionales necesarias para el ingreso de los usuarios verificados al contenido de pago de la aplicación.
04	RF-07	Reproducir videos	Desarrollar la visualización de los videos de contenido.
04	RF-08	Minijuego Carta Memoria	Desarrollar el minijuego Carta Memoria.
04	RF-09	Obtener Estrellas	Desarrollar la característica de ganar estrellas luego de completar un juego.
04	RF-10	Stickerbook RA	Desarrollar RA para visualizar stickers ganados jugando minijuegos.

Elaborado por: el autor

Adicionalmente, se determinaron los puntos funcionales a desarrollar para cada requerimiento, estos se describen en la Tabla 16, en la cual se detalla el requerimiento con el que está relacionado y la complejidad de cada uno.

Tabla 16*Listado de Puntos Funcionales*

Cód. Req.	Cód. PF	Nombre	Complejidad
RF-01	PF-01	Descarga De Contenido Multimedia	Media
RF-01	PF-02	Almacenamiento De Contenido Multimedia	Baja
RF-01	PF-03	Lectura De Contenido Multimedia Guardado	Baja
RF-02	PF-04	Ejecutar Métodos Asíncronos	Media
RF-02	PF-05	Terminar Métodos Asíncronos Ejecutados	Baja
RF-02	PF-06	Controlar La Cantidad De Métodos Asíncronos Ejecu	Media
RF-03	PF-07	Abrir Nuevas Vistas	Media
RF-03	PF-08	Finalizar Vistas Abiertas	Baja
RF-03	PF-09	Regresar A Vistas Anteriores	Baja
RF-04	PF-10	Reproducir Audio	Baja
RF-04	PF-11	Controlar El Volumen De Audio Reproducidos	Baja
RF-04	PF-12	Detener Audio Reproducidos	Baja
RF-05	PF-13	Establecer Contenido A Las Interfaces De Usuario	Media
RF-05	PF-14	Obtener Modelos De Interfaces De Usuario.	Baja
RF-06	PF-15	Ingresar Al Sistema De La Aplicación	Baja
RF-07	PF-16	Reproducir Videos De Franquicias	Baja
RF-08	PF-17	Minijuego Carta Memoria	Alta
RF-09	PF-18	Obtener Estrellas	Baja
RF-10	PF-19	Obtener Stickers	Baja
RF-10	PF-20	Visualizar Stickers Obtenidos	Baja
RF-10	PF-21	Visualizar Sticker En Realidad Aumentada	Alta

Elaborado por: el autor

4.1.2. Requerimientos No Funcionales

Se consideraron los siguientes requerimientos no funcionales, los cuales se describen a continuación:

Tabla 17

Listado de Requerimientos No Funcionales

Cód. Req.	Descripción	Categoría
RNF-01	El aplicativo debe mantener un óptimo rendimiento al mostrar el contenido multimedia.	Rendimiento
RNF-02	El framework debe estar optimizado para mantener el rendimiento de las aplicaciones.	Rendimiento
RNF-03	La aplicación debe presentar mensajes a los usuarios en caso de errores.	Usabilidad
RNF-04	La aplicación debe contar con un manual de usuario.	Usabilidad
RNF-05	La aplicación debe poder continuar, recuperarse o reiniciarse en caso de un error.	Confiabilidad

Elaborado por: el autor

4.1.3. Historias de Usuario

Las historias de usuario permiten entender las necesidades del usuario y cuáles son los criterios de validación. A continuación, se muestra el detalle de cada Historia de Usuario.

La primera historia de usuario hace referencia al requerimiento número 1 “Controlador de Descarga de Contenido”, esta historia explica la necesidad del desarrollador de manipular el contenido al descargarlo, guardarlo y leerlo según lo requiera.

Tabla 18*HU-01 Controlador de Descarga de Contenido*

Historia De Usuario	N.º	01	Cod:	HU-01
Nombre:				
Controlador De Descarga De Contenido				
Como:				
Desarrollador				
Quiero:				
Manipular contenido multimedia, descargarlo, guardarlo en cache y utilizarlo en el futuro.				
Para:				
Descargar contenido. Guardar el contenido descargado. Obtener el contenido guardado.				

Elaborado por: el autor

La siguiente historia de usuario está relacionada con el requerimiento número dos “Controlador de Corrutina”, esta historia busca que el desarrollador pueda realizar métodos asíncronos y ejecutarlos de manera ordenada y limitada según las capacidades de los dispositivos objetivo.

Tabla 19

HU-02 Controlador de Corrutina

Historia De Usuario	N.º	02	Cod:	HU-02
Nombre:				
Controlador De Coroutine				
Como:				
Desarrollador				
Quiero:				
Manipular múltiples métodos asíncronos e hilos, así como también determinar el estado en el que se encuentran.				
Para:				
Ejecutar métodos asíncronos. Terminar métodos asíncronos ejecutados. Controlar la cantidad de métodos asíncronos ejecutados al mismo tiempo.				

Elaborado por: el autor

La historia de usuario número tres detalla la necesidad del desarrollador por controlar la apertura y el cierre de vistas, así como poder regresar a vistas anteriormente abiertas.

Tabla 20

HU-03 Controlador de Vista

Historia De Usuario	N.º	03	Cod:	HU-03
Nombre:				
Controlador De Vista				
Como:				
Desarrollador				
Quiero:				
Manipular las vistas, sean páginas, menús, pop-ups o juegos; así como también sus interacciones y animaciones.				
Para:				
Abrir nuevas vistas. Regresar a vistas anteriores. Finalizar todas las vistas.				

Elaborado por: el autor

La siguiente historia de usuario explica el requerimiento del desarrollador por controlar la reproducción y el volumen de los audios de la aplicación.

Tabla 21

HU-04 Controlador de Audio

Historia De Usuario	N.º	04	Cod:	HU-04
Nombre:				
Controlador De Audio				
Como:				
Desarrollador				
Quiero:				
Manipular múltiples audios, sean efectos de sonido o música.				
Para:				
Reproducir audios. Controlar el volumen de los audios reproducidos. Detener los audios reproducidos.				

Elaborado por: el autor

La historia de usuario número 5 expone el requerimiento del desarrollador por establecer el contenido previamente descargado en las interfaces de usuario, así como también obtener los modelos de las interfaces de usuario a fin de poder utilizarlos en el servicio de gestor de contenido.

Tabla 22

HU-05 Controlador de Interfaz de Usuario

Historia De Usuario	N.º	05	Cod:	HU-05
Nombre:				
Controlador De Interfaz De Usuario				
Como:				
Desarrollador				
Quiero:				
Manipular las interfaces de usuario y asignar contenido multimedia según sea necesario.				
Para:				
Establecer el contenido en las interfaces de usuario. Utilizar los modelos de las interfaces de usuario para gestionar el contenido.				

Elaborado por: el autor

La siguiente historia de usuario explica el requerimiento del cliente suscrito por acceder al sistema de la aplicación.

Tabla 23

HU-06 Ingreso al Sistema

Historia De Usuario	N.º	06	Cod:	HU-06
Nombre:				
Ingreso Al Sistema				
Como:				
Cliente				
Quiero:				
Ingresar al sistema si soy un cliente un suscrito al producto.				
Para:				
Acceder al sistema si las credenciales que ingresa tienen una suscripción activa.				

Elaborado por: el autor

La historia de usuario siete detalla el requerimiento del cliente suscrito por reproducir los videos de cada franquicia de la aplicación.

Tabla 24

HU-07 Reproducir Videos de Franquicias

Historia De Usuario	N.º	07	Cod:	HU-07
Nombre:				
Reproducir Videos De Franquicias				
Como:				
Cliente				
Quiero:				
Acceder al reproductor de videos.				
Para:				
Visualizar videos según cada franquicia.				

Elaborado por: el autor

La siguiente historia de usuario explica el requerimiento del cliente de poder reproducir el minijuego carta memoria y completarlo.

Tabla 25

HU-08 Ejecutar el Minijuego de Carta Memoria

Historia De Usuario	N.º	08	Cod:	HU-08
Nombre:				
Ejecutar El Minijuego De Carta Memoria				
Como:				
Cliente				
Quiero:				
Ingresar a la vista del minijuego de Carta Memoria.				
Para:				
Ejecutar el minijuego Carta Memoria y completarlo.				

Elaborado por: el autor

La historia de usuario nuevo nueve detalla el requerimiento del cliente por obtener estrellas cada vez que completa un minijuego.

Tabla 26*HU-09 Obtener Estrellas*

Historia De Usuario	N.º	09	Cod:	HU-09
Nombre:				
Obtener Estrellas				
Como:				
Cliente				
Quiero:				
Obtener estrellas por completar minijuegos.				
Para:				
Sumar estrellas.				

Elaborado por: el autor

La siguiente historia de usuario explica la necesidad del cliente por obtener un nuevo sticker cada vez que suma la cantidad necesaria de estrellas mientras juega.

Tabla 27*HU-09 Obtener Sticker*

Historia De Usuario	N.º	10	Cod:	HU-10
Nombre:				
Obtener Stickers				
Como:				
Cliente				
Quiero:				
Sumar estrellas hasta obtener un nuevo sticker.				
Para:				
Obtener un nuevo sticker.				

Elaborado por: el autor

La última historia de usuario detalla el requerimiento del cliente por visualizar los sticker que ha obtenido y poder ver cada uno de estos en realidad aumentada.

Tabla 28*HU-11 Visualización de Sticker RA*

Historia De Usuario	N.º	11	Cod:	HU-11
Nombre:				
Visualización de Sticker RA				
Como:				
Cliente				
Quiero:				
Ingresar a la vista Stickerbook.				
Para:				
Visualizar stickers y verlos en realidad aumentada				

Elaborado por: el autor

A continuación, se detalla la trazabilidad en el cual se puede visualizar que puntos funcionales están referenciados a su historia de usuario:

Tabla 29*Trazabilidad Puntos Funcionales/Historias de Usuarios*

HU/PF	PF-01	PF-02	PF-03	PF-04	PF-05	PF-06	PF-07	PF-08	PF-09	PF-10	PF-11	PF-12	PF-13	PF-14	PF-15	PF-16	PF-17	PF-18	PF-19	PF-20	PF-21	
HU-01	X	X	X																			
HU-02				X	X	X																
HU-03							X	X	X													
HU-04										X	X	X										
HU-05													X	X								
HU-06															X							
HU-07																X						
HU-08																	X					
HU-09																		X				
HU-10																			X			
HU-11																				X	X	

Elaborado por: el autor

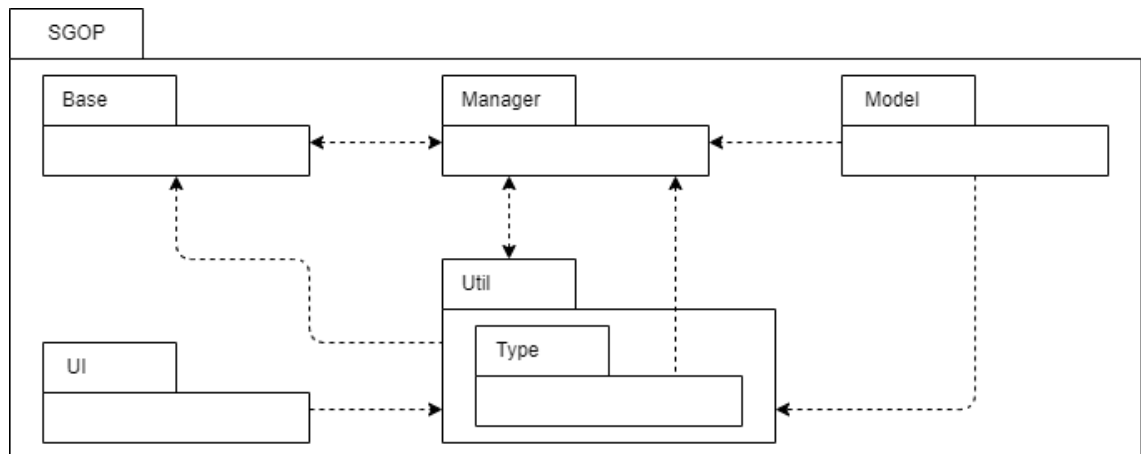
4.2. Análisis

Para la realización del análisis se contó con el apoyo de la empresa la cual proporcionó el código fuente de dos proyectos anteriores con los cuales se procedió a estudiar las funcionalidades que permiten a las aplicaciones realizar las acciones necesarias para su correcta ejecución.

Del análisis de estos proyectos se ha determinado que serán necesarios los siguientes paquetes para el framework.

Figura 4

Diagrama de Paquetes del Framework SGOP



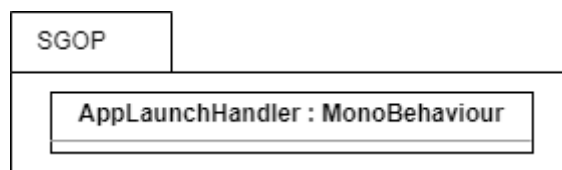
Nota. La figura muestra los paquetes que se han determinado luego del análisis. Elaborado por: el autor

A continuación, se detallan las clases e interfaces que contiene cada uno de los paquetes.

En primer lugar, el paquete SGOP el cual es el contenedor de los demás paquetes y que también contiene la clase “AppLaunchHandler” el cual se encarga del control del inicio de la aplicación.

Figura 5

Diagrama del Paquete SGOP

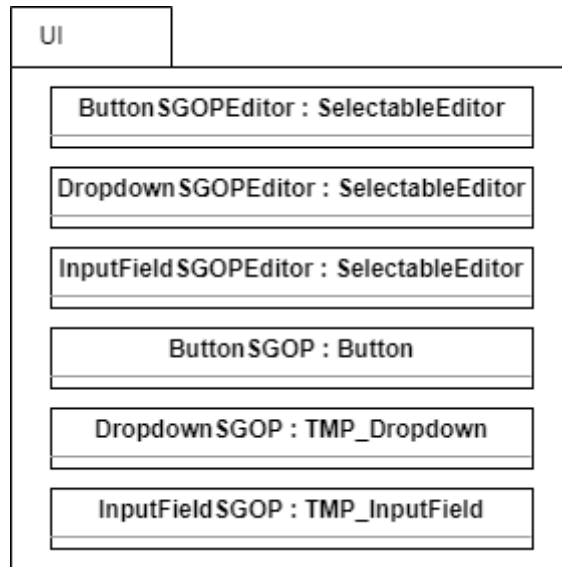


Nota. La figura muestra el paquete SGOP y la clase que contiene. Elaborado por: el autor

El paquete “UI” contiene las clases de interfaces de usuario con propiedades ampliadas para su utilización en el desarrollo de las páginas para las aplicaciones.

Figura 6

Diagrama del Paquete UI

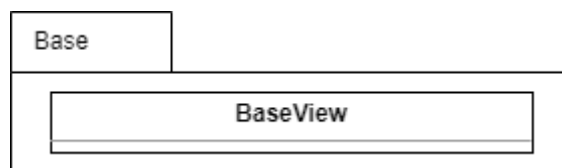


Nota. La figura muestra el paquete UI y las clases que contiene. Elaborado por: el autor

El siguiente paquete es “Base”, el cual contiene la clase “BaseView”, que se utilizará como “padre” de las clases que se creen para las vistas de la aplicación, esta clase contendrá los métodos necesarios para el correcto funcionamiento de las vistas.

Figura 7

Diagrama del Paquete Base

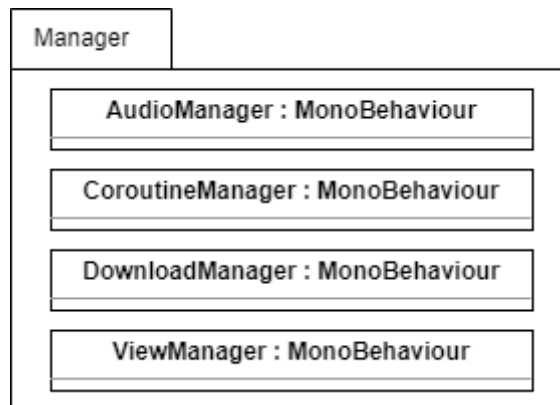


Nota. La figura muestra el paquete Base y la clase que contiene. Elaborado por: el autor

El paquete “Manager” contiene las clases que se utilizarán para implementar los métodos reiterativos a fin de simplificar el desarrollo de nuevas aplicaciones.

Figura 8

Diagrama del Paquete Manager

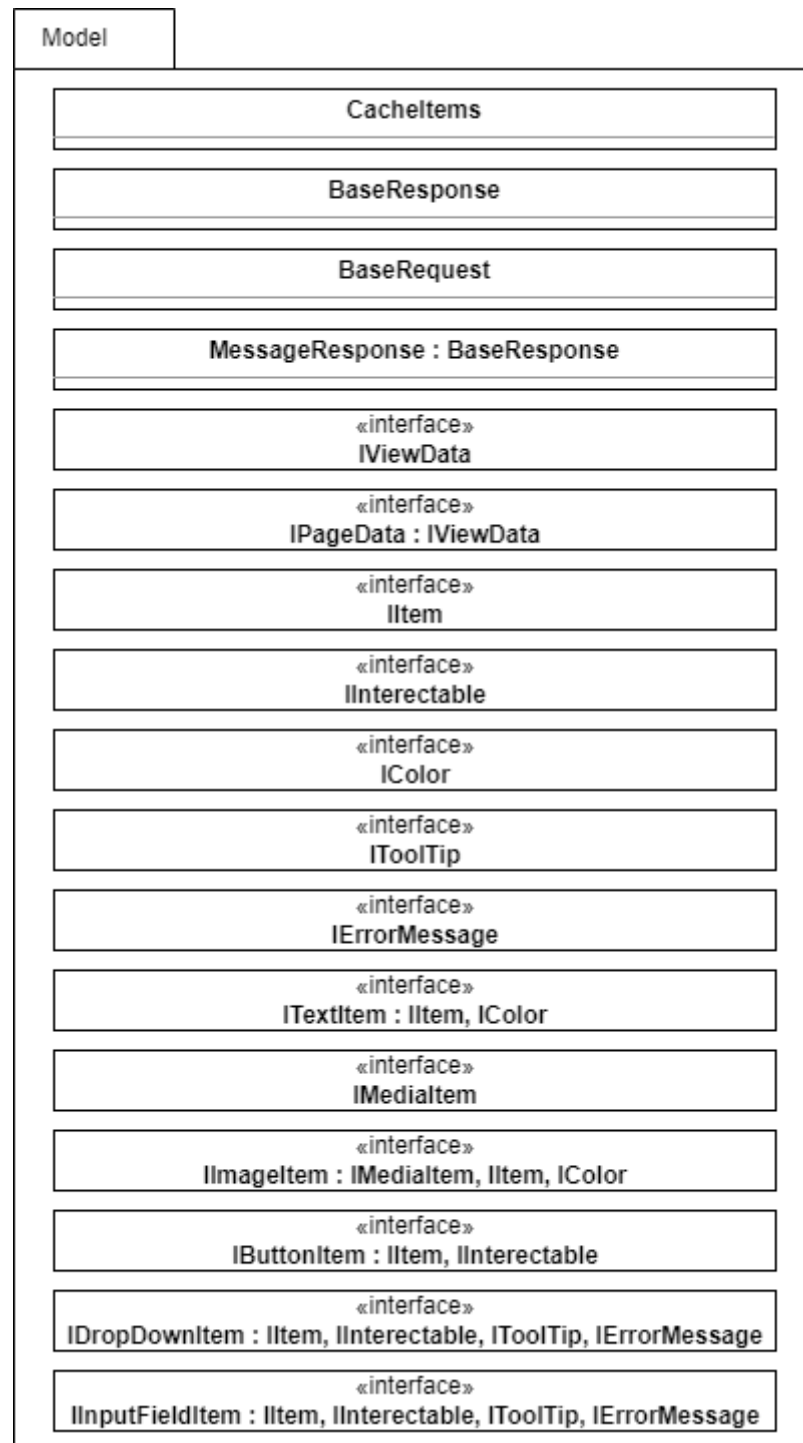


Nota. La figura muestra el paquete Manager y las clases que contiene.
Elaborado por: el autor

En el paquete "Model" se encuentran los modelos que se utilizarán para recibir la información de servicios REST, así como también para interpretar el contenido que se recibirá desde el CMS.

Figura 9

Diagrama del Paquete Model



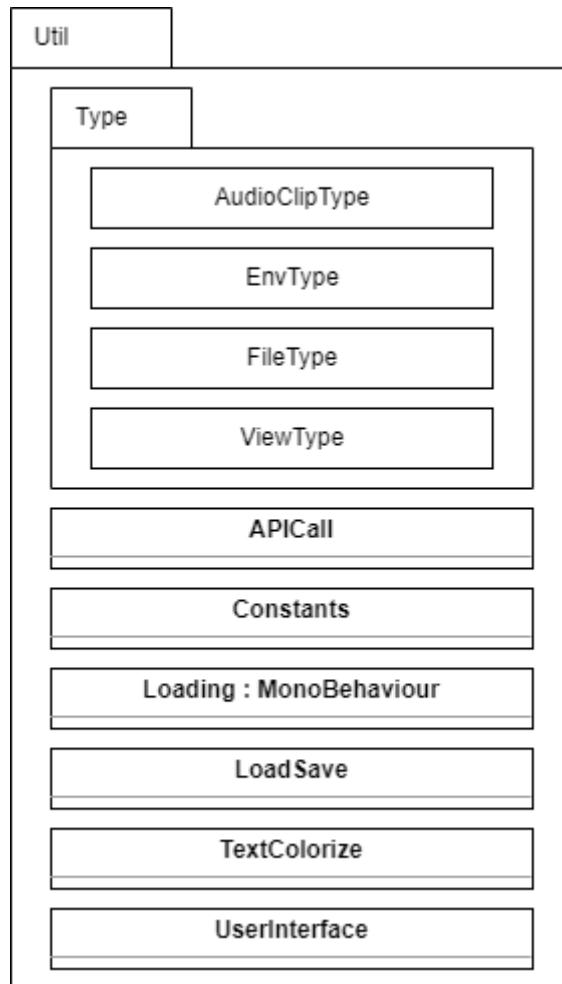
Nota. La figura muestra el paquete Model y las clases que contiene. Elaborado por: el autor

Por último, el paquete “Util” contiene un subpaquete “Type” en el cual se encuentran los Tipos de enumeración, y

también clases que implementan métodos con acciones de utilidades para el resto de los paquetes.

Figura 10

Diagrama del Paquete Util



Nota. La figura muestra el paquete Util, las clases y el sub-paquete que contiene. Elaborado por: el autor

4.3. Diseño

4.3.1. Diagramas de Clases

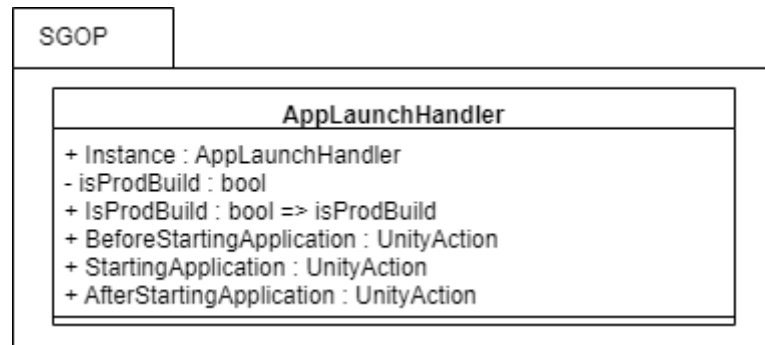
En esta sección se detallarán el diseño de las clases que se han creado para el Framework SGOP.

El primer paquete es el SGOP, el cual contiene la clase "AppLaunchHandler" el cual se encarga de contener la

configuración de inicio de la aplicación, en este caso los atributos son “isProdBuild” para la configuración de producción de la aplicación y las de tipos “UnityAction” los cuales se deben establecer para ser ejecutadas a fin de iniciar la aplicación.

Figura 11

Diagrama de Clases del Paquete SGOP

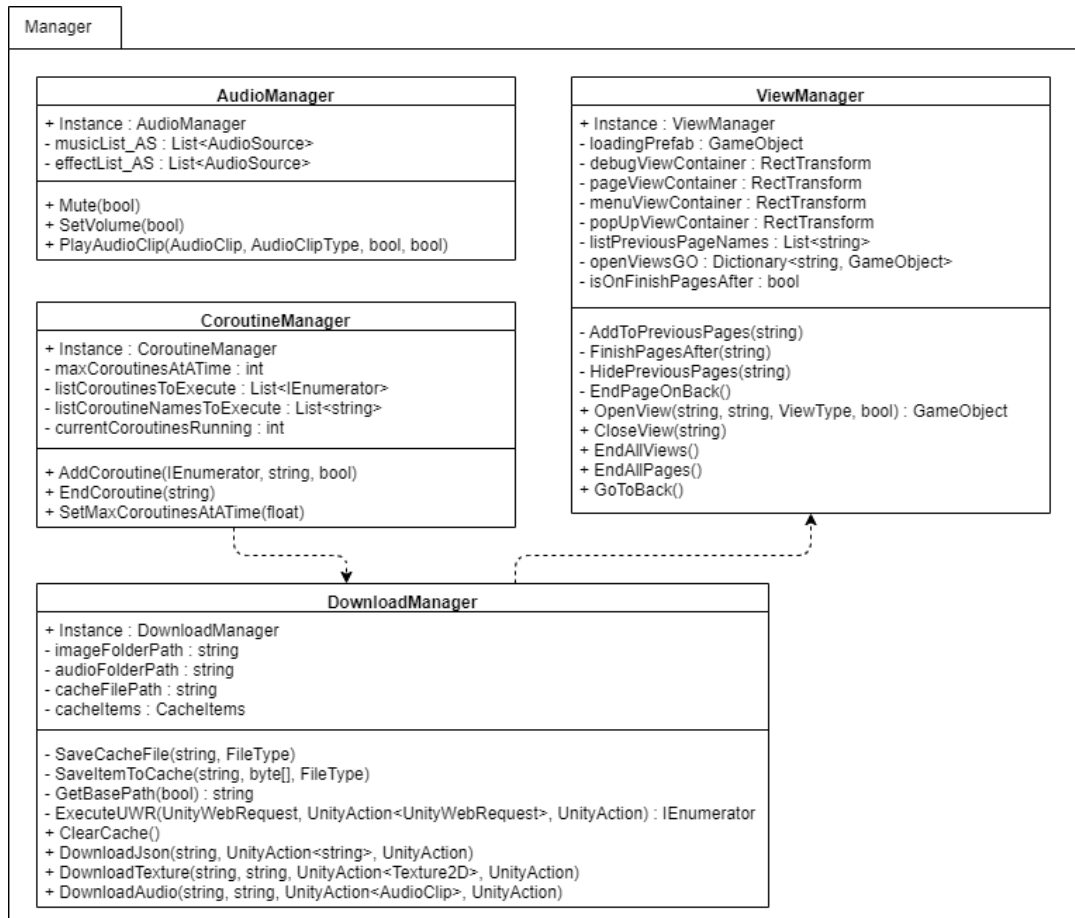


Nota. La figura muestra el detalle de la clase AppLaunchHandler. Elaborado por: el autor

En el siguiente diagrama, se detalla el paquete “Manager” que contiene las clases con las funcionalidades principales que el framework expondrá para la simplificación del desarrollo de nuevas aplicaciones.

Figura 12

Diagrama de Clases del Paquete Manager



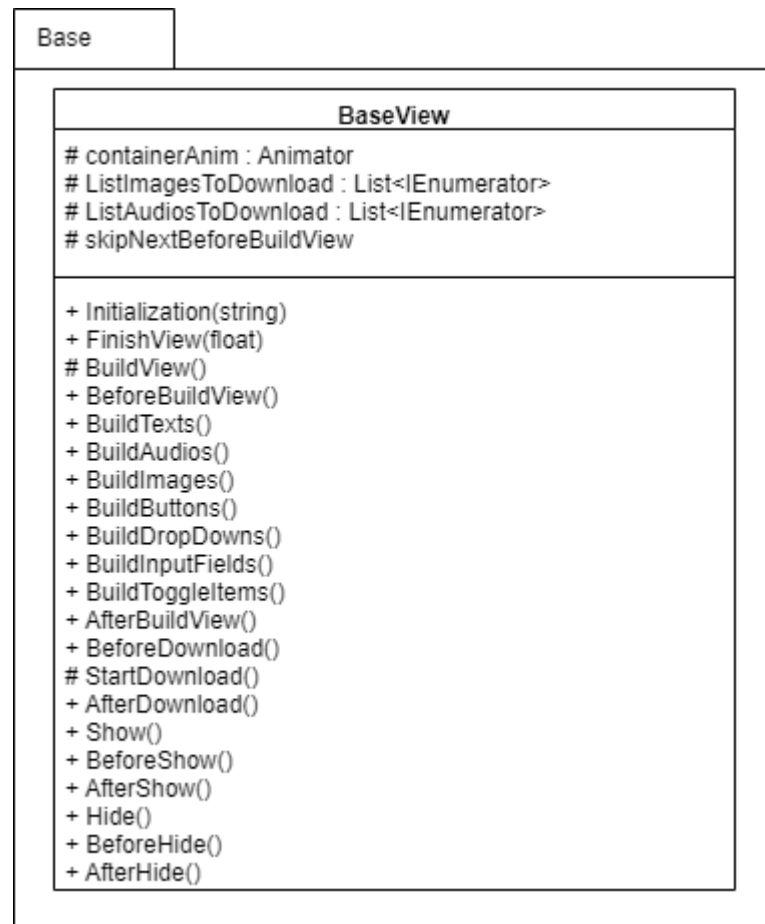
Nota. La figura muestra el detalle de las clases del paquete Manager.

Elaborado por: el autor

El paquete “Base” contiene la clase “BaseView” que será la clase base de la cual las que vistas que se crearán se extenderán para simplificar la interacción con el framework de tal manera que solo se necesite sobrescribir los métodos que se vayan a utilizar por cada vista, dejando el resto al manejo del framework.

Figura 13

Diagrama de Clases del Paquete Base

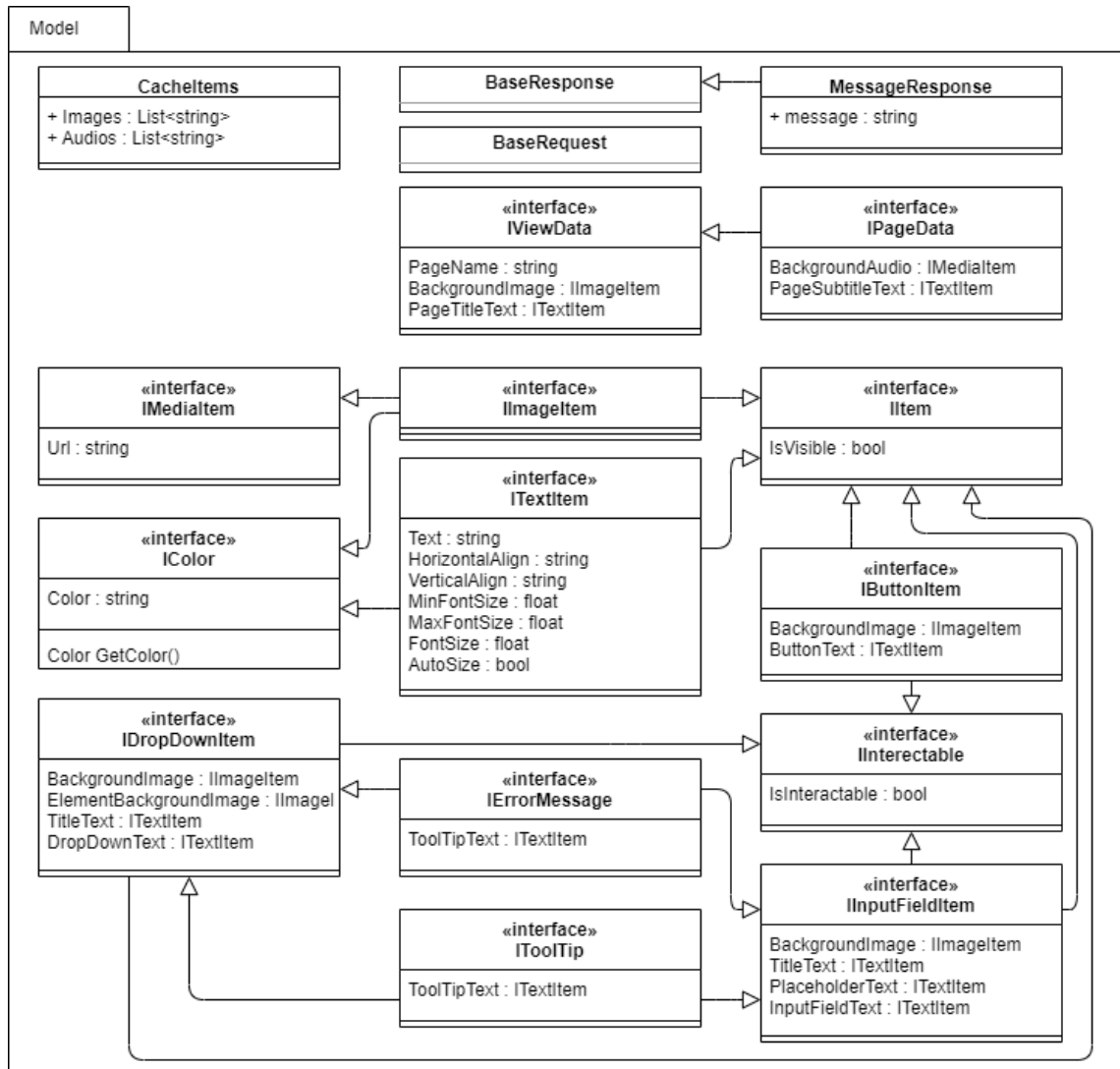


Nota. La figura muestra el detalle de las clases del paquete Base. Elaborado por: el autor

En la **Figura 14** se detalla el paquete “Model” que contiene las interfaces de los modelos que serán utilizados para el manejo de contenido proveniente desde el CMS. También estos modelos están definidos para que su uso sea el más detallado al momento de realizar el establecimiento de cada uno de estos modelos en las interfaces de usuarios, la acción de establecimiento se realizará en la clase “UserInterface”.

Figura 14

Diagrama de Clases del Paquete Model

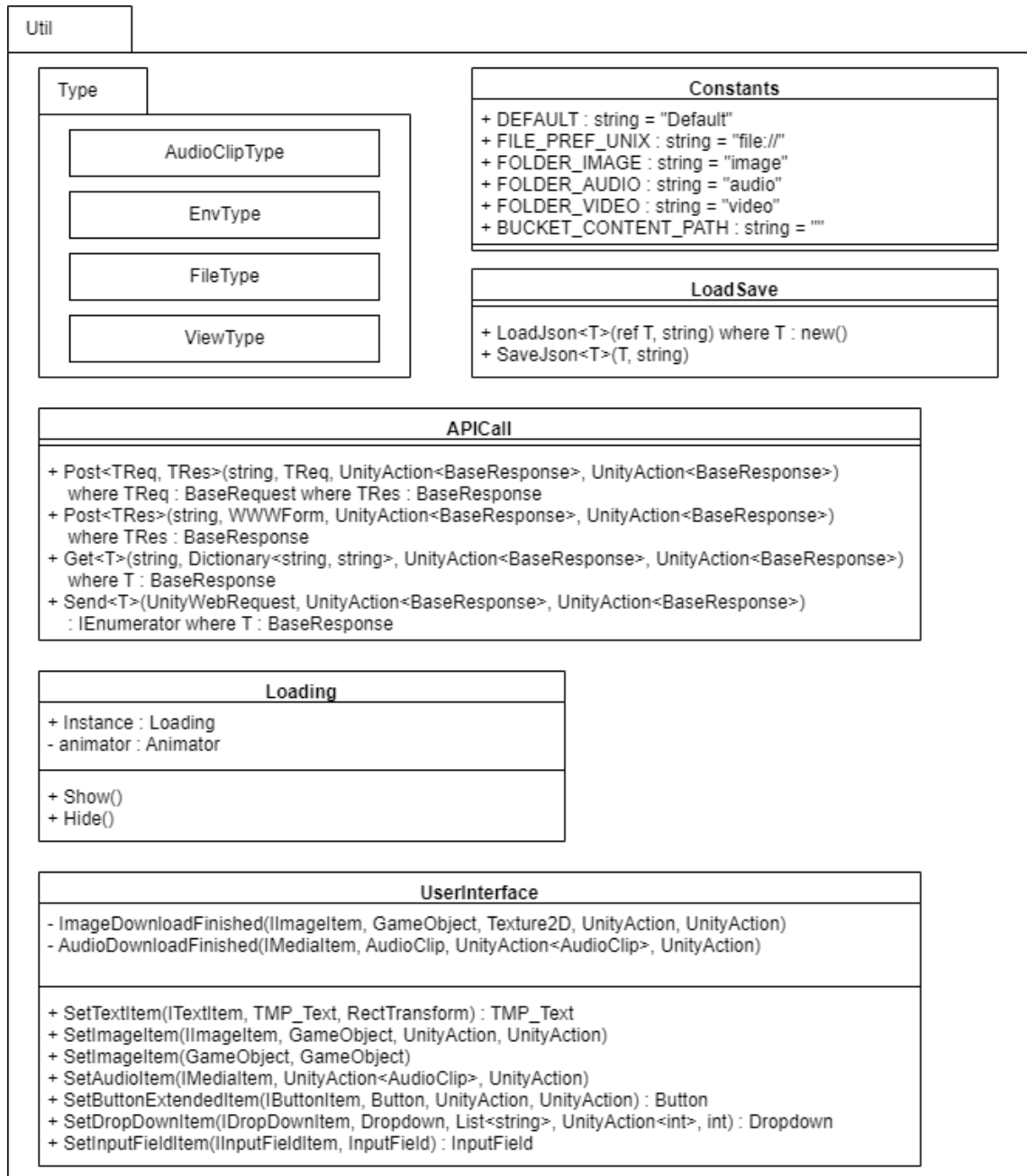


Nota. La figura muestra el detalle de las clases del paquete Model. Elaborado por: el autor

El último paquete es el Util, el contiene el subpaquete “Type” en el cual se encuentran los Tipos de enumeración, también contiene la clase “APICall” el cual contiene los métodos para simplificar las llamadas al servicio de back end y al de contenido. Además, contiene la clase “UserInterface” que expone los métodos encargados de establecer el contenido obtenido desde el servicio de contenido en las interfaces de usuario.

Figura 15

Diagrama de Clases del Paquete Util



Nota. La figura muestra el detalle de las clases del paquete Util. Elaborado por: el autor

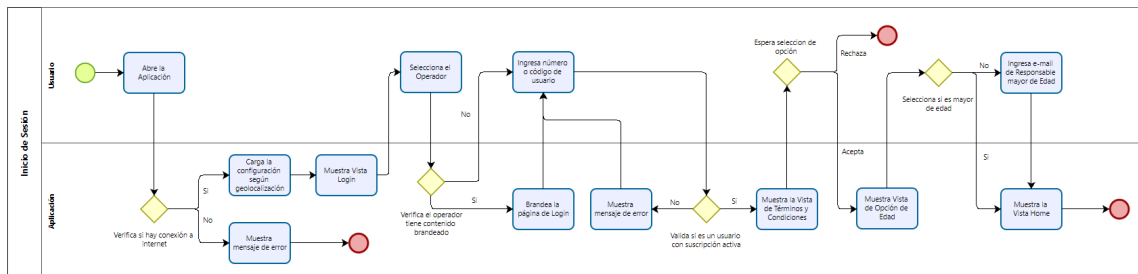
4.3.2. Modelado de Procesos

El principal proceso de la aplicación con la cual la empresa valida el uso de la aplicación es el de Inicio de Sesión, este proceso a nivel de ejecución de software se inicia con el usuario abriendo la

aplicación, lo siguiente la aplicación validará la conexión a internet y de ser validado descargará la configuración de la aplicación según la geolocalización, siguiente a esto el usuario rellena el formulario y dará clic a ingresar. La aplicación validará la suscripción del usuario y de ser validado mostrará la página de términos y condiciones, el usuario lo aceptará, seguido la aplicación mostrará la pantalla de confirmación de edad, el usuario seleccionará si es mayor de edad y rellena el campo email si es menor de edad, por último, la aplicación mostrará la página de Inicio.

Figura 16

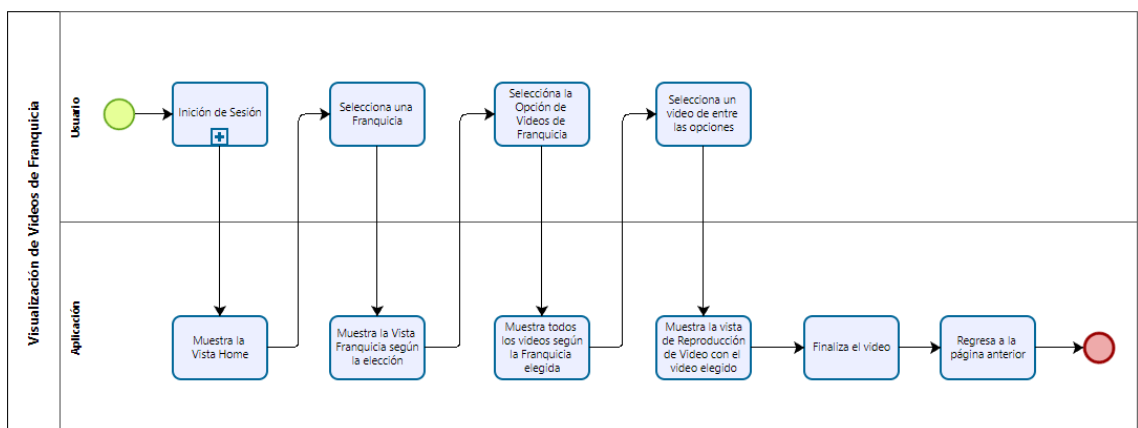
Diagrama de Inicio de Sesión



Nota. La figura muestra el diagrama del proceso por el cual el usuario interactúa con la aplicación para realizar el inicio de sesión. Elaborado por: el autor

Figura 17

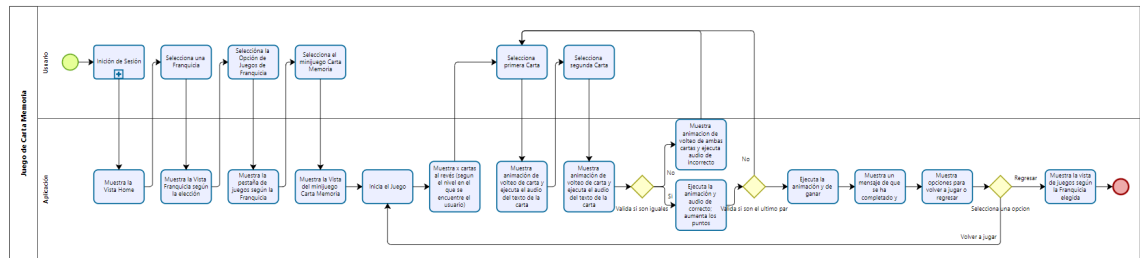
Diagrama de Visualización de Videos de Lección por Franquicia



Nota. La figura muestra el diagrama del proceso por el cual el usuario interactúa con la aplicación para visualizar videos de lección de una franquicia. Elaborado por: el autor

Figura 18

Diagrama de Juego de Carta Memoria

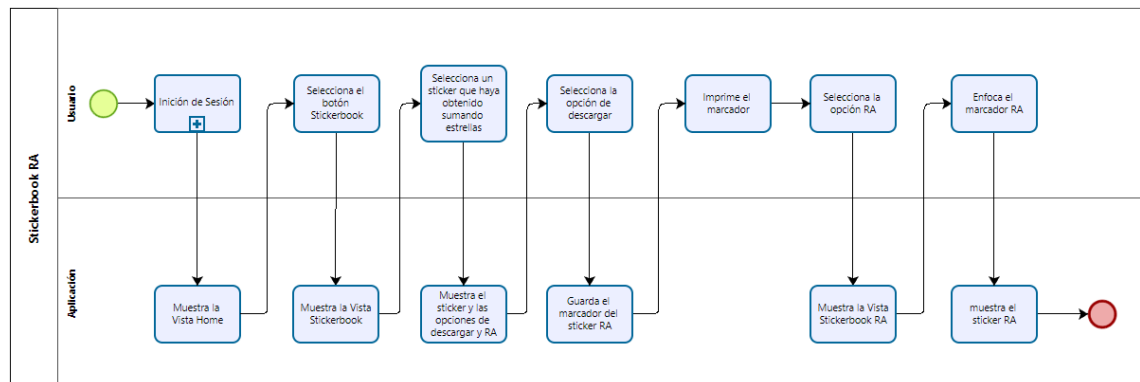


Nota. La figura muestra el diagrama del proceso por el cual el usuario interactúa con la aplicación para realizar el juego de carta memoria. Elaborado por: el autor

En el siguiente diagrama se puede visualizar el proceso por el cual el usuario puede acceder a la función de Stickerbook RA, para esto el usuario ingresa a la vista Stickerbook, tiene que haber conseguido un sticker sumando estrellas mientras juega, selecciona un sticker obtenido la aplicación lo mostrara maximizado con dos botones, Descargar Marcador y RA, si selecciona el primer botón se descargará el marcador para este sticker y con el segundo se ingresa a la vista Stickerbook RA, una vez en esta vista el usuario debe enfocar con la cámara el marcador que haya descargado.

Figura 19

Diagrama de Stickerbook RA



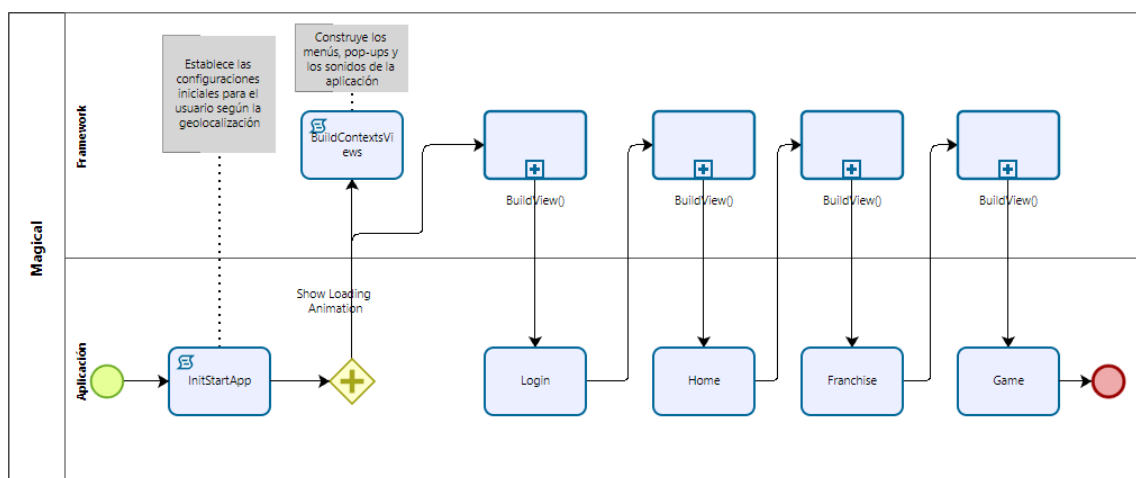
Nota. La figura muestra el diagrama del proceso por el cual el usuario interactúa con la aplicación para visualizar un sticker en realidad aumentada. Elaborado por: el autor

A continuación, se muestra como el proceso principal de la aplicación interactúa con los métodos creados en el Framework.

La **Figura 20** muestra como la aplicación ejecuta constantemente para abrir cada página el proceso reutilizable “BuildView” que está implementado en el Framework, este subproceso se detalla en la **Figura 21**.

Figura 20

Diagrama desde Inicia la Aplicación hasta Abrir un Juego

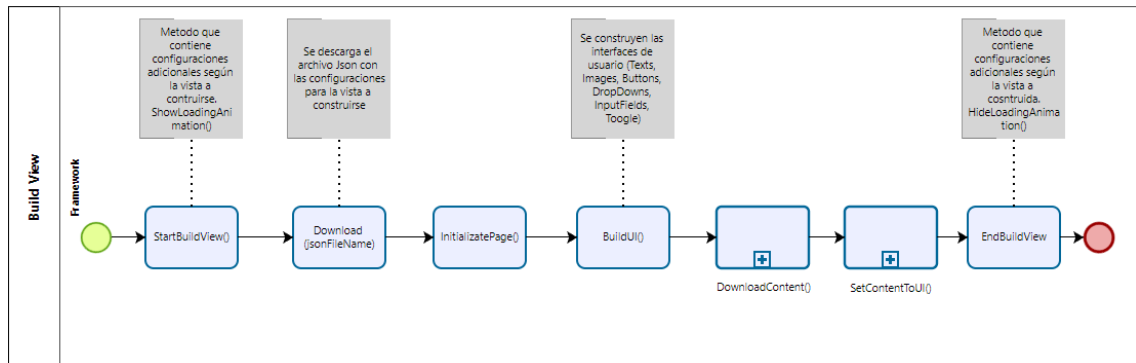


Nota. La figura muestra el diagrama del proceso por el cual la aplicación interactúa con el framework para construir las páginas. Elaborado por: el autor

La siguiente figura detalla el subproceso “BuildView”, que pertenece al framework, este a su vez consume otros subprocesos repetitivos que son el “DownloadContent” y “SetContentToUI”.

Figura 21

Diagrama del Proceso “BuildView” del Framework

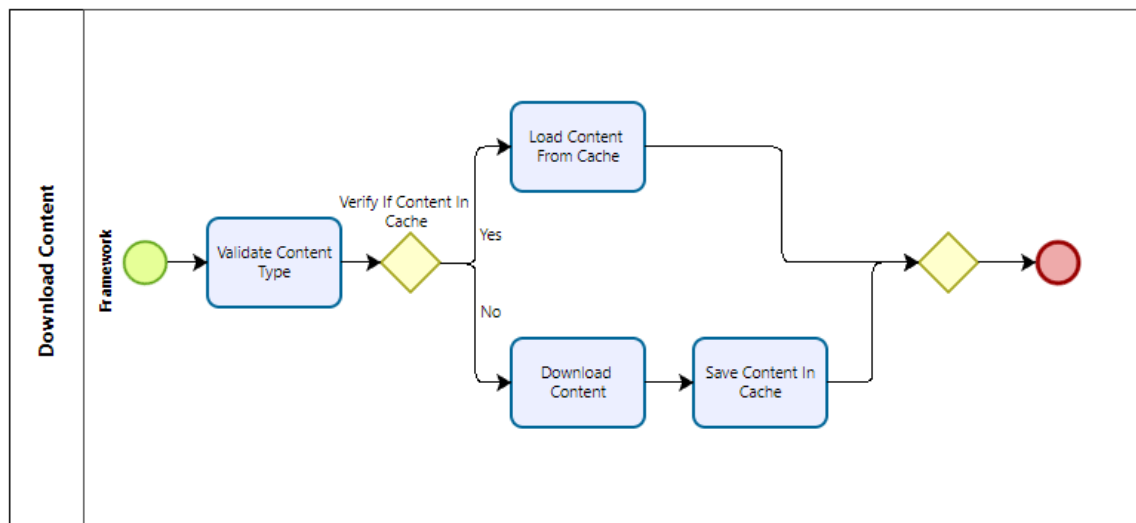


Nota. La figura muestra el diagrama del proceso por el cual el framework construye la página. Elaborado por: el autor

La **Figura 22** muestra como el subproceso repetitivo “DownloadContent” que se encarga de validar los archivos a descargar y manejar si ya han sido descargados o descargarlos y guardarlos en cache.

Figura 22

Diagrama del Proceso “Downloadcontent” del Framework



Nota. La figura muestra el diagrama del proceso por el cual el framework descarga el contenido. Elaborado por: el autor

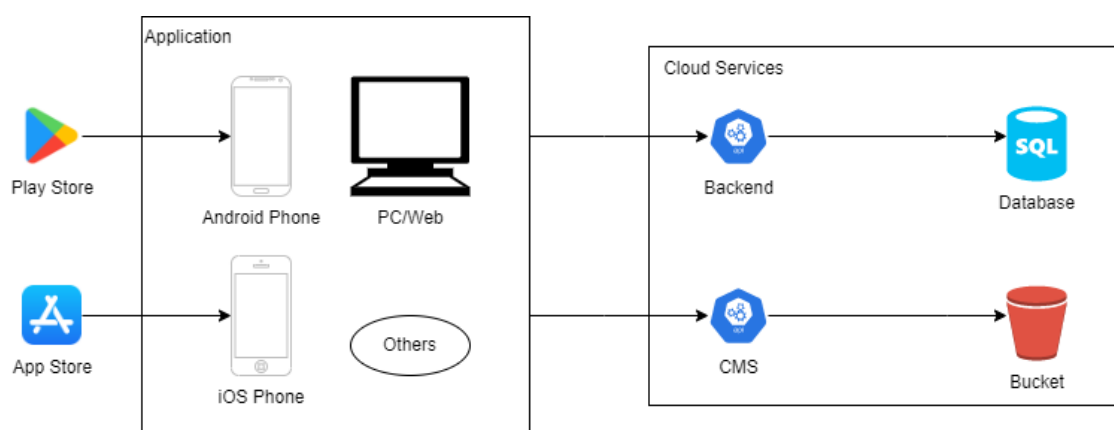
4.3.3. Modelo del Framework

Resultado del análisis de los proyectos concedidos por la empresa se propone un modelo de desarrollo de aplicaciones que implemente el framework, además se llegó a la conclusión de también incluir los componentes físicos de servicios de back end y de gestor de contenido. El servicio de back end es un componente casi obligatorio en la mayoría de los proyectos de software y el gestor de contenido se añade ante la necesidad de localizar la aplicación a nivel mundial con contenido acorde a cada región.

Este modelo de arquitectura para videojuegos formativos propuesto (ver **Figura 23**), parte de una aplicación que ha sido compilada y distribuida, ya sea en las tiendas oficiales de Android y iOS, o para pc, web u otros. Las aplicaciones deberán conectarse con un servicio back end para el manejo de la información del usuario y de la aplicación, este servicio debe estar conectado con su propia base datos separada; por otro lado, un servicio de gestor de contenido que proveerá del contenido multimedia que se utilizará en la aplicación, este servicio almacenará el contenido en un servicio de contenedor (Bucket).

Figura 23

Propuesta de Modelo de Arquitectura para Videojuegos Formativos



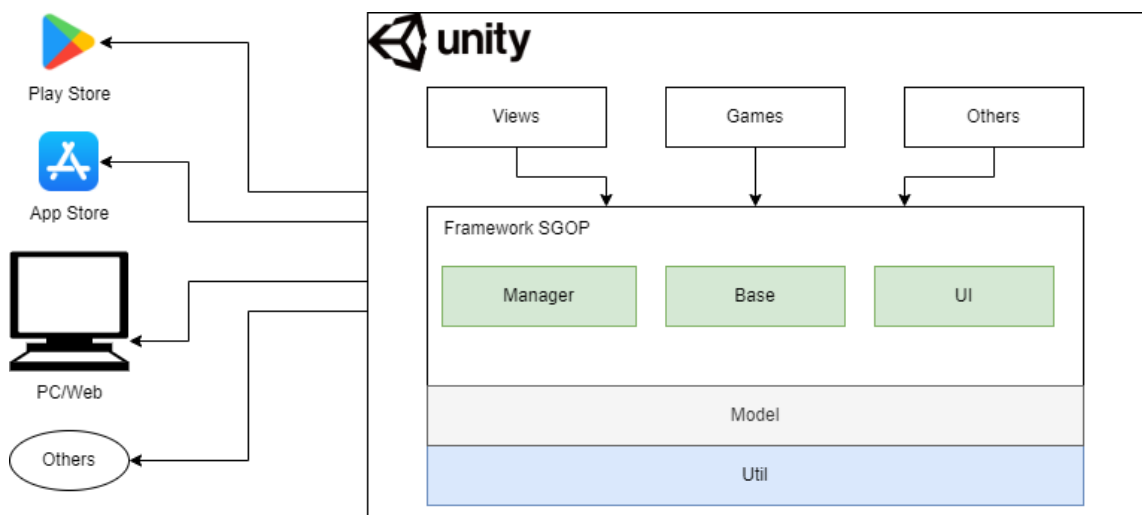
Nota. La figura muestra el modelo de la arquitectura de videojuegos formativos en su despliegue. Elaborado por: el autor

Por la parte del desarrollo de los videojuegos formativos, la propuesta presenta que el motor de videojuegos

sea Unity, en el cual se descargue el Framework SGOP que incluye los paquetes de desarrollo “Base”, “Manager”, “UI” y “Util”, con los cuales el desarrollo de vistas, juegos y otros se realicen utilizando los componentes que exponen las clases de los paquetes del framework. De esta manera, utilizando Unity se podrá compilar los aplicativos para las plataformas objetivo del proyecto en cuestión. En la **Figura 24** se puede ver a detalle el modelo propuesto.

Figura 24

Propuesta de Modelo de Desarrollo de Software de Videojuegos Formativos Utilizando el Framework SGOP

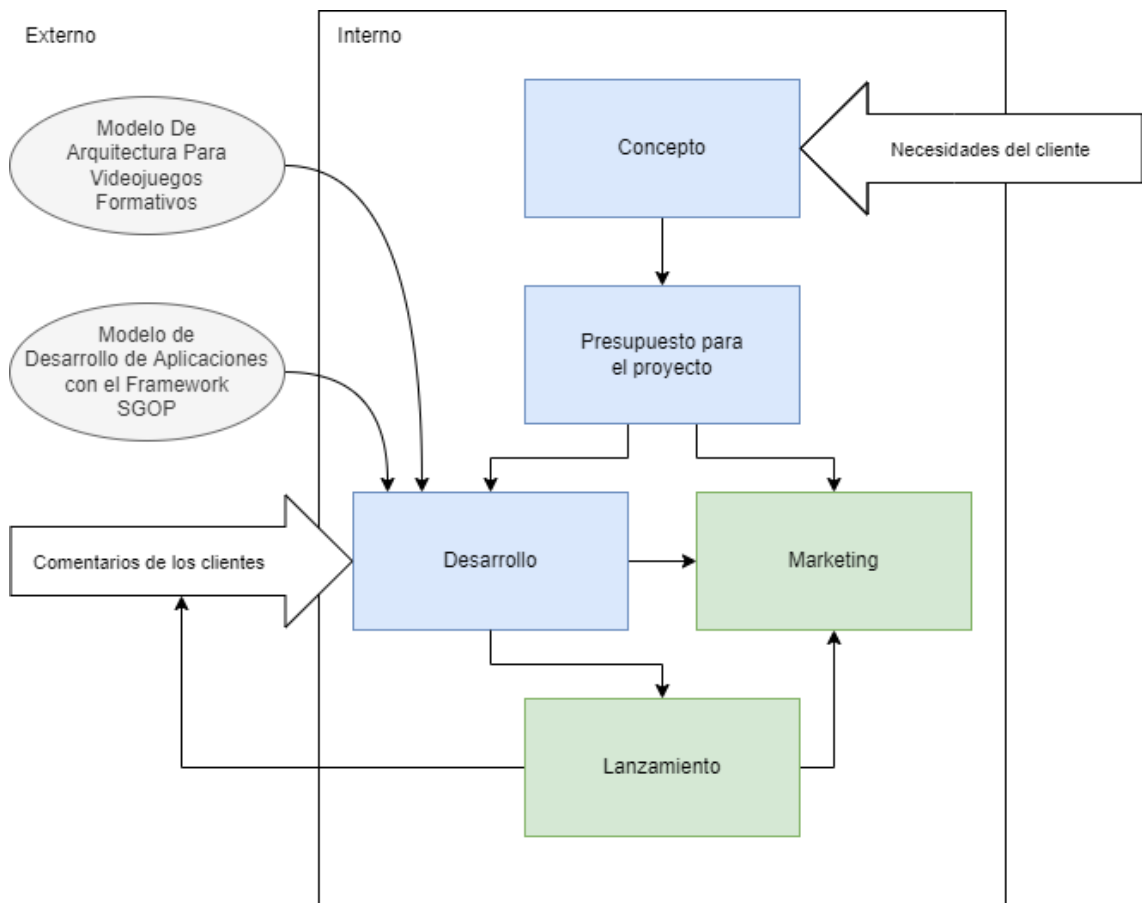


Nota. La figura muestra el modelo del desarrollo de videojuegos formativos utilizando el Framework SGOP. Elaborado por: el autor

Por último, acorde a lo presentado por Ahmad et al. (2017) se ha desarrollado una modificación a su modelo holístico para el desarrollo de proyectos de videojuegos, cambiando fases en nombre y quitando algunos componentes que no se ven necesarios en proyectos de videojuegos formativos, a continuación, se muestra el modelo presentado.

Figura 25

Propuesta de Modelo de Desarrollo de Proyectos de Videojuegos Formativos Utilizando el Framework SGOP



Nota. La figura muestra el modelo de desarrollo de videojuegos formativos y cada una de sus fases. Elaborado por: el autor

a) Fase 1: Concepto

La etapa de concepto es el comienzo del marco e incluye las estrategias generales de desarrollo de productos que utilizan los desarrolladores de videojuegos y la consideración de diferentes géneros de juegos. El primer aspecto que se discutirá es el Ciclo de Vida de Desarrollo de Software (por sus siglas en inglés Software Development Life Cycle, SDLC), para ello se debe definir la metodología con la que se trabajará el proyecto (cascada o ágil). El siguiente aspecto que se discutirá será las características del

videojuego, para ello se tomará en cuenta las necesidades del cliente (de ser un proyecto licitado) o las necesidades del público objetivo.

b) Fase 2: Presupuesto Para El Proyecto

En esta etapa se analiza el presupuesto para el desarrollo del proyecto. Acorde a lo propuesto por Ahmad et al., se sugiere el Análisis De Costos Del Ciclo De Vida (por sus siglas en inglés Life Cycle Cost Analysis, LCCA) para determinar los tipos de costos en los que se incurrirá para desarrollar el videojuego desde la etapa de concepto hasta el lanzamiento posterior.

Para llegar al LCCA más preciso, el desarrollador debe tener en cuenta tres aspectos que afectarán el costo de desarrollar el juego: El SDLC que se analizó en la fase anterior, el ciclo de vida de las plataformas o consolas objetivo, y las ventas y utilidades objetivo.

En lo que respecta a la metodología ya sea que se opte por la cascada o por ágil, habrá cambios en la estructura de la empresa y en las personas, que se deben realizar para incorporar este modelo, y eso eventualmente se traducirá en costos para el desarrollador.

En lo que respecta a las plataformas o consolas objetivo, se debe determinar el estado del ciclo de vida en caso de ser para consolas o en caso de ser dispositivos móviles se deberá tomar en cuenta el modelo de ingresos que tendrá la aplicación desarrollada, si será por suscripción, por venta de la aplicación o por publicidad.

Finalmente, las ventas objetivo y las ganancias que espera el desarrollador también afectan los costos de desarrollo.

c) Fase 3: Desarrollo

Para esta fase según lo definido en el modelo de arquitectura para proyectos se utilizará Unity junto con el Framework SGOP, el cual

proveerá el código fuente que podrán reutilizarle simplificando el proceso de desarrollo.

d) Fase 4: Marketing

Considerar un presupuesto de marketing hoy en día es un paso esencial para sobrevivir en el mercado actual, un videojuego necesita toda la exposición que pueda obtener para llegar a los consumidores objetivo. El desarrollador debe considerar cuidadosamente el presupuesto de marketing utilizando el LCCA en la etapa del presupuesto de desarrollo del proceso.

Los críticos profesionales tienen acceso a los juegos antes de su lanzamiento al mercado y publican sus reseñas (formatos escritos o de video) a los consumidores. El desarrollador de videojuegos se pone en contacto con estos críticos profesionales antes del lanzamiento del juego y les pide que tengan acceso anticipado al juego. Esto viene con la condición de que los revisores publiquen sus reseñas una vez que el desarrollador les dé el visto bueno. El período entre la recepción del acceso anticipado al juego por parte de los revisores y el momento en que se publica la revisión real después de la aprobación del desarrollador se denomina período de embargo. Como la mayoría de los consumidores que juegan videojuegos esperan las reseñas, está claro que una reseña positiva de críticos profesionales sobre un determinado videojuego generará más ventas para el desarrollador.

e) Fase 5: Lanzamiento

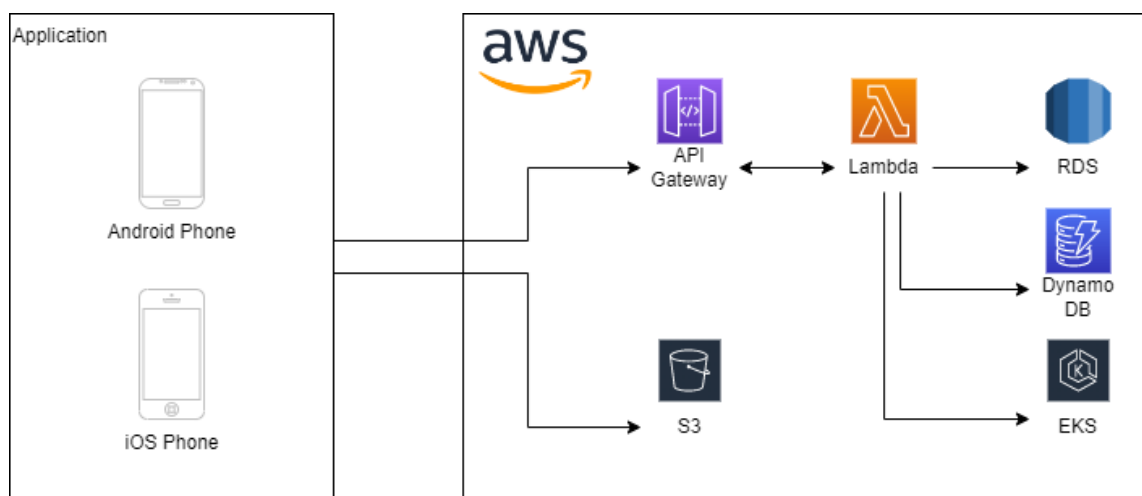
En la última fase se distribuye el juego al público, según sea el tipo de proyecto se deberá tomar la decisión de continuar con el mantenimiento del proyecto, generar nuevo contenido estacional o de la marca de publicidad que se pudiera generar. También se recuperan los comentarios de los clientes a fin de analizar si incluir nuevas características o corregir problemas surgidos durante el lanzamiento del proyecto.

4.3.4. Arquitectura

La arquitectura de la aplicación para la empresa tendrá una infraestructura desarrollada en AWS, utilizando Lambdas para la ejecución del back end, el cual se conectará con la base de datos Dynamo DB y AWS EKS que proveerá servicios extras. Estos lambdas serán expuestos mediante AWS API Gateway. Para el contenido, la aplicación consumirá desde AWS S3.

Figura 26

Arquitectura de la Aplicación de la Empresa



Nota. La figura muestra el modelo de la arquitectura de la aplicación diseñada para la empresa. Elaborado por: el autor

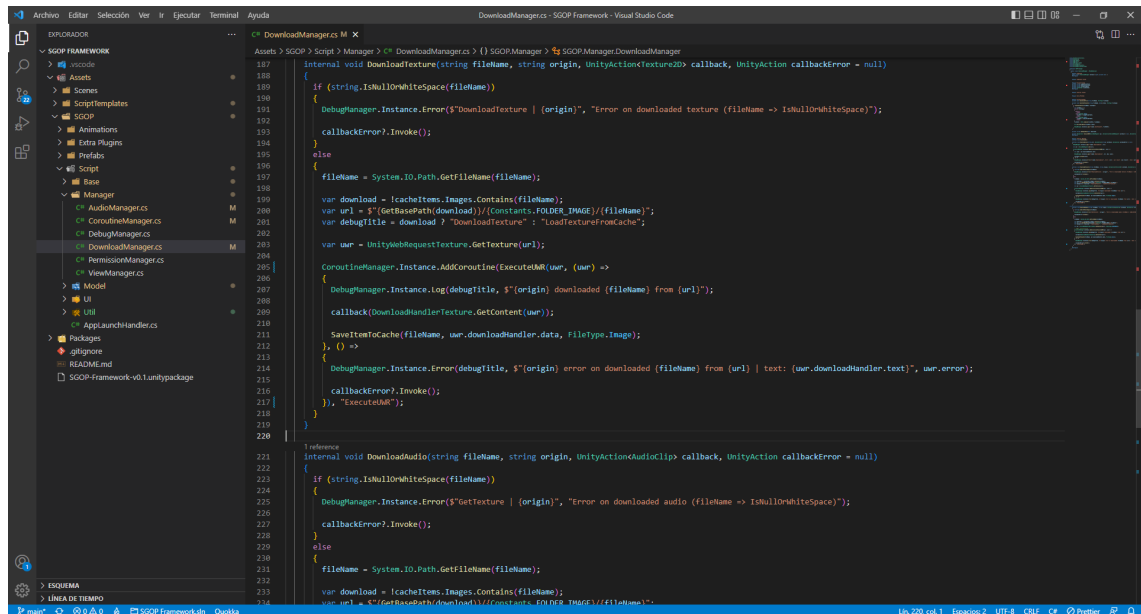
4.4. Desarrollo

En esta sección se muestra algunas partes del desarrollo del Framework:

En la **Figura 27**, se muestra como se ha implementado el método “GetTexture” de la clase “DownloadManager”, este método se encarga de descargar las imágenes.

Figura 27

Captura del Método “DownloadTexture” de la Clase “DownloadManager”



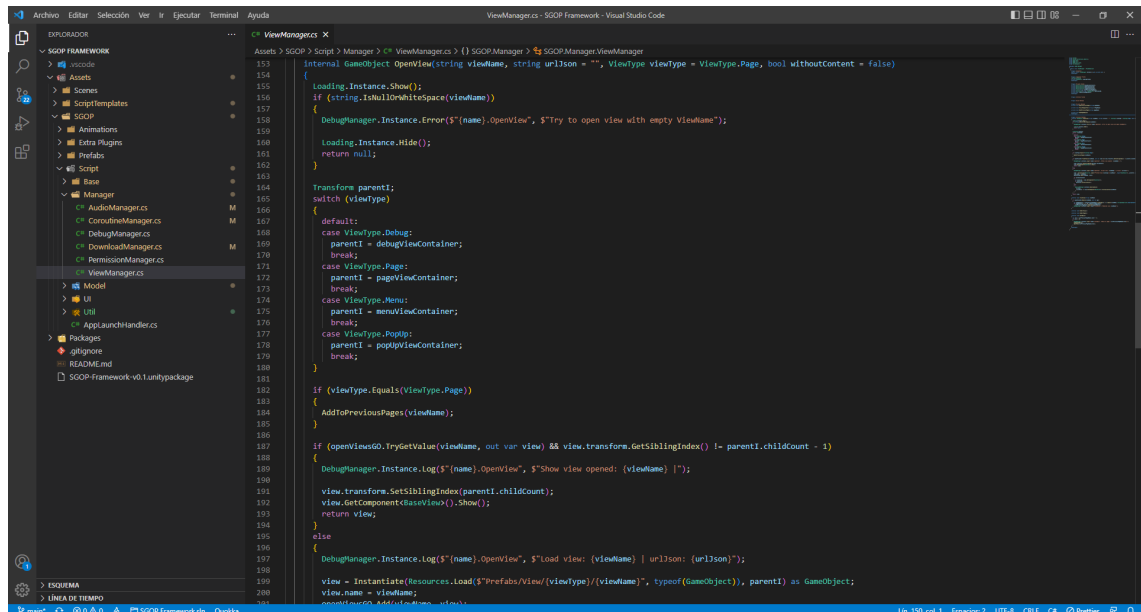
```
Assets > SGOP > Script > Manager > () SGOP.Manager > SGOP.Manager.DownloadManager
187 internal void DownloadTexture(string fileName, string origin, UnityAction<Texture2D> callback, UnityAction callbackError = null)
188 {
189     if (string.IsNullOrEmpty(fileName))
190     {
191         DebugManager.Instance.Error($"DownloadTexture | {origin}", "Error on downloaded texture (fileName => IsNullOrEmpty)");
192     }
193     callbackError?.Invoke();
194     else
195     {
196         fileName = System.IO.Path.GetFileName(fileName);
197         var download = !cacheItems.Images.Contains(fileName);
198         var url = $"(GetBasePath(download))/(Constants.FOLDER_IMAGE)/(fileName)";
199         var debugTitle = download ? "DownloadTexture" : "LoadTextureFromCache";
200         var uwr = UnityWebRequestTexture.GetTexture(url);
201         CoroutineManager.Instance.AddCoroutine(ExecuteDMR(uwr, (uwr) =>
202         {
203             DebugManager.Instance.Log(debugTitle, $"(origin) downloaded (fileName) from (url)");
204             callback(DownloadHandlerTexture.GetContent(uwr));
205             SaveTextToCache(fileName, uwr.downloadHandler.data, FileType.Image);
206         }, () =>
207         {
208             DebugManager.Instance.Error(debugTitle, $"(origin) error on downloaded (fileName) from (url) | text: (uwr.downloadHandler.text)", uwr.error);
209             callbackError?.Invoke();
210         }));
211     }
212 }
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

Nota. La figura muestra el método “DownloadTexture” de la hoja de código de la clase “DownloadManager” desarrollado para el framework. Elaborado por: el autor

En la siguiente figura se muestra la implementación del método “OpenView” de la clase “ViewManager”, el cual se encarga de comprobar que tipo de vista se está abriendo, se agrega la vista para a la pila de páginas previas, se valida si ha sido abierta anteriormente para mostrarlo, en caso contrario crea una nueva instancia del prefab de la vista a mostrarse.

Figura 28

Captura del Método “OpenView” de la Clase “ViewManager”



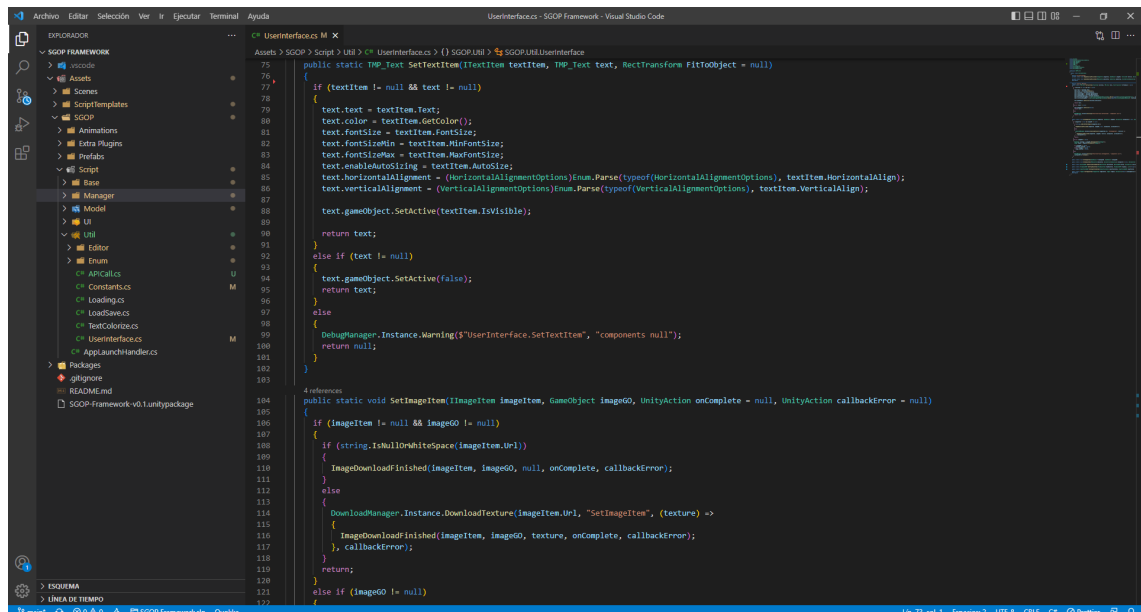
```
Assets > SGOP > Script > Manager > ViewManager.cs > {} SGOP Manager > SGOP Manager > ViewManager
153 Internal_GameObject OpenView(string viewName, string urlJson = "", ViewType viewType = ViewType.Page, bool withoutContent = false)
154 {
155     Loading_Instance.Show();
156     if (string.IsNullOrEmpty(viewName))
157     {
158         DebugManager.Instance.Error($"{name}.OpenView", $"Try to open view with empty ViewName");
159     }
160     Loading_Instance.Hide();
161     return null;
162 }
163
164 Transform parent;
165
166 switch (viewType)
167 {
168     default:
169         case ViewType.Debug:
170             parent = debugViewContainer;
171             break;
172         case ViewType.Page:
173             parent = pageViewContainer;
174             break;
175         case ViewType.Memo:
176             parent = memoViewContainer;
177             break;
178         case ViewType.Popup:
179             parent = popupViewContainer;
180             break;
181 }
182 if (viewType.Equals(ViewType.Page))
183 {
184     AddTopPreviousPages(viewName);
185 }
186 if (openViewSO.TryGetValue(viewName, out var view) && view.transform.GetSiblingIndex() != parent.childCount - 1)
187 {
188     DebugManager.Instance.Log($"{name}.OpenView", $"Show view opened: {viewName}");
189 }
190 view.transform.SetSiblingIndex(parent.childCount);
191 view.GetComponent<ClassView>().Show();
192 return view;
193 }
194 else
195 {
196     DebugManager.Instance.Log($"{name}.OpenView", $"Load view: {viewName} | urlJson: {urlJson}");
197 }
198 view = Instantiate(Resources.Load($"Prefabs/View/{viewType}/{viewName}", typeof(GameObject)), parent) as GameObject;
199 view.name = viewName;
200 parent.transform.AddChild(view);
```

Nota. La figura muestra el método “OpenView” de la hoja de código de la clase “ViewManager” desarrollado para el framework. Elaborado por: el autor

En la siguiente figura se muestra la implementación de los métodos “SetTextItem” y “SetImageItem” de la clase “UserInterface”. El método “SetTextItem” se encarga de configurar los objetos “TMP_Text” que son las instancias de UI de los objetos de textos en las escenas de las vistas. El método “SetImageItem” se encarga de crear una instancia de Corrutina para la descarga del archivo y una vez terminado de descargar ejecutar el método “ImageDownloadFinished” para establecer las imágenes en los objetos de UI.

Figura 29

Captura de los Métodos “SetTextItem” y “SetImageItem” de la Clase “UserInterface”



```
Assets > SGOP > Script > UI > UserInterface > SGOPLUI > SGOPLUIUserInterface
75 public static TMP_Text SetTextItem(ITextItem textItem, TMP_Text text, RectTransform FitToObject = null)
76 {
77     if (textItem != null && text != null)
78     {
79         text.text = textItem.Text;
80         text.color = textItem.GetColor();
81         text.fontSize = textItem.FontSize;
82         text.fontSizeMin = textItem.MinFontSize;
83         text.fontSizeMax = textItem.MaxFontSize;
84         text.enableAutoSizing = textItem.AutoSizing;
85         text.horizontalAlignment = (HorizontalAlignmentOptions)Enum.Parse(typeof(HorizontalAlignmentOptions), textItem.HorizontalAlign);
86         text.verticalAlignment = (VerticalAlignmentOptions)Enum.Parse(typeof(VerticalAlignmentOptions), textItem.VerticalAlign);
87         text.gameObject.SetActive(textItem.IsVisible);
88     }
89     return text;
90 }
91 }
92 else if (text != null)
93 {
94     text.gameObject.SetActive(false);
95     return text;
96 }
97 else
98 {
99     DebugManager.Instance.Warning($"UserInterface.SetTextItem, \"components null\"");
100     return null;
101 }
102 }
103 }
104 }
105 }
106 }
107 }
108 }
109 }
110 }
111 }
112 }
113 }
114 }
115 }
116 }
117 }
118 }
119 }
120 }
121 }
122 }
123 }
124 }
```

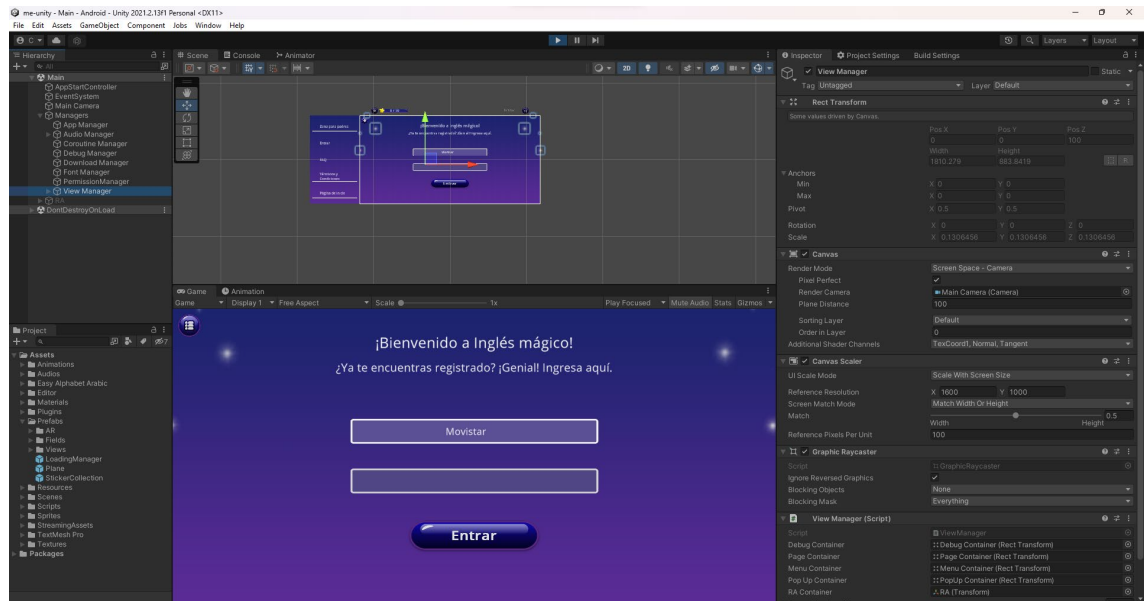
Nota. La figura muestra los métodos “SetTextItem” y “SetImageItem” de la hoja de código de la clase “UserInterface” desarrollados para el framework. Elaborado por: el autor

Con respecto del desarrollo de las páginas de la aplicación para la empresa a continuación se muestran algunas imágenes.

En la **Figura 30** se puede ver la página de Login que se ha desarrollado para la aplicación de la empresa con el formulario de ingreso del usuario.

Figura 30

Página de Login en la Herramienta Unity

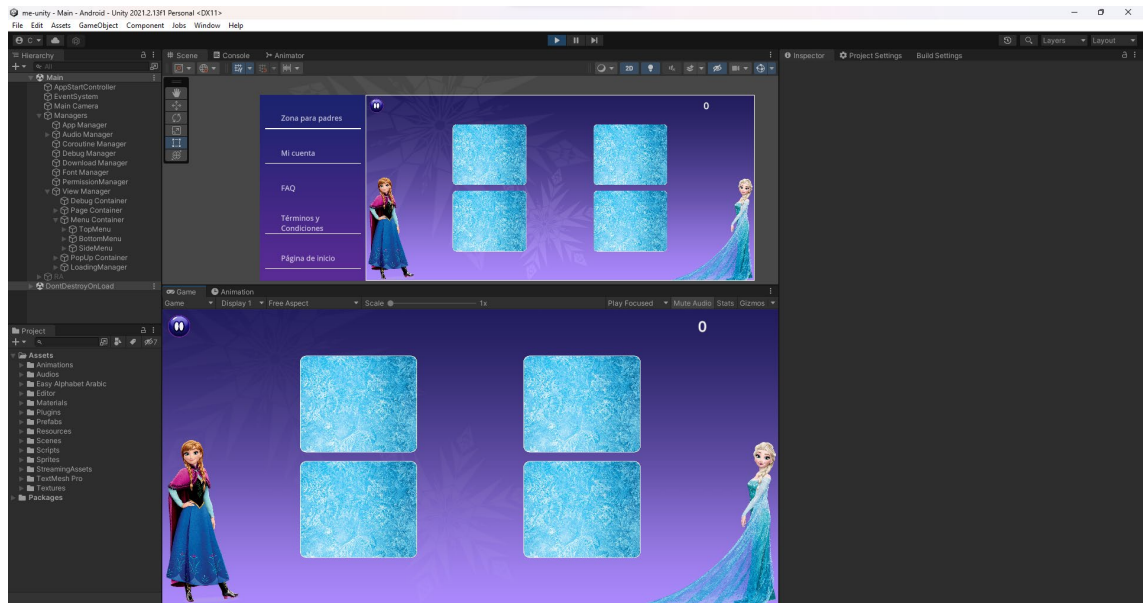


Nota. La figura muestra el desarrollo de la vista de Login en la herramienta Unity. Elaborado por: el autor

En la **Figura 31** se muestra la página de Minijuego Carta Memoria que se ha desarrollado para la aplicación en donde el usuario puede jugar y ganar estrellas.

Figura 31

Página de Minijuego Carta Memoria en la Herramienta Unity

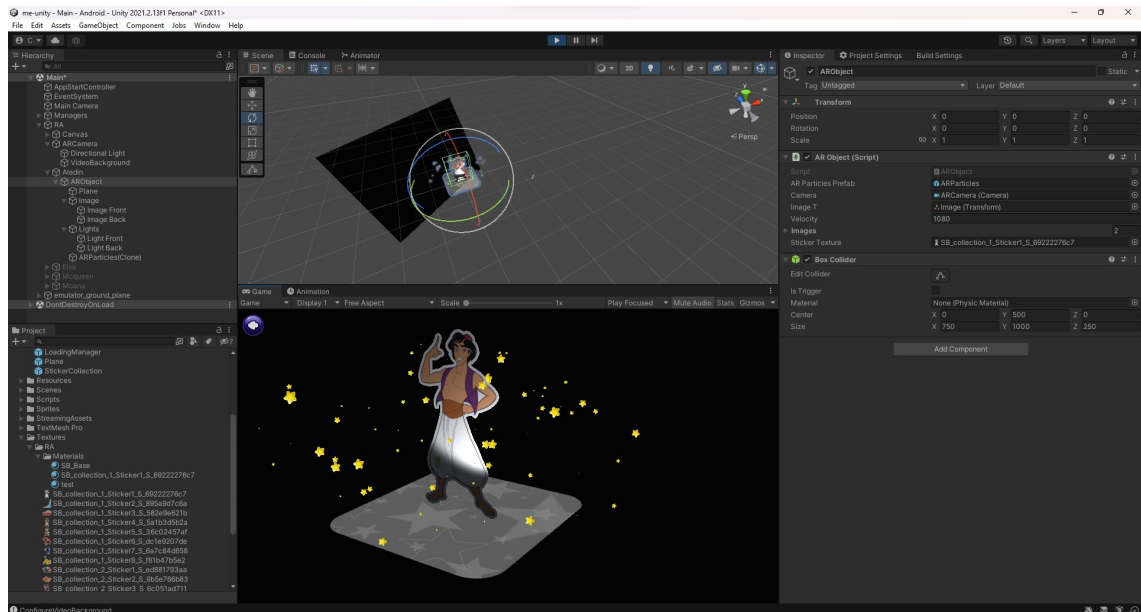


Nota. La figura muestra el desarrollo del minijuego Carta Memoria en la herramienta Unity. Elaborado por: el autor

En la **Figura 32** se puede visualizar el desarrollo de Stickerbook RA en la herramienta Unity.

Figura 32

Desarrollo de Stickerbook RA en la Herramienta Unity



Nota. La figura muestra el desarrollo de la vista Stickerbook RA en la herramienta Unity. Elaborado por: el autor

4.5. Plan de Pruebas

4.5.1. Casos de Pruebas

A continuación, se detallan los casos de prueba para la aplicación de la empresa, se han descrito 4 casos de prueba en los cuales se describe la descripción, precondiciones, el flujo y el resultado esperado de su ejecución.

En la **Tabla 30** se muestra el detalle del CP para ingresar al sistema de la aplicación, para que el CP sea posible de realizarse se requiere que el usuario tenga un usuario con suscripción activa y estar en un país que cuente con servicio de la aplicación. El flujo de la aplicación está detallado para un nuevo usuario, el cual no tenga sesión anterior iniciada. Para un usuario que ya haya ingresado no se llevará a cabo los puntos de TyC y confirmación de edad, con lo cual la aplicación mostrará directamente la página de Home.

Tabla 30**CP-01 Realizar Ingreso al Sistema**

Caso de Prueba	N.º	01	Cod:	CP-01	Realizar ingreso al sistema
Descripción:					
El propósito de esta CP es probar la funcionalidad para realizar el inicio de sesión					
Precondiciones:					
<ul style="list-style-type: none">• Tener un numero de usuario con suscripción activa.• Estar en un país que cuente con servicio.					
Flujo:					
<ul style="list-style-type: none">• La aplicación muestra la vista Login con el formulario de Ingreso.• El usuario selecciona su operador e ingresará su MSISDN.• La aplicación validará los datos del usuario.• La aplicación muestra la vista Términos y Condiciones.• El usuario acepta los TyC.• La aplicación muestra la vista Confirmación de Edad.• El usuario selecciona la opción según su edad.• La aplicación guarda los datos del usuario.• La aplicación muestra la vista Home.					
Resultado Esperado:					
<ul style="list-style-type: none">• La aplicación mostrará la vista Home.					
Resultado Obtenido:					
-					

Elaborado por: el autor

En la **Tabla 31** se detalla el CP para que un usuario visualice un video de lección según la franquicia de su elección.

Tabla 31

CP-02 Ver Video de Lección

Caso de Prueba	N.º	02	Cod:	CP-02	Ver video de lección
Descripción:					
El propósito de esta CP es probar la funcionalidad para de ver video de lección.					
Precondiciones:					
<ul style="list-style-type: none">• Tener un numero de usuario con suscripción activa.• Estar en un país que cuente con servicio.					
Flujo:					
<ul style="list-style-type: none">• La aplicación muestra la vista Home.• El usuario selecciona una franquicia.• La aplicación muestra la vista Franquicia con el contenido de la elección.• El usuario selecciona un video.• La aplicación muestra la vista Reproducción de Video con el video de la lección seleccionada.					
Resultado Esperado:					
<ul style="list-style-type: none">• La aplicación mostrará el video de la lección seleccionada.					
Resultado Obtenido:					
-					

Elaborado por: el autor

En la **Tabla 32** se describe el CP para un usuario que quiera jugar el minijuego Carta Memoria.

Tabla 32

CP-03 Realizar Minijuego Carta Memoria

Caso de Prueba	N.º	03	Cod:	CP-03	Realizar minijuego Carta Memoria
Descripción:					
El propósito de esta CP es probar la funcionalidad del minijuego Carta Memoria.					
Precondiciones:					
<ul style="list-style-type: none">• Tener un numero de usuario con suscripción activa.• Estar en un país que cuente con servicio.					
Flujo:					
<ul style="list-style-type: none">• La aplicación muestra la vista Home.• El usuario selecciona una franquicia.• La aplicación muestra la vista Franquicia con el contenido de la elección.• El usuario selecciona la pestaña de juegos.• La aplicación muestra la pestaña de juegos.• El usuario selecciona el minijuego Carta Memoria.• La aplicación muestra la vista Minijuego Carta Memoria.• El usuario seleccionará dos cartas.• La aplicación reproduce el audio de cada carta seleccionada.• La aplicación validará si las cartas son iguales, si son iguales la aplicación le sumará puntos al usuario, si son diferentes ocultará las cartas.• Se repetirá el proceso este proceso hasta que todas las cartas se oculten.• La aplicación validará si las cartas se ha terminado, cuando se terminen se mostrará el Pop-up de juego completado y sumará estrellas al usuario.					
Resultado Esperado:					
<ul style="list-style-type: none">• El sistema muestra el Pop-up de juego completado y obtención de estrellas para el usuario.					
Resultado Obtenido:					
-					

Elaborado por: el autor

Tabla 33*Cp-04 Ver Stickerbook RA*

Caso de Prueba	N.º	04	Cod:	CP-04	Ver Stickerbook RA
Descripción:					
El propósito de esta CP es probar la funcionalidad Stickerbook RA.					
Precondiciones:					
<ul style="list-style-type: none"> • Tener un numero de usuario con suscripción activa. • Estar en un país que cuente con servicio. • El usuario debe haber obtenido un Sticker sumando 20 estrellas mientras juega. 					
Flujo:					
<ul style="list-style-type: none"> • La aplicación mostrará la página de Home. • El usuario seleccionará el botón Stickerbook. • La aplicación mostrará la página de Stickerbook. • El usuario seleccionará un sticker que haya conseguido. • La aplicación mostrará la vista de Stickerbook RA. 					
Resultado Esperado:					
<ul style="list-style-type: none"> • La aplicación mostrará la vista de Stickerbook RA. 					
Resultado Obtenido:					
-					

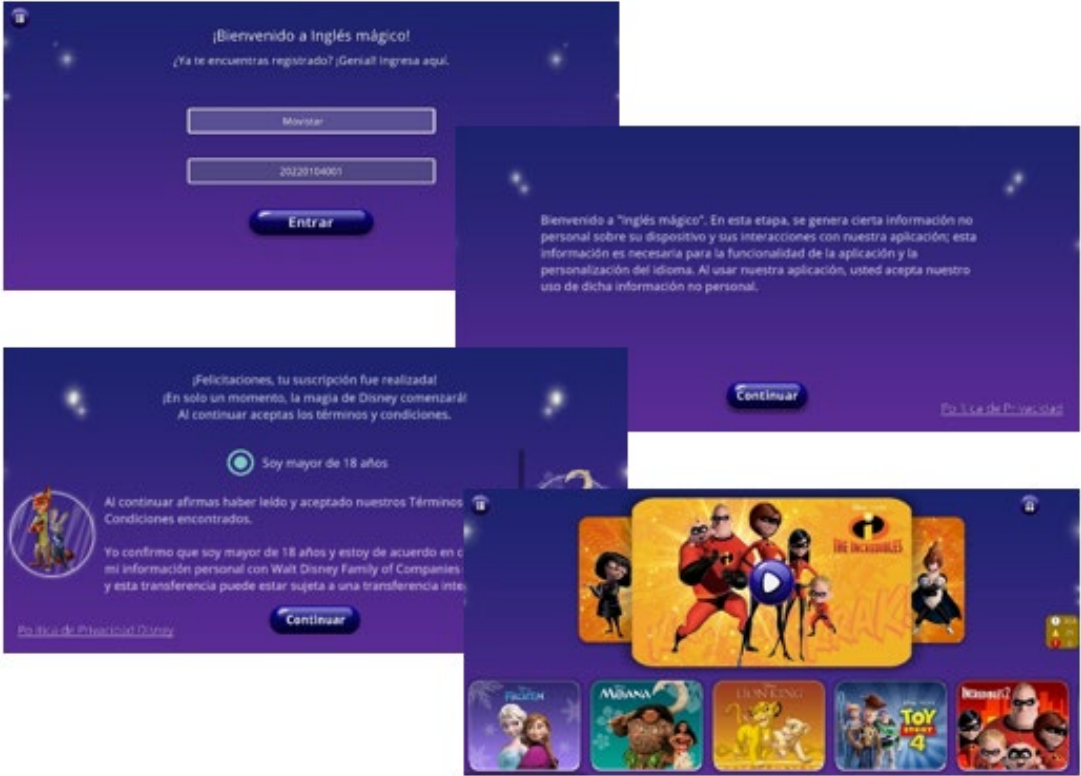
Elaborado por: el autor

4.5.2. Ejecución de Casos de Pruebas

En las siguientes tablas se detallan las pruebas realizadas a la aplicación de la empresa, los cuales se han obtenido los resultados esperados según se describió en el punto anterior

Tabla 34

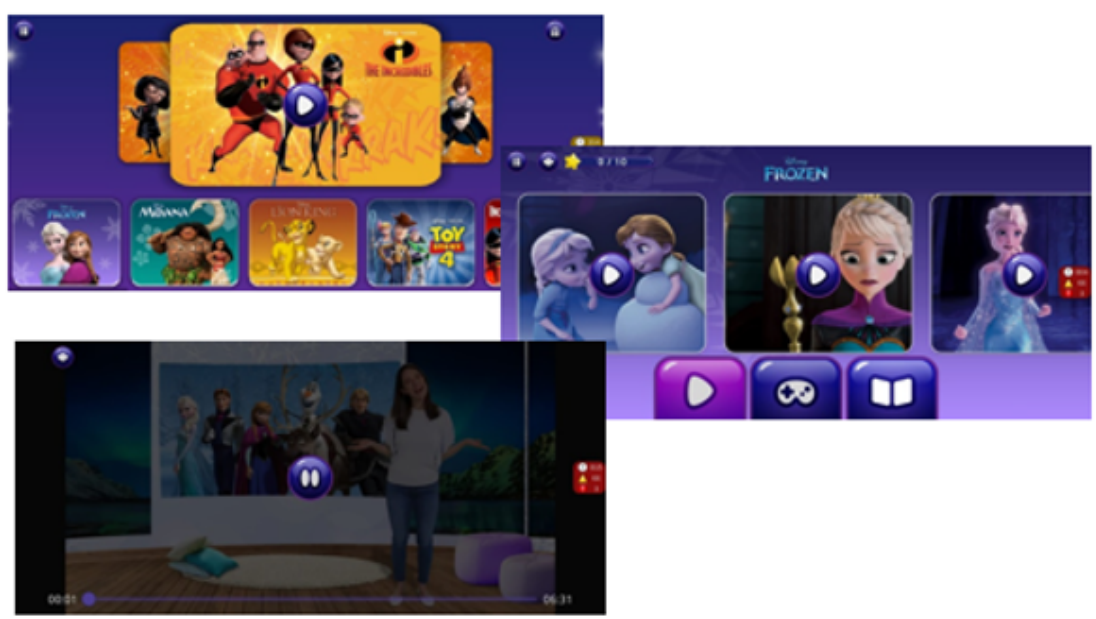
Ejecución de CP-01 Realizar Ingreso al Sistema

Caso de Prueba	N.º	01	Cod:	CP-01	Realizar ingreso al sistema
Descripción:					
El propósito de esta CP es probar la funcionalidad para realizar el inicio de sesión					
Precondiciones:					
<ul style="list-style-type: none"> • Tener un numero de usuario con suscripción activa. • Estar en un país que cuente con servicio. 					
Flujo:					
<ul style="list-style-type: none"> • La aplicación muestra la vista Login con el formulario de Ingreso. • El usuario selecciona su operador e ingresará su MSISDN. • La aplicación validará los datos del usuario. • La aplicación muestra la vista Términos y Condiciones. • El usuario acepta los TyC. • La aplicación muestra la vista Confirmación de Edad. • El usuario selecciona la opción según su edad. • La aplicación guarda los datos del usuario. • La aplicación muestra la vista Home. 					
Resultado Esperado:					
<ul style="list-style-type: none"> • La aplicación mostrará la página de Home. 					
Resultado Obtenido:					
 <p>The image displays four screenshots from the application. The top-left screenshot shows the login screen with the text '¡Bienvenido a Inglés mágico!' and '¿Ya te encuentras registrado? ¡Genial! Ingresá aquí.' It features input fields for 'Mostrar' and '20220104001', and an 'Entrar' button. The top-right screenshot shows a privacy policy notice: 'Bienvenido a "Inglés mágico". En esta etapa, se genera cierta información no personal sobre su dispositivo y sus interacciones con nuestra aplicación; esta información es necesaria para la funcionalidad de la aplicación y la personalización del idioma. Al usar nuestra aplicación, usted acepta nuestro uso de dicha información no personal.' with a 'Continuar' button. The bottom-left screenshot shows a subscription confirmation: '¡Felicitaciones, tu suscripción fue realizada! ¡En sólo un momento, la magia de Disney comenzará! Al continuar aceptas los términos y condiciones.' It includes a 'Soy mayor de 18 años' checkbox and a 'Continuar' button. The bottom-right screenshot shows the home screen with a central video player for 'The Incredibles' and a row of movie thumbnails including 'Frozen', 'Moana', 'La Princesa', 'Toy Story 4', and 'Navegator 2'.</p>					

Elaborado por: el autor

Tabla 35

Ejecución de CP-02 Ver Video de Lección

Caso de Prueba	N.º	02	Cod:	CP-02	Ver video de lección
Descripción:					
El propósito de esta CP es probar la funcionalidad para de ver video de lección.					
Precondiciones:					
<ul style="list-style-type: none"> • Tener un numero de usuario con suscripción activa. • Estar en un país que cuente con servicio. 					
Flujo:					
<ul style="list-style-type: none"> • La aplicación muestra la vista Home. • El usuario selecciona una franquicia. • La aplicación muestra la vista Franquicia con el contenido de la elección. • El usuario selecciona un video. • La aplicación muestra la vista Reproducción de Video con el video de la lección seleccionada. 					
Resultado Esperado:					
<ul style="list-style-type: none"> • La aplicación mostrará el video de la lección seleccionada. 					
Resultado Obtenido:					
 <p>The image displays three screenshots from a mobile application. The top-left screenshot shows the 'Home' screen with a grid of franchise cards including 'The Incredibles', 'Frozen', 'Moana', 'Toy Story 3', and 'The Lion King'. The top-right screenshot shows the 'Frozen' franchise selection screen with three video thumbnails and a bottom navigation bar with play, home, and book icons. The bottom screenshot shows a video player interface with a woman presenting in front of a screen displaying the Frozen characters, with a play button overlay and a progress bar at the bottom.</p>					

Elaborado por: el autor

Tabla 36

Ejecución de CP-03 Realizar Minijuego Carta Memoria

Caso de Prueb	N.º	03	Cod:	CP-03	Realizar minijuego Carta Memoria
Descripción:					
El propósito de esta CP es probar la funcionalidad del minijuego Carta Memoria.					
Precondiciones:					
<ul style="list-style-type: none"> • Tener un numero de usuario con suscripción activa. • Estar en un país que cuente con servicio. 					
Flujo:					
<ul style="list-style-type: none"> • La aplicación muestra la vista Home. • El usuario selecciona una franquicia. • La aplicación muestra la vista Franquicia con el contenido de la elección. • El usuario selecciona la pestaña de juegos. • La aplicación muestra la pestaña de juegos. • El usuario selecciona el minijuego Carta Memoria. • La aplicación muestra la vista Minijuego Carta Memoria. • El usuario seleccionará dos cartas. • La aplicación reproduce el audio de cada carta seleccionada. • La aplicación validará si las cartas son iguales, si son iguales la aplicación le sumará puntos al usuario, si son diferentes ocultará las cartas. • Se repetirá el proceso este proceso hasta que todas las cartas se oculten. • La aplicación validará si las cartas se ha terminado, cuando se terminen se mostrará el Pop-up de juego completado y sumará estrellas al usuario. 					
Resultado Esperado:					
• El sistema muestra el Pop-up de juego completado y obtención de estrellas para el usuario.					
Resultado Obtenido:					

Elaborado por: el autor

Tabla 37

Ejecución de CP-04 Ver Stickerbook RA

Caso de Prueba	N.º	04	Cod:	CP-04	Ver Stickerbook RA
Descripción:					
El propósito de esta CP es probar la funcionalidad Stickerbook RA.					
Precondiciones:					
<ul style="list-style-type: none"> • Tener un numero de usuario con suscripción activa. • Estar en un país que cuente con servicio. • El usuario debe haber obtenido un Sticker sumando estrellas mientras juega. 					
Flujo:					
<ul style="list-style-type: none"> • La aplicación muestra la vista Home. • El usuario selecciona el botón Stickerbook. • La aplicación muestra la vista Stickerbook. • El usuario selecciona un sticker que haya conseguido. • La aplicación muestra el sticker maximizado junto con los botones Descargar Marcador y RA. • El usuario descarga e imprime el marcador. • El usuario da clic al botón RA. • La aplicación muestra la vista Stickerbook RA. • El usuario enfoca con el celular el marcador que ha impreso. • La aplicación muestra el sticker RA. 					
Resultado Esperado:					
• La aplicación muestra el sticker seleccionado en RA.					
Resultado Obtenido:					

Elaborado por: el autor

4.6. Producto Mínimo Viable

4.6.1. Versión Final del Producto

Para la ejecución mínima del proyecto se ha desplegado conjunto con la empresa una back end y un CMS de gestión del contenido multimedia ambos en AWS. Con respecto al desarrollo ejecutado en el proyecto se ha desarrollado dos proyectos en cascada, primero el framework que contiene todos los códigos que optimizan el desarrollo y el segundo el proyecto de la aplicación de la empresa.

Este segundo es el cual ha culminado su desarrollo con la creación de un producto que puede ser consumido por el usuario final, la versión final del producto se muestra en imágenes en el siguiente punto.

4.6.2. Manual de Usuario

Para el uso de la aplicación el usuario debe haberse suscrito con su operador para poder activar su periodo de uso. Con un número de MSISDN suscrito al servicio el usuario debe ingresar a la aplicación para iniciar sesión, en la **Figura 33** se puede ver el formulario de inicio de sesión en el cual el usuario debe seleccionar su operador e ingresar su número MSISDN y dar clic a Entrar.

Figura 33

Captura de Pantalla de la Vista Login en el Celular



Nota. La figura muestra una captura de pantalla del celular de la vista Login.
Elaborado por: el autor

En el primer inicio de sesión del usuario en cada dispositivo el usuario deberá aceptar los términos y condiciones del uso de la aplicación, para ello la aplicación mostrará la Vista TyC, esta se puede observar en la **Figura 34**.

Figura 34

Captura de Pantalla de la Vista TyC en el Celular



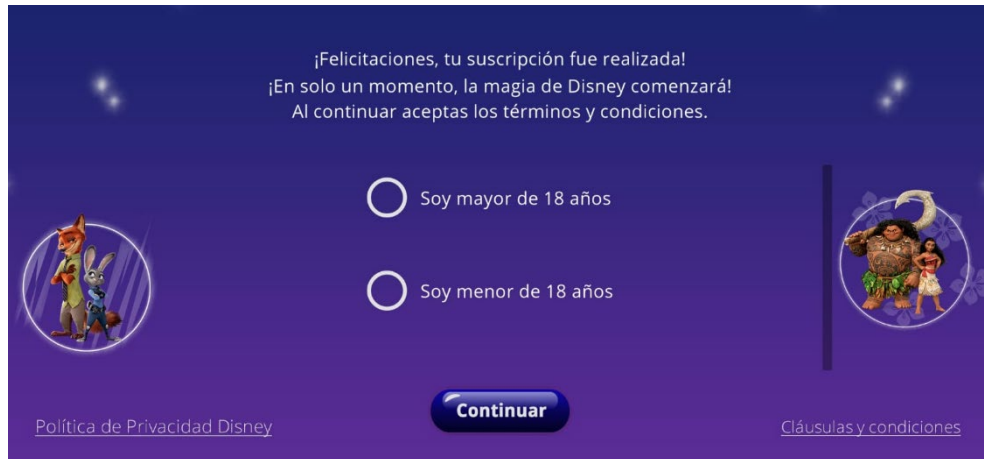
Nota. La figura muestra una captura de pantalla del celular de la vista TyC.
Elaborado por: el autor

Luego de dar clic en el botón Continuar de la página TyC, la aplicación guardará la selección y mostrará la Vista

Confirmación de Edad. En esta vista se puede observar dos opciones para seleccionar según la edad del usuario.

Figura 35

Captura de Pantalla de la Vista “Confirmación de Edad” en el Celular



Nota. La figura muestra una captura de pantalla del celular de la vista “Confirmación de Edad”. Elaborado por: el autor

Si el usuario es mayor de 18 años debe seleccionar la primera opción, caso contrario debe seleccionar la segunda opción, rellenar el campo email, en ambos casos para terminar debe dar al botón Continuar. Luego de esto la aplicación actualizará la información del usuario y mostrará la Vista Home según la **Figura 36**.

En la Vista Home se observa un carrusel en la parte superior con contenido acerca de las franquicias en la aplicación, en la parte inferior hay una lista de botones que redirigen a la Vista Franquicia con el contenido seleccionado.

Figura 36

Captura de Pantalla de la Vista Home en el Celular

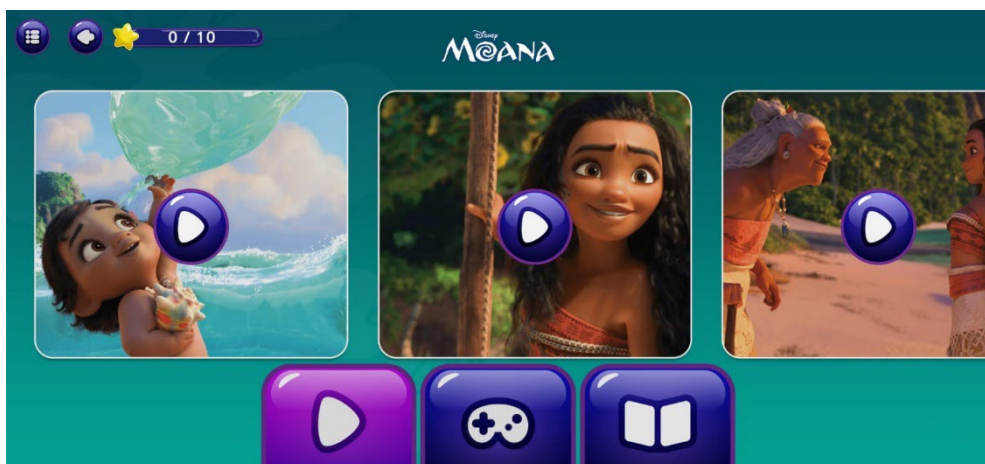


Nota. La figura muestra una captura de pantalla del celular de la vista Home. Elaborado por: el autor

Al seleccionar alguna franquicia se mostrará la Vista Franquicia con la pestaña de videos seleccionado, como se puede observar en la **Figura 37**, además de esta pestaña también se puede seleccionar la pestaña de juegos y el botón para ir a la Vista Stickerbook.

Figura 37

Captura de Pantalla de la Vista Franquicia - Videos en el Celular

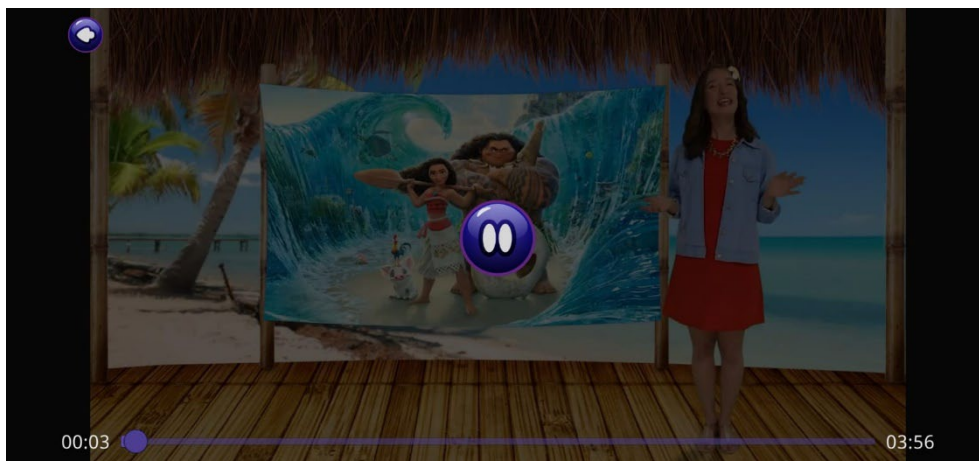


Nota. La figura muestra una captura de pantalla del celular de la vista "Franquicia - Videos". Elaborado por: el autor

Si el usuario selecciona un video la aplicación abrirá la Vista Reproductor de Video y se reproducirá el video seleccionado, esto se puede observar en la **Figura 38**.

Figura 38

Captura de Pantalla de la Vista Reproductor de Video en el Celular

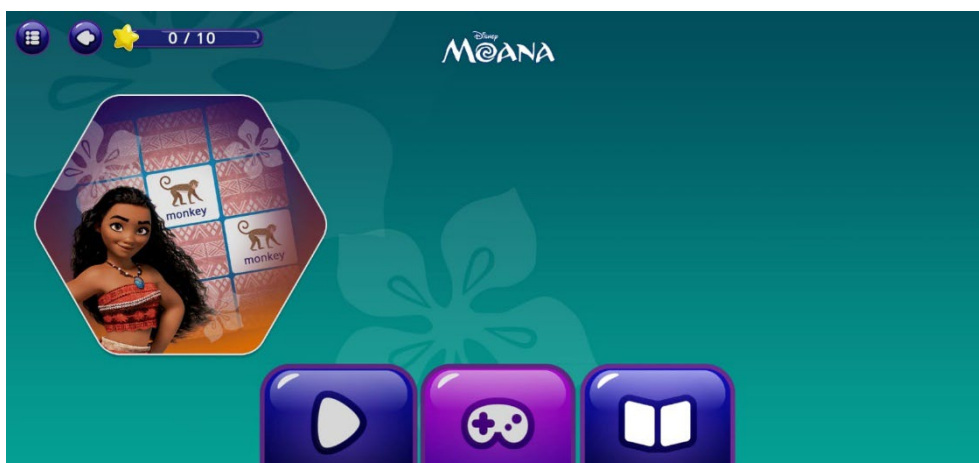


Nota. La figura muestra una captura de pantalla del celular de la vista “Reproductor de Video”. Elaborado por: el autor

Si el usuario quiere acceder los minijuegos tiene que dar clic al botón juegos en la Vista Franquicia, el cual mostrará los minijuegos que contenga la franquicia, según se muestra en la **Figura 39**.

Figura 39

Captura de Pantalla de la Vista Franquicia - Juegos en el Celular

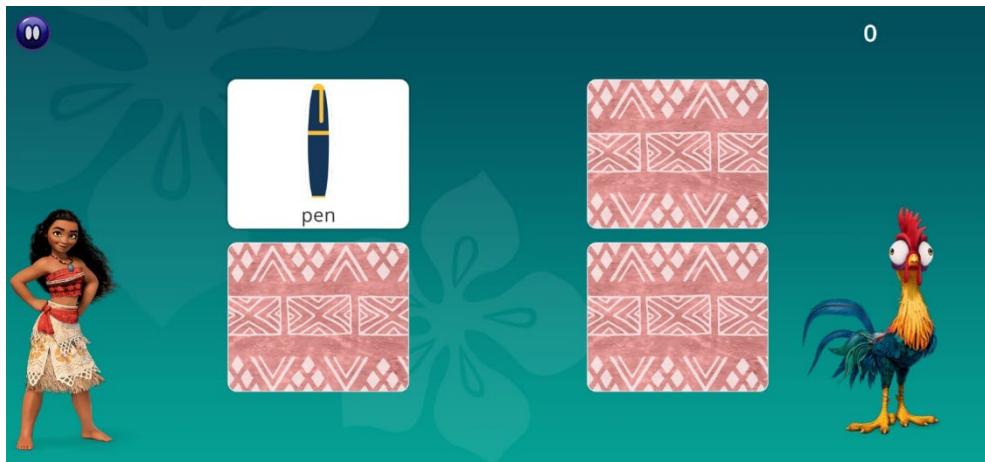


Nota. La figura muestra una captura de pantalla del celular de la vista “Franquicia - Juegos”. Elaborado por: el autor

Quando el usuario de clic al botón del minijuego Carta Memoria, la aplicación mostrará la Vista Carta Memoria e iniciará el minijuego. En este minijuego el usuario debe ir seleccionando pares de cartas, cada vez que seleccione una carta la aplicación girará la carta y reproducirá un audio con el texto de la carta. Si el usuario selecciona un par de cartas diferentes ambas cartas se voltearán. Si por el contrario las cartas son iguales, la aplicación realizará una animación de “Correcto”, desaparecerán las cartas y se sumarán puntos al usuario. Cuando todas las cartas se hayan desaparecido se mostrará un pop-up con los puntos y estrellas obtenidos.

Figura 40

Captura de Pantalla de la Vista Minijuego Carta Memoria en el Celular



Nota. La figura muestra una captura de pantalla del celular de la vista Minijuego Carta Memoria. Elaborado por: el autor

Quando el usuario consiga obtener más de 10 estrellas la aplicación de además de mostrar el pop-up de resumen de puntos y estrellas, mostrará otro pop-up con un nuevo sticker que se haya obtenido, según se muestra en la **Figura 41**.

Figura 41

Captura de Pantalla de la Obtención de un Nuevo Sticker en el Celular



Nota. La figura muestra una captura de pantalla del celular al obtener un nuevo sticker por sumar estrellas, en la vista Minijuego Carta Memoria al completar el minijuego. Elaborado por: el autor

Si el usuario desde la Vista Home o Vista Franquicia da clic al botón Stickerbook, la aplicación mostrará la Vista Stickerbook, ver **Figura 42**, en la cual se observa todos los stickers que el usuario haya obtenido al conseguir más de 10 estrellas mientras juega.

Figura 42

Captura de Pantalla de la Vista Stickerbook en el Celular

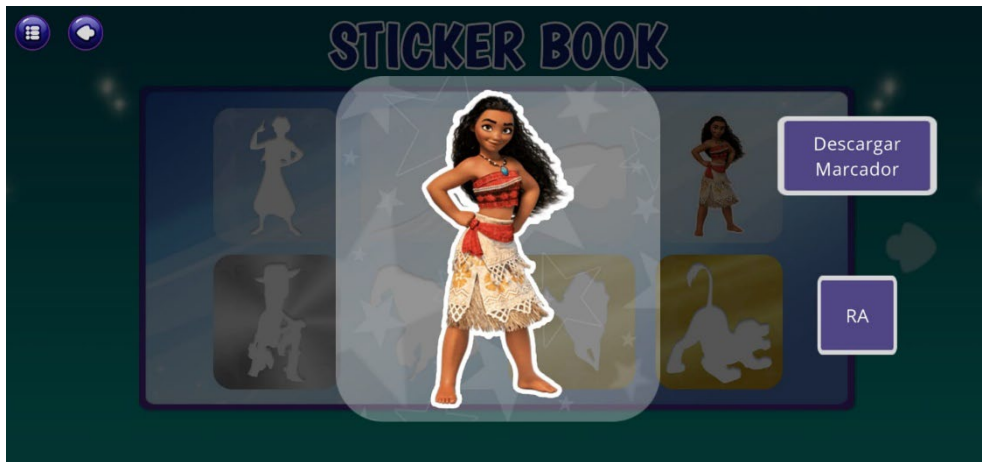


Nota. La figura muestra una captura de pantalla del celular de la vista Stickerbook. Elaborado por: el autor

Al dar clic en cualquier sticker que se haya obtenido, este se mostrará de manera maximizada junto con dos botones Descarga Marcador y RA según se muestra en la **Figura 43**. Si es usuario da clic al botón Descargar Marcador, la aplicación descargará el marcador del sticker maximizado, luego si el usuario desea acceder al Sticker en RA deberá imprimir el marcador descargado.

Figura 43

Captura de Pantalla de un Sticker Maximizado

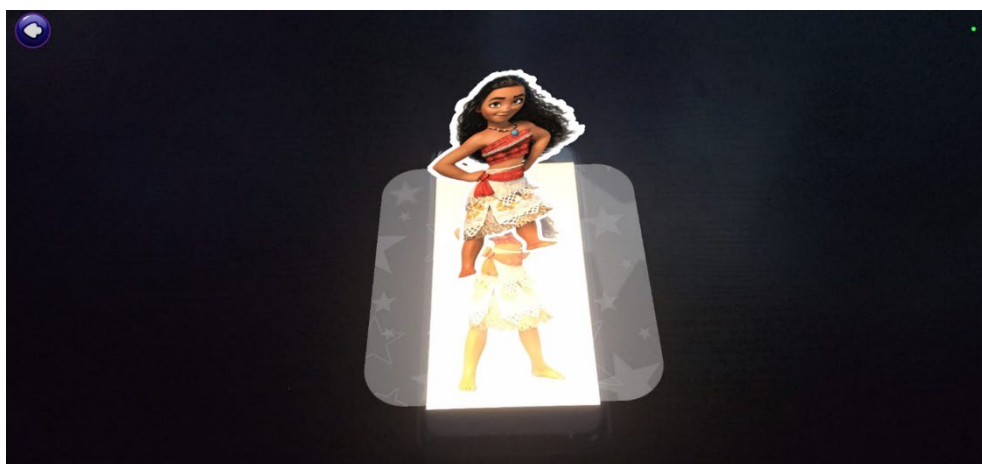


Nota. La figura muestra una captura de pantalla del celular al maximizar un sticker en la vista Stickerbook. Elaborado por: el autor

Si el usuario ha impreso el marcador del sticker seleccionado y da clic al botón RA, la aplicación abrirá la Vista RA, el cual mostrará lo que se enfoque con la cámara del usuario, aquí el usuario puede enfocar al sticker y se mostrará el Sticker RA según se observa en la **Figura 44.**

Figura 44

Captura de Pantalla de la Vista Sticker RA en el Celular



Nota. La figura muestra una captura de pantalla del celular visualizando un sticker en RA de la vista Stickerbook RA. Elaborado por: el autor

CAPÍTULO V

RESULTADOS

Los resultados de esta investigación se presentan a través del logro del objetivo general y los objetivos específicos planteados. Se tiene como resultado del objetivo general la creación del Framework SGOP el cual redujo en un 75.00 % del tiempo estimado para el desarrollo de un producto mínimo viable.

A continuación, se presentan los resultados obtenidos por cada uno de los objetivos específicos propuestos.

5.1. Respecto al Primer Objetivo Específico

Este objetivo indica: “Determinar el código fuente reutilizable para optimizarlos e implementarlos en futuros proyectos de videojuegos formativo”. Como resultado se obtuvo la definición de los paquetes y clases que contienen el código fuente reiterativo y las funciones de soporte al crear videojuegos formativos en páginas, estos paquetes y clases se pueden ver a detalle en la **Figura 45**.

En el análisis de los proyectos proporcionados por la empresa se determinó que del código fuente las partes que se involucraban con mayor predominancia en el proyecto eran las

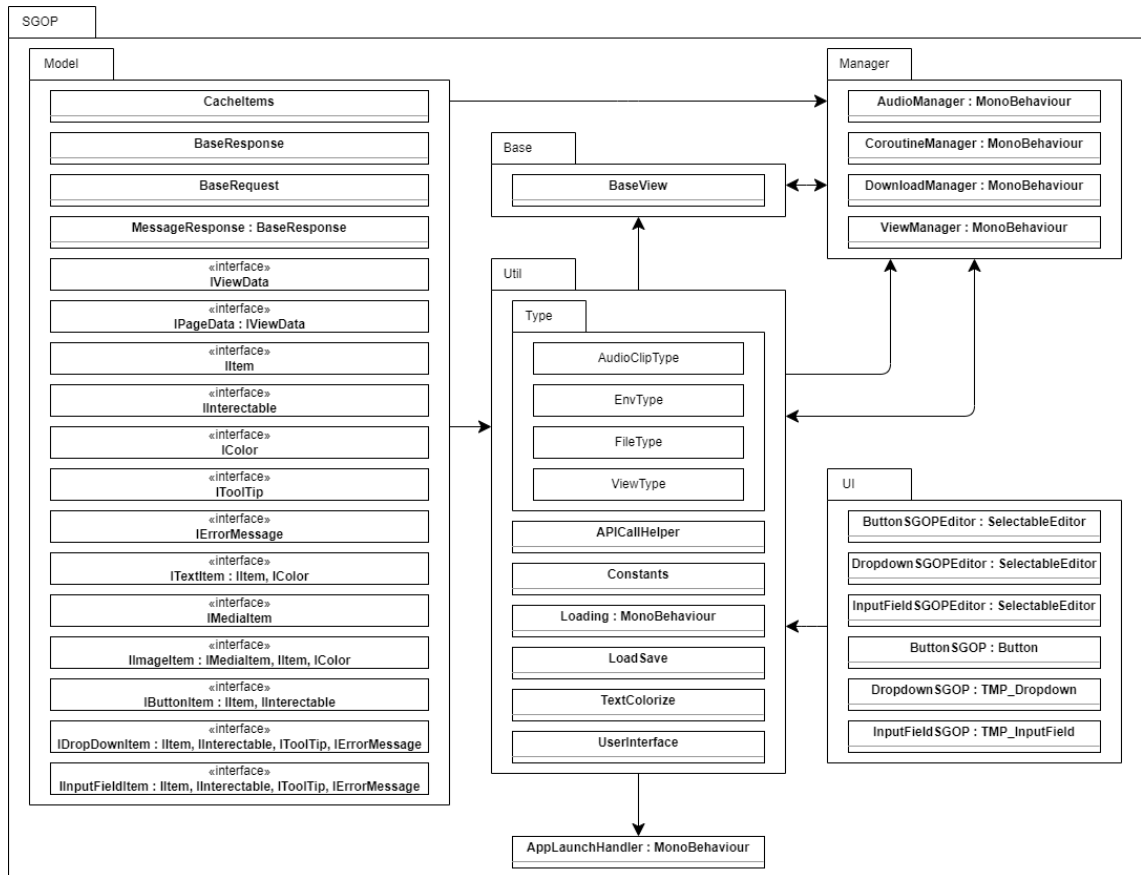
correspondientes a los “Managers”, por lo cual se encontró que “AudioManager”, “CoroutineManager”, “DownloadManager” y “ViewManager” son los que contenían la mayor cantidad de código fuente que se está utilizando de manera reiterativa.

Para el desarrollo del framework se definió que los scripts sean la base sobre la cual se armarán nuevos proyectos de videojuegos formativos basados en páginas, para lo cual también se han determinado componentes adicionales necesarios para optimizar el código fuente a fin de encapsular de manera eficiente los métodos de manipulación de los datos internos. Estos métodos de manipulación se añadieron al proyecto dentro del paquete “Util”, de este paquete se puede mencionar principalmente dos scripts, “APICallHelper” el cual es necesario para el desarrollador en función de simplificar las llamadas a Apis externas y “UserInterface” el cual funciona directamente con el script “ViewManager” para la manipulación de las interfaces de usuario de la aplicación.

También se crearon clases de interfaces de usuario para extender las funcionalidades básicas que presentan las proporcionadas por la herramienta Unity. Estos scripts exponen componentes adicionales necesarios para una mejor personalización de las interfaces de usuario. Por último, se ha creado el paquete “Model” con el fin de contener los modelos que interactuarán tanto con las interfaces de usuario, como con el contenido descargado desde el CMS.

Figura 45

Diagrama de Clases del Framework SGOP



Nota. La figura muestra el diagrama de clases del Framework SGOP

Elaborado por: el autor

5.2. Respecto al Segundo Objetivo Específico

El segundo objetivo planteado fue “Diseñar la metodología que implementará el marco de trabajo para el desarrollo para videojuegos formativos”, como resultado de cumplir este objetivo se obtuvo el modelo de implementación del Framework SGOP, el cual se puede observar en la **Figura 25**.

Esta metodología diseñada se compone de tres figuras las cuales explican el desarrollo de videojuegos formativos basados en páginas en diferentes contextos.

En un primer contexto está la arquitectura de la aplicación (**Figura 23**), en la cual se muestra como la aplicación debe interactuar en su despliegue, siendo esta de la siguiente manera: la aplicación se deberá desplegar su plataforma objetivo, pueda ser móvil, web o escritorio, una vez la aplicación esté en ejecución esta deberá interactuar con dos componentes independientes, el primero será la parte back end el cual contendrá los servicios que se requiera según el proyecto y el segundo será el CMS el cual contendrá el contenido multimedia que se presentará en la aplicación.

En un segundo punto está el cómo deberá funcionar el desarrollo de nuevos videojuegos formativos basados en páginas (**Figura 24**), el cual se realizará implementando como base de proyecto el Framework SGOP mediante el cual, durante el desarrollo de vistas, juegos u otros, se implementarán los métodos expuestos del código fuente del framework de manera que se evite realizar codificación reiterativa.

Por último, para el desarrollo de nuevos proyectos de videojuegos formativos en páginas se ha generado un modelo en el cual se especifican las fases desde el concepto hasta el lanzamiento de nuevos proyectos (**Figura 25**).

5.3. Respecto al Tercer Objetivo Especifico

En el tercer objetivo de determino el “Desarrollar el código fuente reutilizable optimizando su funcionamiento y encapsulándolos en un mismo proyecto”, como resultado de cumplir este objetivo se obtuvo el proyecto SGOP-Framework el cual se ha publicado en GitHub y empaquetado en el archivo “SGOP-Framework-v*” para que pueda ser utilizado por cualquier desarrollador, lo cual se puede visualizar en la **Figura 46**.

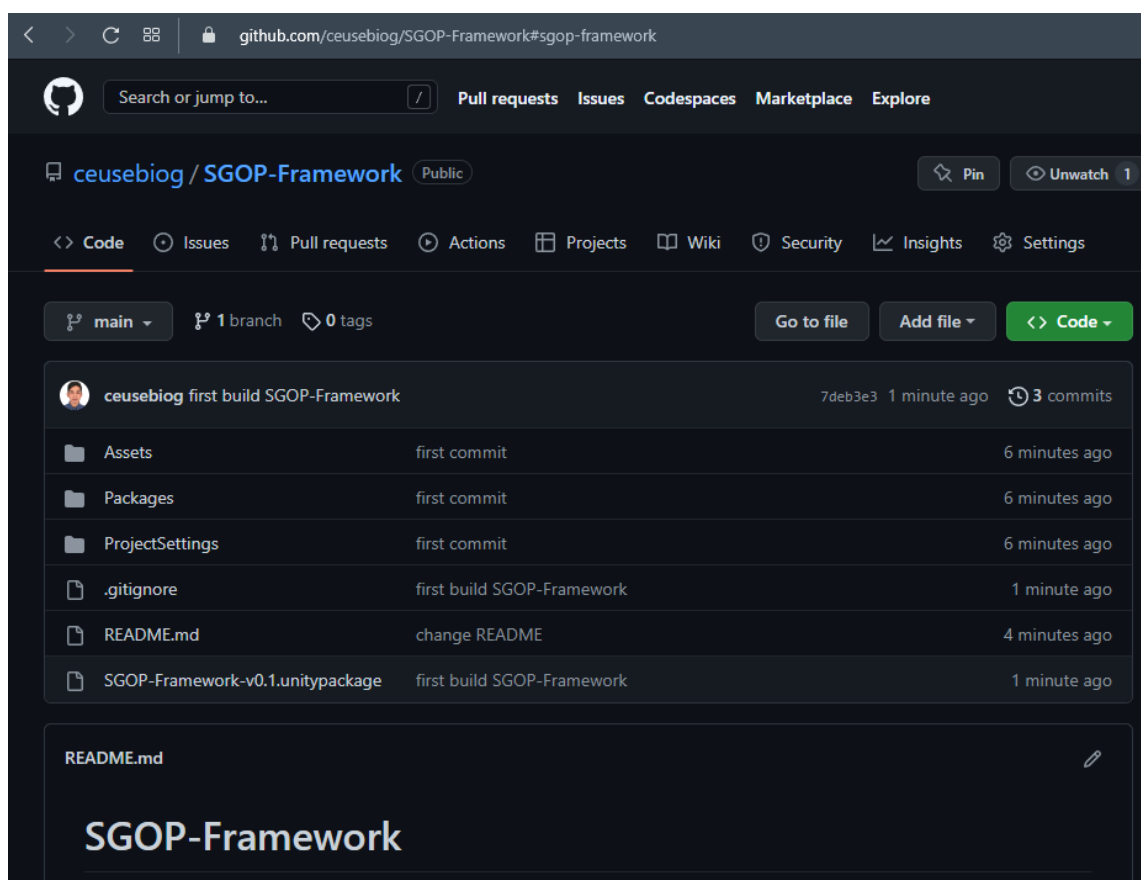
En el desarrollo del código fuente del framework se ha utilizado el resultado del primer objetivo, con lo cual el proyecto obtenido ha mejorado su funcionamiento gracias al análisis previo y

a seguir las directivas presentadas por el desarrollador de Unity en la sección de “Fixing Performance Problems” de su página web.

Lo presentado en el “Diagrama De Clases Del Framework SGOP” está diseñado a fin de cumplir con la abstracción detallada de los métodos y funciones, esto conlleva que durante el desarrollo no se encontrarán problemas de rendimiento al utilizar concurrentemente los métodos expuestos hacia el usuario.

Figura 46

Captura de Pantalla de la Página de GitHub del Proyecto SGOP-Framework



Nota. La figura muestra una captura tomada de [GitHub.com/ceusebiog/SGOP-Framework](https://github.com/ceusebiog/SGOP-Framework) en donde se encuentra alojado el proyecto realizado. Elaborado por: el autor

5.4. Respecto al Cuarto Objetivo Específico

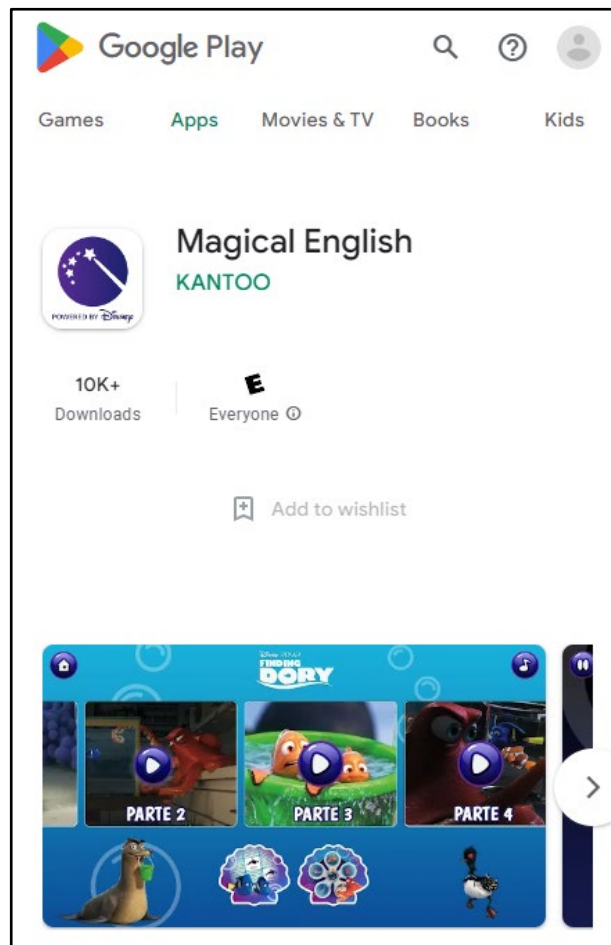
El último objetivo específico planteado fue “Validar que el framework creado cumple con las necesidades y optimiza el tiempo de desarrollo de un videojuego formativo”, como resultado de cumplir este objetivo se obtuvo se desarrolló una aplicación para la empresa La-Mark Visión utilizando el Framework SGOP, este desarrollo culminó con la publicación de la aplicación de la empresa en Google Play para su descarga abierta al público, el cual se puede visualizar en la **Figura 47**.

Este desarrollo se ha limitado a la creación de una aplicación como producto mínimo viable, en este sentido, el desarrollo previo del producto mínimo viable de la aplicación Magical English en la empresa tomo alrededor de cuatro meses con un equipo de desarrollo de tres personas, encargándose cada uno respectivamente del CMS, back end y del front end.

Para el caso del desarrollo del front end de la aplicación de la empresa utilizando el Framework SGOP, se ha determinado una reducción del tiempo de desarrollo de tres meses con cual se traduce en un 75.00 % del tiempo de desarrollo previsto.

Figura 47

Captura de Pantalla de Google Play - Magical English



Nota. La figura muestra una captura tomada de <https://play.google.com/store/apps/details?id=com.lamark.MagicalEnglish&hl=es> en donde se visualiza el despliegue de la aplicación de la empresa en la tienda de aplicaciones Google Play. Elaborado por: el autor

CAPÍTULO VI

DISCUSIÓN

El primer objetivo planteado en la investigación “Identificar el código fuente reutilizable con el fin de optimizarlos e implementarlos en futuros proyectos de videojuegos formativos” hace referencia a las fases de elicitación de requerimientos, análisis y diseño de software. En cuanto al resultado obtenido de estas actividades se puede determinar cómo artefacto principal el diagrama de clases.

En su artículo Hernández Vega, Verona Marcos, & Pérez Lovelle (2015), menciona que el diseñar el software con UML tiene la ventaja de ser un lenguaje universal, pero que también tiene la desventaja de ser muy general en alguno de sus términos, por lo cual no se puede representar adecuadamente las situaciones particulares (2015, pág. 81). En este sentido el diseño de diagrama de clases obtenido de la primera actividad ha sido capaz de incluir de manera concreta las necesidades del Framework SGOP, si bien se encontró que algunos términos específicos para Unity no son posibles de representar en UML, como las etiquetas de atributos para las variables que son útiles para exponerlas y poder modificarlas desde el inspector de componentes de Unity, estas han sido reemplazadas y representadas con las capacidades de UML.

El segundo objetivo de la investigación es “Diseñar la metodología que implementará el framework para el desarrollo para videojuegos formativos” esta actividad se define por la creación de los modelos de arquitectura, de desarrollo de software y desarrollo de proyectos de videojuegos formativos.

El modelo por el cual se parte es el presentado por Ahmad et al. (2017, pág. 10), el cual considera las fases de prelanzamiento y post lanzamiento de videojuegos, en la misma línea se modificó el modelo presentado para agregarle el enfoque que requiere el desarrollo de videojuegos formativos el cual dista de un videojuego comercial, a este modelo se le denominó “Modelo De Desarrollo De Proyectos De Videojuegos Formativos Utilizando el Framework SGOP”. Los cambios realizados al modelo presentado por Ahmad et al. constan de la definición de la fase de concepto en donde el original menciona el obtener del público objetivo cuales con los géneros de videojuegos más solicitados y seleccionar alguno de ellos, en el presentado por la investigación se añade el input de si el videojuego formativo es licenciado por alguna empresa con lo cual se manejará el género y público objetivo especificado según la definición del proyecto. En la fase de presupuesto se mantiene según el original. Las fases en adelante se modificaron para poner en paralelo la fase desarrollo y de marketing, y por último la fase de lanzamiento, de estas tres fases solo se ha mantenido la fase de Marketing propuesto por Ahmad et al. con respecto a la fase de desarrollo se cambió ya que no contará con la elección de motor videojuegos, es así debido a que el Framework SGOP está desarrollado para Unity. Por último, el modelo original mencionaba tres fases en paralelo que están dedicadas a un videojuego convencional, por lo cual no se encuentra necesario para un videojuego formativo, así mismo la interacción entre la fase de lanzamiento y de desarrollo se mantiene para realizar las correcciones o mejoras futuras según la retroalimentación proporcionada por los usuarios finales de las aplicaciones.

Adicionalmente, se creó el modelo de desarrollo/implementación del Framework SGOP el describe de manera visual como este debe interactuar con lo que el desarrollador final genere en su propio aplicativo y el modelo de

arquitectura el cual define como interactuará la aplicación con los servicios externos cuando sea desplegada y esté disponible a los usuarios finales.

El tercer objetivo de la investigación es “Desarrollar el código fuente reutilizable optimizando su funcionamiento y encapsulándolos en un mismo proyecto” está relacionado con el desarrollo, pruebas y despliegue del ciclo de vida del software. El resultado de esta actividad fue la generación del paquete del framework el cual podrá ser utilizado por el público en general ya que está publicado en un repositorio público de GitHub.

En el mismo sentido Apaza Yllachura (2019) desarrolló un framework para la creación de aplicaciones de realidad aumentada utilizando las tecnologías Angular y Google Blockly que permiten implementar las tecnologías de RA como Vuforia. De la misma manera, el framework desarrollado incluye la capacidad de administrar la visualización de contenido en RA con los framework de RA compatibles con Unity.

El último objetivo de la investigación es “Validar que el framework creado cumple con las necesidades y optimiza el tiempo de desarrollo de un videojuego formativo”, este ha involucrado la implementación de los modelos de arquitectura y desarrollo propuestos para el Framework SGOP, así como la implementación en el desarrollo del paquete de código de este. El resultado obtenido ha sido el despliegue de la aplicación de la empresa en la tienda de aplicativos de Android. También se ha obtenido como resultado que el tiempo de desarrollo de la aplicación se ha reducido de cuatro meses sin el uso de un framework a un mes con el uso del Framework SGOP. Por lo cual se puede concluir que con esta implementación se ha reducido el tiempo de desarrollo en un 75%.

CONCLUSIONES

Las siguientes cuatro conclusiones son la síntesis del objetivo general de la investigación, a las cuales se llegó luego del análisis, desarrollo y pruebas realizadas con el Framework SGOP y su implementación en el desarrollo de una aplicación.

1. Se extrajo código reutilizable para su implementación en el desarrollo del Framework SGOP, a este código fuente se le realizó un análisis a fin de aumentar el rendimiento y extender las funcionalidades. Este código fuente fue la base para el diseño del Framework SGOP.
2. Se diseñó el modelo metodológico que se recomienda implementar en el desarrollo de nuevos proyectos de videojuegos formativos basados en páginas. Este modelo incluye las fases de Concepto, Presupuesto, Desarrollo, Marketing y Lanzamiento, de esta manera se fija un punto de partida y estándar para el desarrollo de nuevos proyectos de videojuegos formativos basados en páginas.
3. Se desarrolló y publicó el Framework SGOP en el repositorio público GitHub, el cual incluye el código fuente y la primera versión del proyecto en un archivo “unitypackage”.
4. El desarrollo del front end del producto mínimo viable de la aplicación Magical English duró un mes utilizando el Framework SGOP en comparación con los cuatro meses previstos sin su

implementación. Por lo tanto, se concluyó que el Framework SGOP redujo en un 75% el tiempo de desarrollo de la mencionada aplicación, con ello se validó que su uso optimiza y disminuye el tiempo de creación de nuevos videojuegos formativos basados en páginas.

RECOMENDACIONES

Luego de analizar las conclusiones se procedió a dar las siguientes recomendaciones.

1. Continuar con la investigación siguiendo con lo propuesto en la arquitectura de los videojuegos formativos, creando metodologías tanto para el back end como para el gestor de contenido.
2. Implementar el Framework SGOP en más proyectos para obtener una mayor retroalimentación. Esto aumentaría los datos sobre el nivel de optimización que puede proporcionar la implementación del Framework SGOP, asimismo puede repercutir positivamente en sus correcciones y mejoras.
3. Implementar el “Modelo de Desarrollo de Proyectos de Videojuegos Formativos Utilizando el Framework SGOP” en futuros proyectos, a fin de sofisticarlo y mantenerlo actualizado con las nuevas lógicas que se desarrollen a futuro.

FUENTES DE INFORMACIÓN

- Ahmad, N. B., Abdul Rahman, S., Abou Shahada, T. M., & Ayman Anabtawi, Z. (2017). How to Launch a Successful Video Game: A Framework. *Entertainment Computing*, 23, 1-11. doi:<http://dx.doi.org/10.1016/j.entcom.2017.08.001>
- Apaza Yllachura, Y. G. (2019). *Framework para diseño de contenidos en aplicaciones móviles de realidad aumentada orientado a personal no-experto [Tesis de grado, Universidad Nacional de San Agustín de Arequipa]*. <http://hdl.handle.net/20.500.12773/11978>
- Cuervo Gómez, W. O., & Ballesteros-Ricaurte, J. A. (2017). Framework para desarrollo de aplicaciones educativas móviles, basado en modelos de enseñanza. *Praxis & Saber*, 8(17), 125-153. doi:<https://doi.org/10.19053/22160159.v8.n17.2018.7204>
- Cwalina, K., & Brad, A. (2009). *Framework Design Guidelines Conventions, Idioms, and Patterns for Reusable .NET Libraries* (2 ed.). Addison-Wesley Professional.
- Garita Hernández, F. A., Lizano Madriz, F., & Cordero Esquivel, C. M. (2019). Metodologías para el Desarrollo de Videojuegos Serios: Una Revisión

de Literatura. *Tecnología Educativa*, 6(1), 103-114.
doi:<https://doi.org/10.32671/terc.v6i1.55>

Hernández Vega, J. R., Verona Marcos, S., & Pérez Lovelle, S. (2015). Modelando con UML el proceso de evaluación de productos de software utilizando el enfoque GQM. *Revista Cubana de Ciencias Informáticas*, 9(1), 76-84.

Jopen, G., Gomez, W., & Olivera, H. (2014). Sistema educativo peruano: balance y agenda pendiente. *Departamento de Economía, PUCP*.
<https://departamento.pucp.edu.pe/economia/wp-content/uploads/DDD379.pdf>

Juul, J. (2005). *Half-Real: Video Games between Real Rules and Fictional Worlds*. MIT Press.

Lozada, J. (2014). Investigación Aplicada: Definición, Propiedad Intelectual e Industria. *CienciAmérica*, 3(1), 34-39.
<https://cienciamerica.edu.ec/index.php/uti/article/view/30>

Michael, D., & Chen, S. (2005). *Serious Games: Game That Educate, Train, and Inform*. Cengage Learning PTR.

Muñoz Izquierdo, C. (2012). Tres problemas fundamentales del sistema educativo. *Perfiles Educativos*, 34, 154-159.
<https://www.scielo.org.mx/pdf/peredu/v34nspe/v34nspea14.pdf>

Preciado Cossío, G., & Silva Paucar, E. (2018). *Buenas Prácticas para el Desarrollo de Videojuegos Educativos Aplicados a Historia y Arqueología Usando Realidad Aumentada para la Educación Primaria en el Perú [Tesis de grado, Universidad de San Martín de Porres]*.
<https://hdl.handle.net/20.500.12727/3943>

Qustodio. (2021). Más conectados que nunca. *Qustodio*.
<https://www.observatoriodelainfancia.es/oia/esp/descargar.aspx?id=7131&tipo=documento>

- Schwaber, K., & Sutherland, J. (2016). *La Guía de Scrum™ La Guía Definitiva de Scrum: Las Reglas del Juego*. Scrum Guides.
- SCRUMstudy. (2016). *La Guía para el Cuerpo de Conocimiento de Scrum (Guía SBOK™)*.
- Sisa Yataco, H. R. (2020). *Técnica para validar el diseño de software durante la elicitación de requisitos empleando especificaciones de usabilidad [Tesis de grado, Universidad Nacional de San Agustín de Arequipa]*. <http://hdl.handle.net/20.500.12773/13232>
- Unity3d. (s.f.). *Unity Manual: Asset packages*. Recuperado el 3 de Octubre de 2022, de <https://docs.unity3d.com/Manual/AssetPackages.html>
- Unity3d. (s.f.). *Unity Manual: Coroutines*. Recuperado el 3 de Octubre de 2022, de <https://docs.unity3d.com/Manual/Coroutines.html>
- Unity3d. (s.f.). *Unity Manual: GameObjects*. Recuperado el 3 de Octubre de 2022, de <https://docs.unity3d.com/Manual/GameObjects.html>
- Unity3d. (s.f.). *Unity Manual: Prefabs*. Recuperado el 3 de Octubre de 2022, de <https://docs.unity3d.com/Manual/Prefabs.html>
- Vuforia. (s.f.). *Vuforia: Vuforia Engine Overview*. Recuperado el 3 de Octubre de 2022, de <https://library.vuforia.com/getting-started/vuforia-features>