

Informe Proyecto Programación II

Numero de Grupo: 16

Alumnos:

Cesar Andres Franco Mindiola (2021459561)

Sebastián Ignacio García Péndola (2021443321)

Fecha: 14-12-2022

Enunciado de Nuestro Problema: Animación de la física de lanzamiento de misiles desde un avión

De Este enunciado se nos solicitaba:

- Un Escenario compuesto por un avión, un blanco y un misil animados
- el misil debe mostrar su ruta mediante las clases Angular y RellenaConpuntos
- se debe poder elegir la altura, dirección y velocidad del avión
- el blanco se mueve en una cierta altura de forma horizontal
- el control de Disparo se debe realizar mediante el uso del mouse o teclado o con controles GUI
- El misil tendrá una distancia máxima de detección del blanco en caso de no detectar sigue su dirección
- si el misil detecta el blanco este ajusta su dirección hacia este
- se puede cambiar la dirección del blanco con el teclado

Lo que decidimos agregar fue el cambio de dimensiones de la ventana y el reajuste de los parámetros de la simulación para encajar con estos mediante una escala

Casos de Uso:

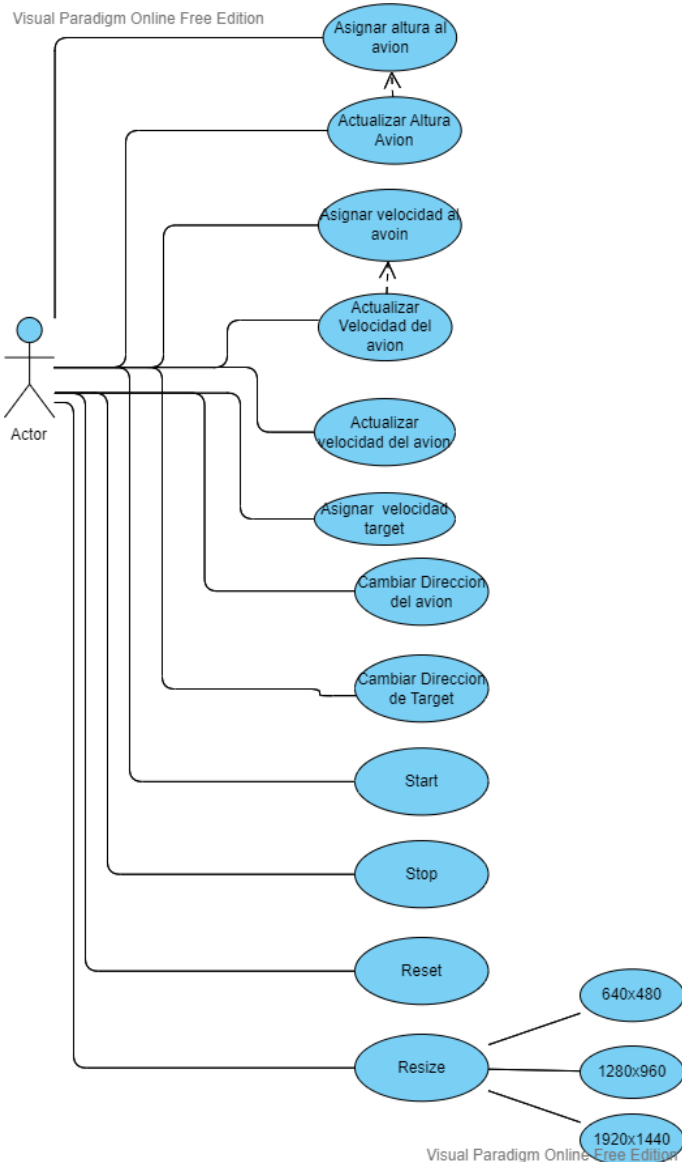
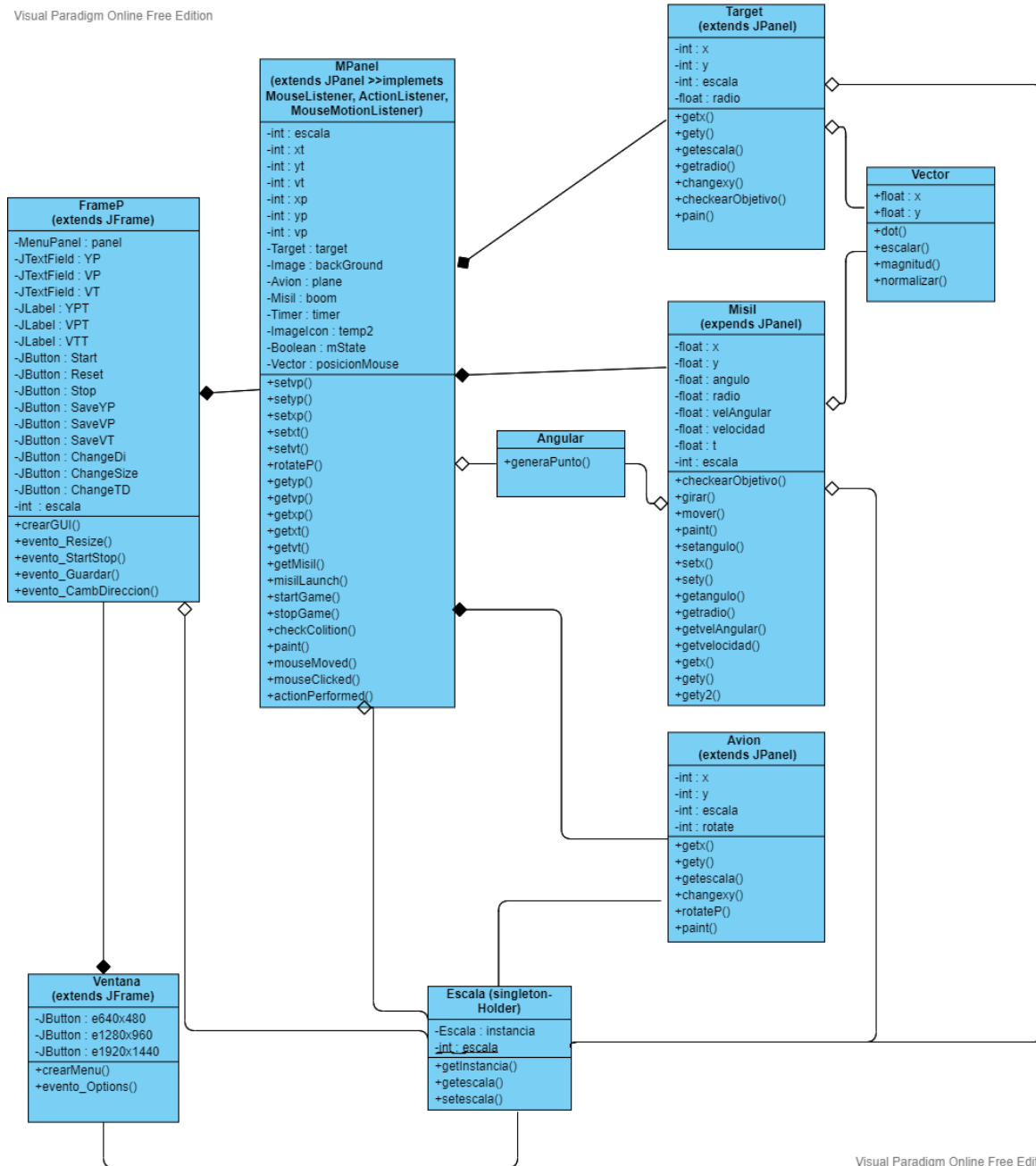


Diagrama UML:

Visual Paradigm Online Free Edition

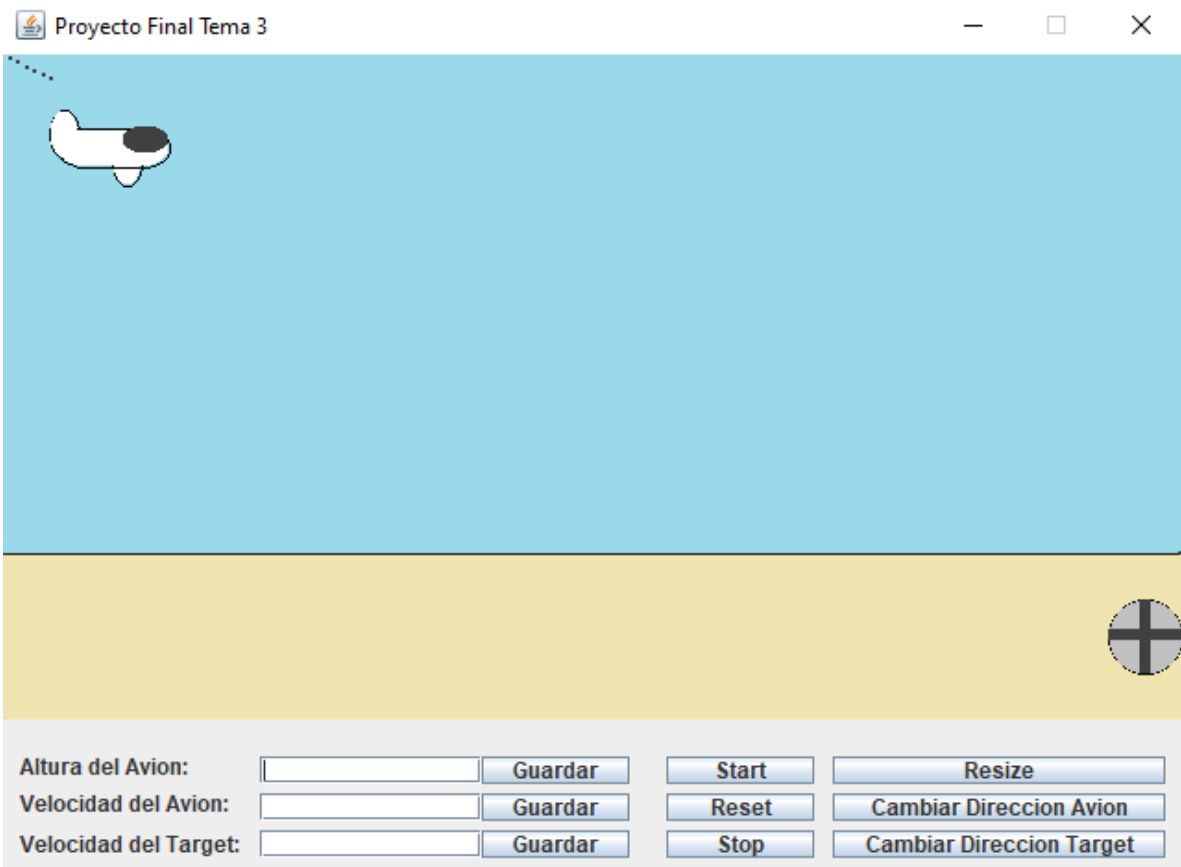
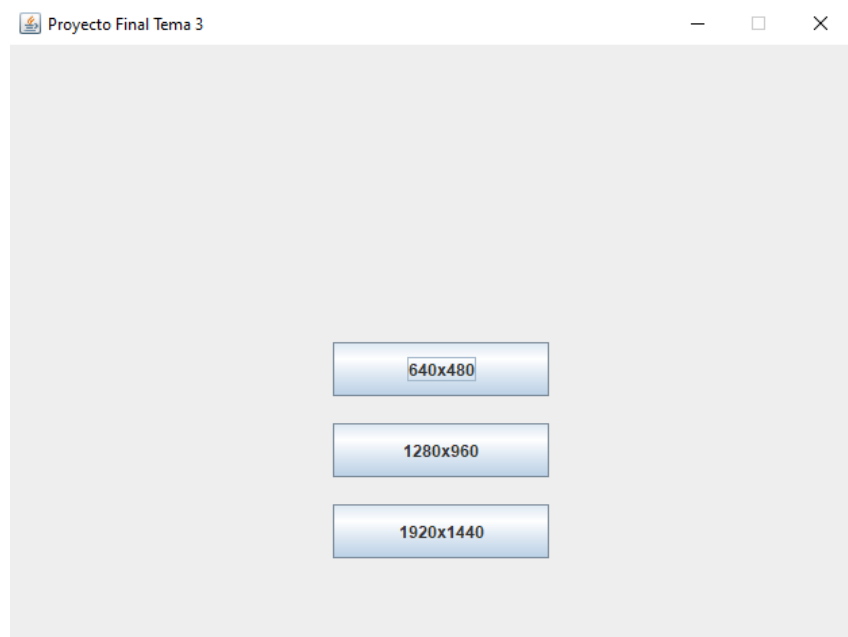


Visual Paradigm Online Free Edition

Patrones Utilizados:

Singleton-Holder en la Clase Escala, Buscábamos el que cada clase tuviera la escala en sus variables, pero al igual que queríamos que la escala fuera única para todos por eso en vez de que cada clase recibiera una escala por el constructor se decidió que se creara una única clase Escala que contendría la escala y esta cuando se creara una clase que la utilizara la pudiera sacar de aquí

Captura de la Interfaz:



Decisiones tomadas durante el proyecto:

- Por el conocimiento de que distintos monitores tienen distintas resoluciones y que para ciertos computadores una resolución de la aplicación sería muy grande o muy pequeña decidimos implementar una escala que permitiría modificar la resolución del programa y mantener toda la interfaz visual en la misma posición relativa.
- Por diferencias en la forma de programar al inicio se crearon 2 branch una para cada integrante donde se reescribía de una a otra y cada uno lo ajustaba a su forma de escribir esto hasta que ambos estuviéramos en el mismo punto de avance donde decidimos hacer merge una sola branch y continuar en esta.
- Decidimos usar el ejemplo de misil entregado por el Ayudante (Marcelo Vargas) pero con adaptaciones a nuestro código al igual que simplificación de algunas partes.
- Implementación de 2 ventanas para facilitar el ajuste de la Resolución
- Decidimos usar interfaz GUI en vez del Teclado para cambiar la Dirección del Blanco/Objetivo

Problemas encontrados durante el proyecto y autocritica:

- Dificultad en las mecánicas del misil por lo que decidimos usar como base el ejemplo del Ayudante
- Problemas para organizar nuestro tiempo causando el atraso de la entrega del proyecto
- Problema de organización de que hacer generando descoordinación de tiempo, ideas y la implementación de las mismas causando por ejemplo el no uso de escala en los inicios o Confusión a la hora de leer el código
- Aun habiendo Visto la materia la Falta de Practica con Patrones, Javadoc y Unit Test Causaron que su uso fuera más básico del que creemos que sería ideal
- Diversas Complicaciones al desarrollar la mecánica de Detección de Target y Colisión con Target causando un mayor atraso.