

# **DOCUMENTATION**

## **LOGICAL REASONING GAME**

**TRIF ORIANA MARIA**

## **CONTENTS**

1.	Assignment Objective .....	3
2.	Problem Analysis, Modeling, Scenarios, Use Cases .....	3
3.	Design .....	3
4.	Implementation.....	4
5.	Results .....	5
6.	Conclusions .....	8
7.	Bibliography.....	8

## **1. Assignment Objective:**

The primary objective of the project is to develop an educational game that enables players to solve mathematical and logical problems. Players can earn tokens, badges, and climb as they progress through the game.

Users have to solve different logical reasoning and mathematical challenges in each game, if the first game is complete and you have 3 tokens you can level up to game 2, if this game is finished successfully, you earn the final badge.

To create this, there are some sub-objectives to fulfill, among which were:

- a) Input validation: The calculator should validate the user's input to ensure that it is in the correct
- b) Result checking: Check if user introduce the correct answer
- c) Result display: After the input validation and computation the result should be displayed
- d) Graphing: The calculator could also have a graphical user interface that allows the user to go through app and see the number of tokens, badges and rank every time

## **2. Project Analysis, Use Cases**

**Objectives:** The objective of the project is to develop a Java Window App game that allows players to solve logical and mathematical problems in an engaging and interactive way. The game aims to improve the player's cognitive skills and logical reasoning ability.

**Target Audience:** The target audience for the game could be students of all ages who are interested in improving their logical reasoning and mathematical skills. It can also be used by educators to enhance their students' learning experience.

### **a) Scenarios and use cases**

The game will have different levels of difficulty, and each level will have a set of mathematical and logical problems for the player to solve. The game will have a scoring system that will allow the player to earn tokens and badges for their achievements.

## **3. Design**

The game's user interface will have an appealing design and be user-friendly. The user interface will include different features such as badges, tokens, and rank and when you reach a higher rank you receive a message/

## Packages

A package in Java is used to group related classes. Think of it as a folder in a file directory. We use packages to avoid name conflicts, and to write a better maintainable code.

The model designs based on the MVC architecture follow MVC design pattern. The application logic is separated from the user interface while designing the software using model designs.

**The architecture of the MVC pattern comprises three layers:**

**Model:** This layer represents the business logic of the application and includes objects that carry data as well as any necessary logic for updating the controller if data is modified.

Model contains only one class, which is Game, class which model de problem and is the main important class

**View:** The presentation layer of the application, responsible for visualizing the data contained in the model.

View package contains a class with the elements for the user interface like buttons text fields, labels.

**Controller:** This layer interacts with both the model and the view, managing the flow of the application by facilitating data flow within the model object and updating the view when necessary.

Controller is the link between view and model. This package contains 3 classes:

- ControllerGame: all inner classes with action listeners and link with buttons
- 

When implementing the MVC pattern in Java programming, the Model layer consists of simple Java classes, the View layer is responsible for displaying the data, and the Controller layer contains servlets.

## 4. Implementation

The whole project is based on the MVC ( Model, View and Controller) and consist of four parts:

- a) Model
- b) View
- c) Controller
- d) Main Class

- a) Model:

### **Game**

Consist of one class called Game, which is the base class of the project:

Variables: name of the game and integers the number of tokens, badges and rank

Constructor : Default constructor initializes the elements of the game and numbers with 0 and rank with 1.

#### Methods:

- *public void addTokensBadgesRank(int T, int B, int R)* – in this method we have as parameters T, B and R which are the number of Tokens, Badges and Rank which will be added to the current one when it is needed

#### b) View:

In the Model-View-Controller (MVC) architecture, the view class is responsible for presenting data from the model to the user and for accepting user input, which is then passed back to the controller. In a Java application using MVC, the view is typically implemented as a user interface (UI) component such as a JFrame.

### **MainView Class**

#### Variables:

There are 2 buttons for each game such as, 3 textFields for Tokens, Badges and Rank and Labels.

#### Constructor:

*public MainView()*, in this constructor, the buttons, labels, and text fields are initialized, resized, and colored appropriately.

#### Methods:

- *public void addFromMainToG1( ActionListener action)*: there are 2 of this method type which represents the action listeners for the buttons
- *public void showErrorRank( String message)*: display an error
- 
- **ViewG1 and ViewG2 Classes**

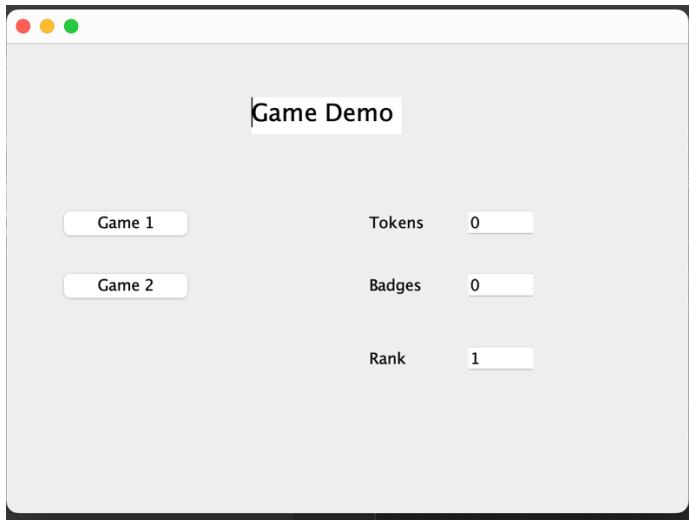
#### Variables:

There are 2 buttons for each game such as, 7 textFields for data input and 2 buttons for going back and result checking

- *public void addG1toMain( ActionListener action)*; action listener for the back button
- *public void addcheckG1( ActionListener action)*; action listener for check button
- *public void refreshEx1()*: after checking the result, all the fields are cleared

## **5. Results:**

Main View with the buttons and rank 1, and 0 tokens and badges

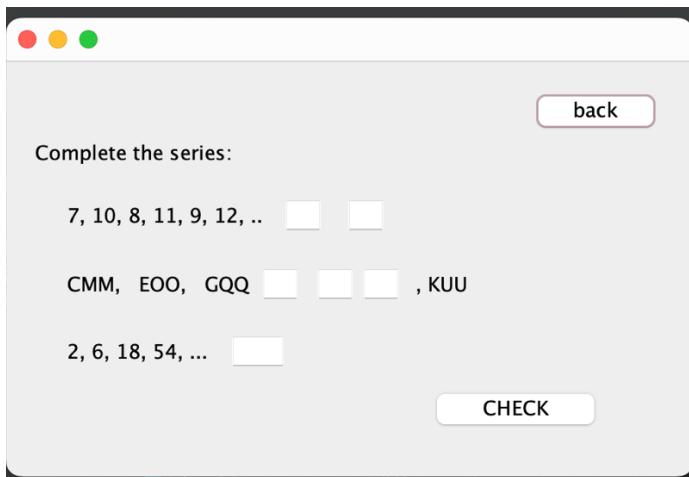


This is game 1,

First array uses the pattern add 3 subtract 2 and so on.

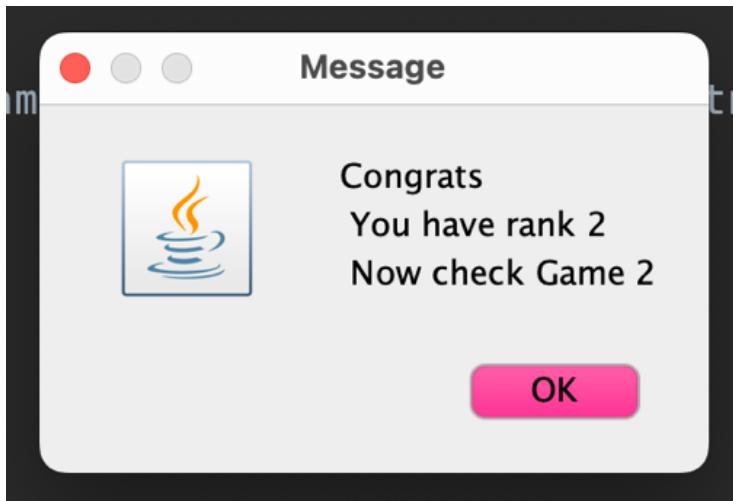
Second array first letters are from 2 to 2 starting from C and the same for the last 2 letters so the answer is “I S S”.

Last array is a geometrical progression, each termen is obtain by multiplying previous onne with 3.



You receive as many tokens as the correct answer you give. In this case all the result were correct.

If the first step is successfully completed, you receive this message and now you are allowed to play the second game.



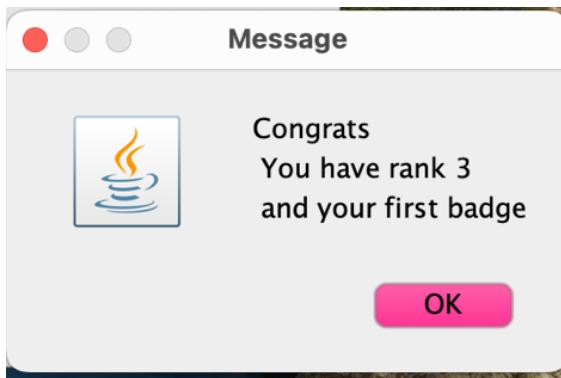
This is the Second Game,  
The first one is an alternating 2 and 3 subtraction,  
The second one is group of consecutive 3 letters but written in another order.  
The last one is an alternating division and addition series: First, divide by 2, and then add 8.  
If the input is correct, you receive the previous message too.

A screenshot of a game interface titled 'Complete the series:'. It shows three puzzles:

- A number sequence: 42, 40, 38, 35, 33, 31, 28, ...  
Input fields: 26, 24
- A letter sequence: QPO, NML, KJI  
Input fields: H, G, F, EDC
- A number sequence: 664, 332, 340, 170, ...  
Input field: 78  
Input field: 89

Buttons include 'back', 'CHECK', and a scroll bar.

This is the last Message received, with the leveling up the receiving the final badge.



## 6. Conclusions and Further Improvement

To further enhance the user experience, the project could be expanded to include additional games with diverse problem sets. The communication system could also be improved by introducing interactive and visually pleasing message formats. The ranking system could be refined by incorporating additional metrics to determine player proficiency, while the graphical user interface could be further optimized to improve the overall game experience.

## 7. Bibliography

<https://www.indiabix.com/logical-reasoning/number-series/004001>