Dec 26, 2019

DESIGN DOCUMENT

Project Title: 2048 Game

Members	Seat Numbers
SYEDA SARA AKIF	CS-17002
MAHRUKH KHAN	CS-17003
AYESHA NADEEM AKHTER	CS-17030

Department of Computer & Information Systems Engineering

NED University of Engineering and Technology

Submitted To: Miss Fakhra Aftab

Contents

CHAPTER # 1: INTRODUCTION

The Software Design Document provides aid in the development of the software by providing details about how it is to be built. Within this document, there are different narratives and graphical representations of the software design for our project (2048 Game) including sequence diagrams, collaboration diagrams, class diagram and other supporting requirement information.

1.1) Purpose:

The purpose of this document is to signify the system design that is essential for the software production with an understanding of the fundamental pathway which is to be followed while the software is developed. Moreover, it also provides the details of the software and the system to be produced.

1.2) Scope:

This software application (2048 game) is an interactive computer game with a textual interface which entertains the user by helping the user to pass a little time without getting bored. 2048 is a single-player game where the player makes actions using the four arrow keys of a keyboard on a 4 x 4 grid. Initially, the grid is empty except two to three tiles placed on the board randomly. As the player makes a move, the program spawns a tile on the grid randomly while all tiles slide towards the edge of the chosen side if able. If two tiles with the same numerical value collide, they merge into one tile having twice the merged value. The player wins if a tile with value 2048 is spawned and loses if no more moves are available i.e. the grid is filled completely.

(a) State-based Definition:

Player input: actions (up, down, left and right) based on the current state Computer input: adds tiles on the grid randomly based on the current state Output: result state, score (instant) while playing the game and a final list of player scores at the end of the game. Terminal state (success): One '2048' tile Terminal state (failure): All adjacent tiles occupied and have different values, i.e. no more moves available.

(b) Evaluation Metrics:

The main goal of this project is to get a '2048' tile. However, that is not the only evaluation metric in our project. Besides achieving a 2048 tile in the grid, there is also a bar displaying player score that keeps adding up whenever the

player matches two tiles into one. The winning of the game is a 2048 tile but the score value serves as the icing on the cake.

1.3) Acronyms and Abbreviations:

Acronyms and Abbreviations	Description
DFD	Data Flow Diagram
DD	Design Document

1.4) References and Acknowledgement:

- Adams, Ernest; Rollings, Andrew (2003). Andrew Rollings and Ernest Adams on game design. New Riders Publishing. ISBN 1-59273-001-9.
- Bates, Bob (2004). Game Design (2nd ed.). Thomson Course Technology. ISBN 1-59200-493-8.
- Bethke, Erik (2003). Game development and production. Texas: Wordware Publishing, Inc. ISBN 1-55622-951-8.
- Brathwaite, Brenda; Schreiber, Ian (2009). Challenges for Game Designers. Charles River Media. ISBN 1-58450-580-X.
- Moore, Michael E.; Novak, Jeannie (2010). Game Industry Career Guide. Delmar: Cengage Learning. ISBN 978-1-4283-7647-2.
- Oxland, Kevin (2004). Gameplay and design. Addison Wesley. ISBN 0-321-20467-0.
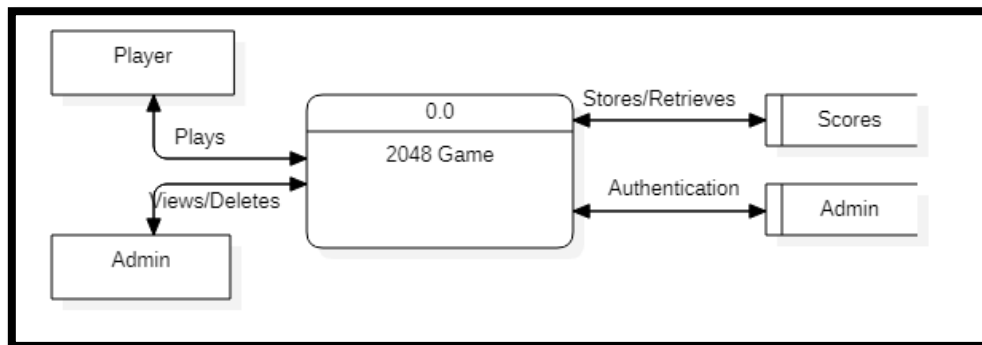
1.5) Remainder of the Document:

Furthermore, this document reveals the graphical representation of the design of this software application including;

- ✓ **DFDs** which represent the flow of data of the processes involved in this system while providing the inputs/outputs of each entity as well.
- ✓ **Class Diagram** which describes the structure of this software by showing its classes, their attributes, operations (or methods) and the relationship between different objects.
- ✓ **Data Dictionary** which defines the contents, format and structure of the database used.
- ✓ **State Transition Diagram** which gives an abstract description of the behavior of this software.
- ✓ **Sequence Diagram** which shows the object interactions arranged in time sequence.
- ✓ **Collaboration Diagram** which models the interactions between the objects of this software in terms of sequenced messages.
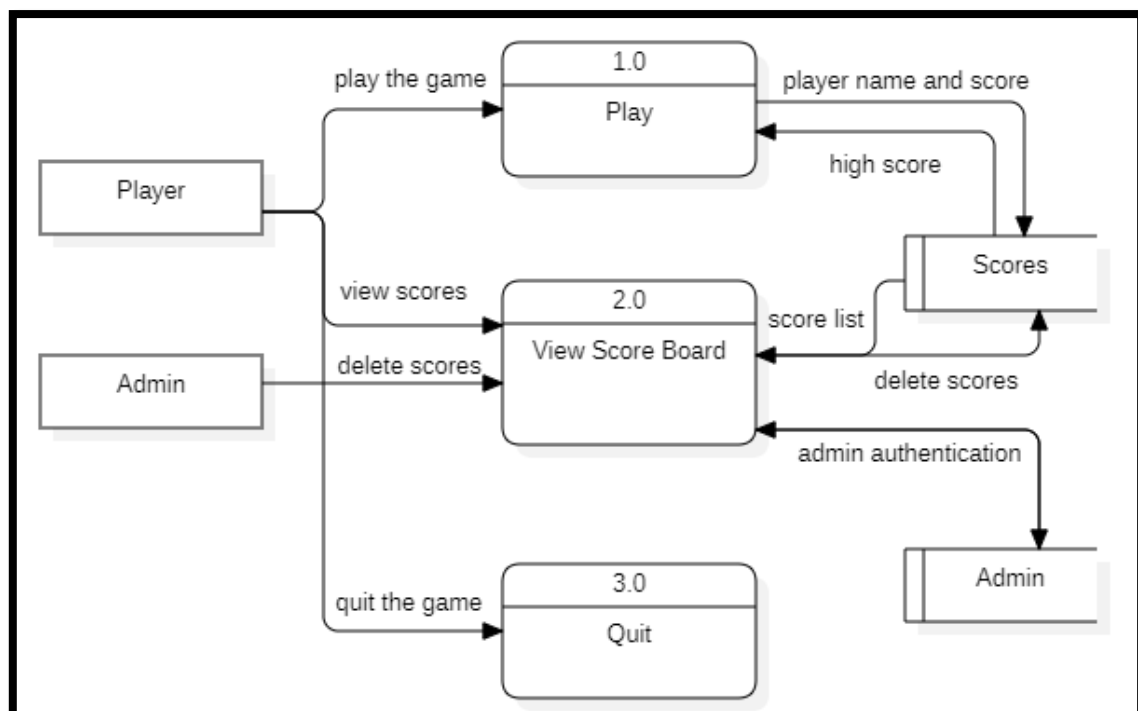
- ✓ **Deployment Diagram** which depicts the execution architecture of the entire system including nodes, such as the hardware and software execution environments, and the middleware connecting them.
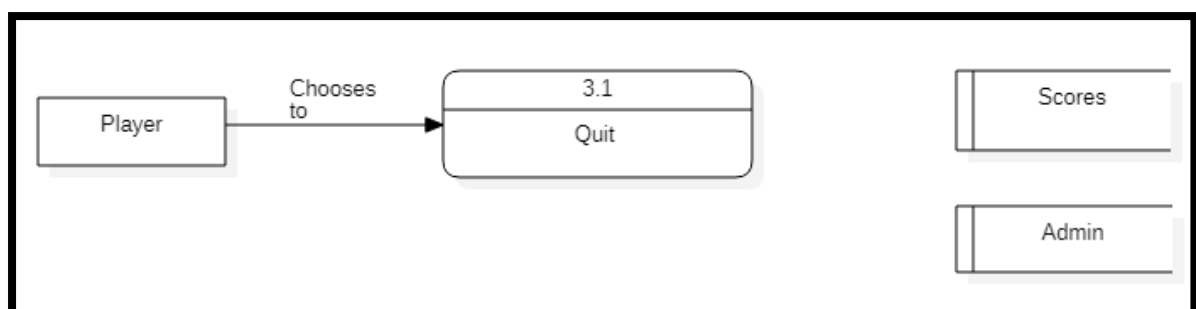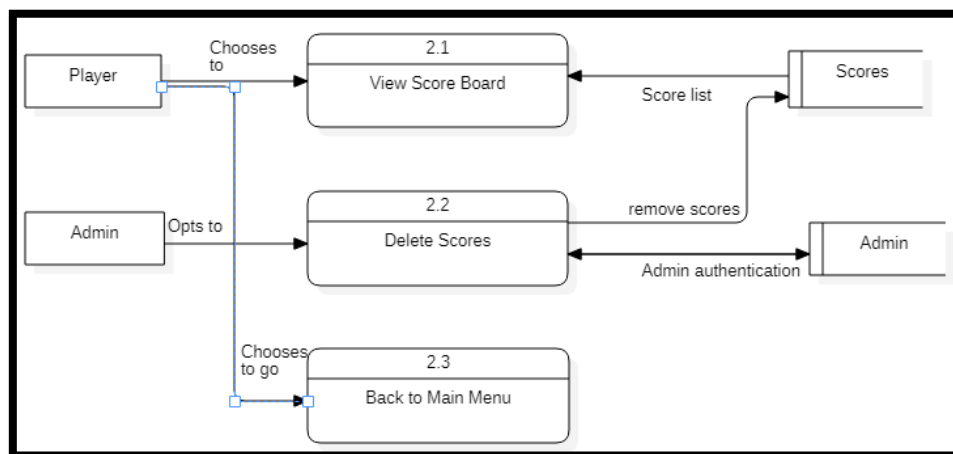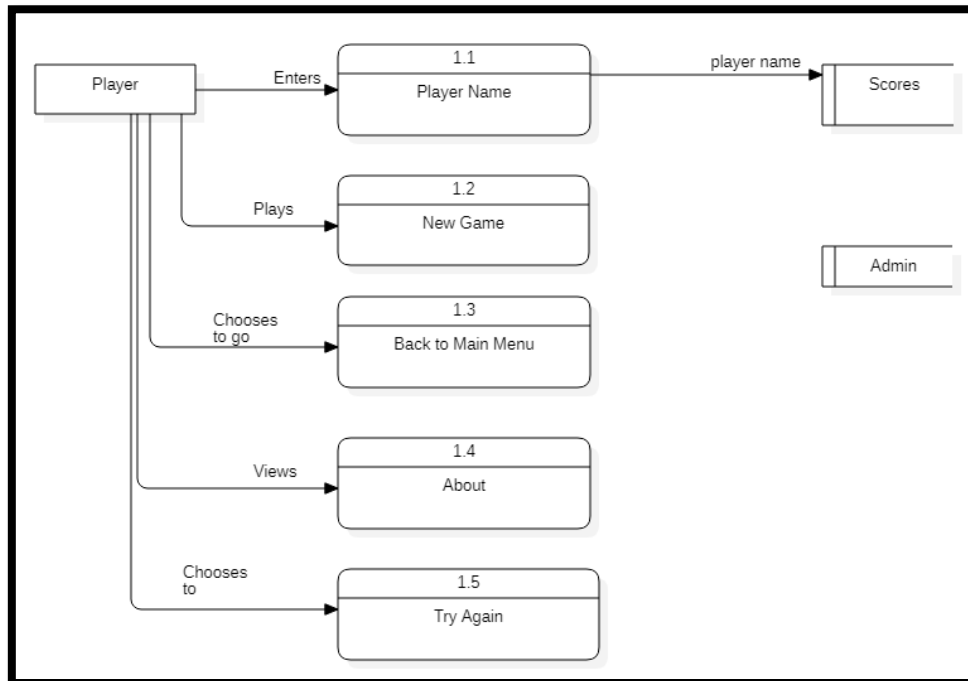
CHAPTER # 2: FUNCTIONAL MODELLING

2.1) LEVEL-0 DFD:

2.2) LEVEL-1 DFD:

2.3) LEVEL-2 DFDs:

| Player | Enters → | **1.1** Player Name | player name → | Scores |

Player
- Enters → **1.1** Player Name → player name → Scores
- Plays → **1.2** New Game
- Chooses to go → **1.3** Back to Main Menu
- Views → **1.4** About
- Chooses to → **1.5** Try Again

Admin

Player — Chooses to → **2.1** View Score Board ← Score list ← Scores

Admin — Opts to → **2.2** Delete Scores — remove scores → Admin

Admin authentication ← Admin

Player — Chooses to go → **2.3** Back to Main Menu

Player — Chooses to → **3.1** Quit

Scores

Admin
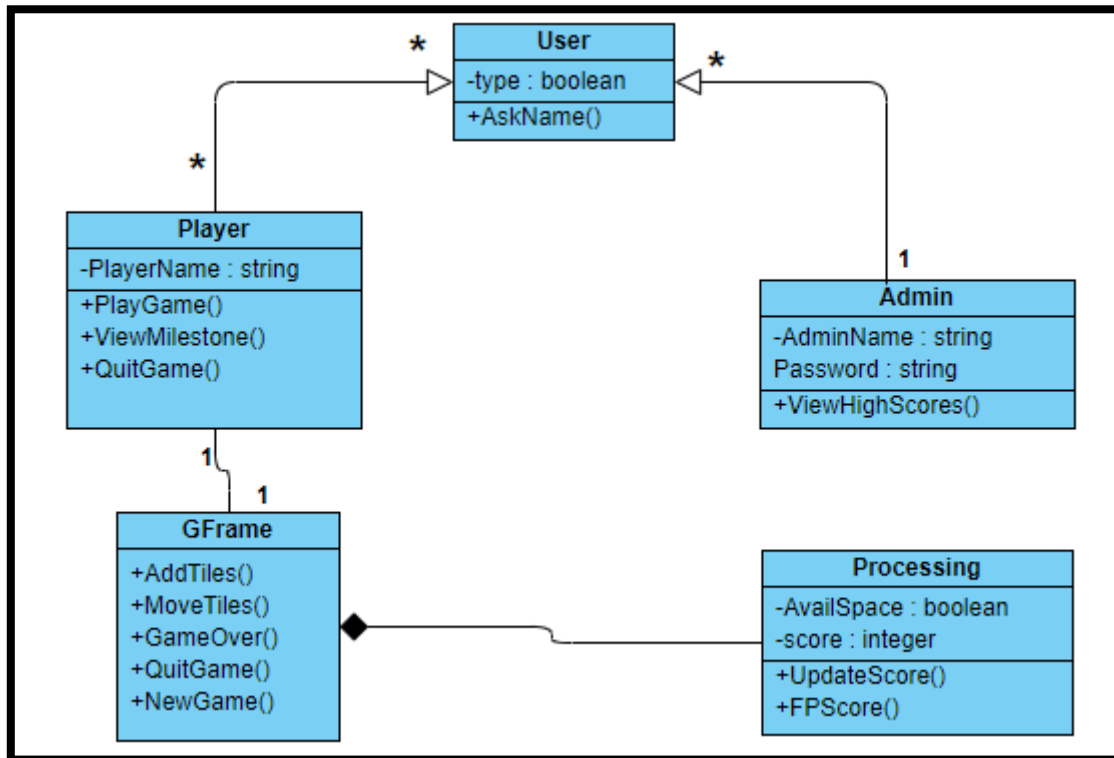
CHAPTER # 3: OBJECT ORIENTED DESIGN

3.1) Class Diagram:

3.2) Data Dictionary:

Field_Name	Field_DataType	Constraint	Range	Size (in Bytes)	Examples
EntryID	Integer	Not Null. Primary Key. Unique.	1-100,000	-	1234
PlayerName	String	Not Null	-	10	Ayesha
Milestone	Integer	Not Null	-	10	2000
HighScore	Integer	Not Null	-	10	3900

CHAPTER # 4: BEHAVIORAL MODELLING

4.1) State Transition Diagram:

4.2) Sequence Diagram:

interaction SequenceDiagram1

| Player | Game Application | Scores | Admin |

1 : execute
2 : Main Menu
3 : Play
4 : Player name entry
5 : Player name
6 : player name
7 : Game screen
8 : New Game
9 : Game screen
10 : About
11 : About message
12 : Try Again
13 : Game screen
14 : Back
15 : Main Menu
16 : View Score Board
17 : fetch scores
18 : score list
19 : Score Board
20 : Delete scores
21 : Admin authentication
22 : Admin name and password
23 : authentication successful
24 : updated Score Board
25 : Back
26 : Main Menu
27 : Quit
28 : terminate game app
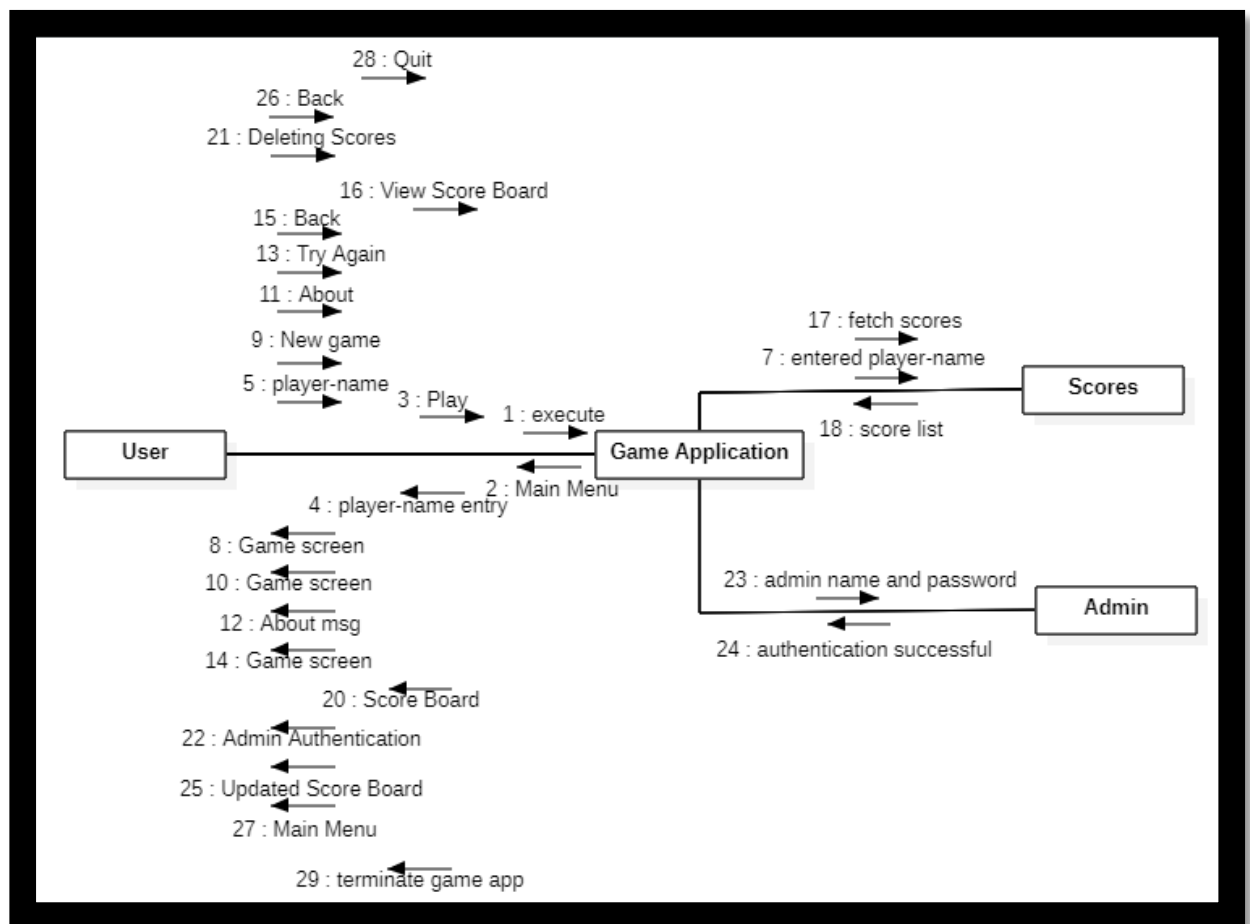
4.3) Collaboration Diagram:

CHAPTER # 5: DEPLOYMENT VIEW

5.1) Components in Deployment View: