

**Proyecto 1 -Datos II – Juego de Memoria**

Instituto Tecnológico de Costa Rica

Área Académica Ingeniería en Computadores

CE2103 – Algoritmos y Estructuras de Datos II

Alessandro Hidalgo Prendas – 2020099003

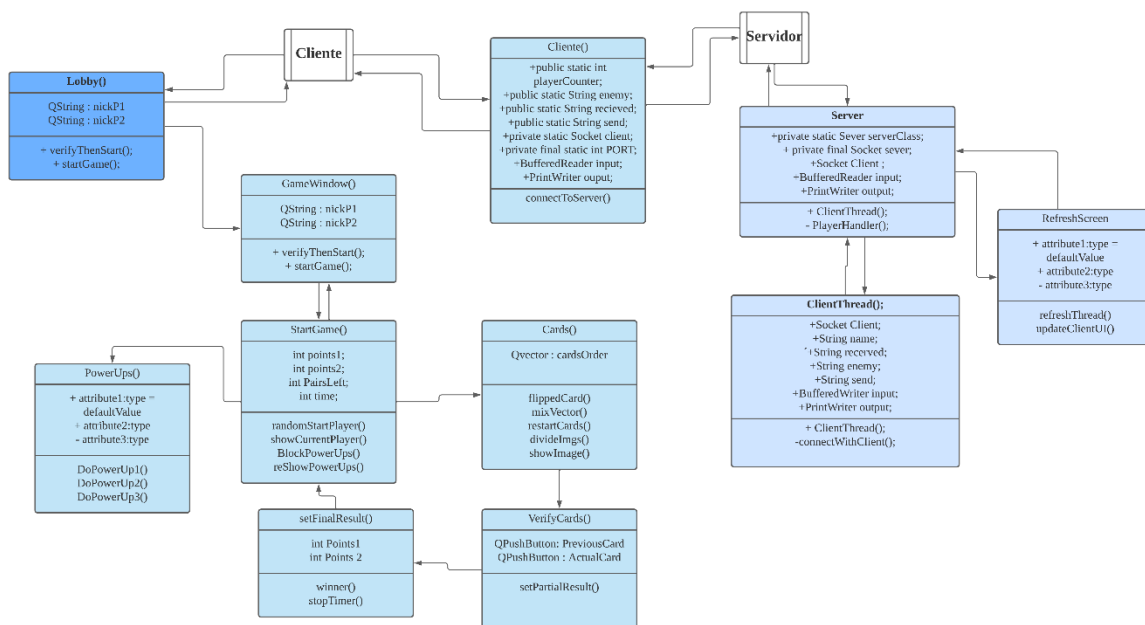
**Descripción Breve**

El proyecto tiene como objetivo principal introducirnos al lenguaje de programación C++ en el que para concluir se debe de investigar por cuenta propia como implementar soluciones para poder cumplir los requerimientos que se nos solicitan.

El juego consiste en crear una aplicación para dos jugadores que se basa en un juego de memoria de cartas en el cual habrá un solo ganador que será el que tenga el mayor puntaje, el juego consiste básicamente en que se deben de buscar parejas de cartas para así ganar puntos, si se voltean 2 cartas y estas coinciden el jugador podrá seguir con su turno y además se le añadirán puntos al mismo, por otro lado, si las tarjetas volteadas no coinciden se le restarán puntos y el turno pasa al otro jugador.

Cada jugador posee tres power ups o ventajas que podrá usar a su favor, una consiste en añadir más tiempo que puede ser útil para que tenga más oportunidades de sumar puntaje, otra ventaja es disminuir el tiempo, esta sirve por ejemplo si tienes mayor puntaje al jugador rival, restas tiempo para que así el otro jugador tenga menos oportunidades de igualar el marcador, luego la última ventaja que tendrá cada jugador es que puede sumarse 100 puntos al marcador que es el equivalente al haber encontrado una pareja.

Cuando se acaban las parejas o cuando termina el tiempo finaliza el juego y quien tenga mayor puntaje ganará la partida y se puede decidir si jugar de nuevo o cerrar la aplicación.

**Diagrama de Clases**

### Descripción de las estructuras de datos desarrolladas

La estructura de datos que se utilizó para crear la GUI del juego fue la biblioteca de QtCreator que es una herramienta que facilita la manipulación de los elementos de la interfaz y trae varias opciones prediseñadas que pueden ser útiles a la hora de crear una interfaz, respecto a el resto del proyecto como la lógica entre las principales estructuras de datos que se manejaron eran las librerías de QT como QString, QVector y otras más, para la creación de variables globales se utilizaba principalmente el tipo de datos int o QString.

Para el manejo de los QPushButton se hizo un vector al que posteriormente con una función se le asignaban aleatoriamente las imágenes de las cartas que iba a corresponder.

### Protocolo de Comunicación entre cliente y Servidor

Esta parte del proyecto no se logró concretar, pero básicamente consistía en mediante el uso de sockets que son un sistema de comunicación entre procesos de diferentes maquinas conectadas a una red y un puerto específico, se quería lograr comunicar 2 componentes principales: el cliente y el servidor, en el cliente sería la GUI del juego que se encarga únicamente de renderizar lo que se ordene en el servidor. Cada acción que realice el o los usuarios en la interfaz grafica del juego deberá enviar comandos al servidor para que se ejecuten y se actualice su estado y el del cliente.

### Problemas y Bugs

En el desarrollo del juego iban surgiendo varios bugs y problemas que se fueron resolviendo, pero también quedaron problemas que resolver como el no poder integrar la arquitectura de cliente-servidor y algunos problemas con la paginación requerida o el manejo de memoria esperado. Enlistando algunos bugs que hubo pueden ser los siguientes:

- Se buggeaban la obtención de los nicknames, no podía enviarlos a la pantalla del juego.
- Los puntajes no se actualizaban cada que ganará o perdiera puntos.
- El tiempo se quedaba congelado depende de la función que usaba.
- Los power ups eran infinitos y no se desactivaban los del rival.
- No se realizó la conexión de cliente servidor, no se pudo correr el juego como cliente.
- Las cartas se quedaban volteadas aunque no fueran parejas, no se refrescaban al instante.
- Cuando se terminaba el juego y se quería volver a jugar las cartas no se reiniciaban.
- Se podían voltear más de 2 tarjetas a la vez y si se hacía esto quedaban dos cartas congeladas aunque no fueran pareja.

### Planificación y Administración en Jira

En el siguiente enlace se puede acceder a la planificación y descomposición de historias de usuario manejado en Jira: <https://ahidalgoprendas.atlassian.net/jira/software/projects/P1D2/issues/>

### Repositorio de GitHub

El siguiente enlace es el repositorio del proyecto: <https://github.com/A-HiDALGO/Proyecto1>