Chat-Room Adventure

San José State University

Computer Engineering 133: Software Engineering II

Group I

Leader: Uyen Nguyen

Members: Brian Albert Redoloza, Dhirtitapa(Risha) Ray, Yenni Lam

Overview

We are making a choose-your-own-adventure game that is a web application.

We will be looking into:

- Our cost estimates
- Our class relations
- Our sequence diagrams

Based on the following use cases:

Login/Logout, Play Game, Save Game

Group 1

Login/Logout: Unadjusted FP Calculations

	Count	**Simple**	Count	**Medium**	Count	**Complex**	SubTotal
EI	2	3	0	4	0	6	6
EO	1	4	0	5	0	7	4
EIN	1	3	0	4	0	6	3
IFL	1	7	0	10	0	15	7
EIF	0	5	0	7	0	10	0

Comments:

EI: password, username

EO: menu display

EIN: verify user type

IFL: data about user

Total: 20

Play Game: Unadjusted FP Calculations

	Count	**Simple**	Count	**Medium**	Count	**Complex**	SubTotal
EI	1	3	1	4	1	6	13
EO	1	4	0	5	0	7	4
EIN	1	3	0	4	0	6	3
IFL	1	7	1	10	0	15	17
EIF	0	5	0	7	0	10	0

Comments:

EI: player name, player choice, save file input

EO: game display

EIN: game menu option

IFL: data about user and save file

Total: 37

Save Game: Unadjusted FP Calculations

	Count	**Simple**	Count	**Medium**	Count	**Complex**	SubTotal
EI	0	3	0	4	1	6	6
EO	0	4	0	5	0	7	0
EIN	0	3	0	4	0	6	0
IFL	0	7	1	10	0	15	10
EIF	0	5	0	7	0	10	0

Comments:

EI: save input

IFL: data about save file

Total: 16

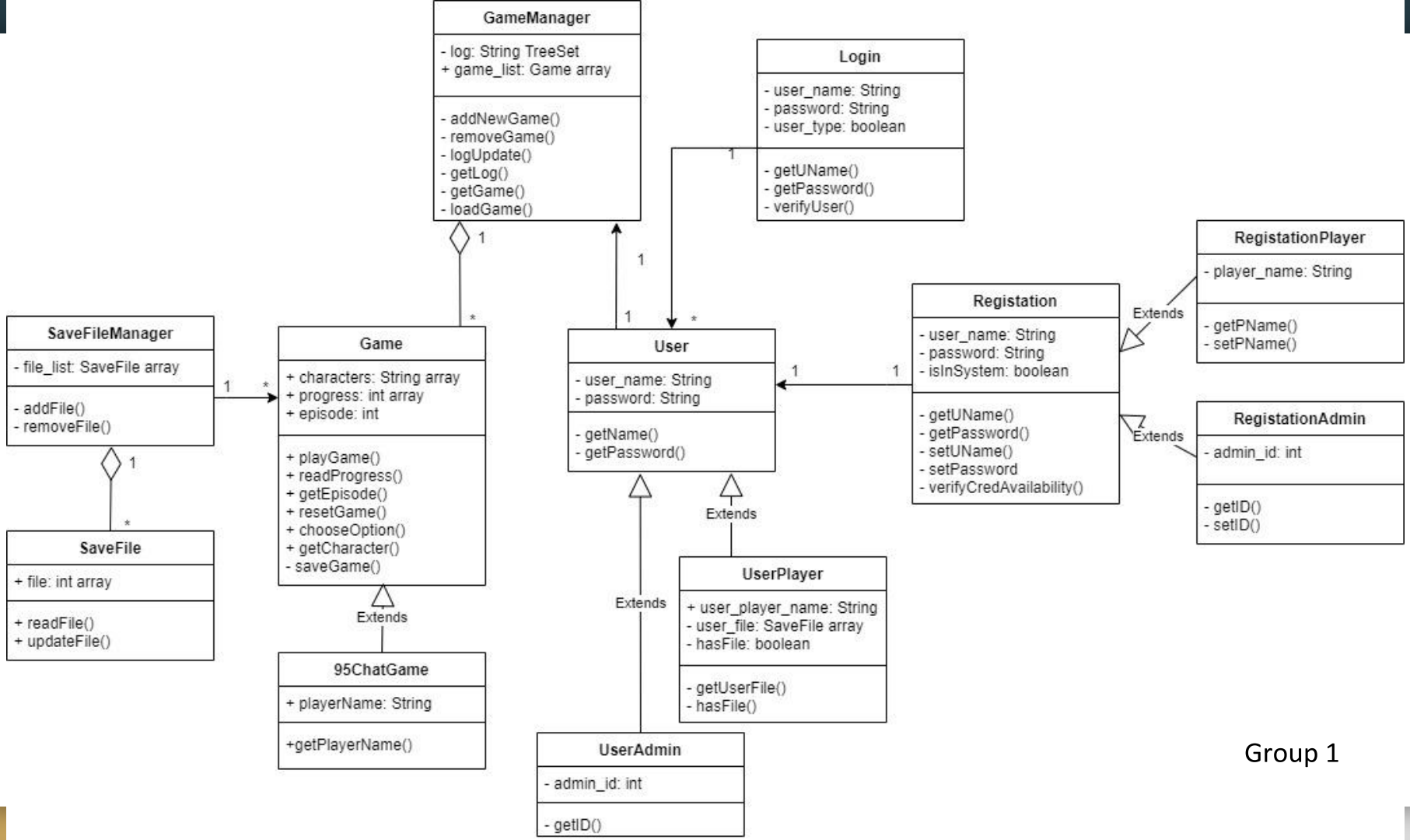
General Characteristics Calculations

Characteristics	Score	Characteristics	Score
1	5	8	1
2	1	9	1
3	1	10	1
4	5	11	3
5	1	12	1
6	1	13	1
7	1	14	2

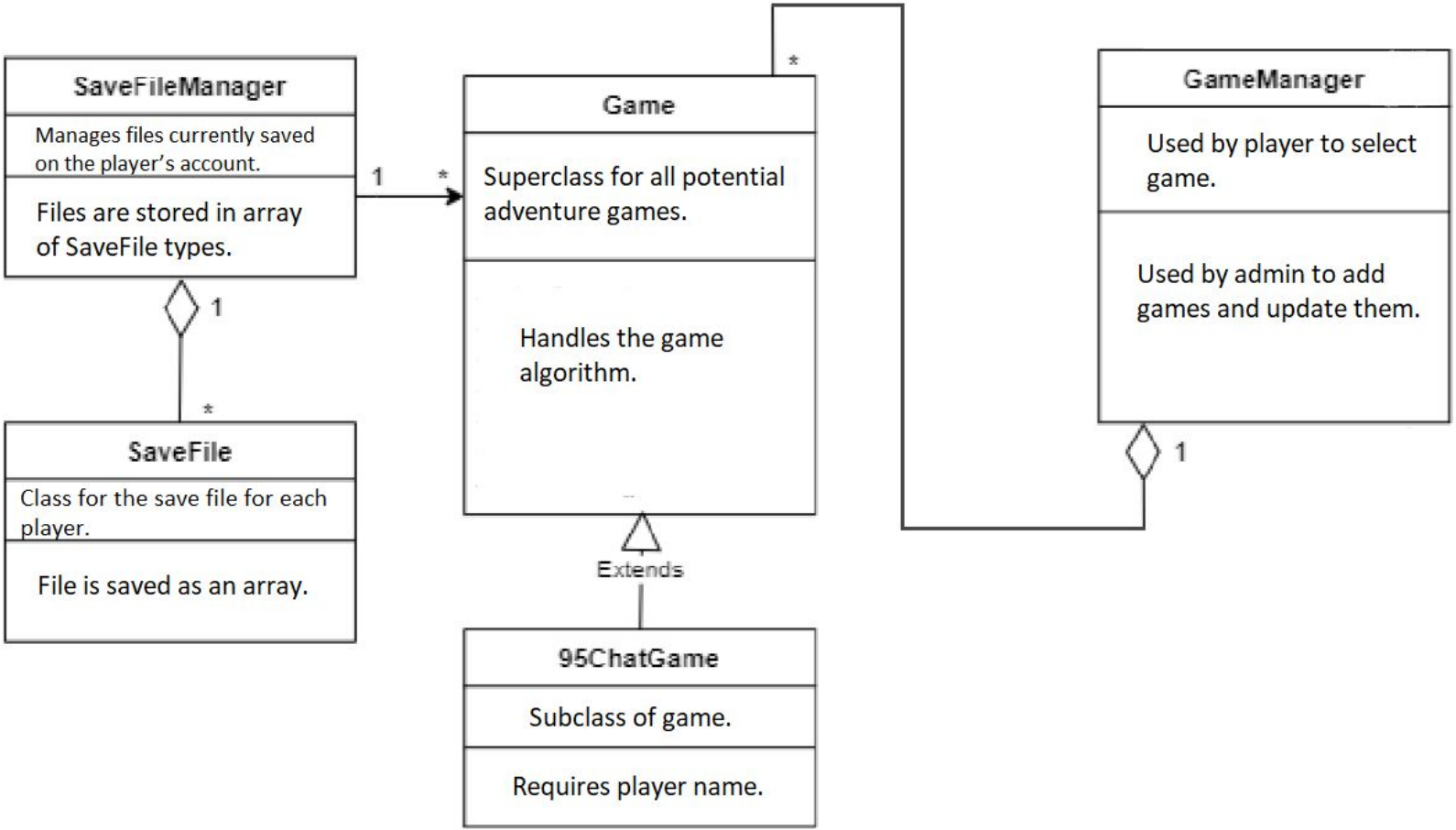
Total: 25

Adjusted Function Points (AFP)

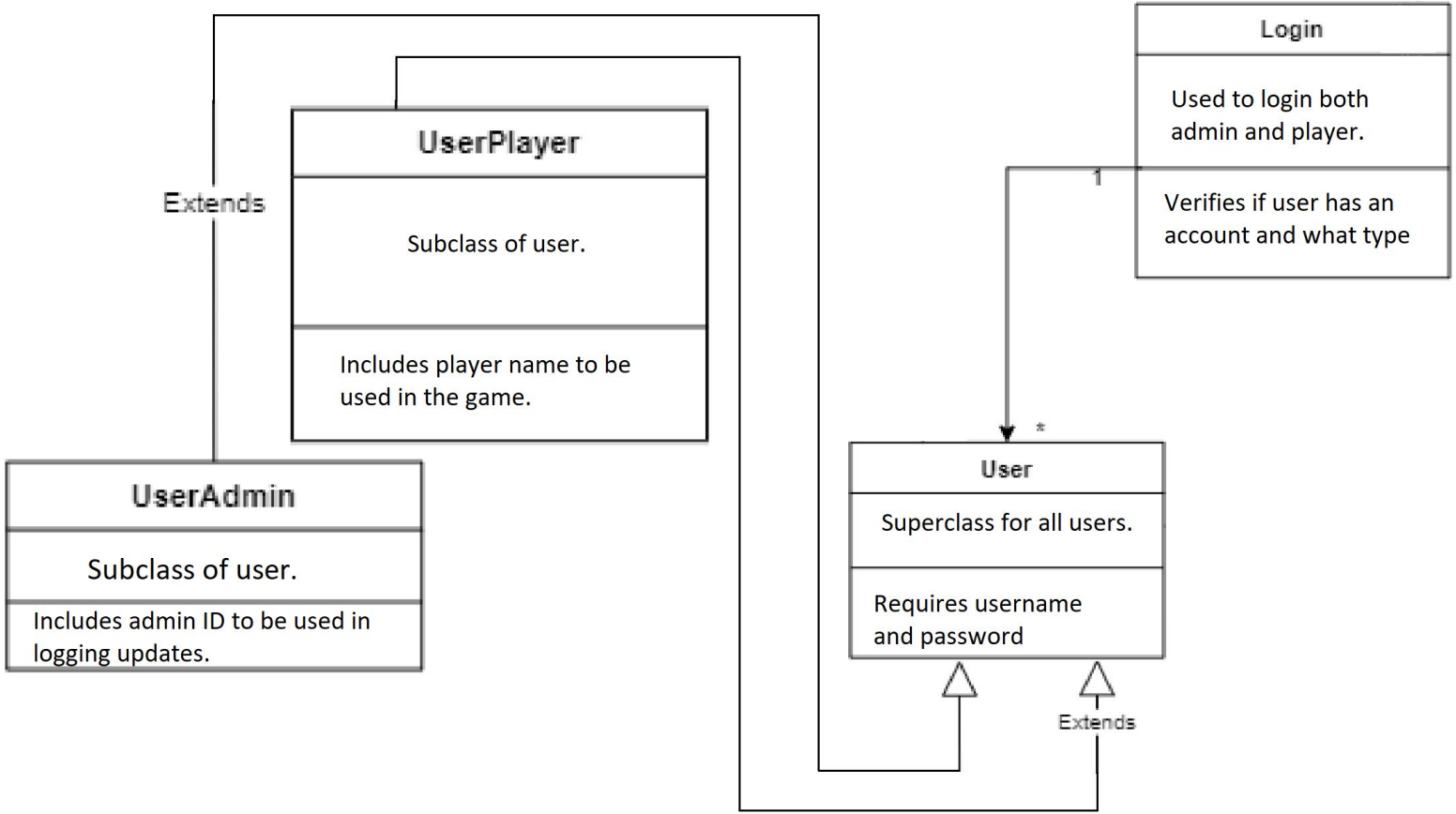
- AFP = (UFP) X (0.65 + 0.01 X GC)
- AFP = (20+37+16) X (0.65 + 0.01 X 25)
- AFP = 65.7


GameManager
- log: String TreeSet
+ game_list: Game array
- addNewGame()
- removeGame()
- logUpdate()
- getLog()
- getGame()
- loadGame()
Login
- user_name: String
- password: String
- user_type: boolean
- getUName()
- getPassword()
- verifyUser()
RegistationPlayer
- player_name: String
- getPName()
- setPName()
Registation
- user_name: String
- password: String
- isInSystem: boolean
- getUName()
- getPassword()
- setUName()
- setPassword
- verifyCredAvailability()
RegistationAdmin
- admin_id: int
- getID()
- setID()
SaveFileManager
- file_list: SaveFile array
- addFile()
- removeFile()
SaveFile
+ file: int array
+ readFile()
+ updateFile()
Game
+ characters: String array
+ progress: int array
+ episode: int
+ playGame()
+ readProgress()
+ getEpisode()
+ resetGame()
+ chooseOption()
+ getCharacter()
- saveGame()
95ChatGame
+ playerName: String
+getPlayerName()
User
- user_name: String
- password: String
- getName()
- getPassword()
UserPlayer
+ user_player_name: String
- user_file: SaveFile array
- hasFile: boolean
- getUserFile()
- hasFile()
UserAdmin
- admin_id: int
- getID()
Extends
Extends
Extends
Extends
1
*

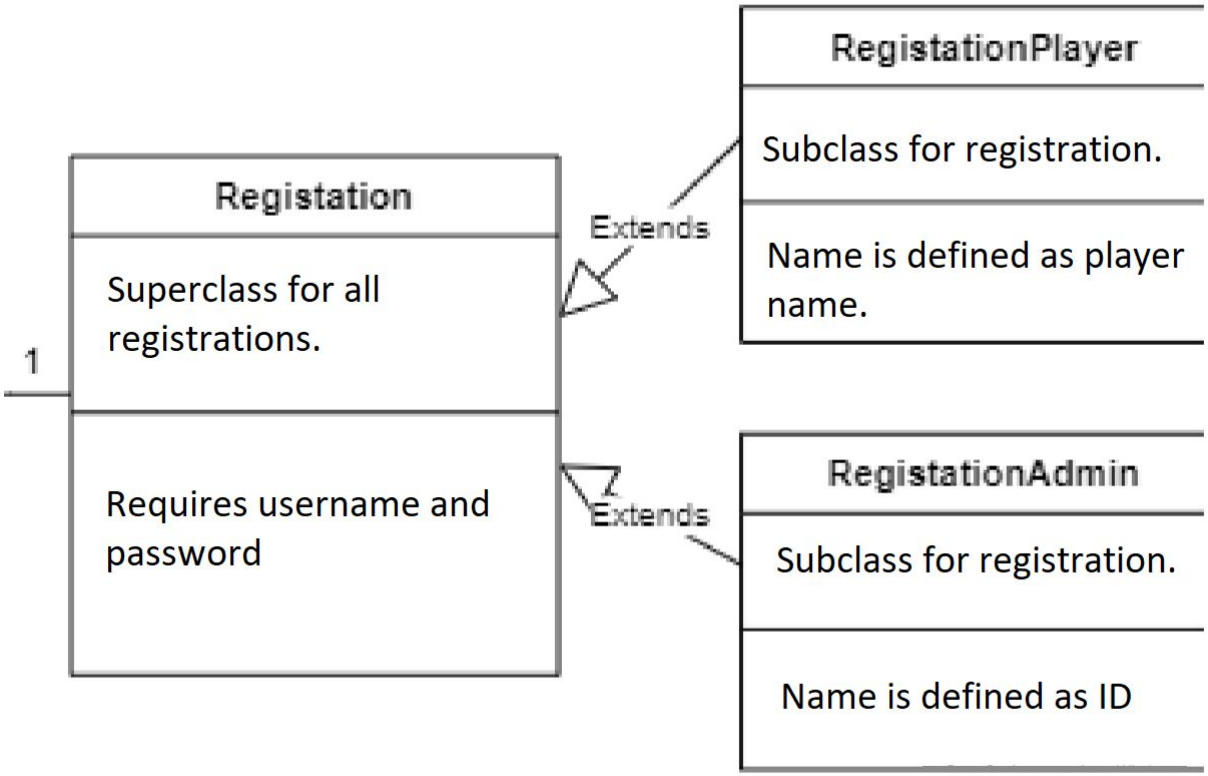
Group 1


SaveFileManager
Manages files currently saved on the player's account.
Files are stored in array of SaveFile types.
1
*
1
*
SaveFile
Class for the save file for each player.
File is saved as an array.
Game
Superclass for all potential adventure games.
Handles the game algorithm.
*
Extends
95ChatGame
Subclass of game.
Requires player name.
GameManager
Used by player to select game.
Used by admin to add games and update them.
1

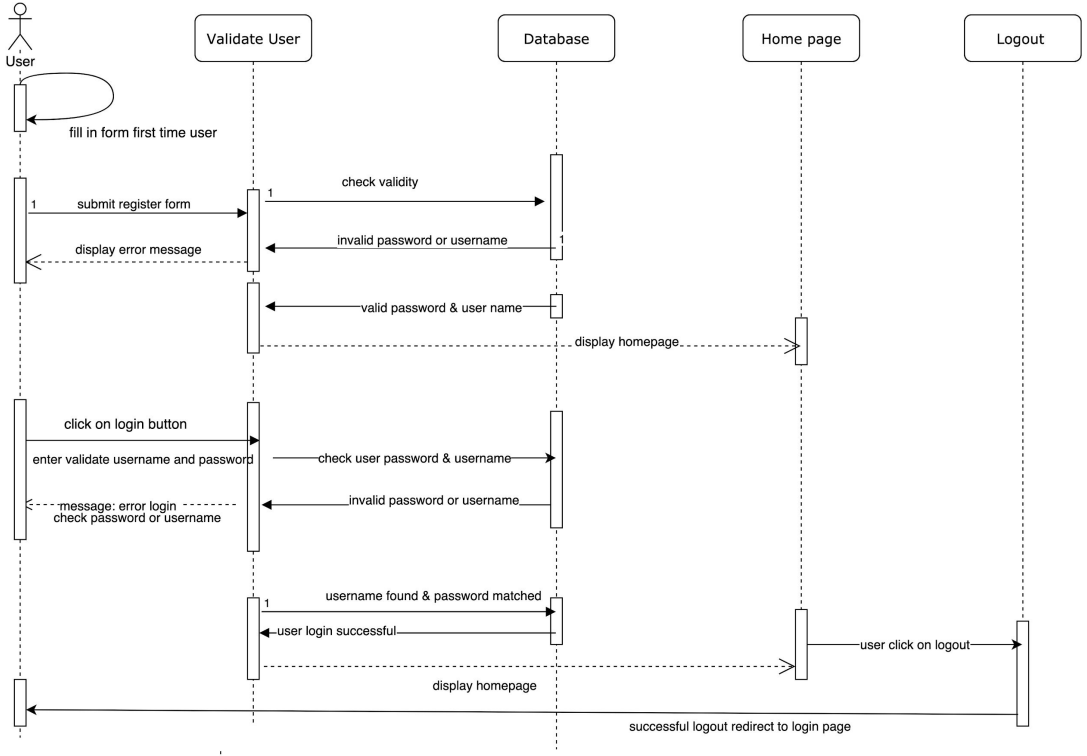
Group 1


Login
Used to login both admin and player.
Verifies if user has an account and what type
1
*
UserPlayer
Subclass of user.
Includes player name to be used in the game.
Extends
UserAdmin
Subclass of user.
Includes admin ID to be used in logging updates.
User
Superclass for all users.
Requires username and password
Extends

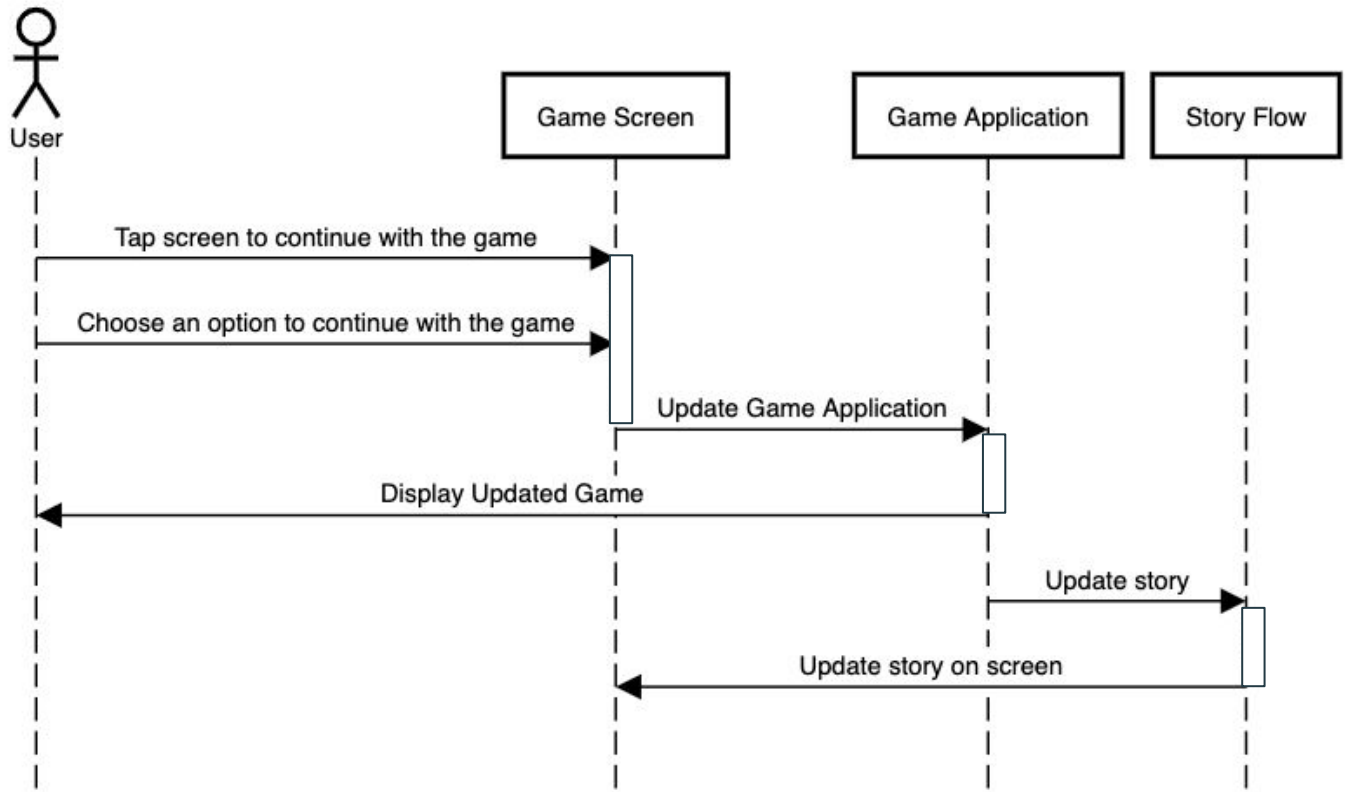
Group 1


Registation
Superclass for all registrations.
Requires username and password
1
Extends
Extends
RegistationPlayer
Subclass for registration.
Name is defined as player name.
RegistationAdmin
Subclass for registration.
Name is defined as ID

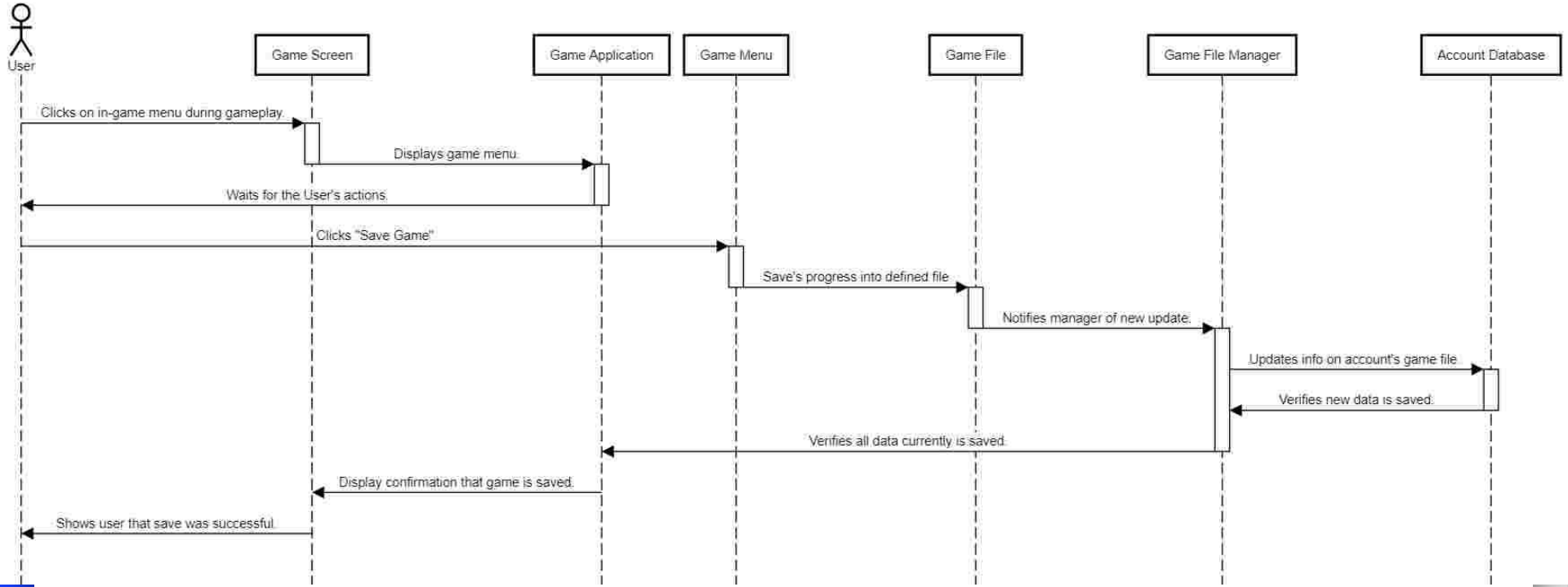
Login/ Logout Sequence Diagram



Play Game Sequence Diagram



Game Save Sequence Diagram



QUESTIONS?