UNO

Documentation

Team #4 - UNO

Malhotra, Shubham

Periaswami, Siddhesh

Chen, Joe

Date:12/08/2019

unaccepted Revision: 0.1

Table of Contents

Revision History

Name	Date	Reason for Change	Version
Joe Chen	12/08/2019	Adding login GUI info and Design document	0.1
Siddhesh P	12/11/2019	Networking - client side	0.1
Shubham M	12/11/2019	Networking - Server , Design document	0.1
Joe Chen	11/27/2019	Jar files	0.1
Siddhesh P	11/23/2019	Functioning - Removing and adding the cards on panels	0.1
Shubham M	11/18/2019	Validation and java docs	0.1
Joe Chen	10/23/2019	Cleaning the code (removing the unnecessary code)	0.1
Siddhesh P	10/22/2019	Player class was implemented	0.1
Shubham M	10/20/2019	Images were sent directly to the Server	0.1
Joe Chen	10/17/2019	Unnecessary login information asked while logging in the game was removed.	0.1
Siddhesh P	10/15/2019	Images were taken out from the GUI portion	0.1
Shubham M	10/10/2019	Card class was implemented to get the attributes of the card	0.1

Executive overview

UNO!!! Have you played UNO with your family or friend? If yes, you should know how fun it is. UNO is one of the most famous card games in the world. It is easy and simple to play with family and friends. According to research, an estimated 60 percent of households play the UNO card game at home. Therefore, we as a team are trying to make an online version of the game for the people who want to play UNO. Our version of the game only allows four players to play the game.

This Game is easy to play asking the users to place an appropriate choice card , also referred to as validation. In this, we are just doing the validation part when the user is having the same color card or when the number is same to the card in the middle, otherwise the user needs to take a card from the middle of the stack and wait for his next turn.

The user who ends up finishing his/her cards first will be declared as the winner and rest all the users will see a message over there screen and the game will be ended.

Audience

It is a Multi threaded Game designed using Sockets. It is a multiplayer game with having some modifications to the original UNO version. It allows the user to replace the card available in the middle by any card of the same color or the card having the same number of the card in the middle. Everytime a card is clicked, an object of a player class is instantiated and sent to the server. Server broadcasts and all the clients read this and update their Graphical User Interface.

The client who will be finishing up all his cards from the panel in the bottom will be declared as the winner. Also , the users having the option to chat with the other clients connected over the Server.

Application intentions

The purpose of this application - it lets multiple(4) clients play UNO.

The target audience for this application is everyone who likes to play UNO and who is interested in UNO.

Assumptions made for this project

- Exactly four players can play this game.
- Port number is 16789.
- Wait for all the clients to connect before the game starts.
- Default number of cards -> 6.

Gantt chart

Gantt task descriptions:

WBS	M	Tasks	Date (if known)
1	M	Assigned project ideas	09/6/2019
2		Individually think about possible projects	
3	M	Project ideas discussed in class	09/09/2019
3.1		Discuss projects with possible team(s) / team member(s)	09/11/2019
	M	Teams formed	09/13/2019
N	M	Design Document assigned	10/16/2019
N.1		*Design discussions/Team Meeting*	10/18/2019
N.2		*Discussions on sending and receiving the UNO card images over the server and client*	10/20/2019
N.3		*Designed the user interface (GUI)*	10/20/2019
		System Design	
	M	Team present project topic to class	10/23/2019
		GUI Prototype	10/30/2019
	M	Write project plan for lab, update design document	11/02/2019
	M	Design Documents due for review, end of class	11/09/2019
		Design discussions/Team Meeting	11/11/2019
		Design discussions/Team Meeting	11/14/2019
	M	Mid-project review: GUI with integrated chat	11/18/2019
		Teams write test document – due in class	11/25/2019
		System testing – update test document with results	
	M	Grid testing *{tentative}*	12/04/2019
		Design discussions/Team Meeting	
		Team by team review/evaluations with instructor	w10 d29-30
		Prepare presentation	
	M	Trade show (System Deployment)	*12/13/2019*

M = Milestones

IDEAS:

->Graphical User Interface: - The basic layout of the interface.

- The interface from each client's perspective.

- Components of the GUI.

->Networking: - Should the Client be threaded?

- Other classes(Player class)

- Should the client validate moves?

- What should the server do?

Class and method overview

Overview of the classes and functionality.

Project parameter Interface

- Contains Port number constants, used by Client and Server
- Contains Author and About messages

Client

- Main method
  - Can log on to the server
  - Can chat on server
  - Can play game without awkwardness
- Client constructor
  - Create GUI

  -Have a user-friendly GUI interface

  -Maintain a game board that will maintain state of game as provided by server

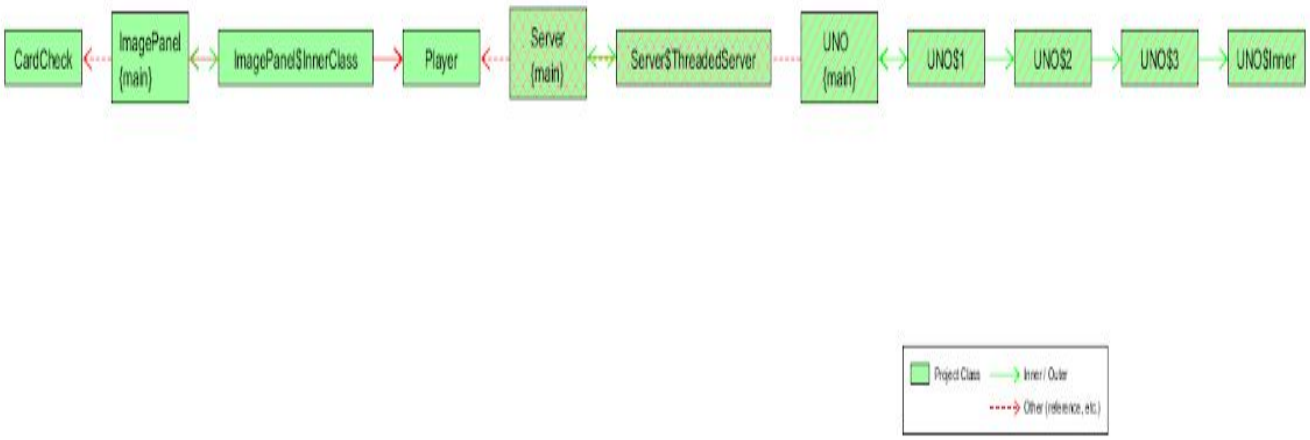
  -Have a readme file or GUI window pop-up to learn how to play the game

Server

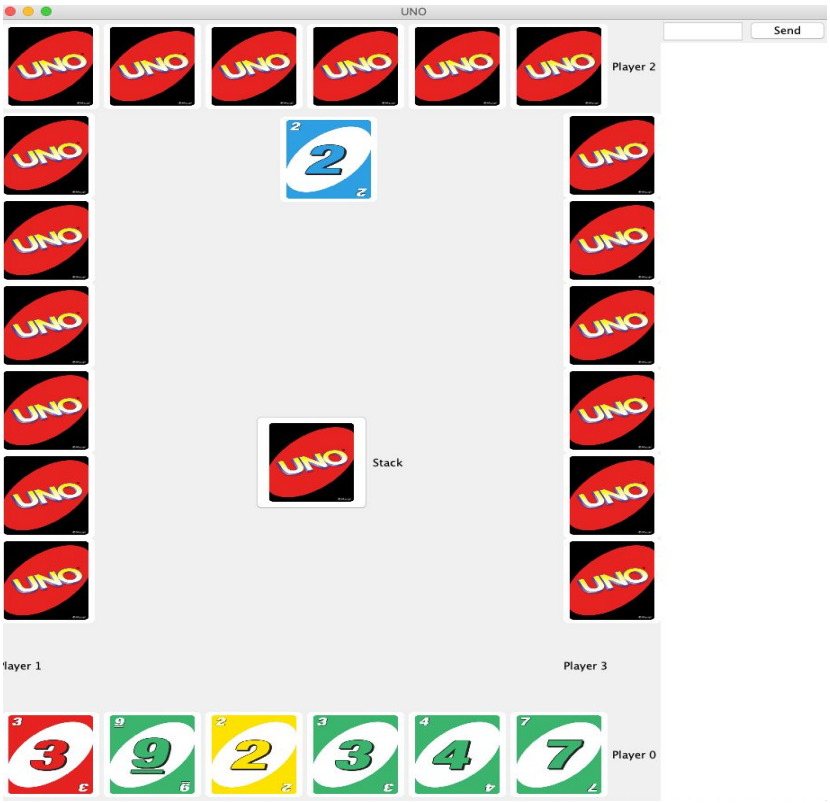
- Main method

    - Multiple Connections from client
    - Display a list of current clients logged on
    - Allow for public chat among clients
    - Maintains information about the game such that it can validate some moves.
    - Decides and Announces winner for game
- Server GUI
    - Displays information about the Users connected
    - Displays the winner
    - Shows messages being sent back and forth from the server.

<UNO> UML


CardCheck
ImagePanel (main)
ImagePanel$InnerClass
Player
Server (main)
Server$ThreadedServer
UNO (main)
UNO$1
UNO$2
UNO$3
UNO$Inner
Project Class
Inner / Outer
Other (reference, etc.)

Client GUI



Networking connections & Protocols

Client connection to server: 127.0.0.1

Port number(s): the localhost number is 16789

Protocol interface code for both client and server:

- Client uses TCP/IP protocol to communicate with the server.
- Thread is being generated whenever a new Client is connected to the server.
- Run method within the inner class is reading the objects from the clients and doing appropriate task accordingly.
- Server is accepting the client over with the object of class ServerSocket which is accepting the client details with predefined method called accept().
- Object of class Player is sent over the server through client which is Serializable as well to perform the various tasks.
- readObject and flush are the two main methods which are further used over the Client as well as the Server helping to send and receive the Objects.

Communication class

Chat interaction between Clients and Server.

The server starts first, then multiple clients connect to the server. Every client needs to input the IP Address and their nickname then press the login button to connect to the server. After that, it will jump to the UNO game panel. On the left side is the game and the right side is the chat. Players can chat among themselves in the game. The server receives player chat messages and information like which player has played what card. The server broadcasts these messages to all the clients. The clients get data back from the server and act accordingly. For example, if the clients receive a string, it adds it to the text area.

Client	Communication	Server
		Startup
		Waits for client to connect
Client connection	Connection, no data	Accept connection
Client sends an object	Instance of Player class	Server reads info
Clients receive an object	Object	Server broadcasts
Client sends a String	Text message to add it in the text area	Server reads info
Clients receive a String	String	Server broadcasts
Client sends a Double	Double for Welcome message	Server reads info
Clients receive a Double	Double	Server broadcasts
Client sends an Integer	Integer for deciding winner	Server reads info
Clients receive an Integer	Integer	Server broadcasts

Data used

We have 40 UNO images in the folder.

Data files

An external folder we use is the folder of UNO images (files).

Punch List

To do:

- UNO special cards like draw 2, wild, draw 4, UNO button

Done:

So far, we have done almost 90% of the project.

- Client receives message from another client and send the message back – fixed on 11/25/19

- Card button when pressed, get updated in the middle of the GUI. Card is sent to the center – fixed 11/25/19

- Game starts once all users are connected – fixed 11/26/19

- Login to the client with the correct IP address and also can input their name/nickname – fixed 11/26/19

- GUI menu -> about me -> read me: explain to the user how to play "UNO". – fixed 12/8/19

- Announces whose turn to play the game in the beginning (which player go first/play first) – fixed 12/7/19

-Functionality of stack of cards button 12/9 / 19

- Card removed (not show up) from other clients' GUI when someone plays. – fixed 12/ 10/19

- Decides and announces winner - fixed 12/10 /19

Unresolved Issues

- UNO special cards like draw 2, wild, draw 4, UNO button.