



**INSTITUTO TECNOLÓGICO DE COSTA RICA
ÁREA DE INGENIERÍA EN COMPUTADORES
ALGORITMOS Y ESTRUCTURAS DE DATOS II**

DOCUMENTACIÓN EXTERNA DEL PROYECTO

Profesor:

Diego Noguera Mena

Estudiante:

Sofía Valverde Ureña – 2020161005

20 de abr. de 22

I Semestre 2022

| | |
|--|---|
| <i>Descripción del problema</i> | 3 |
| <i>Descripción de las estructuras de datos desarrolladas</i> | 4 |
| <i>Descripción de los algoritmos de paginación</i> | 5 |
| <i>Protocolo de comunicación entre el cliente y servidor</i> | 5 |
| <i>Problemas y bugs</i> | 6 |

Descripción del problema.

En este proyecto se solicita implementar un juego de memoria, en donde se deben hacer parejas con cartas que estará dispuestas al azar.

Se utilizan conceptos de paginación, pues no todas las cartas están cargadas en la memoria.

El juego debe tener un servidor y un cliente conectados por medio de sockets, el juego cuenta con dos jugadores, que sumará puntos con cada pareja que se logre hacer. Saldrá la puntuación que tienen los jugadores. Si un jugador logra encontrar una pareja, estas cartas tendrán un indicador de que ya no están en juego, además de que el servidor cambiará aleatoriamente las tarjetas que se encuentran en memoria.

Se contará con tres power ups, que darán ventaja al jugador que los use.

El servidor tendrá una interfaz gráfica en donde se encuentran las cartas, el puntaje de los jugadores y el consumo de memoria, que esta no debe aumentar cuando se cargan nuevas tarjetas.

El jugador tendrá un punto extra, si hace parejas con cartas que se encuentran en memoria.



Se vería algo como esta imagen.

Descripción de las estructuras de datos desarrolladas.

4

Se creó el CustomCommand, que tiene como uso usar strings como comando, pasándolo del server al cliente.

```
struct CustomCommand {  
    std::string command;  
    std::string btn;  
};
```

La clase Tarjeta reconoce cuál tarjeta es, dependiendo de la imagen, además de que encuentra el lugar en el que está en la matriz.

```
Demoserver  
#include <fstream>  
class Tarjeta  
{  
public:  
    std::string imagen;  
    int fila;  
    int columna;  
private:  
};
```

Descripción de los algoritmos de paginación.

En este proyecto no se pudo implementar los algoritmos de paginación, entonces no existe la función de las tarjetas guardadas en la memoria.

Protocolo de comunicación entre el cliente y servidor.

Para comunicar el cliente y el servidor, se utilizaron sockets, en específico el Winsock.

El Winsock es un tipo de interfaz, utilizado para comunicar el software de la red y Windows y los servicios de red. En otras palabras, se puede explicar como un traductor para los servicios de red básicos, haciendo solicitudes de envío y de recepción, que normalmente son genéricas y simples, pero se convierten en solicitudes con un protocolo, que logra que se cumple la función deseada.

Normalmente, viene preinstalado en todas las computadoras con SO de Windows.

El servidor se inicia con el localhost 127.0.0.1 en el puerto 5555.

Problemas y bugs.

Se encuentran varios problemas en este proyecto, por ejemplo:

- Al hacer una pareja, no se vuelven a poner al azar y se puede volver a elegir esas dos cartas, haciendo que el jugador pueda tener puntos infinitos.
- Se inició con el repositorio “<https://github.com/Sof-valur/TEC.ProyectoDatos>”, pero por un problema con el tamaño de los archivos, se tuvo que iniciar un nuevo repositorio, que es el actual. En el read.me se puede ingresar al primer repositorio para ver los commits.
- No se pudo implementar los algoritmos de paginación en este proyecto.