

Examination Seat No: _____

CERTIFICATE

**Mumbai Educational Trust
Institute of Computer Science**

**M.C.A. Sem-III
2020-2021**

This is to certify that Ms./Mr. Paras Gherwada of
MCA Sem-III (CBCGS) Roll no 1013 has successfully completed
practical assignments in the subject of MCAPR301 MINI PROJECT
during academic year 2020-2021 as specified by the UNIVERSITY
OF MUMBAI.

Faculty in-charge

Principal

External examiner

INDEX

Topics	Page No
1. Introduction	1
a. Introduction of the project (SRS)	1
b. Problem definition	1
c. Objective of Project	1
d. Scope of Project	1
2. System Study	2
a. Existing System	2
b. Disadvantages of Existing system	2
c. Proposed System	2
d. Use Cases	2
3. Analysis & Design	3
a. Software/Hardware Requirement Specification	3
i. Software requirement	3
ii. Hardware requirement	3
b. GANTT Chart	3
c. Flowchart	4
d. Module design and organization	5
4. Testing & Validation	6
a. Test cases and Report	6
5. User Manual	7
a. Explanation of key functions	7
b. Method of implementation	8
i. Forms	8
ii. Output screens	9
6. Conclusion	21
a. Project conclusion & Future enhancement	21
7. References	21

1. Introduction

a. Introduction of the project (SRS)

"Space Shooter" is a classic game that has been popular around the world for more than 3 decades. It has been developed for various gaming platforms like PC, Xbox, Play Station, Sega, Nintendo and for cell phones as well. The game revolves around a space ship which has embarked on a mission to eliminate as many asteroids and alien spaceships as possible and gather points for each successful elimination.

b. Problem definition

The project *"Space Shooter"* has been undertaken with the aim to get started and pursue further interests in the field of Game development. The project is fully developed using Java and uses Postgres as database.

c. Objective of Project

- Gain knowledge about java graphics libraries (example: javafx)
- Create a space shooter game using java.
- Randomly spawn asteroids and enemy ships towards the protagonist's space ship.
- Entertainment.

d. Scope of Project

The end product of this project will be a miniature version of the classic *"Space Shooter"* game developed using java. The project can further be improved by hosting it over a server to provide web-based gameplay. Also, improvement can be made by allowing user to customize their space ship, bullet and changing the background.

2. System Study

a. Existing System

Existing Systems do not possess any inconvenience to the player but lack one aspect of further improvements and improvisations. In contrast, this project fully allows the user to modify every aspect of the game as per their desires (the user can also add cheat codes).

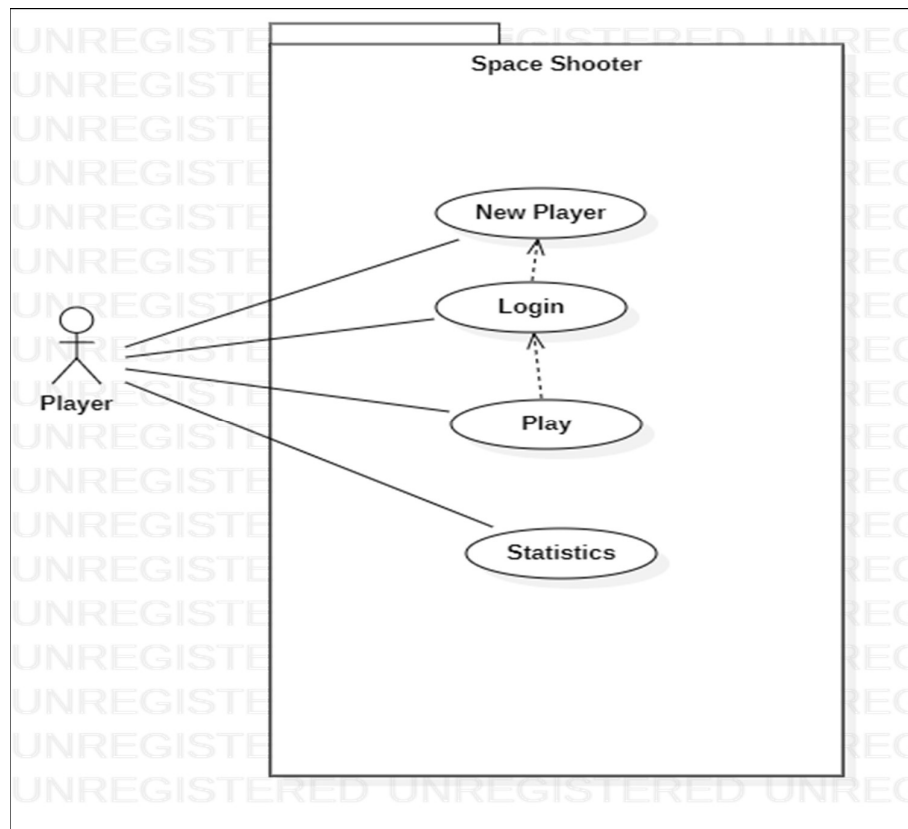
b. Disadvantages of Existing system

- Limited/Lack of customization

c. Proposed System

- User is free to customize the game to their liking
- Open source
- Rich database connectivity
- Doesn't require high end support specifications

d. Use Cases



3. Analysis & Design

a. Software/Hardware Requirement Specification

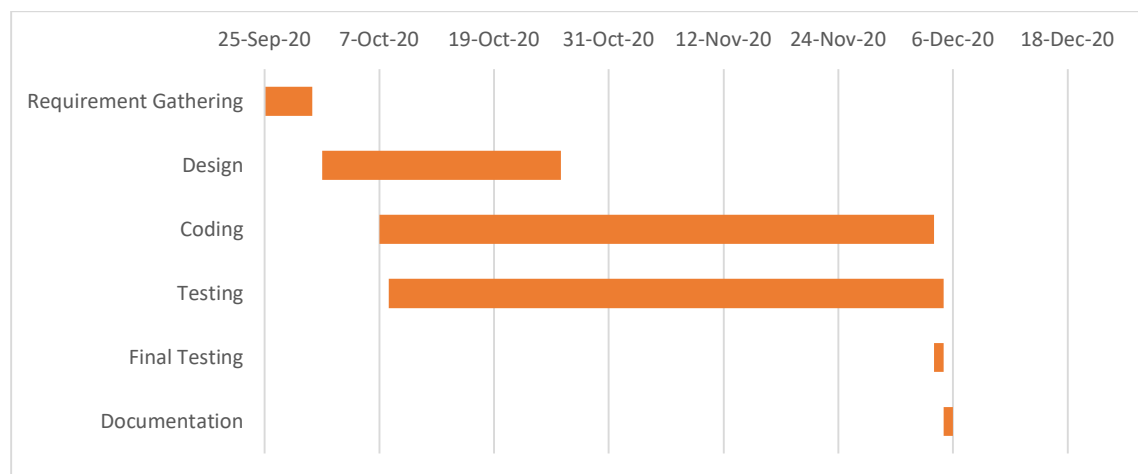
i. Software requirement

- Any OS
- JavaSE (jdk 14.0.2 or jre 8u261) or above
- Javafx external library (11.0 or above)
- Postgres database (11.0 or above)

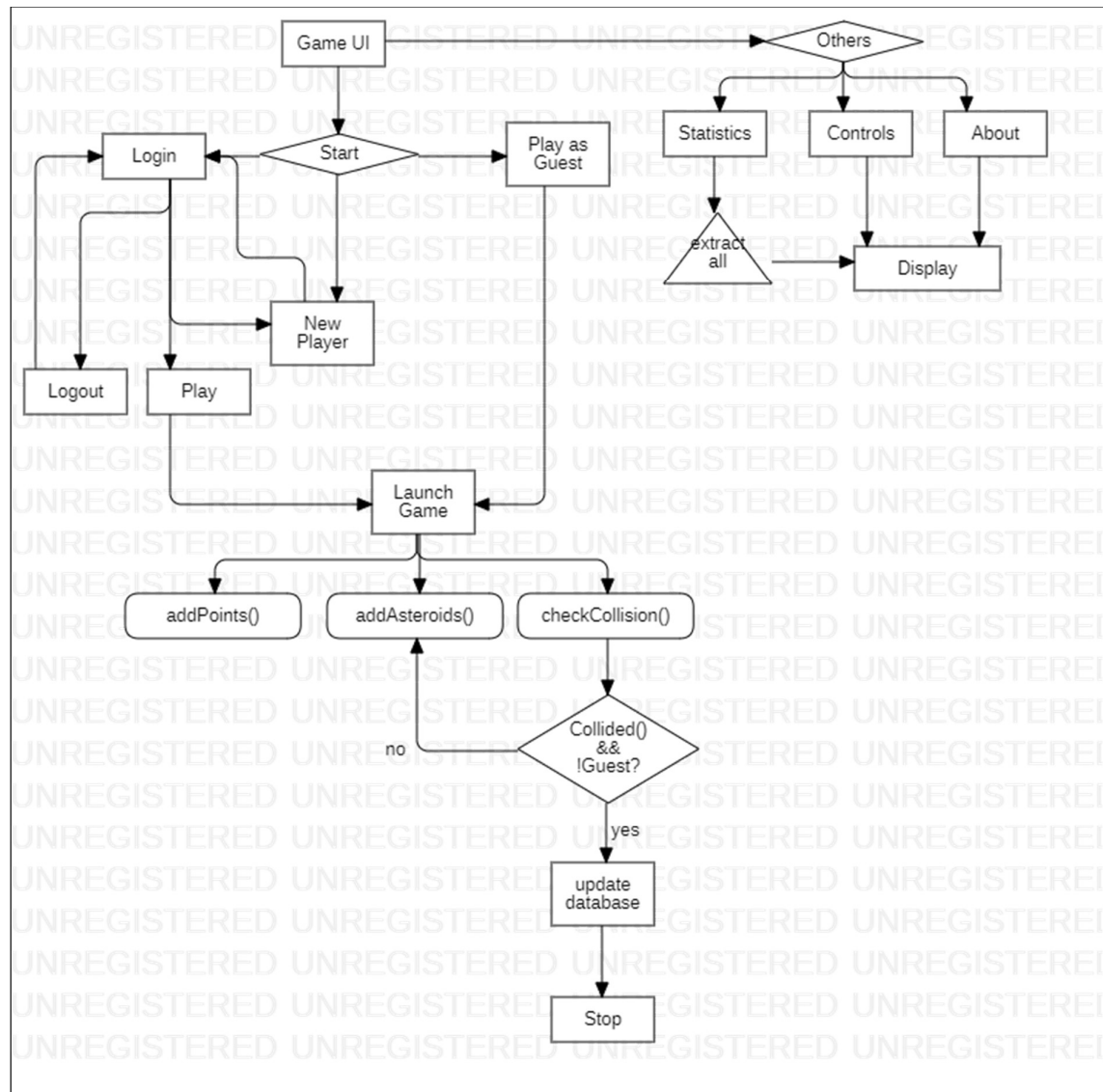
ii. Hardware requirement

- A computer/laptop

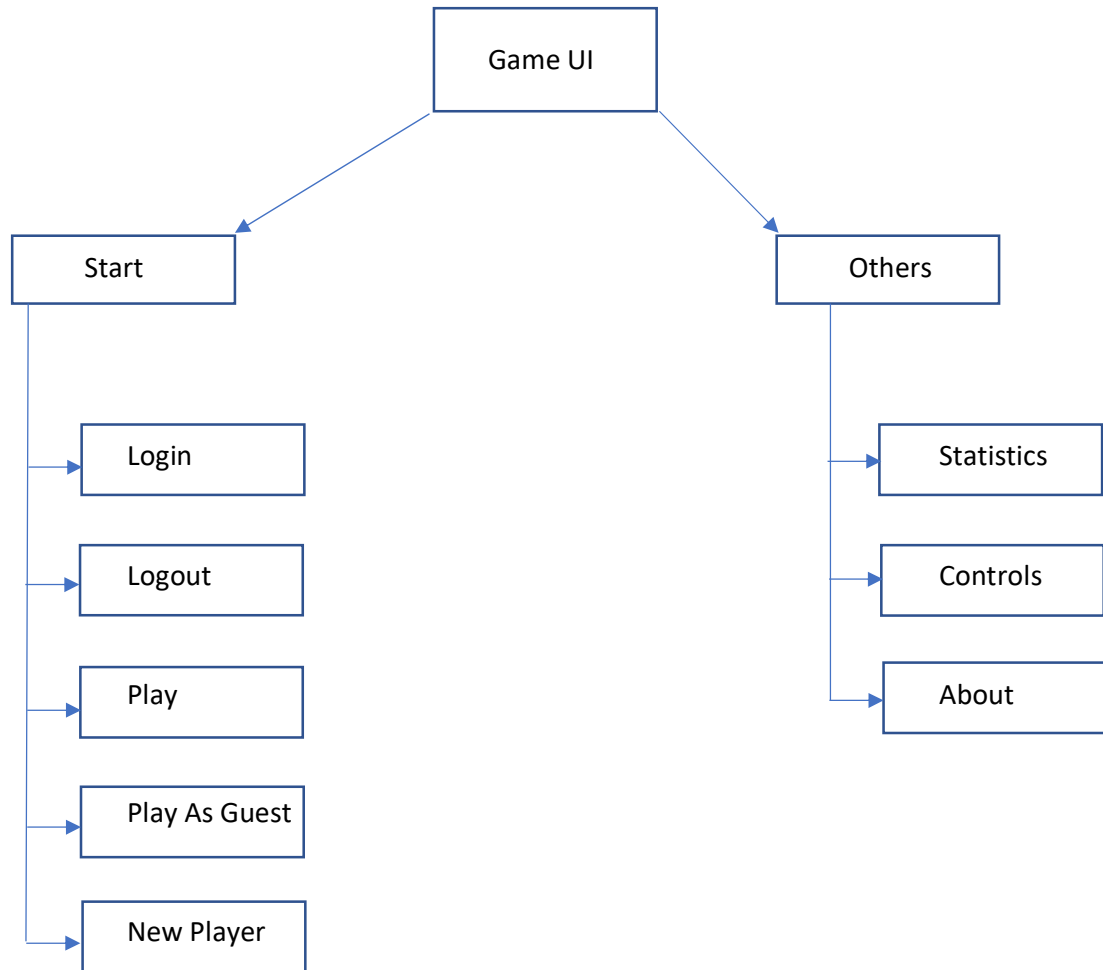
b. GANTT Chart



c. Flowchart



d. Module design and organization



4. Testing & Validation

a. Test cases and Report

Id	Pre-Condition	Input	Expected Output	Actual Output	Status
1	Login	Invalid credentials	Display error message	Display error message	Pass
2	Login	Valid credentials	Update title with username	Update title with username	Pass
3	New User	No values	Display error message	Display error message	Pass
4	New User	Valid values	Add user to database	Add user to database	Pass

5. User Manual

a. Explanation of key functions

Class	Method	Description
SpaceShooterMain	display(String)	Takes file name as parameter in string format. Displays contents of file based on selection from menu items. Example: Controls, About
	addPlayer()	Prompts user for username and password which can later be used to login and track progress.
	viewStatistics()	Displays data of all users (except Guest) in descending order of their high scores.
	getLoginInformation()	Prompts user for login credentials to play game. Updates title bar of application with username if login is success.
	initGame(String)	Launches main Space Shooter application. Takes username as parameter.
DatabaseConnector	createConnection()	Creates connection with the postgres database
	validateUsername(String) validatePassword(String)	Both methods take username and password respectively and validate them. Returns true if valid else false.
	getPoints(String) getAsteroids(String) getDeathCount(String) getStatistics()	All methods take username as parameter and return high score, asteroids destroyed & death count of user. getStatistics() returns set containing progress of players.
	updatePoints(String, int,int,int)	Takes 4 parameters with first one as username and other three as points, asteroid count and death count. Updates the points, asteroids & death count of user.
GameLoader	start(Stage)	Method of Application class overridden to imply custom actions and gameplay. Is automatically called by the launch(args) method provided by Application class.
	addAsteroid(Pane)	Takes pane (similar to JPanel in SWING) as argument. Adds an asteroid(polygon) to the pane.
	addPoints(int)	Keeps a count of points scored by the player while in-game.

b. Method of implementation

i. Forms

Adding new player

The first screenshot shows an 'Input' dialog box titled 'Provide Username'. It contains a green question mark icon, a text input field with 'paras' entered, and 'OK' and 'Cancel' buttons.

The second screenshot shows an 'Input' dialog box titled 'Provide Password'. It contains a green question mark icon, a text input field with 'paras1998' entered, and 'OK' and 'Cancel' buttons.

The third screenshot shows a 'Message' dialog box with a purple information icon and the text '1 player added'. It has an 'OK' button.

Login

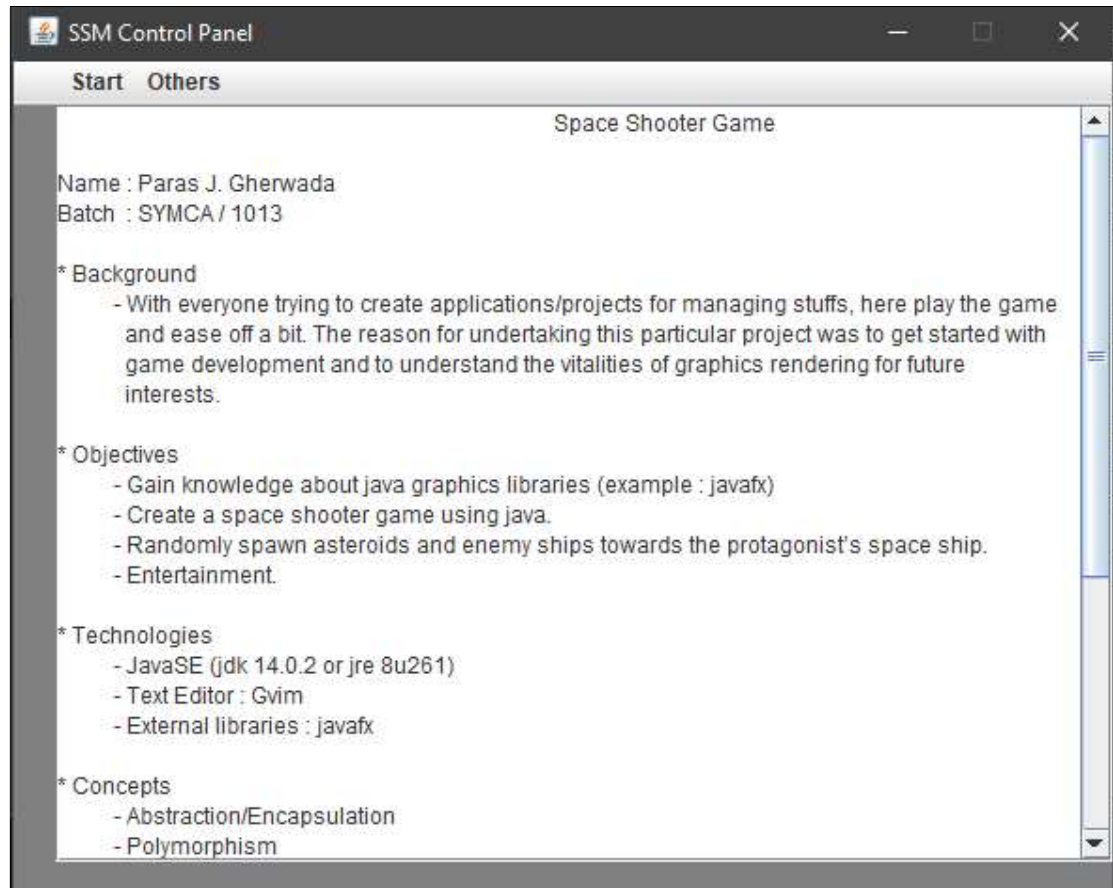
The first screenshot shows an 'Input' dialog box titled 'Username'. It contains a green question mark icon, a text input field with 'paras' entered, and 'OK' and 'Cancel' buttons.

The second screenshot shows an 'Input' dialog box titled 'Password'. It contains a green question mark icon, a text input field with 'paras1998' entered, and 'OK' and 'Cancel' buttons.

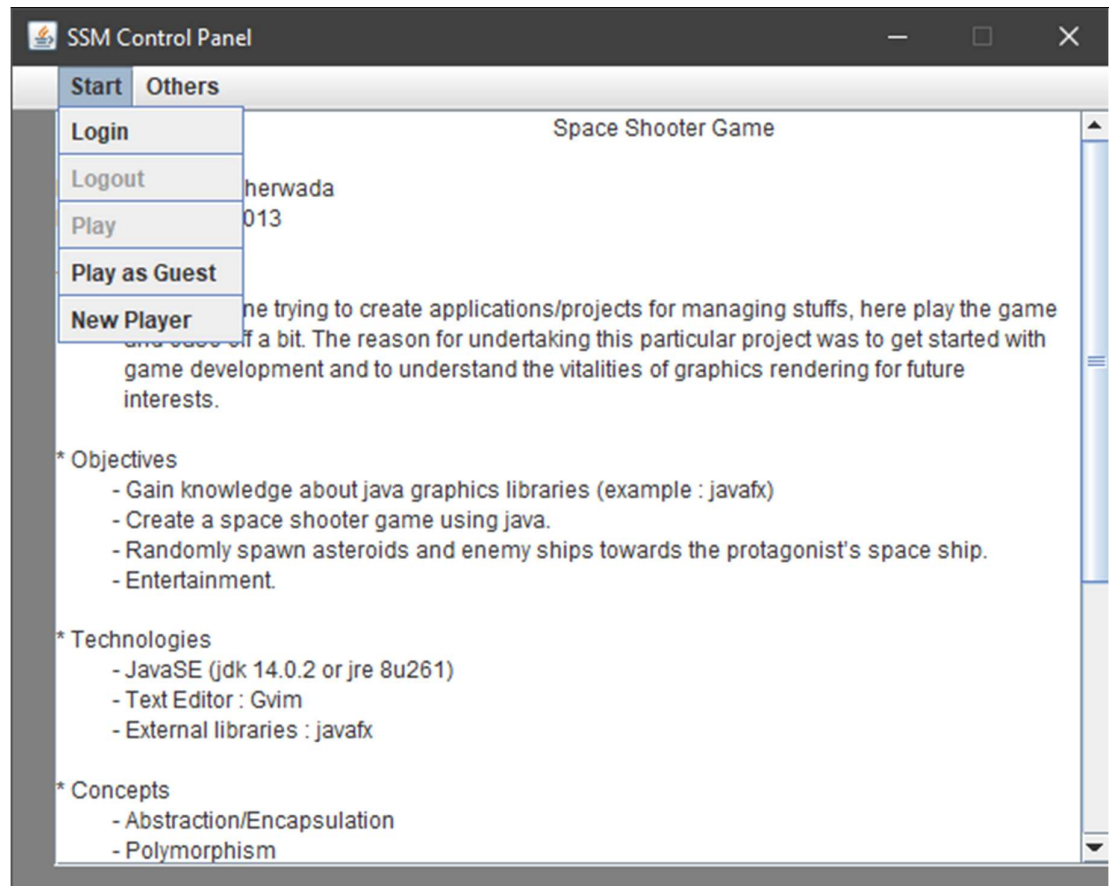
The third screenshot shows the application window titled 'SSM Control Panel - [paras]'. It has a menu bar with 'Start' and 'Others'.

ii. Output screens

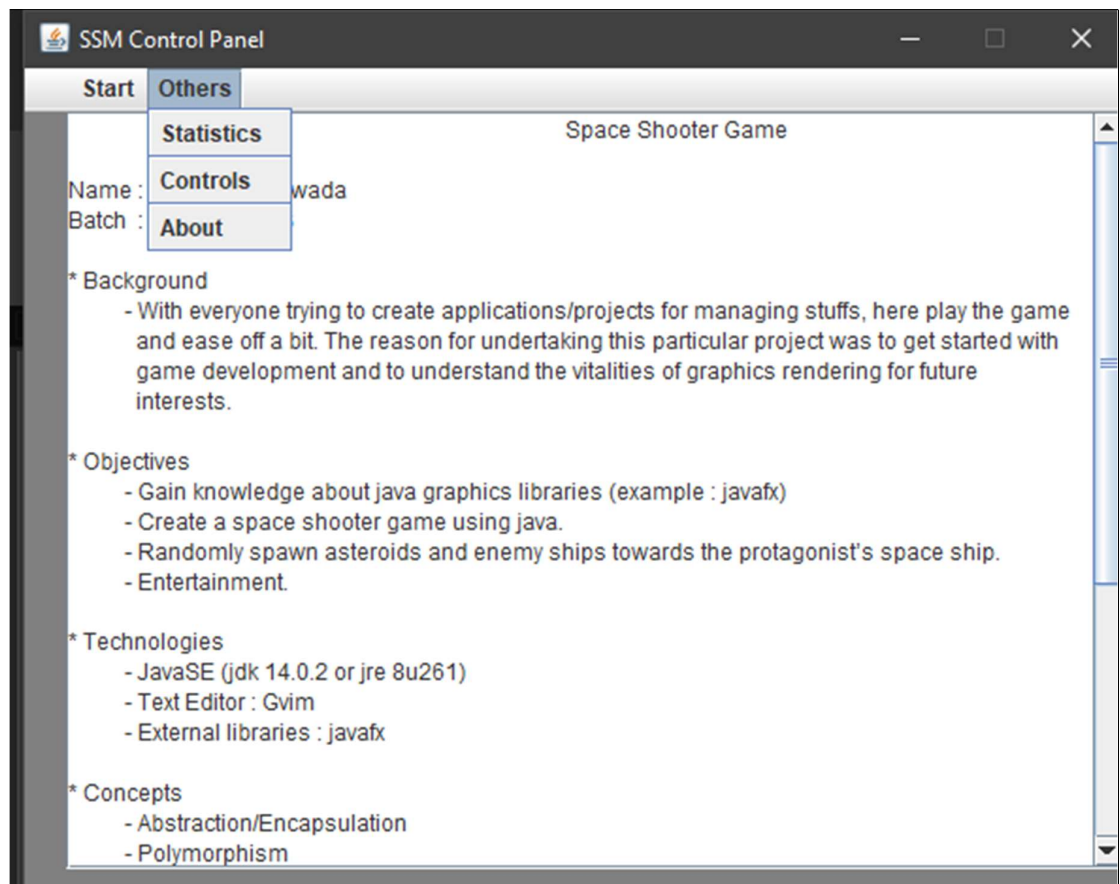
Start Up Screen



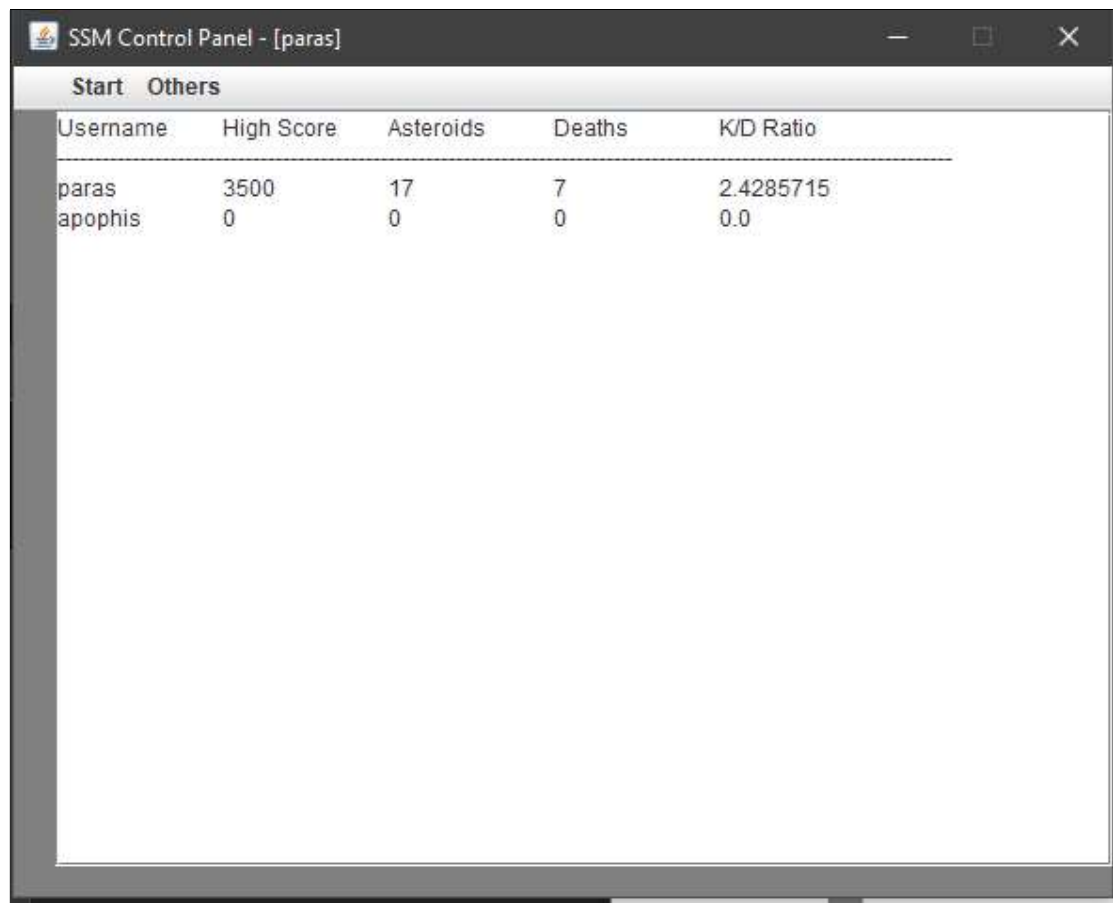
Start Menu



Others Menu



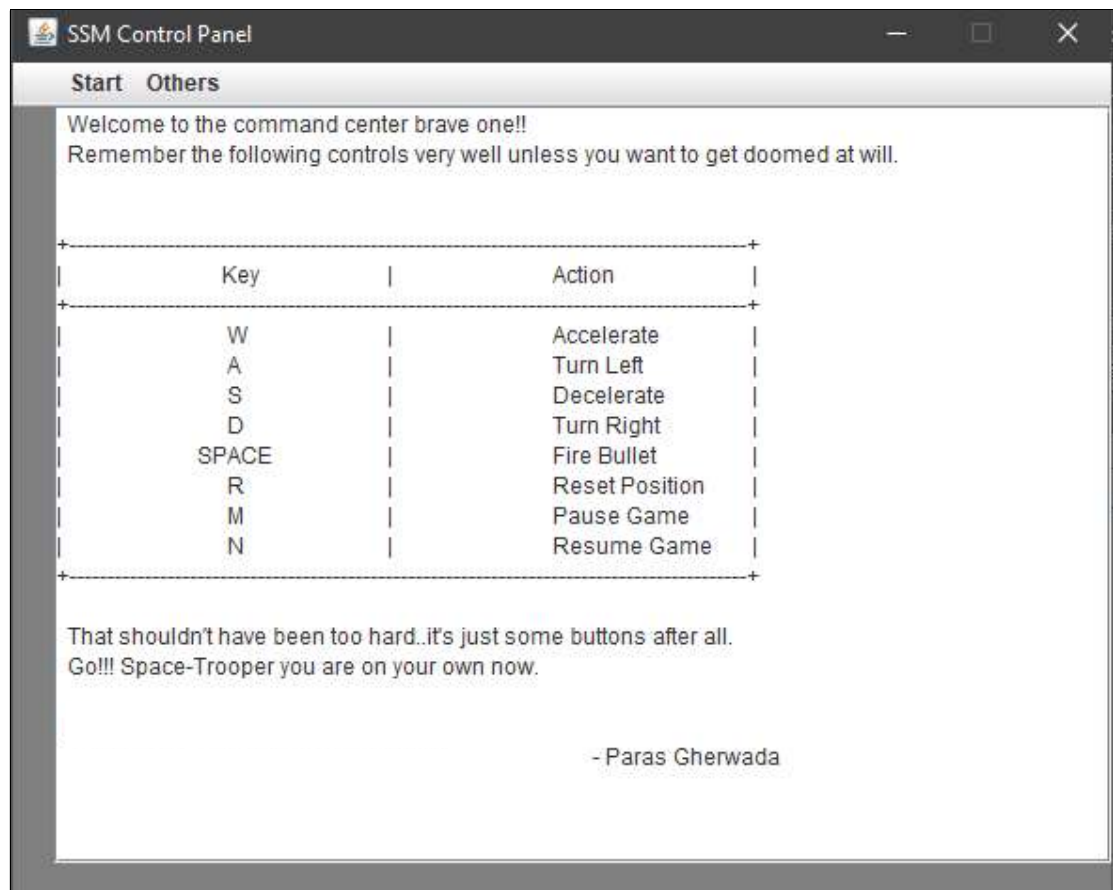
Others Menu Item – Statistics



The screenshot shows a window titled "SSM Control Panel - [paras]" with a dark title bar. Inside, there are two tabs: "Start" and "Others". The "Others" tab is active, displaying a table with five columns: "Username", "High Score", "Asteroids", "Deaths", and "K/D Ratio". The table contains two rows of data. The first row shows "paras" with a High Score of 3500, 17 Asteroids, 7 Deaths, and a K/D Ratio of 2.4285715. The second row shows "apophis" with a High Score of 0, 0 Asteroids, 0 Deaths, and a K/D Ratio of 0.0.

Username	High Score	Asteroids	Deaths	K/D Ratio
paras	3500	17	7	2.4285715
apophis	0	0	0	0.0

Others Menu Item – Controls



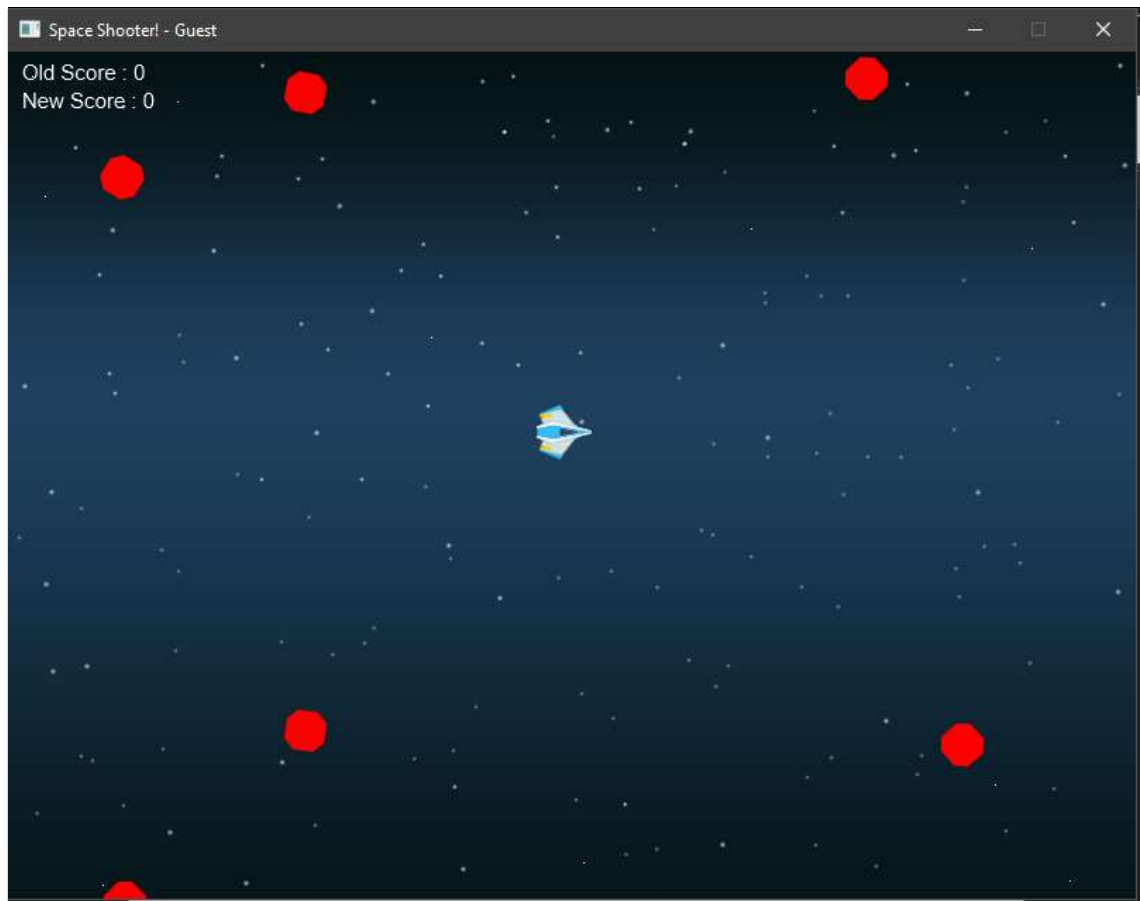
Others Menu Item – About

** Same as Start Up Screen **

Start Menu Item – New Player

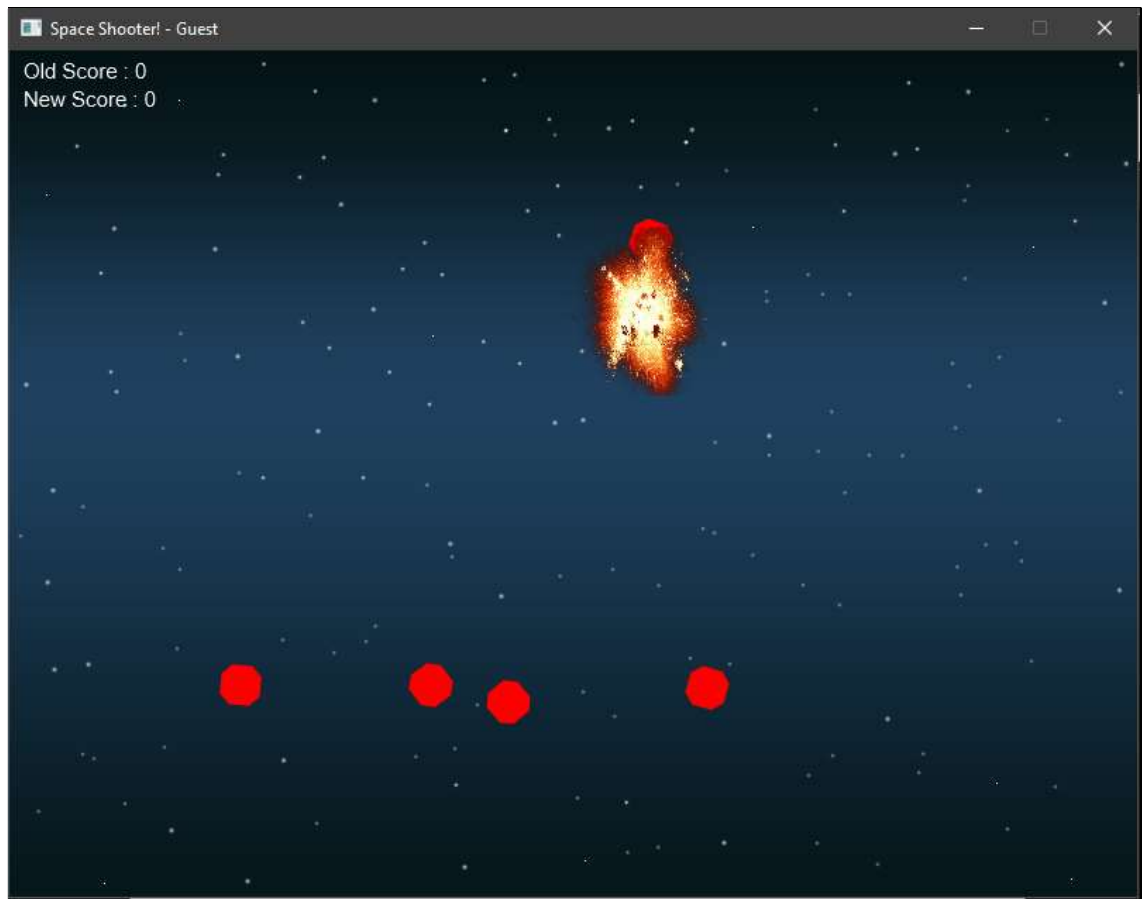
** Refer section 5.b.i **

Start Menu Item – Play As Guest (in-game)



Note: The application does not store any kind of progress of Guest user.

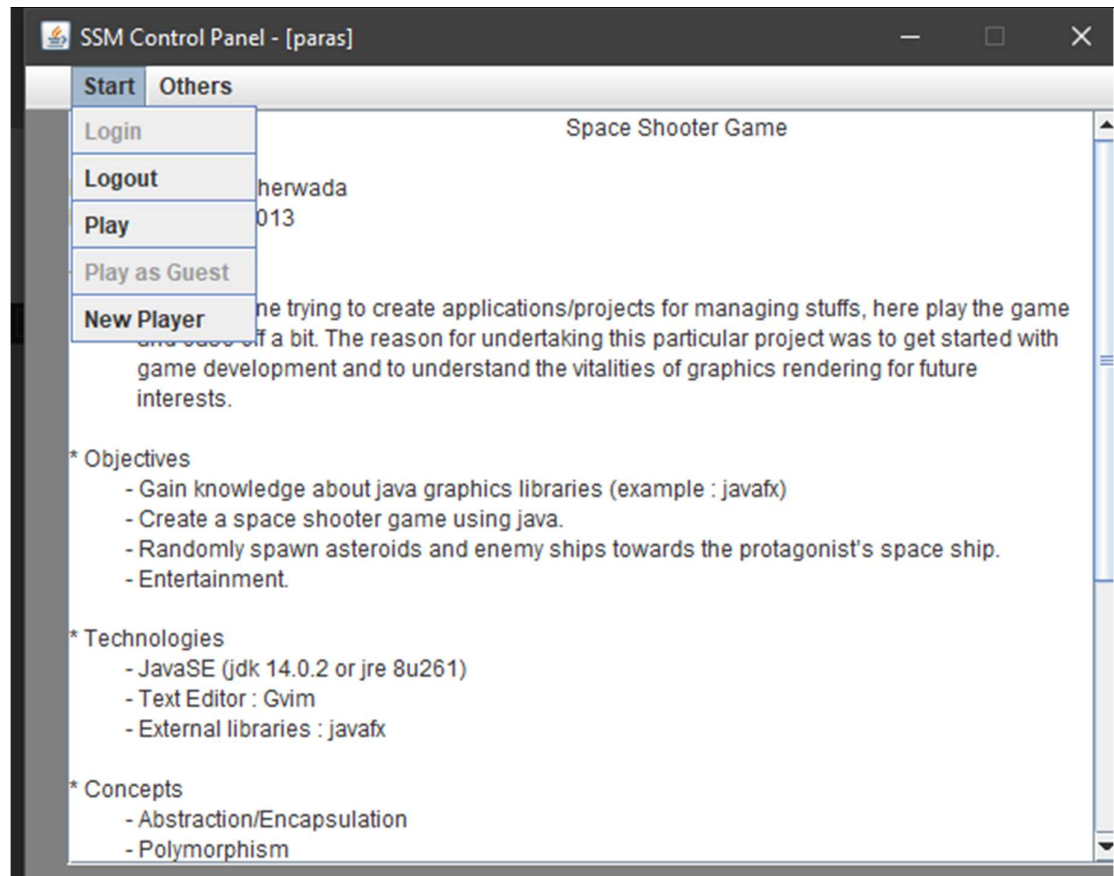
Start Menu Item – Play As Guest (on collision with asteroid)



Start Menu Item – Login

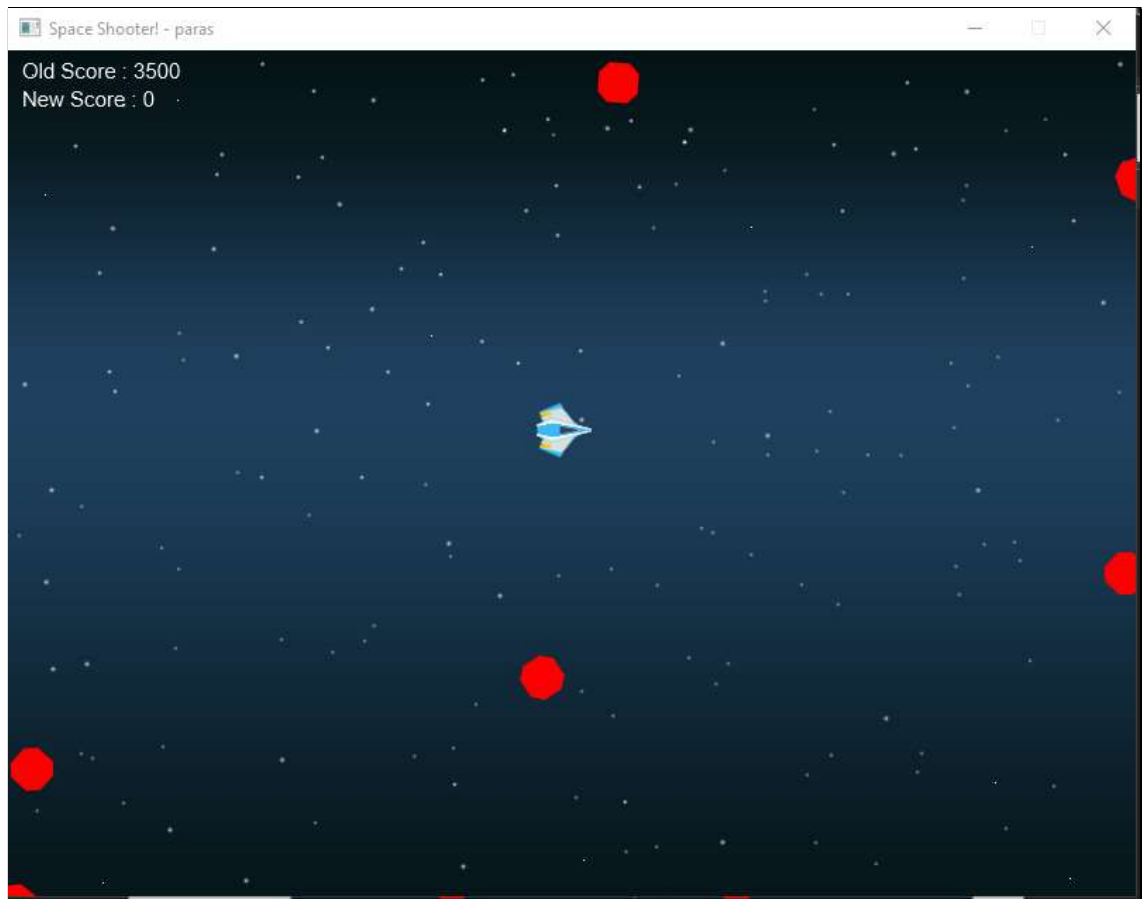
** Refer section 5.b.i **

Start Menu Item – After Login

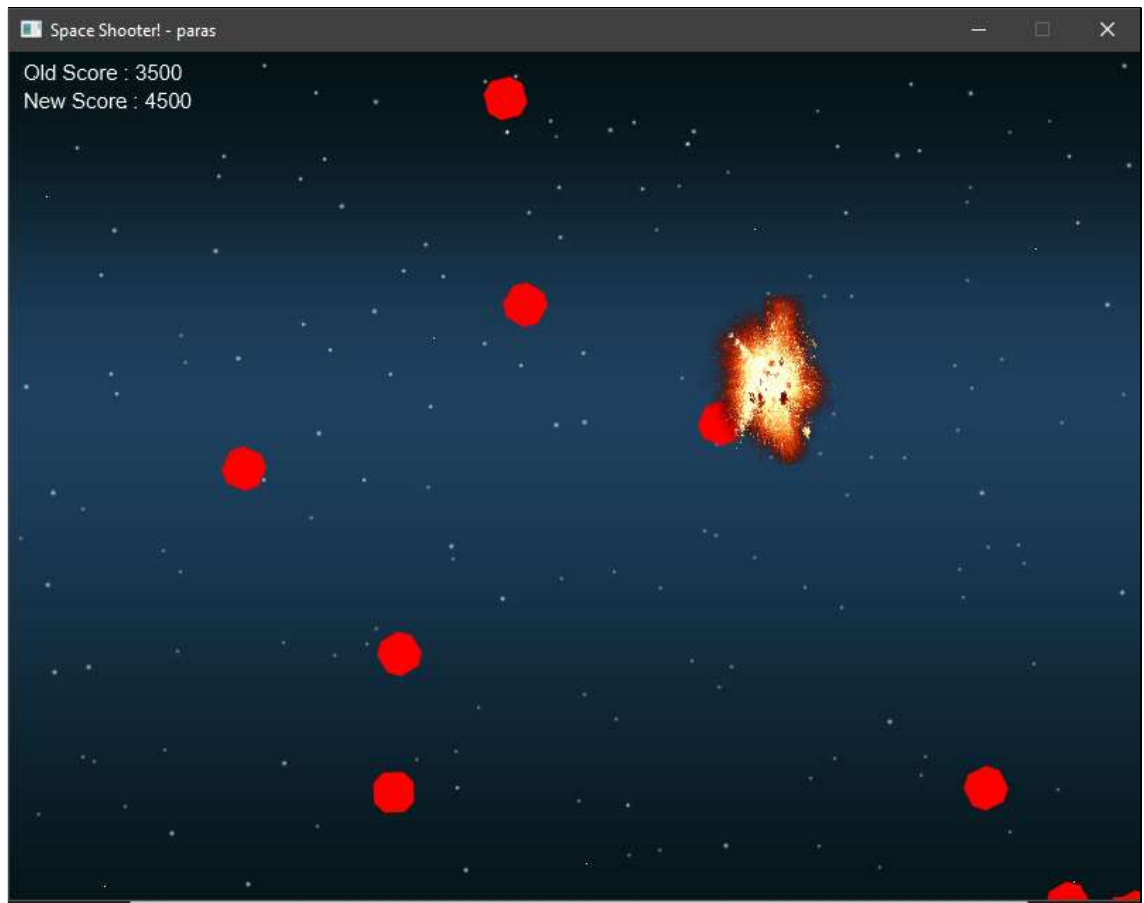


Note: The start options have now changed after login. Refer section 5.b.ii Start Menu screenshot for comparison. Also notice that the title bar now shows the logged-in user.

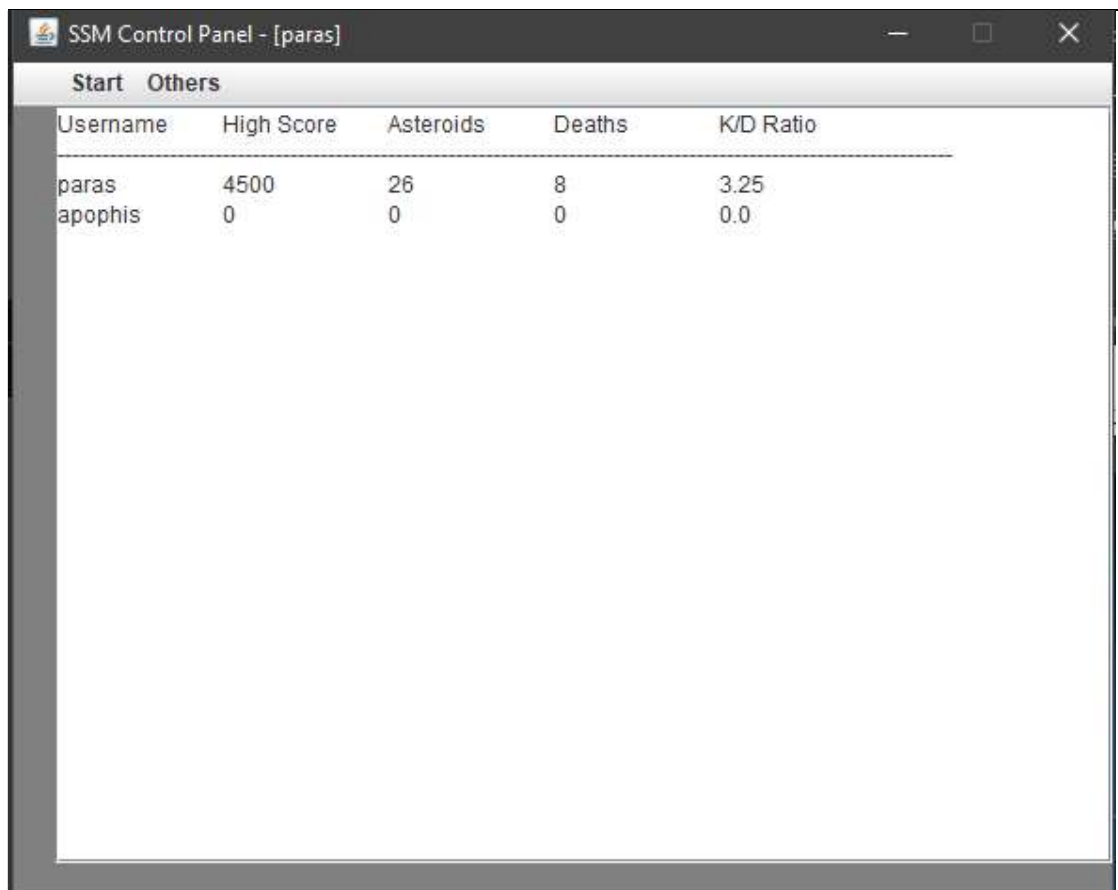
Start Menu Item – Play (in-game)



Start Menu Item – Play (After collision)



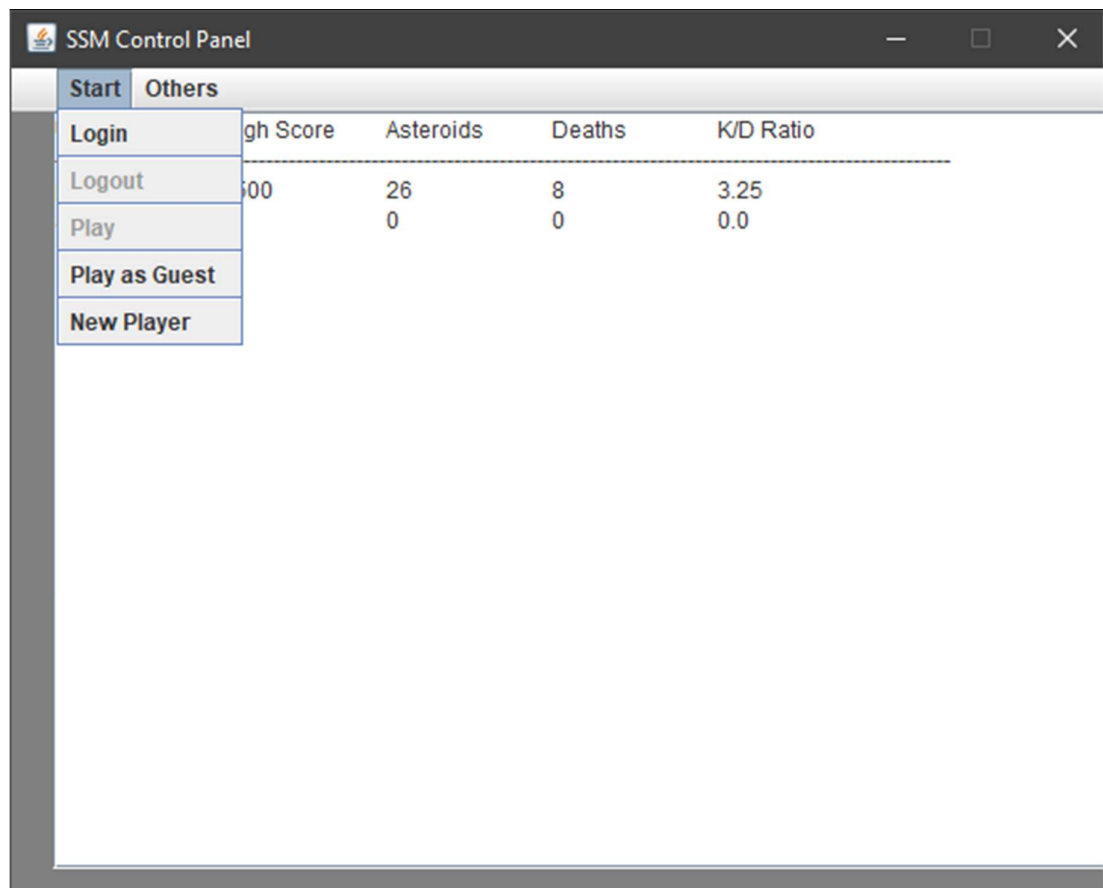
Others Menu Item – Statistics (After collision)



The screenshot shows a window titled "SSM Control Panel - [paras]". It has a tabbed interface with "Start" and "Others" tabs. The "Others" tab is active, displaying a table with the following data:

Username	High Score	Asteroids	Deaths	K/D Ratio
paras	4500	26	8	3.25
apophis	0	0	0	0.0

Start Menu Item - Logout



Note: After logout the Start Menu items have been set to their default i.e., the way they were before login. Also notice that the username has been removed from title.

6. Conclusion

a. Project conclusion & Future enhancement

The end product of this project will be a miniature version of the classic “*Space Shooter*” game developed using java. Enhancements can be made by allowing user to customize their space ship, bullet and changing the background. In future, this project can be modified to be hosted over a server to provide web-based single/multi player gameplay.

7. References

- <http://tutorials.jenkov.com/javafx/stage.html>
- <https://docs.oracle.com/javase/8/javafx/api/index.html>
- <https://www.geeksforgeeks.org/javafx-polygon-with-examples/>
- <https://www.programcreek.com/java-api-examples/>
- <https://ss64.com/nt/errorlevel.html>