CPSC 319
Team 2

Dr. Pat Mirenda

Software Design Specification Document

Version: 1.2 Date: (03/17/2006)

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Wei Lin	Initial Draft	Feb. 20th, 2006
1.1	Wei Lin	Revised Draft	Feb 27th, 2006
1.2	Wei Lin	Reference Version	March 17th, 2006

Contents

1 Introduction

1.1 System Overview

This SDS will cover the software Autistic Conversational Skills Software or ACSS. This name is simply a developmental project name and in no ways reflects the final marketable name for the product.

ACSS will be a tool to help conversationally-impaired users learn and understand the proper protocol and timing of interrupting conversation. These users are children between ages 6-19 who are autistic or have Asperger's disorder and have difficulty knowing when to interrupt an ongoing conversation

ACSS will provide multimedia examples of real conversations in order to illustrate the two main types of legitimate conversation interruption:

i) An appropriate pause in the conversation. The length of this pause will depend on the rate at which the conversation is taking place (rapid, relaxed, moderate, etc.)

ii) An emergency situation in which immediate attention of one or more of the people involved in the conversation. Although emergencies can be relative, the software will focus on examples of emergencies that are universally recognized (e.g. fire, injury, etc.)

ACSS will provide an interface in which a user must indicate when it is appropriate to interrupt the onscreen conversation with regard to either of the situations presented above. Likewise, ACSS must relay back to the user whether or not the chosen moment of interruption was correct or incorrect, along with a reason and or suggestion.

ACSS will provide a statistical history of correct and incorrect choices along with sub-categories of each. ACSS will provide a profile based on these statistics for at least one main user.

ACSS will NOT provide any examples in which the exceptional clause "excuse me" is required or permitted to interrupt a conversation. This clause is beyond the scope and purpose of the software.

The application of ACSS is as learning software as well as possibly being used in research.

ACSS is targeted towards autistic children of age 6-19 with conversational impairments, specifically difficulties in detecting appropriate ways/times to interrupt an ongoing conversation.

The objective of the software is to simulate a real life conversation and to allow the user the ability to "interrupt" the ongoing conversation. The software will then determine whether or not the interruption choice was appropriate or not and why.

The ultimate goal of the software is to improve the ability of the user to determine how and when to interrupt a real-life conversation based on his/her performance with the software. A secondary goal is to allow an observer to monitor the progress of the user based on statistical data gathered by the software. Any inferences, conclusions, and relevant studies made based on this data will be handled independently and the software is not required to make any opinion nor is it responsible for any that are made.

1.2 Supporting Materials

The PowerPoint on Autism given by Dr. Pat Mirenda (the client) is available at http://www.umobileco.com/cs319/project/autismintro.06.HO.ppt

The recorded project description conversation given by Dr. Mirenda is available at http://www.umobileco.com/cs319/clientmeeting-jan24.mp3.

The general team website is located at http://www.umobileco.com/cs319/

Final version of Software Requirements Specification Document available at http://www.umobileco.com/cs319/project/SRS.pdf

1.3 Definitions, Acronyms, and Abbreviations

ACSS – Autistic Conversational Skills Software, the software that this SRS describes, simulating a real life conversation and allowing users to choose when to interrupt the given conversation

SRS – Software Requirements Specification, this document which outlines the requirements that the software must fulfill. Entirely design independent.

InterruptED – The temporary internal developmental name for the ACSS, used for the simplicity of labeling windows/title bars, etc. The completed project will be named according to marketing and research needs.

User – any person who uses the program, with the general case being children ages 6-19 with autism or having Asperger's disorder who have conversational difficulties.

Administrator – a person who has administrative privilege/access to the system

GUI – graphical user interface

Main Menu Page – the initial menu to allow users to register/login

Game Menu Page – the initial menu for users who have logged in

Registration Page – the page that allows the user to create new accounts

Login Page – the page that allows the user to login if they have a valid login

UML – Unified Modeling Language. Refer to http://www.uml.org/

OS – operating system

SDS – Software Design Specification

RC – Release Candidate

CVS – Concurrent Versioning System

Client – Dr. Pat Mirenda, with possible inclusion of her department and/or research staff.

2 Design Considerations

2.1 Assumptions

It is assumed that the hardware designated for ACSS needs to have a CD-ROM or a DVD-ROM drive for installing the software.

2.2 Constraints

- Platform:
  - must work on Macintosh and PC
- Operating system:
  - for Macintosh: must have OS X.4
  - for PC: must have at least Windows 98 or above
- Hardware:
  - the computer must have an audio output system and speakers or headphones
  - at least 128 MB of RAM
  - minimum 800x600 screen resolution with 256 colors

2.3 System Environment

The system runs a Java Virtual Machine to provide the ACSS to service on any operating system with Java installed, along with Java Media Framework. A XML database will reside on the system to store user and game information.

There is no specific hardware requirement for the system, other than the assumptions made in section 2.1 and the constraints made in section 2.2.

3 Architecture

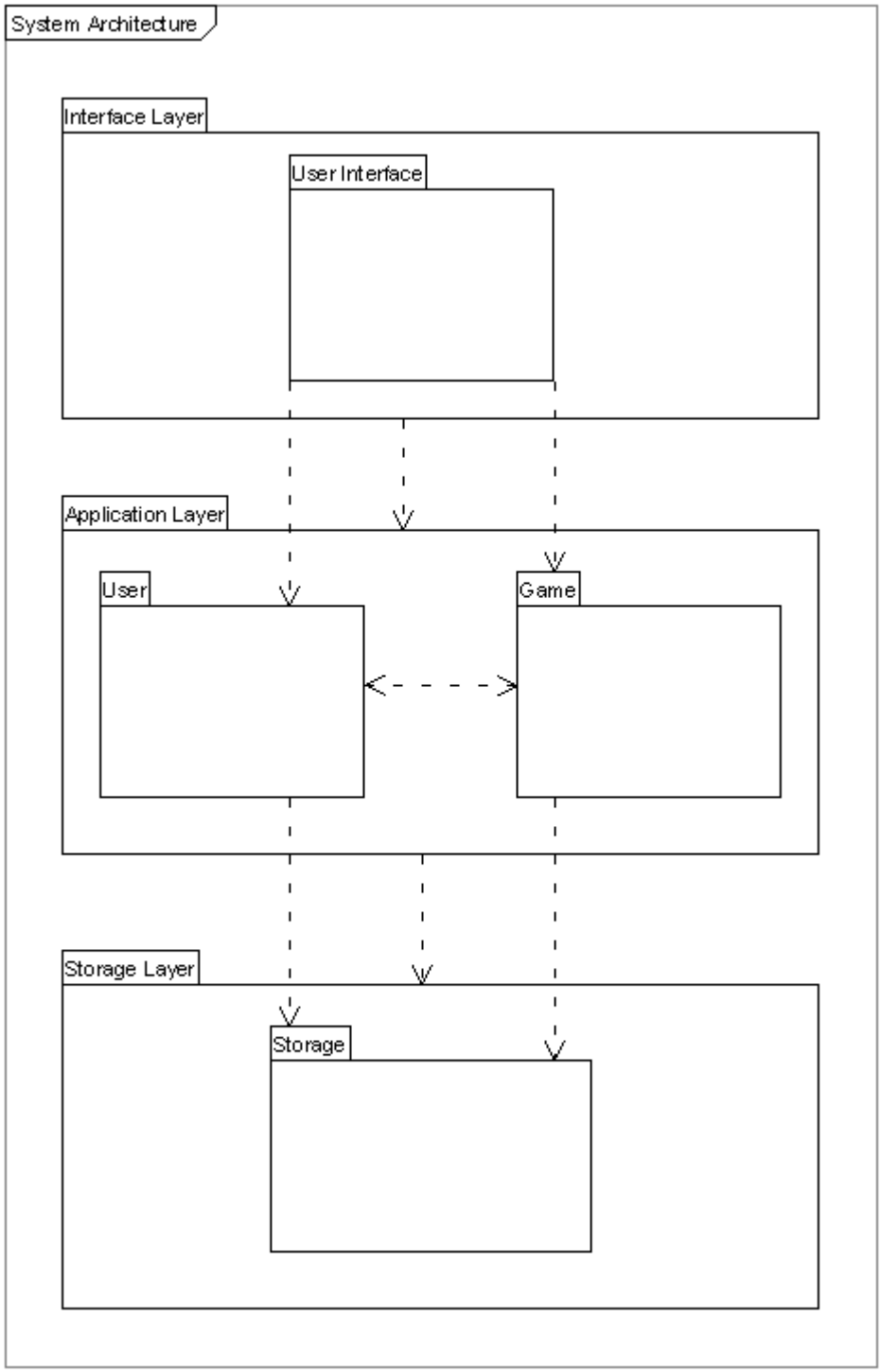


Figure 3.1.1 An Architectural View of the ACSS System

3.1 Overview

The system will follow the three-tier architectural style and be organized into three layers: the interface layer, application layer and the storage layer. The interface layer will be the graphical user interface that allows the users to interact with the system. It will be implemented using the Java Media Framework and the Java Swing Package, and will contain the video player and the all menus. The application layer will contain the logic and rules for storing data in the database layer and also retrieving it in accordance with the user's needs. This is the layer that will contain the data file parsers and will allow controlled access to the data files. Finally, the storage layer will store the metadata required for the system.

3.2 Rationale

The three-tier architecture style shall be used because it not only separates the user interface and the metadata, but also provides an application logic layer. The application layer provides a middle layer that allows the data files and the GUI components to be loosely coupled. The application layer has to be modified if there are any changes to the format of the data files and the interface layer will need little or no modification. This will make it easy for clients of this software to modify the data file format and attributes for further research purposes if they wish to do so. This layer makes the system more maintainable and reusable and also hides the complexity of processing data from the users.

4 High Level Design

4.1 Conceptual View

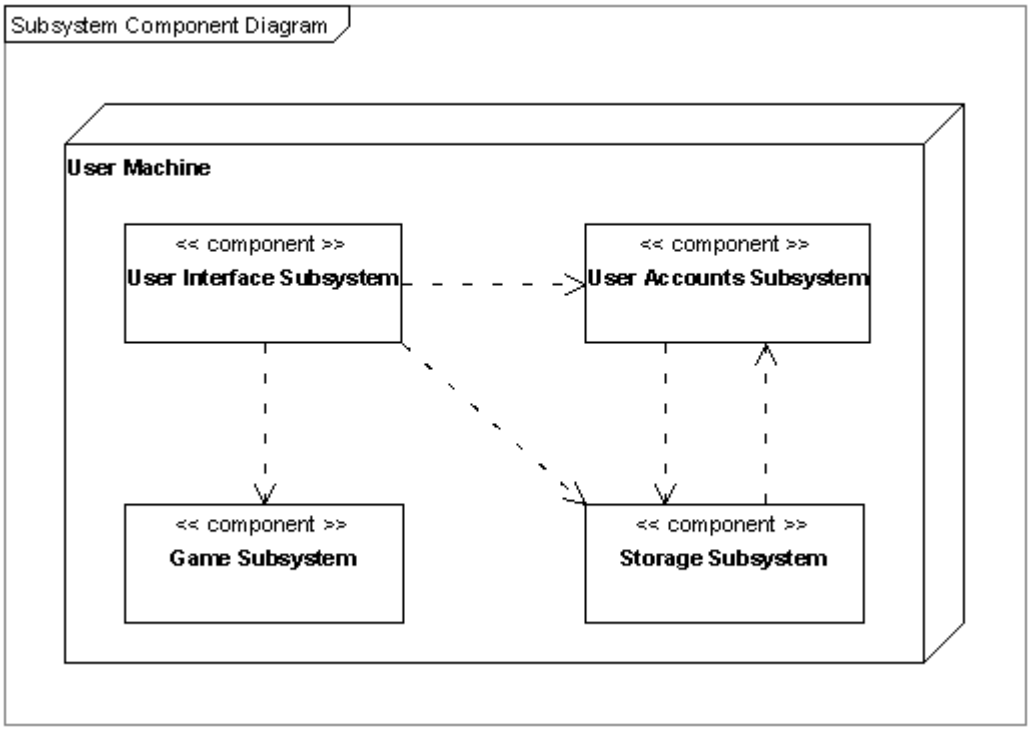


Fig 4.1.1 The Conceptual Diagram for the ACSS System

4.2 Physical View

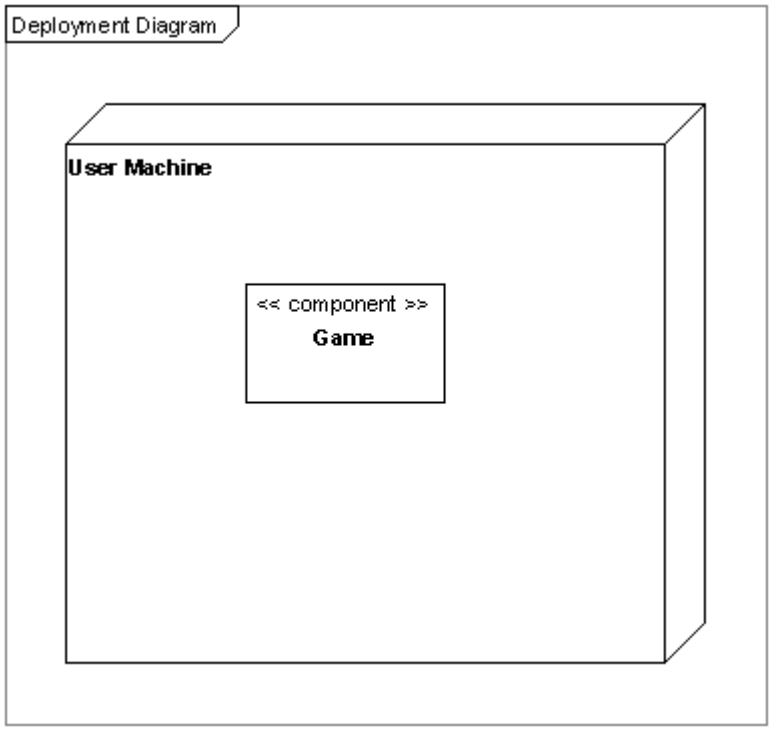


Fig 4.2.1 The Deployment Diagram for the ACSS System

5 Low Level Design

5.1 Module

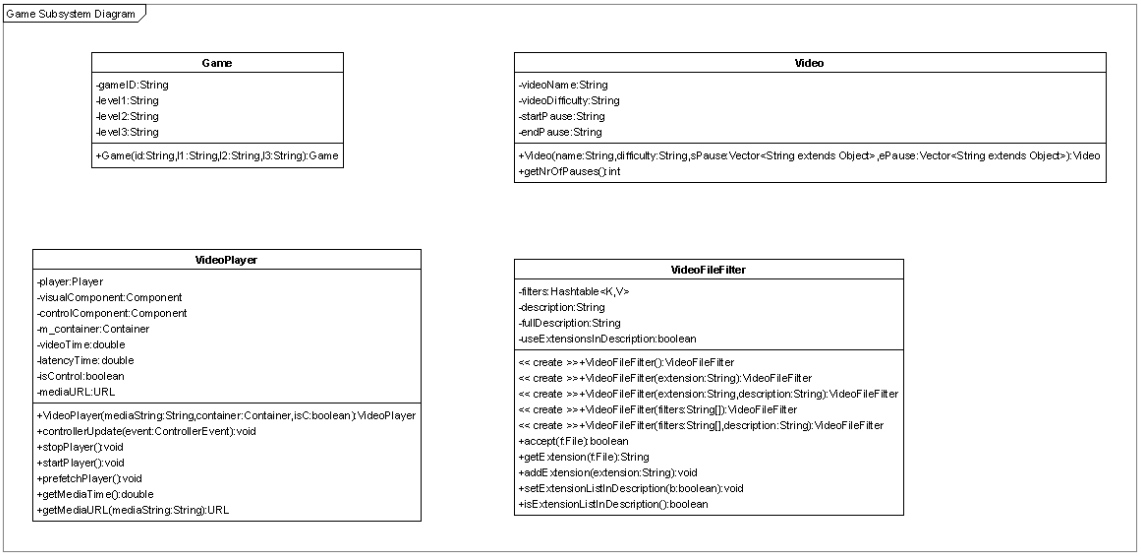
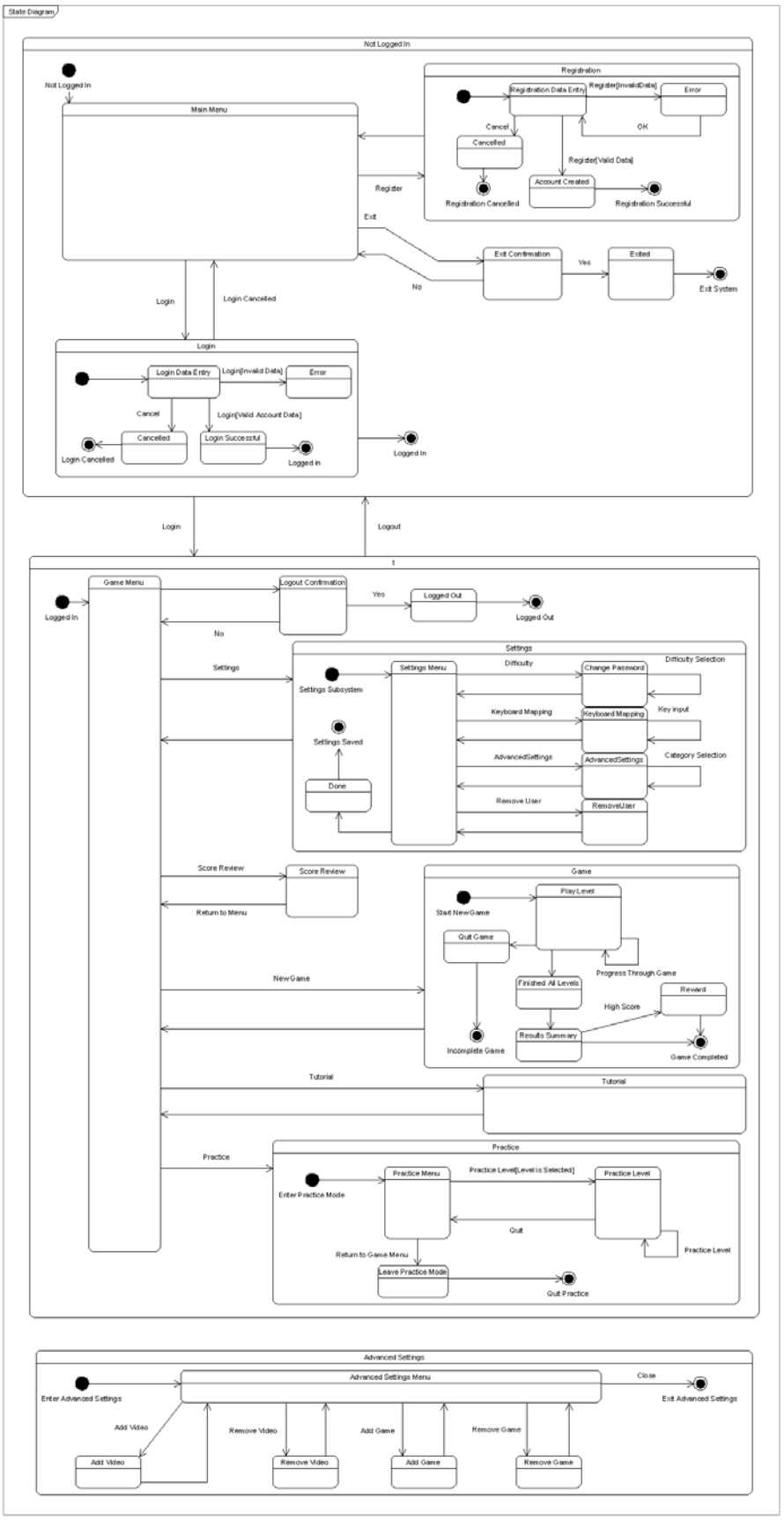


Fig 5.1.1 The Game Subsystem Diagram

Fig 5.1.2 (Next Page) The State Diagram


State Diagram
Not Logged In
Not Logged In
Registration
Registration Data Entry
Register[InvalidData]
Error
Cancel
OK
Cancelled
Register[Valid Data]
Account Created
Registration Cancelled
Registration Successful
Main Menu
Register
Exit
Exit Confirmation
Yes
Exited
No
Exit System
Login
Login Cancelled
Login
Login Data Entry
Login[Invalid Data]
Error
Cancel
Login[Valid Account Data]
Cancelled
Login Successful
Login Cancelled
Logged In
Logged In
Login
Logout
t
Game Menu
Logout Confirmation
Yes
Logged Out
Logged In
Logged Out
No
Settings
Settings
Settings Subsystem
Settings Menu
Difficulty
Change Password
Difficulty Selection
Keyboard Mapping
Keyboard Mapping
Key input
Settings Saved
AdvancedSettings
AdvancedSettings
Category Selection
Done
Remove User
RemoveUser
Score Review
Score Review
Return to Menu
Game
Start NewGame
Play Level
Quit Game
Progress Through Game
NewGame
Finished All Levels
Reward
High Score
Results Summary
Incomplete Game
Game Completed
Tutorial
Tutorial
Practice
Practice
Enter Practice Mode
Practice Menu
Practice Level[Level is Selected]
Practice Level
Quit
Practice Level
Return to Game Menu
Leave Practice Mode
Quit Practice
Advanced Settings
Advanced Settings Menu
Close
Enter Advanced Settings
Exit Advanced Settings
Add Video
Remove Video
Add Game
Remove Game
Add Video
Remove Video
Add Game
Remove Game

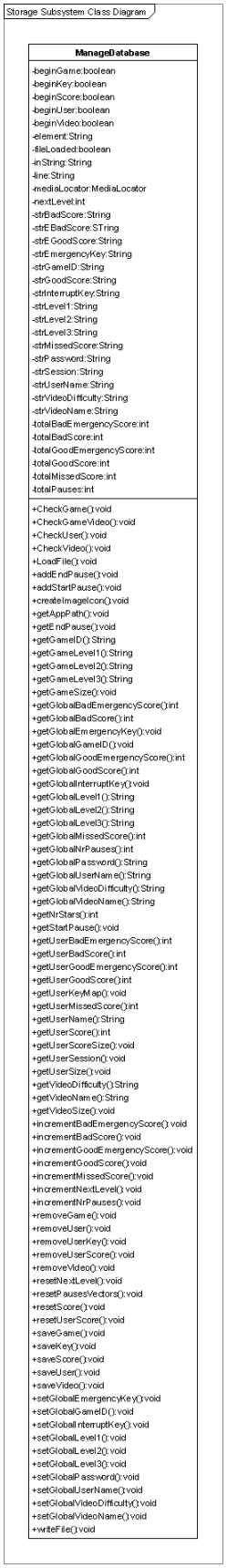


Fig 5.1.3 The Storage Subsystem Class Diagram

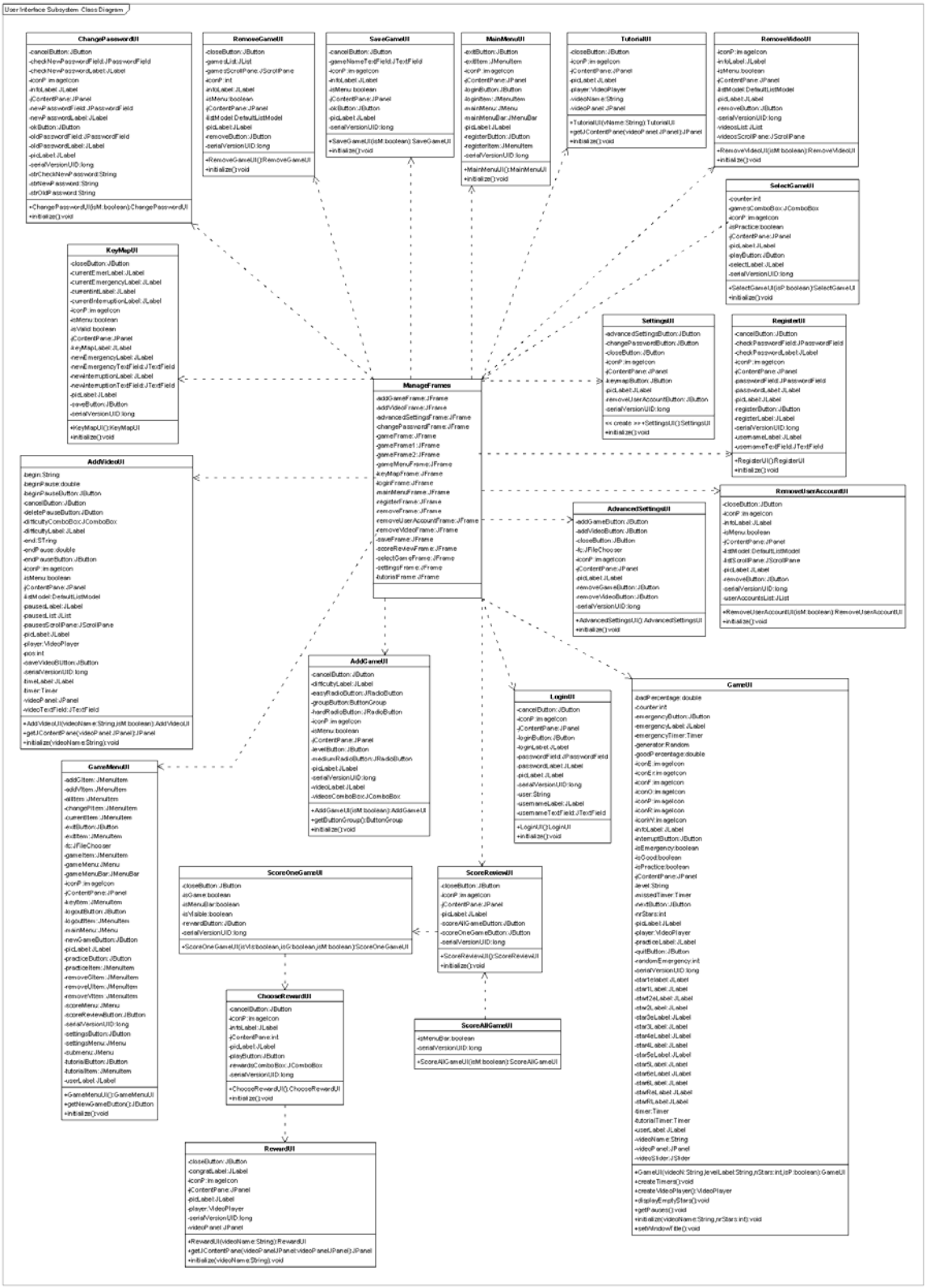


Fig 5.1.4 The User Interface Subsystem Class Diagram

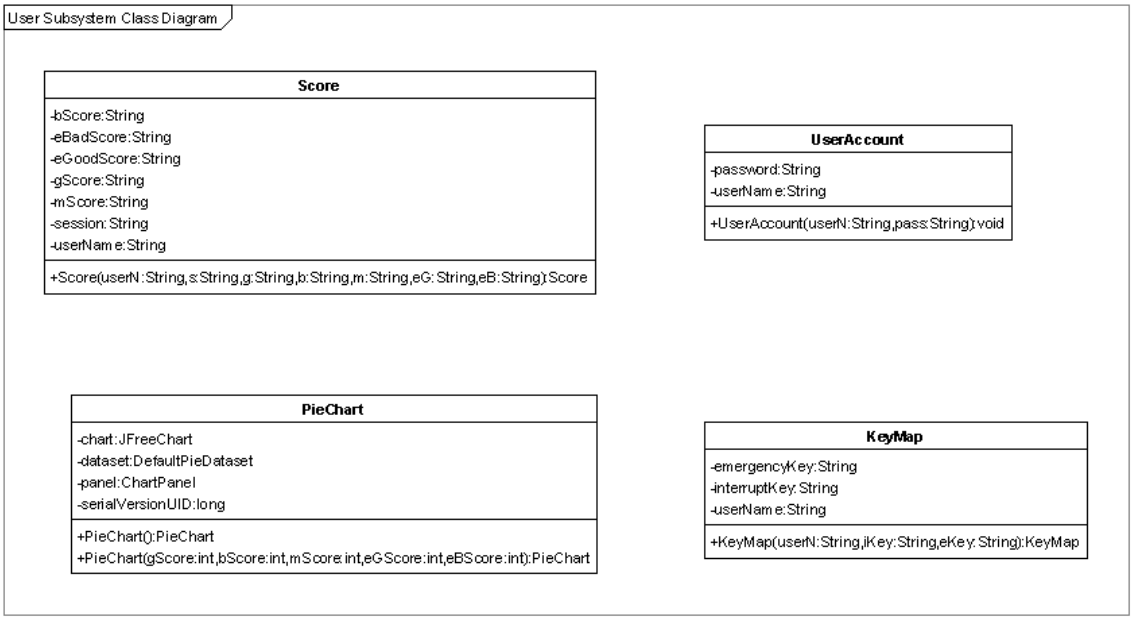
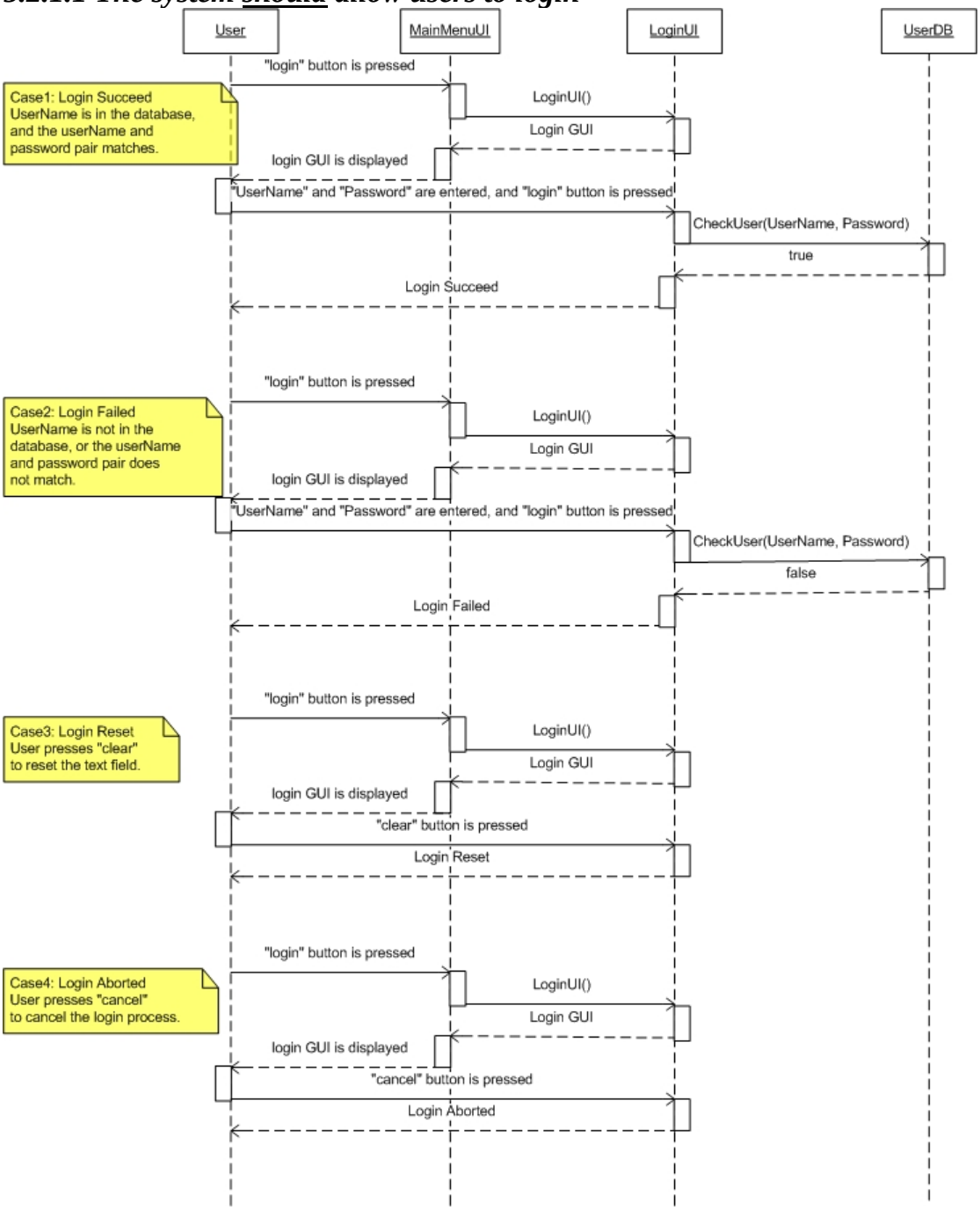


Fig 5.1.5 The User Subsystem Class Diagram

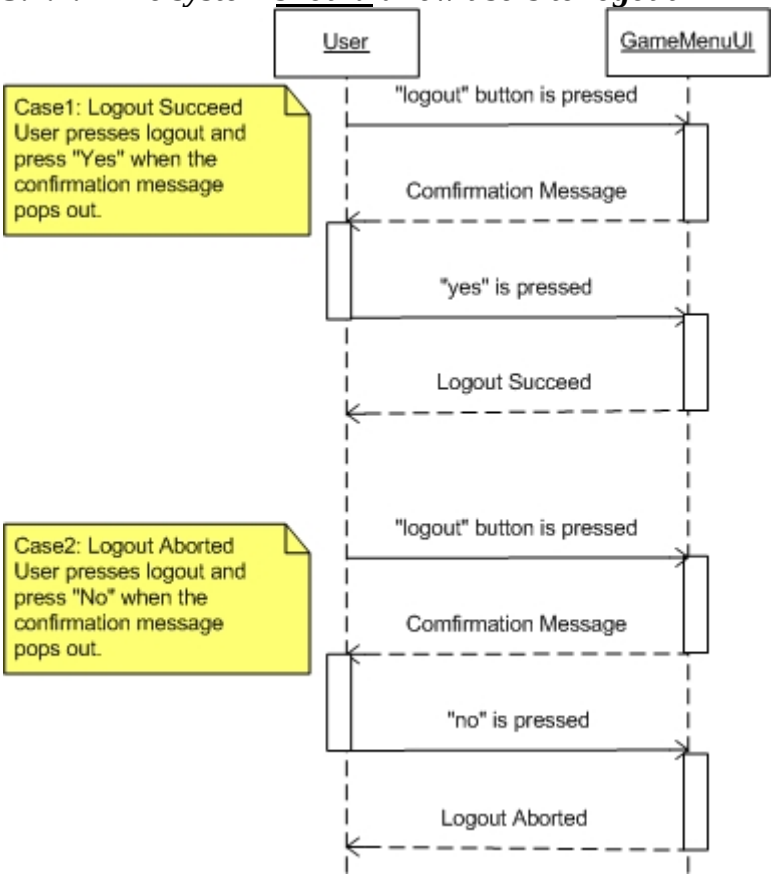
5.2 Sequence Diagrams

5.2.1 Accounts System

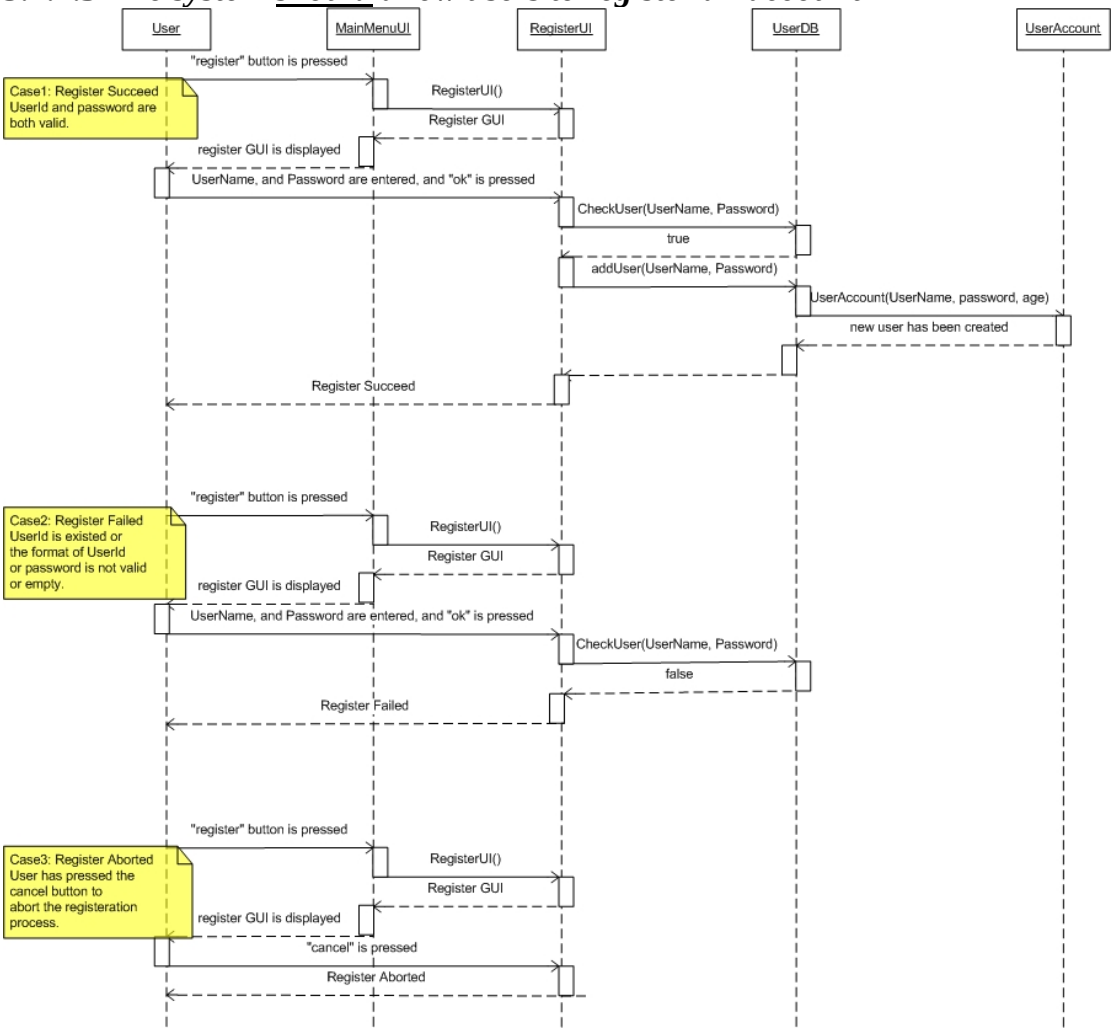
5.2.1.1 The system should allow users to login



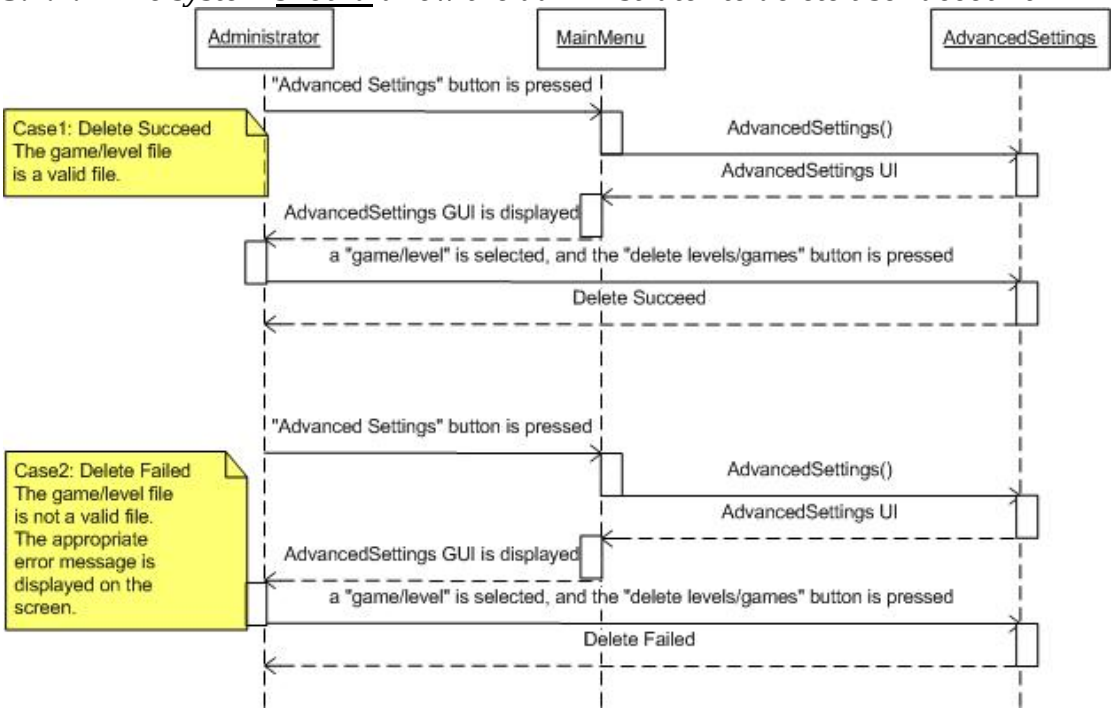
5.2.1.2 The system <u>should</u> allow users to logout



5.2.1.3 The system <u>should</u> allow users to register an account

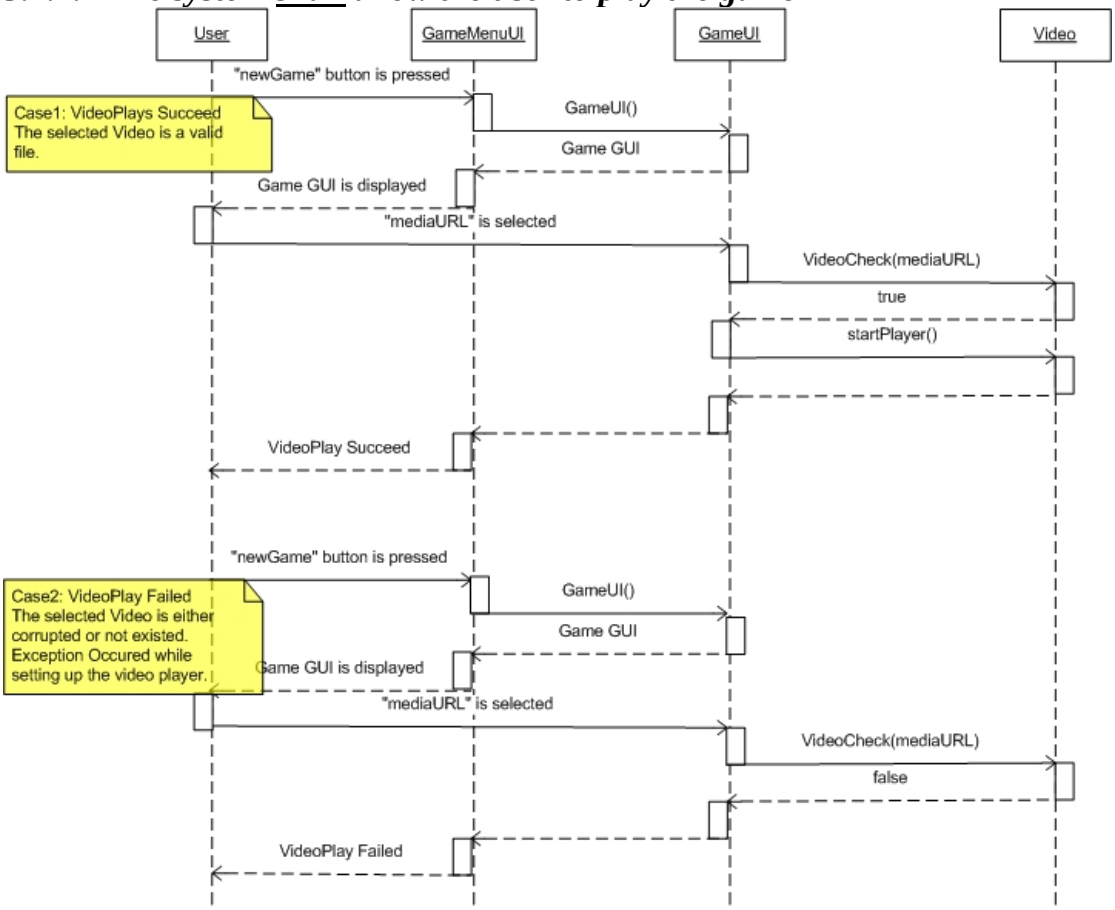


5.2.1.4 The system <u>should</u> allow the administrator to delete user account

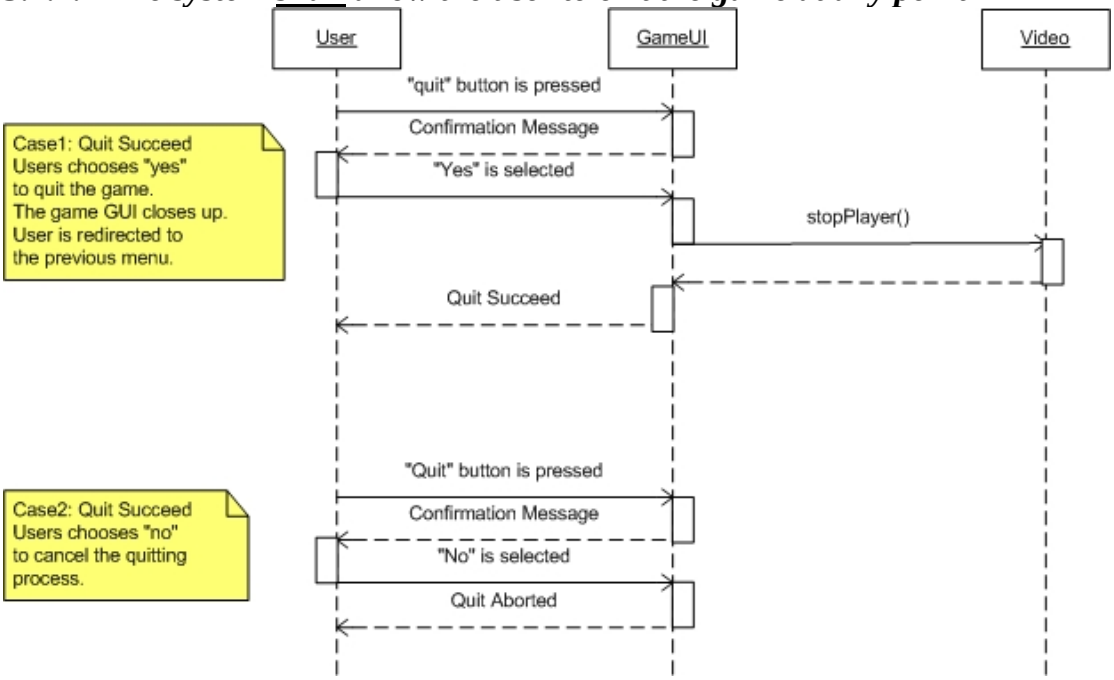


5.2.2 Game System

5.2.2.1 The system shall allow the user to play the game

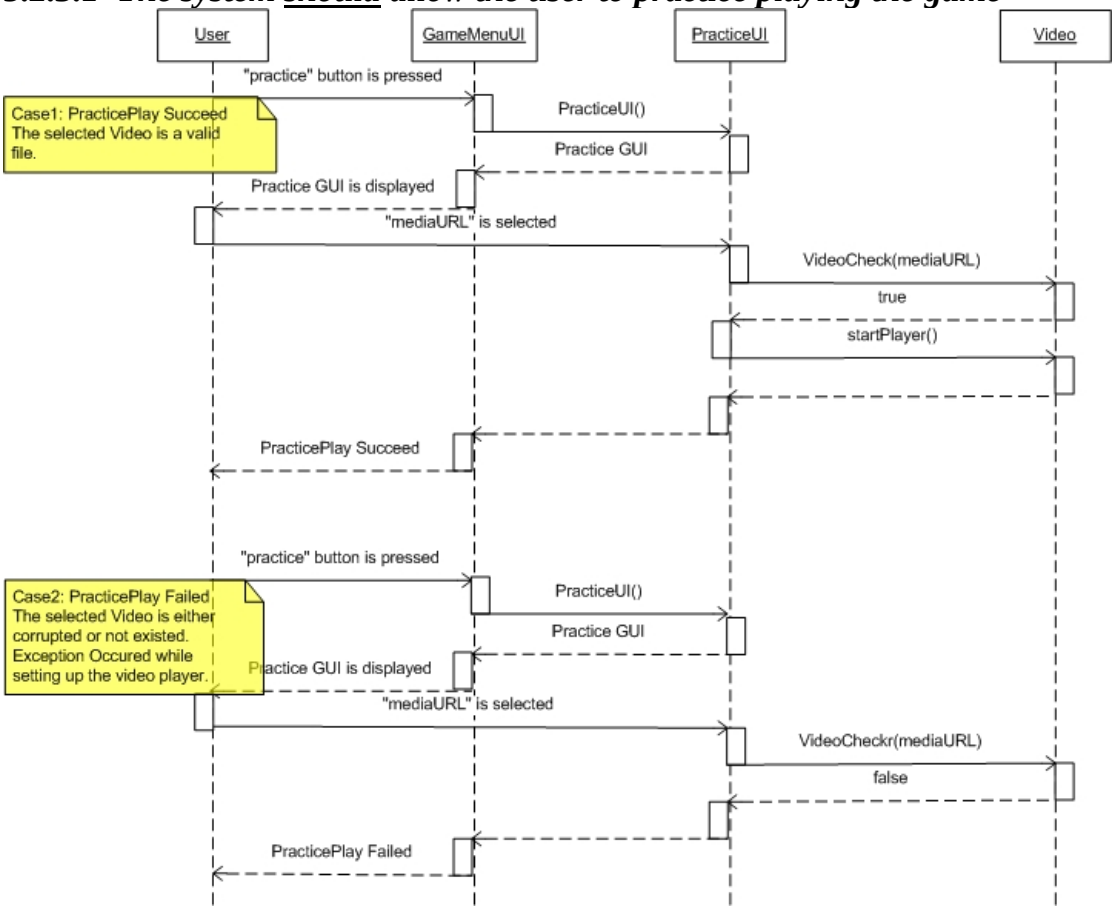


5.2.2.2 The system shall allow the user to exit the game at any point

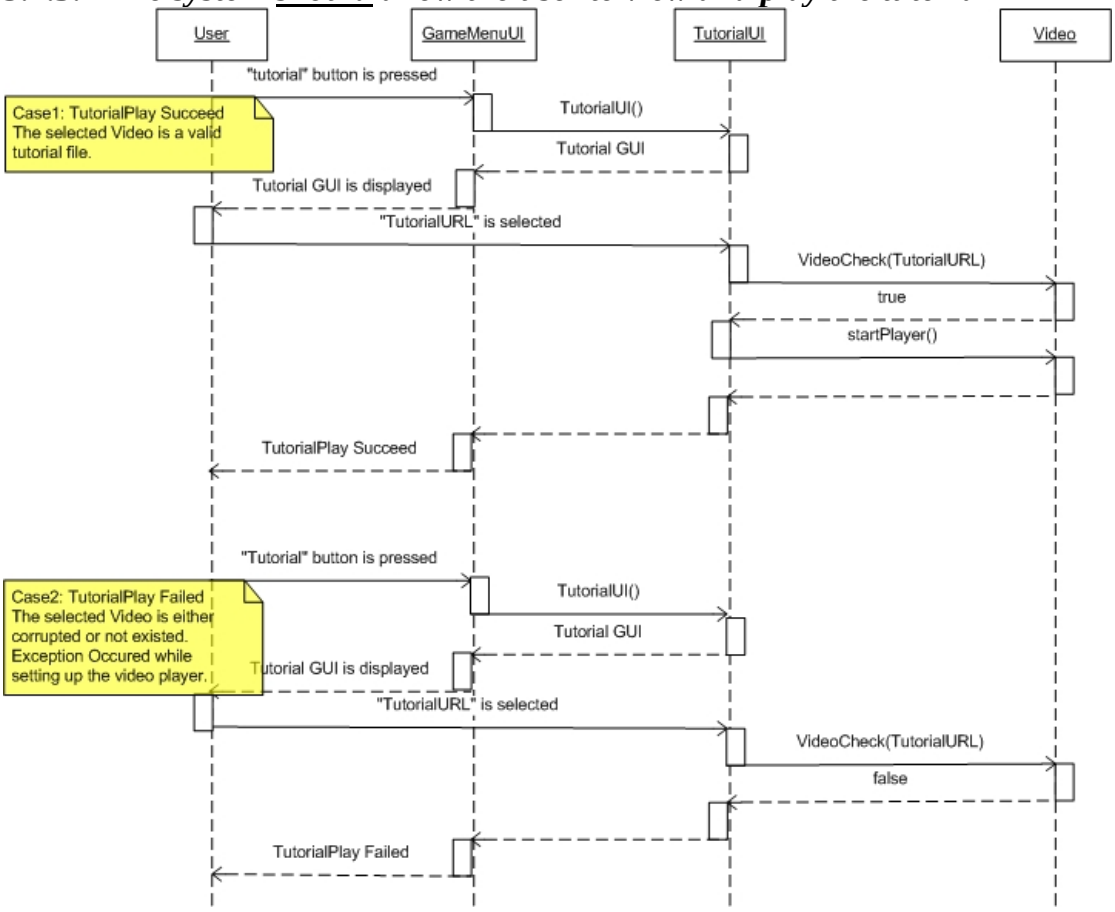


5.2.3 Training System

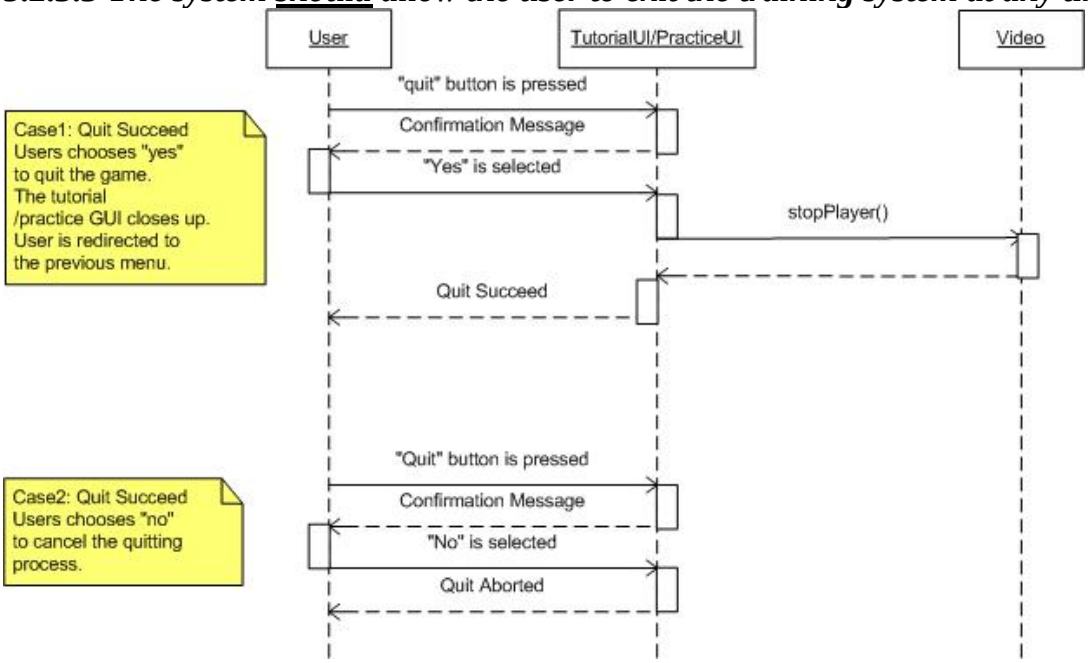
5.2.3.1 The system <u>should</u> allow the user to practice playing the game



5.2.3.2 The system <u>should</u> allow the user to view and play the tutorial

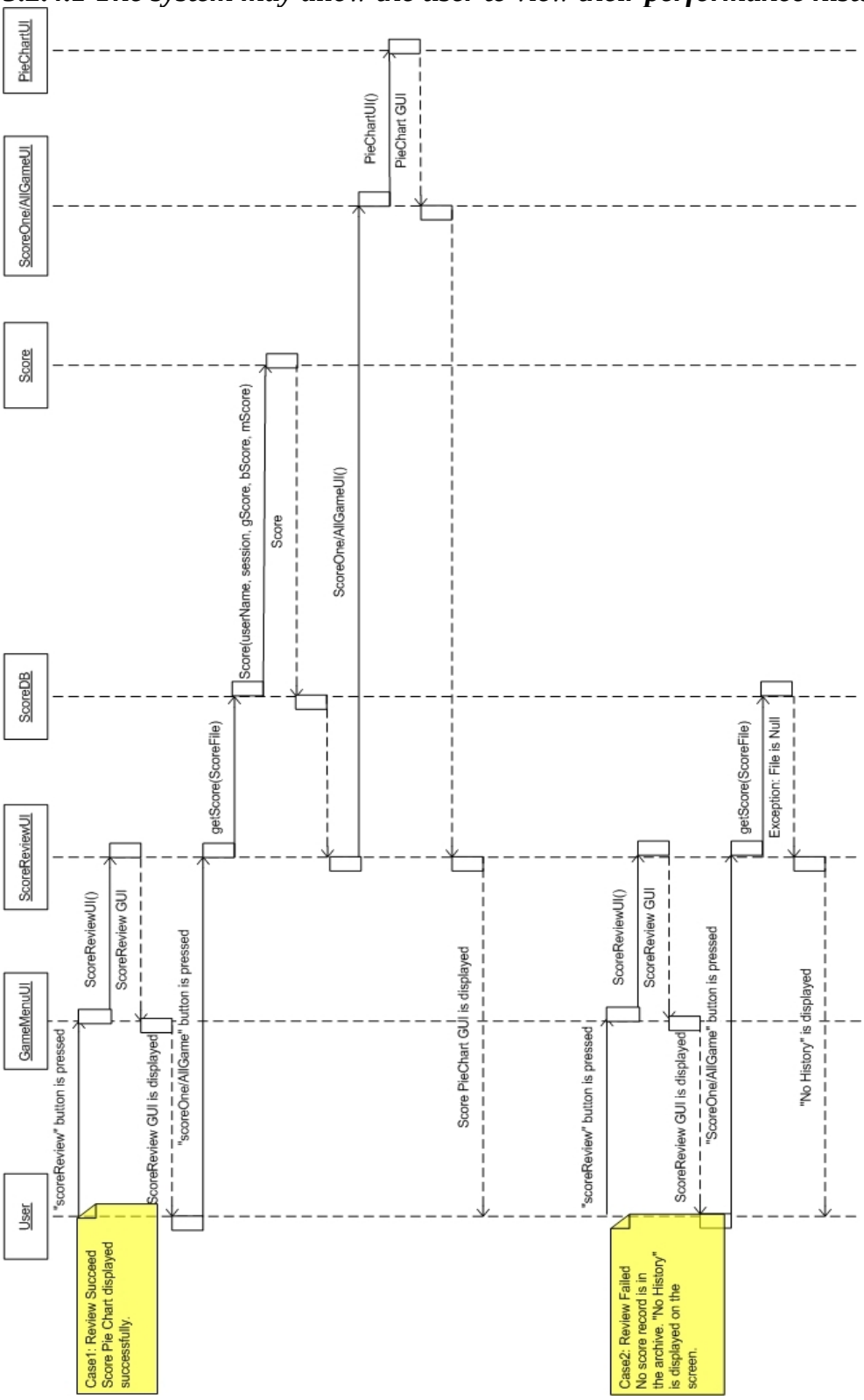


5.2.3.3 The system <u>should</u> allow the user to exit the training system at any time

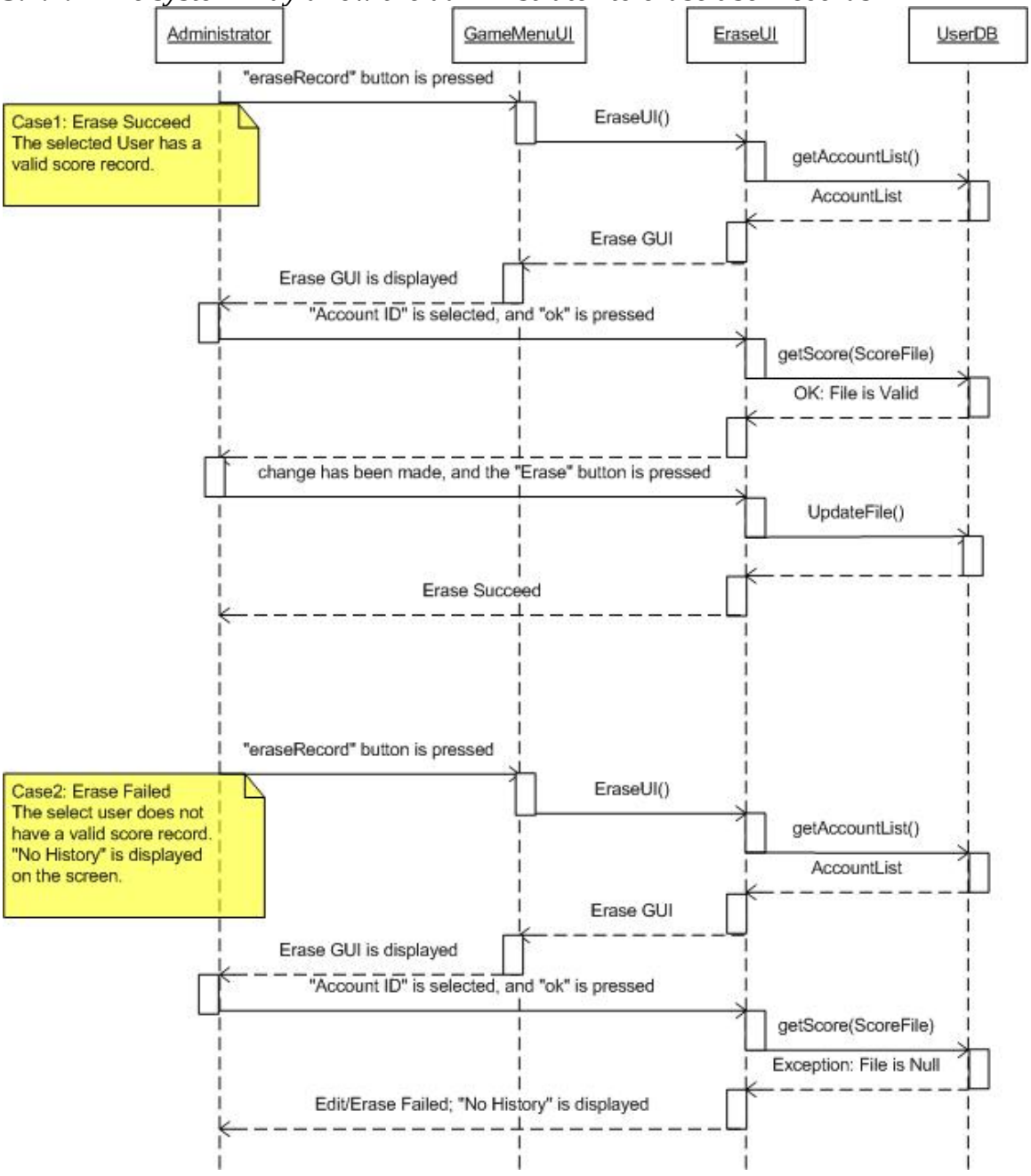


5.2.4 User Data System

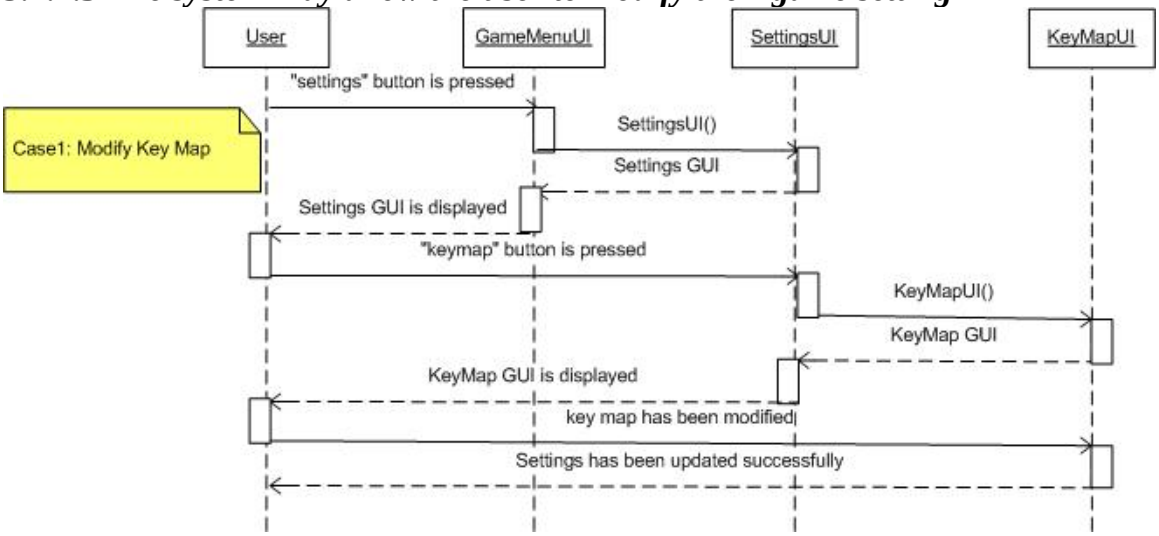
5.2.4.1 The system may allow the user to view their performance history



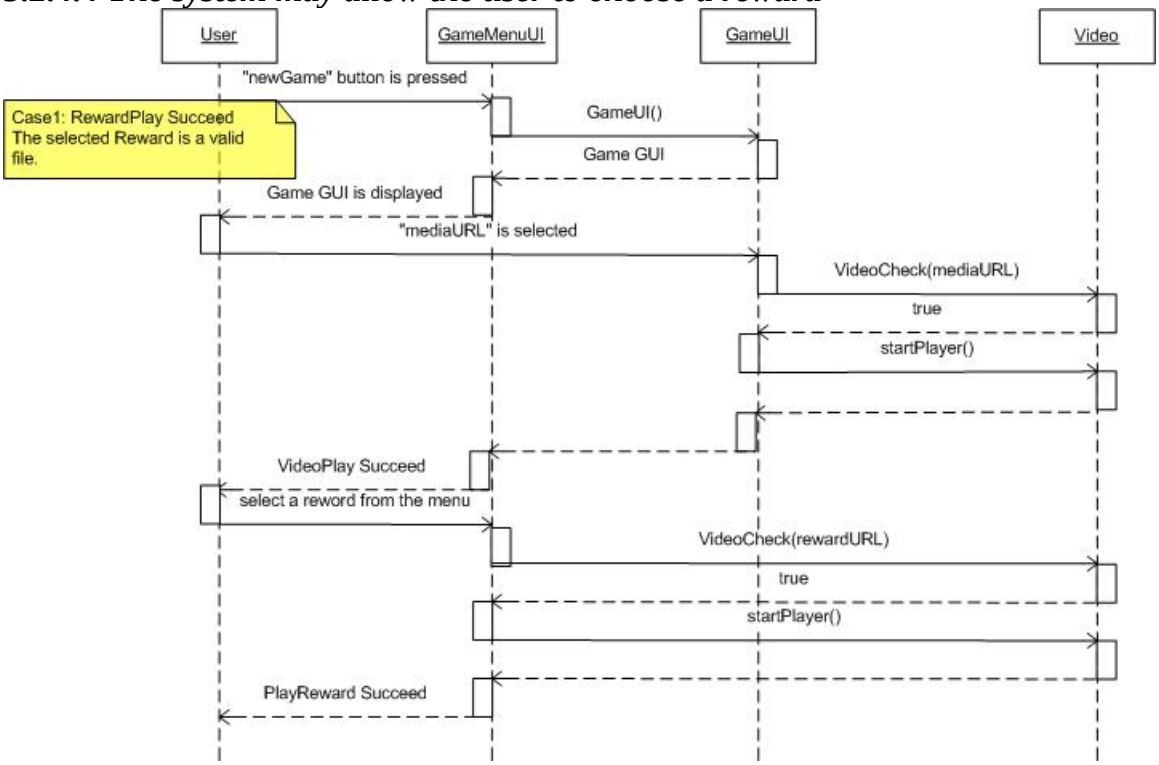
5.2.4.2 The system may allow the administrator to erase user records



5.2.4.3 The system may allow the user to modify their game setting



5.2.4.4 The system may allow the user to choose a reward



6 User Interface Design

6.1 Application Control

Common Look & Feel
All windows will be Java/Swing style windows. These windows will have no additional toolbars or buttons except for the *minimize*, *maximize*, and *close* buttons found in all application windows.

The window appearance on any given system may change depending on the current system desktop theme and operating system (i.e. OS X.4 or Windows).

Minimum Requirements
The user must be running either Macintosh OS X.4 or Windows 98 or higher. Users will be required to use a minimum screen resolution of 800 x 600.

6.2 Screenshots

Fig. 6.2.1 The Main Menu

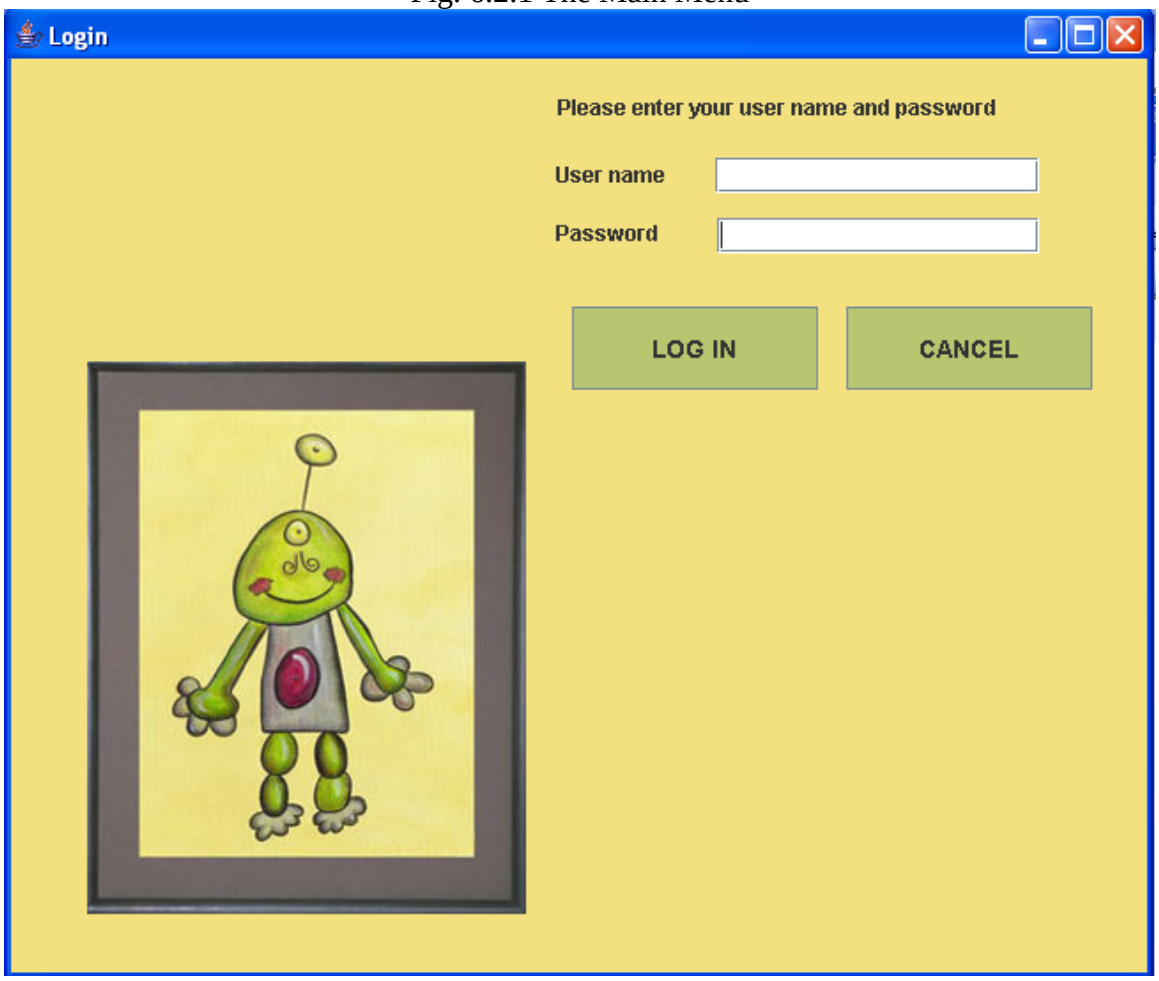


Fig 6.2.3 Login Menu

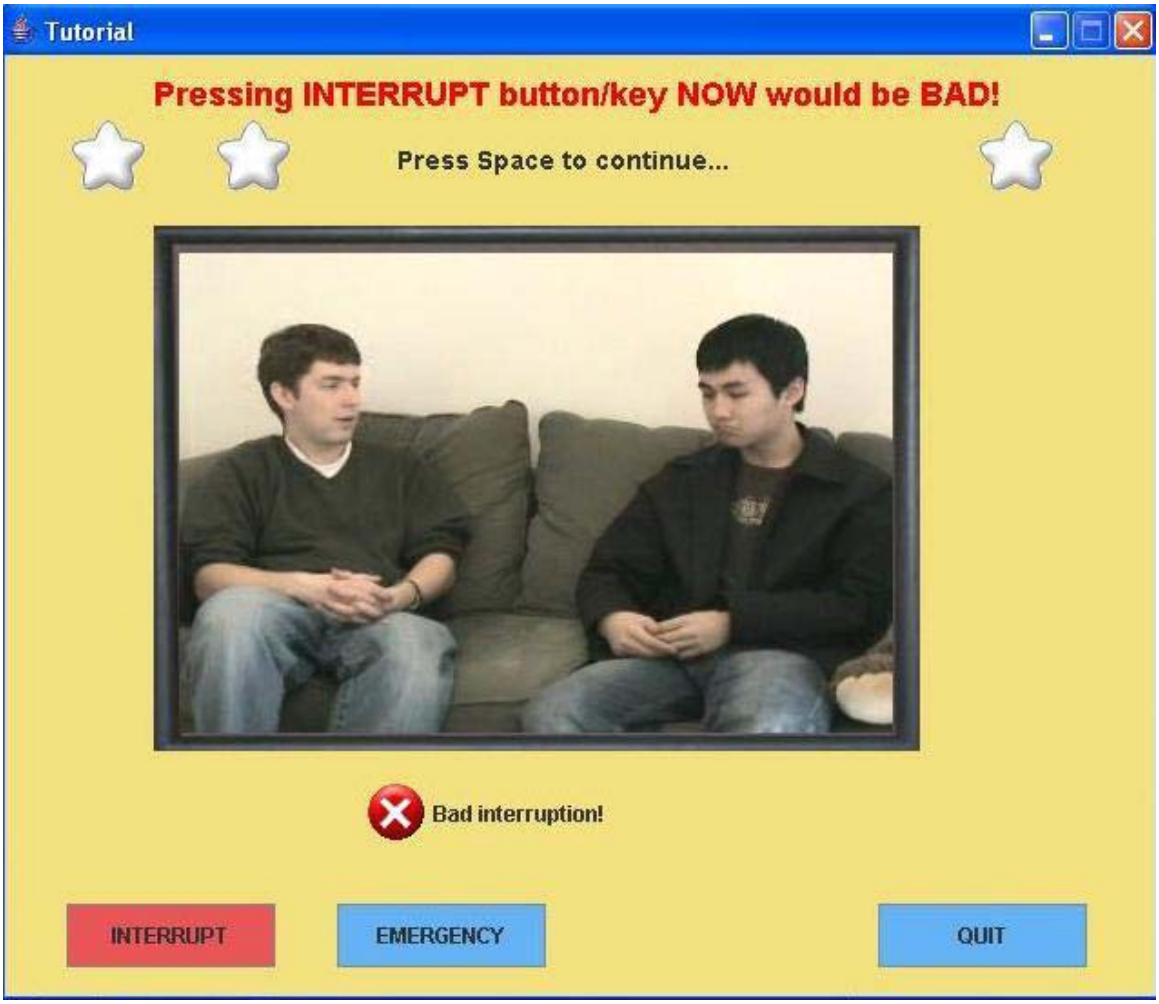


Fig 6.2.4 Tutorial

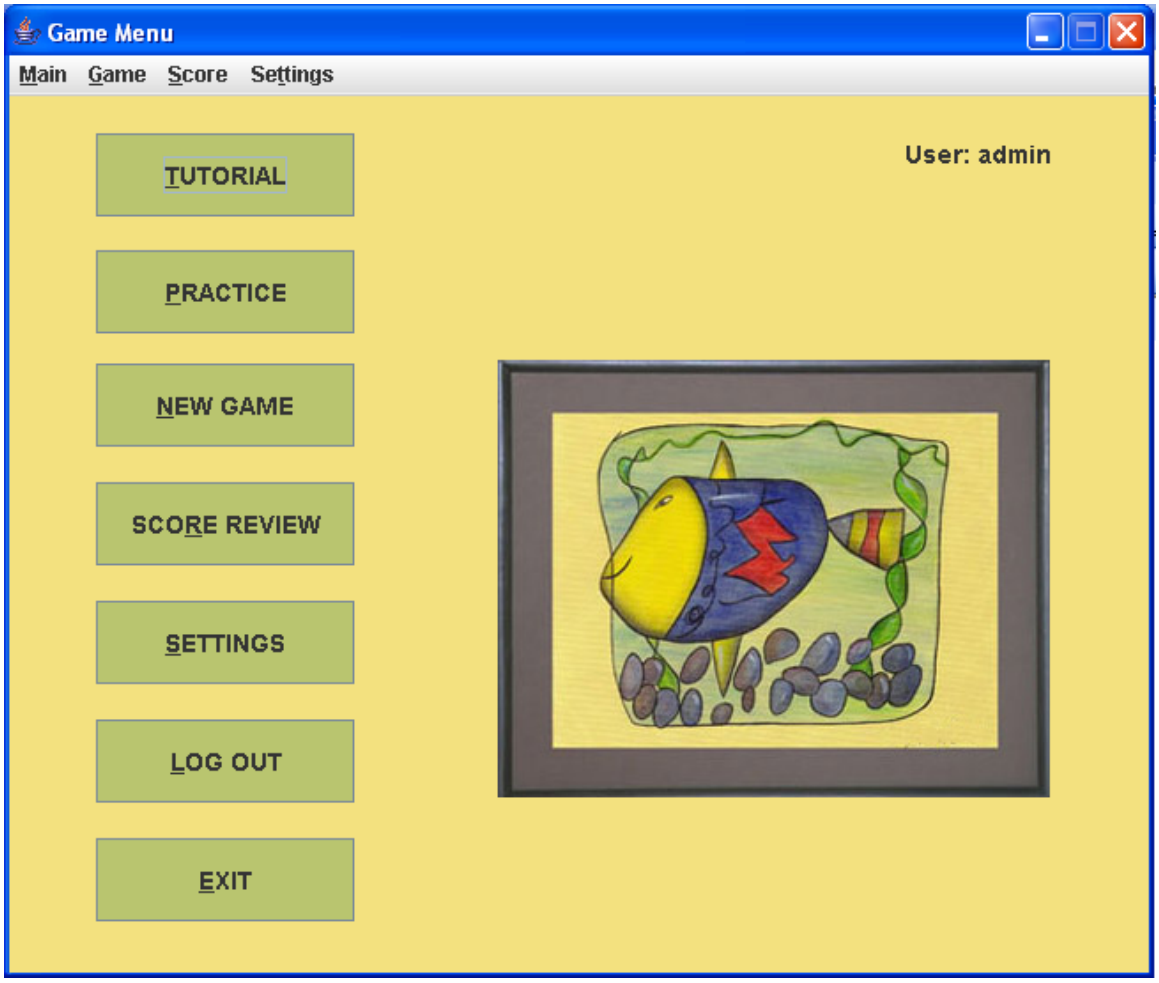


Fig 6.2.5 Game Menu

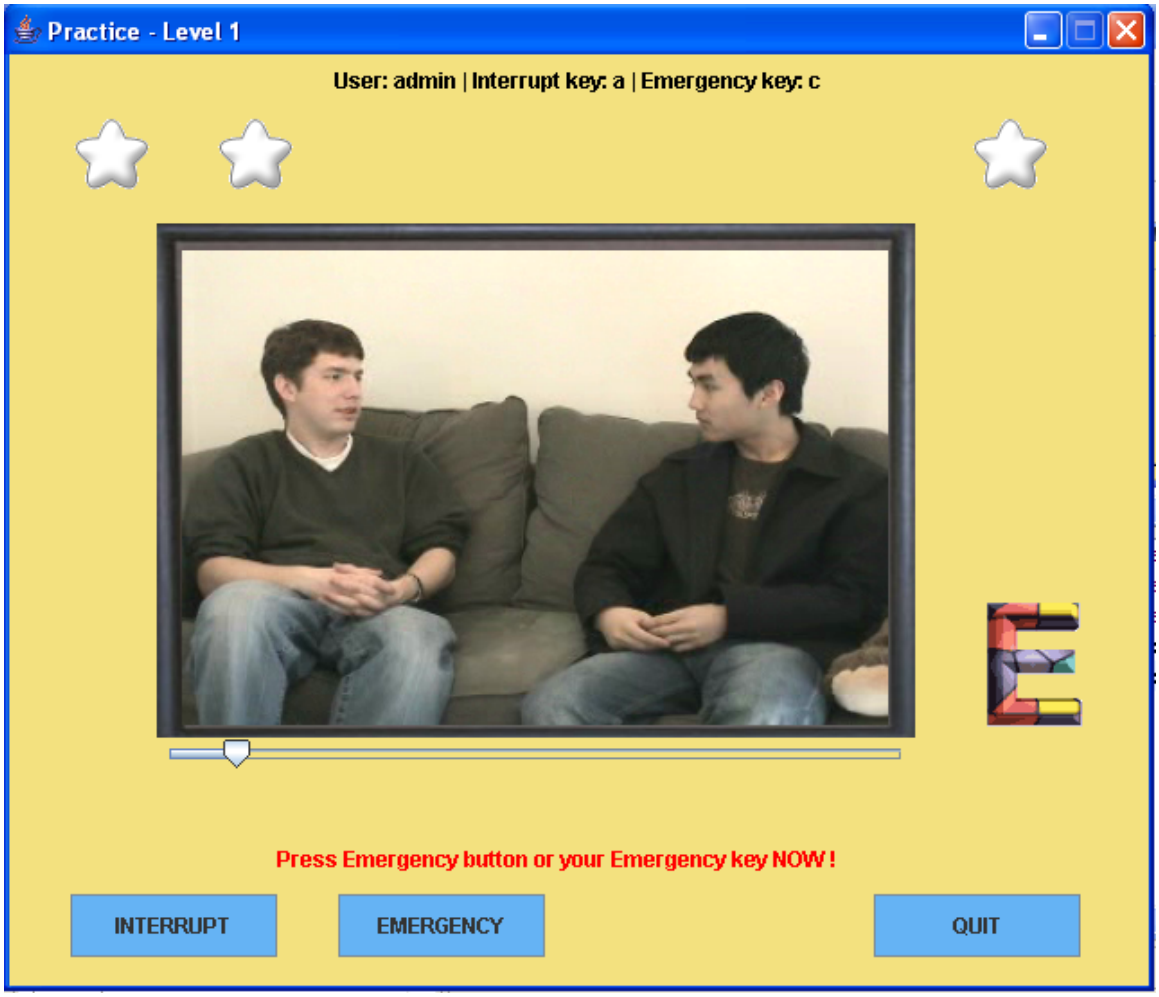


Fig. 6.2.6 Practice


Select game
Please select your game:
default easy
PLAY
CANCEL

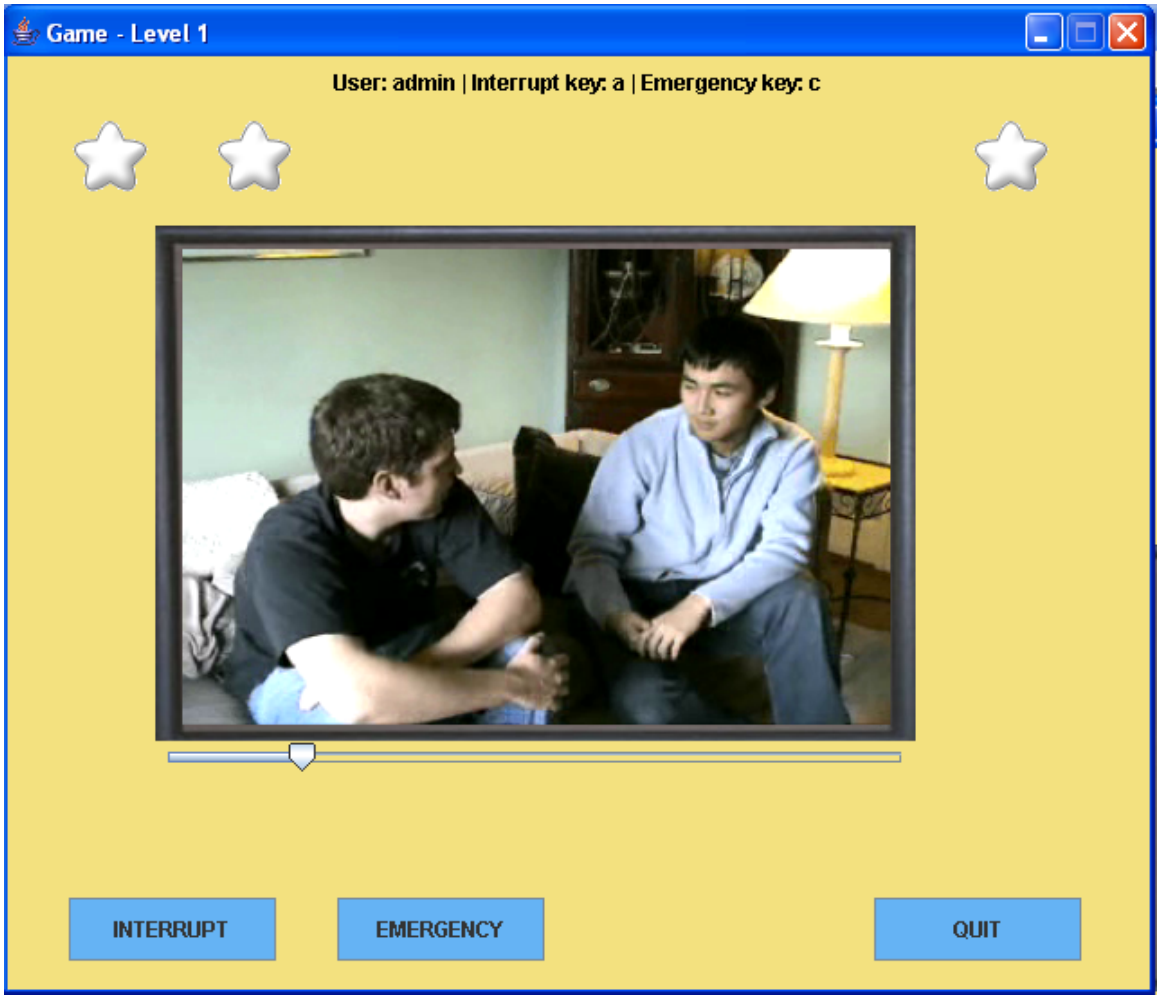


Fig. 6.2.8 Game Screenshots

Fig. 6.2.9 Interrupt

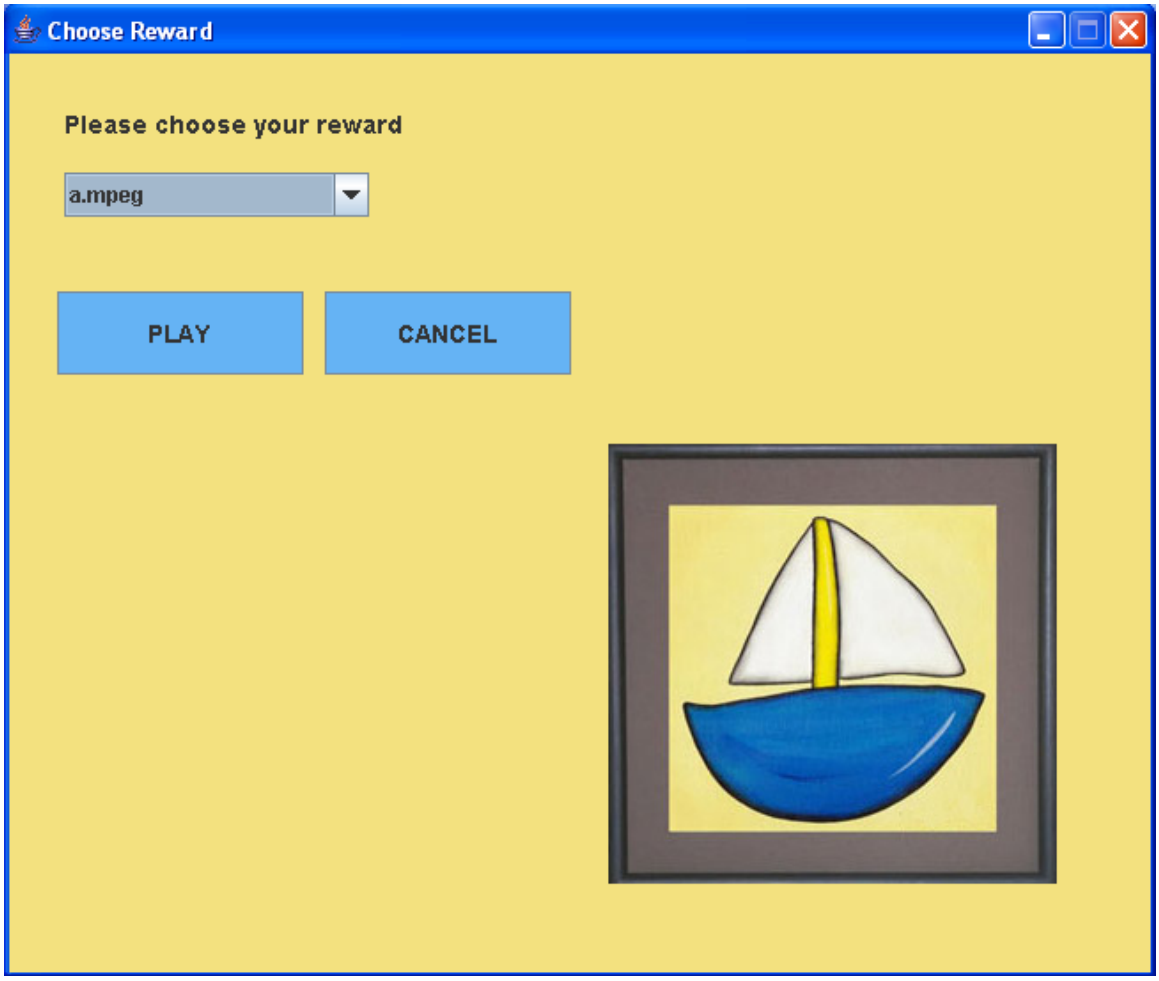


Fig. 6.2.10 Choose Reward

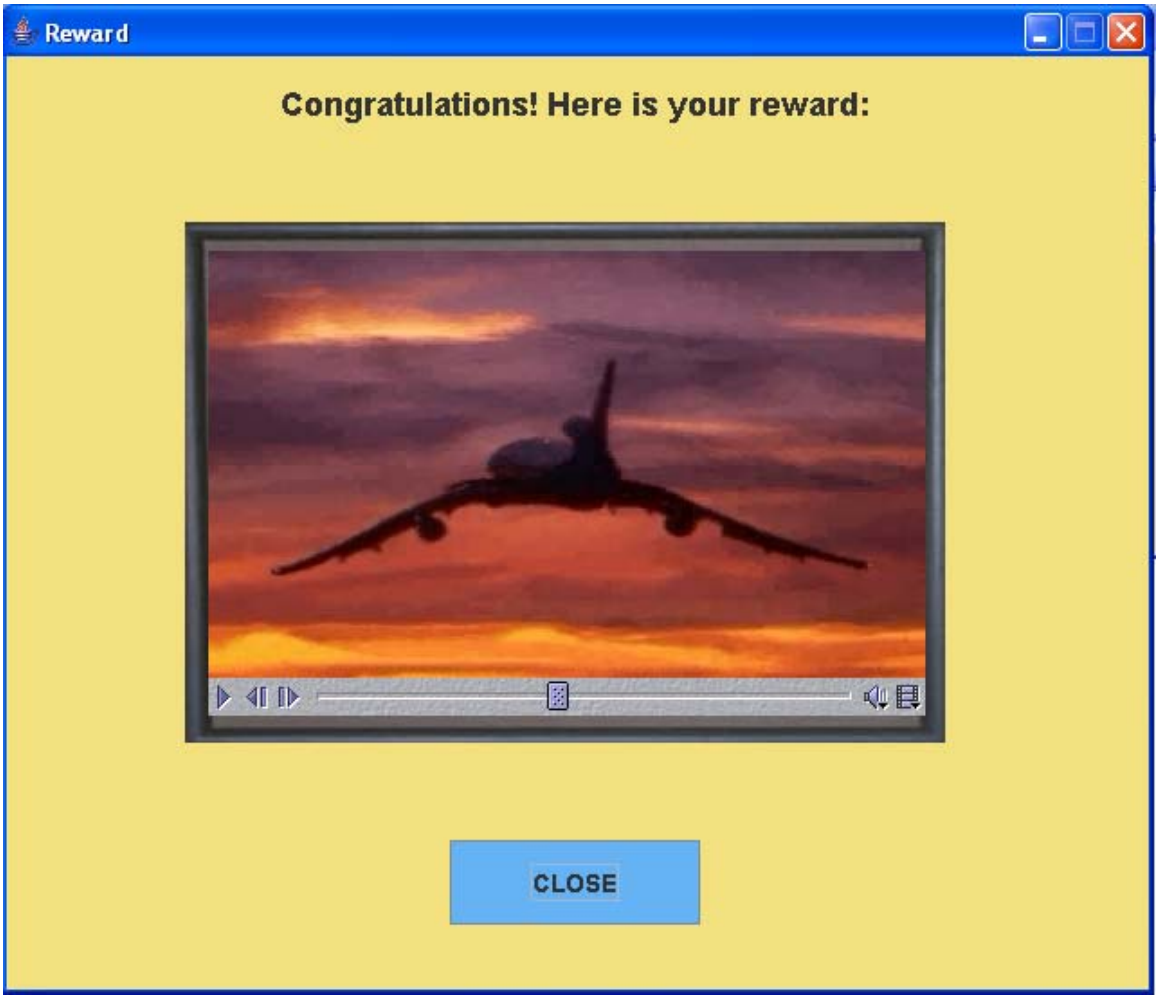


Fig 6.2.11 Reward

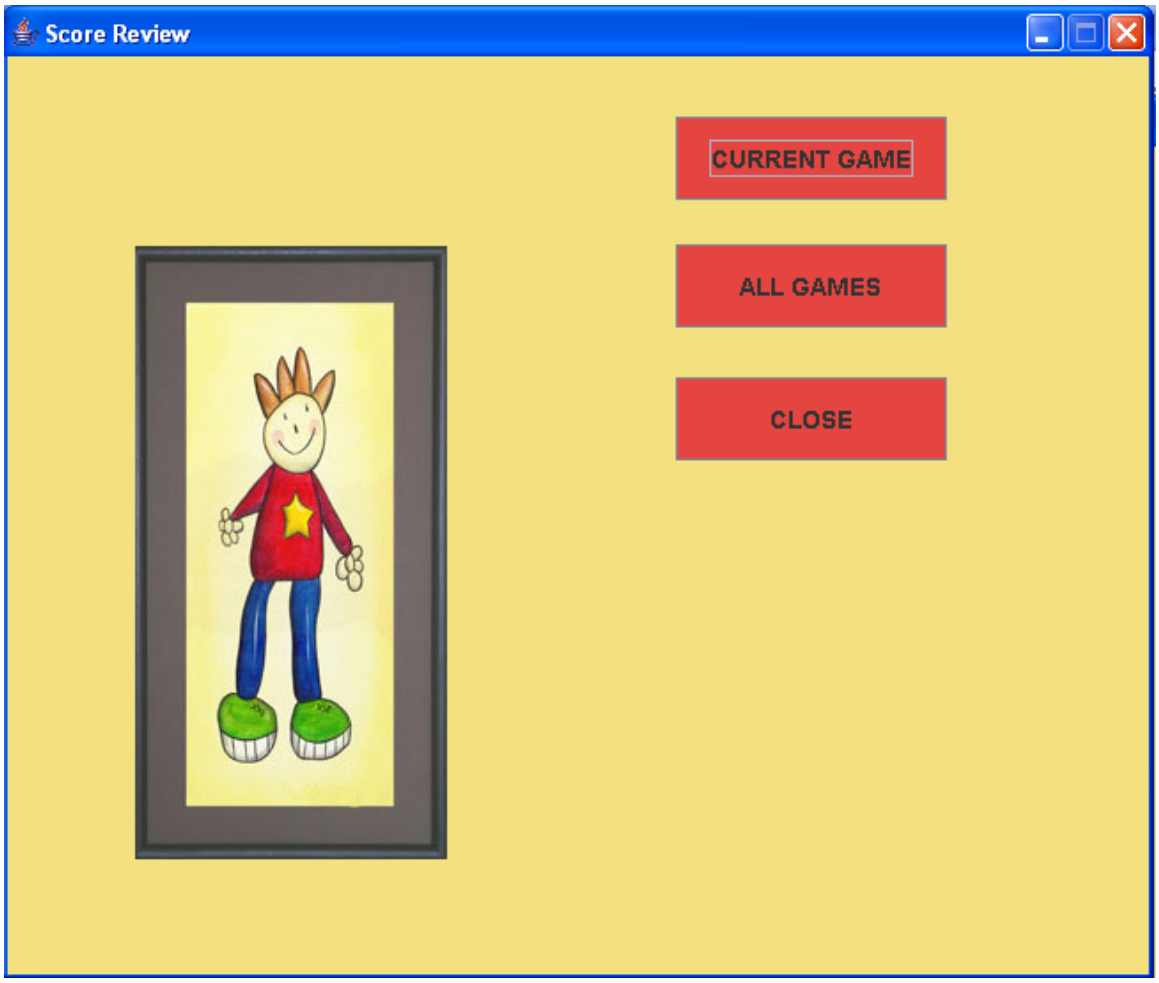


Fig 6.2.12 Score Review Menu

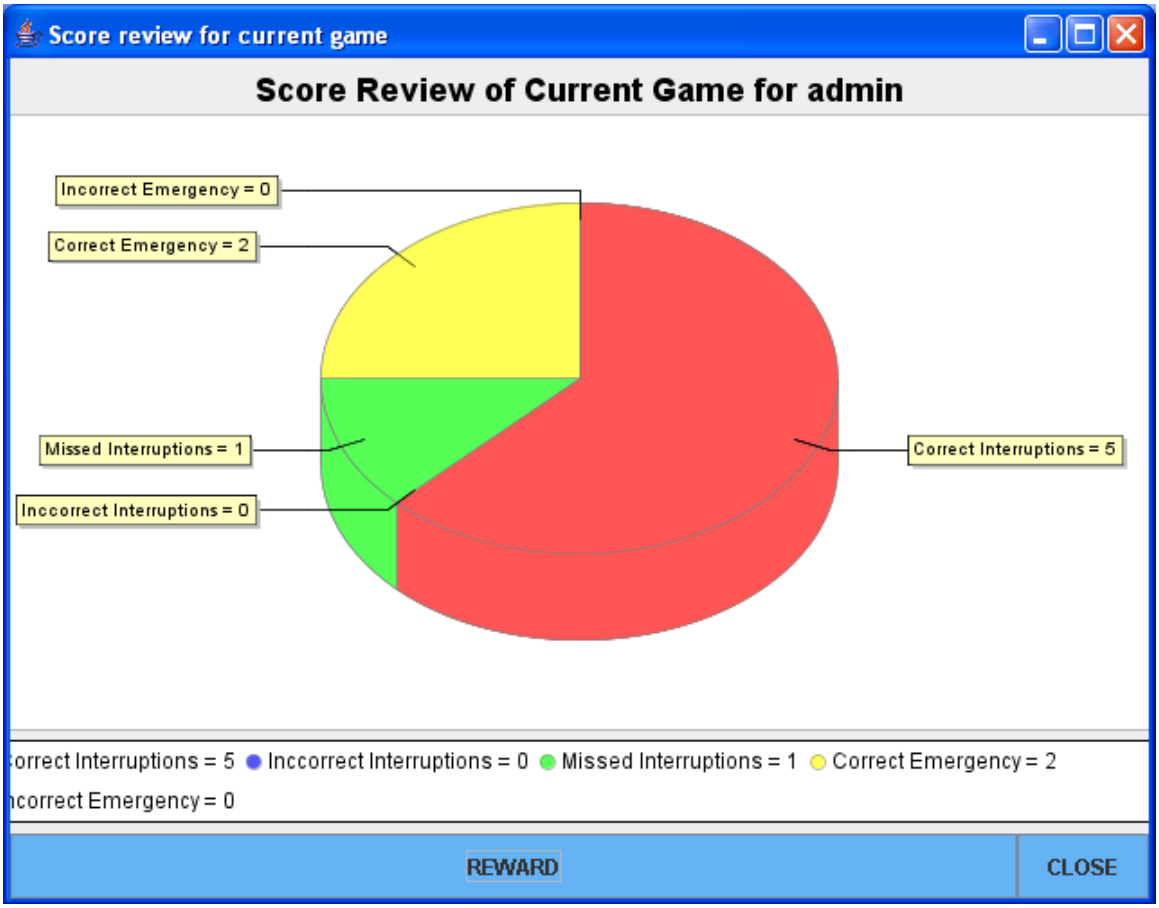


Fig. 6.2.13 Score Review for Current Game

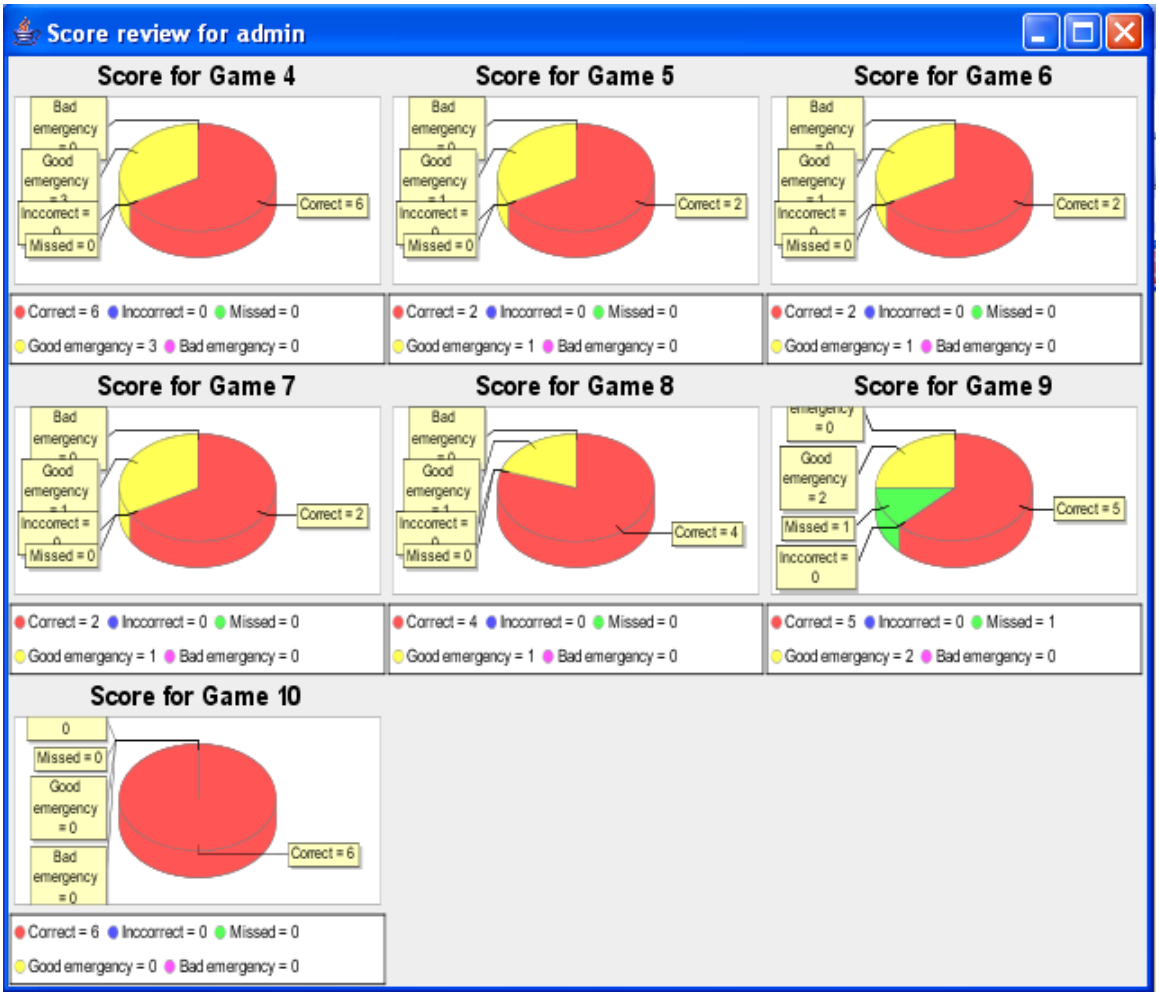


Fig. 6.2.14 Score Review for All Games

Fig. 6.2.15 Settings

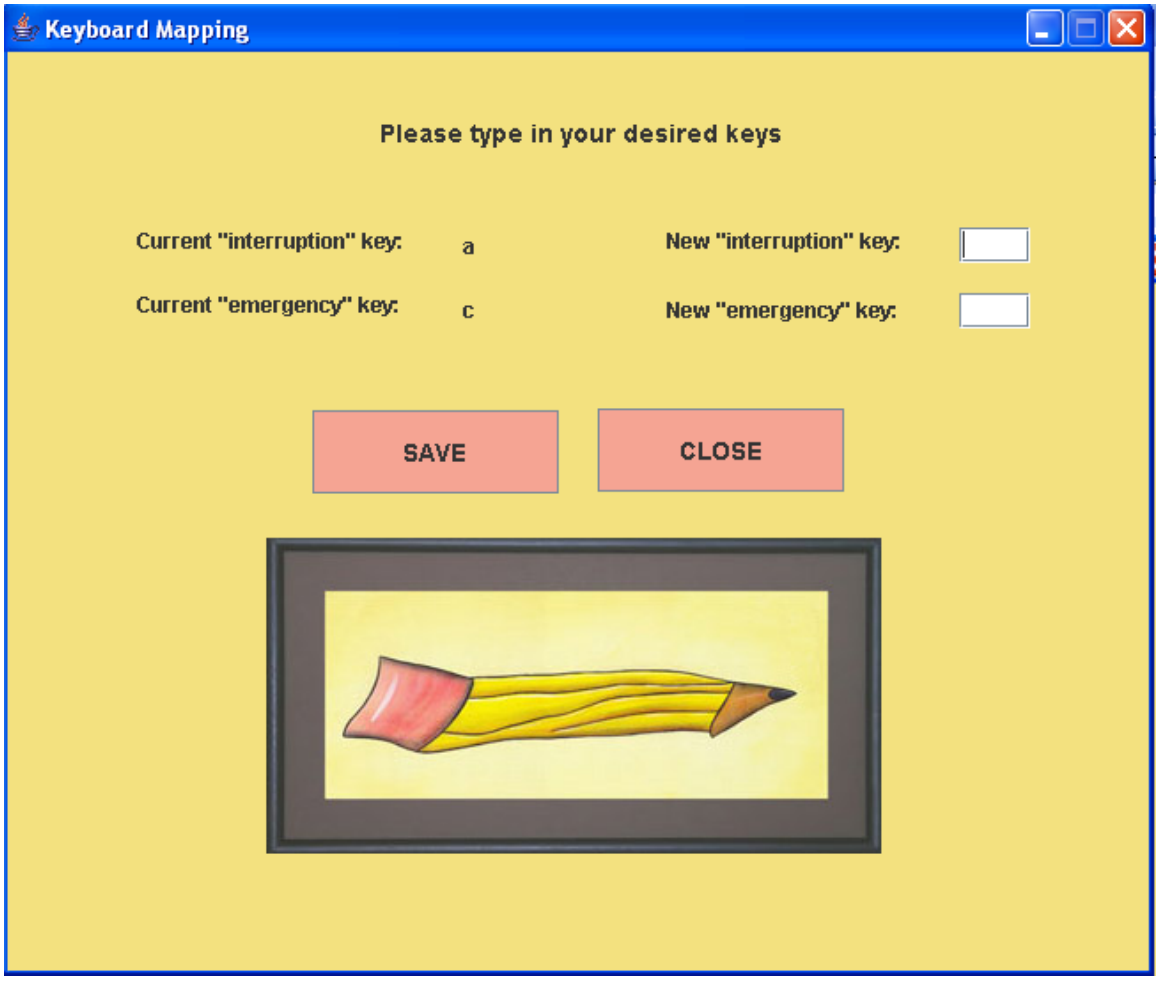


Fig. 6.2.17 Keyboard Mapping Menu

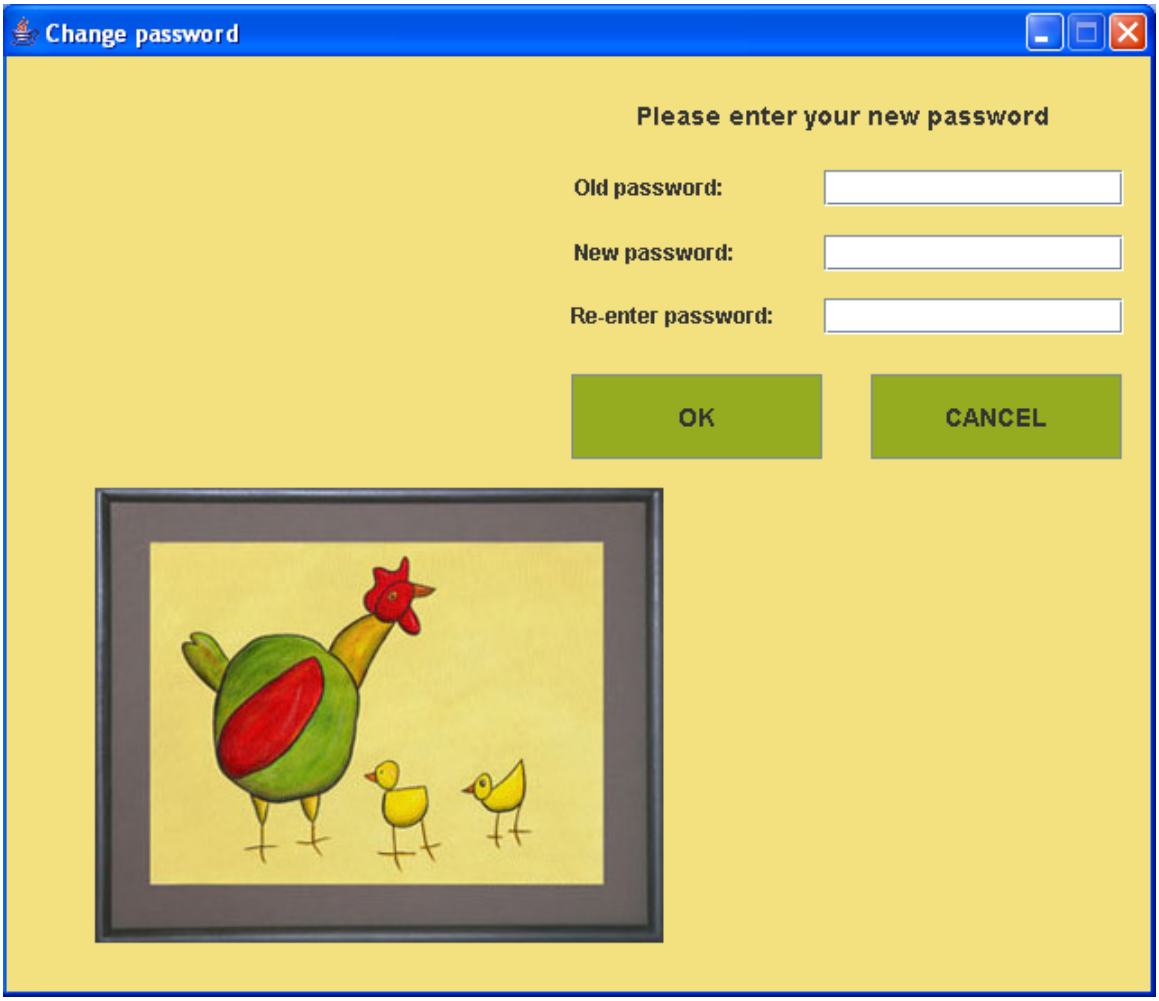


Fig. 6.2.18 Password Change

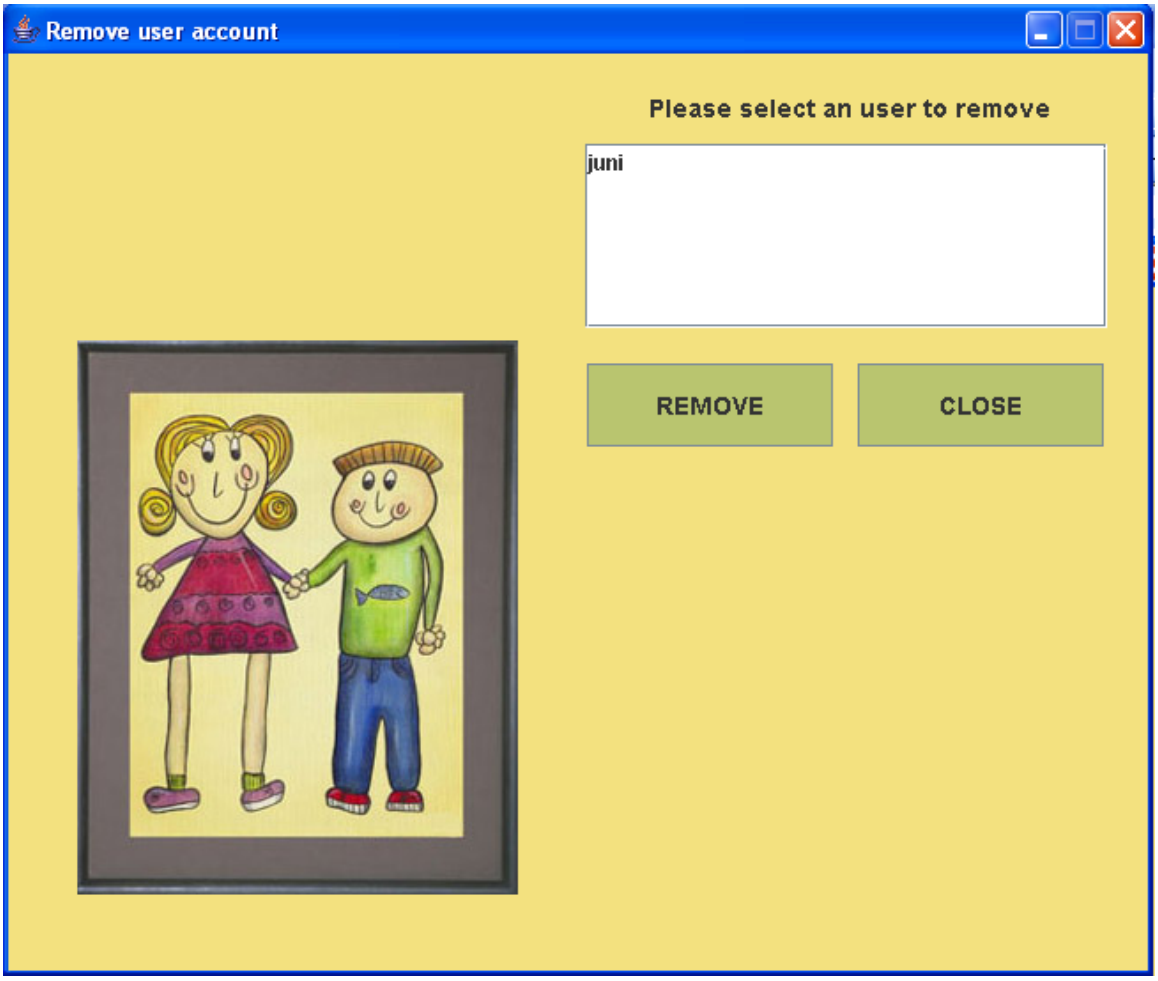


Fig. 6.2.19 Remove User

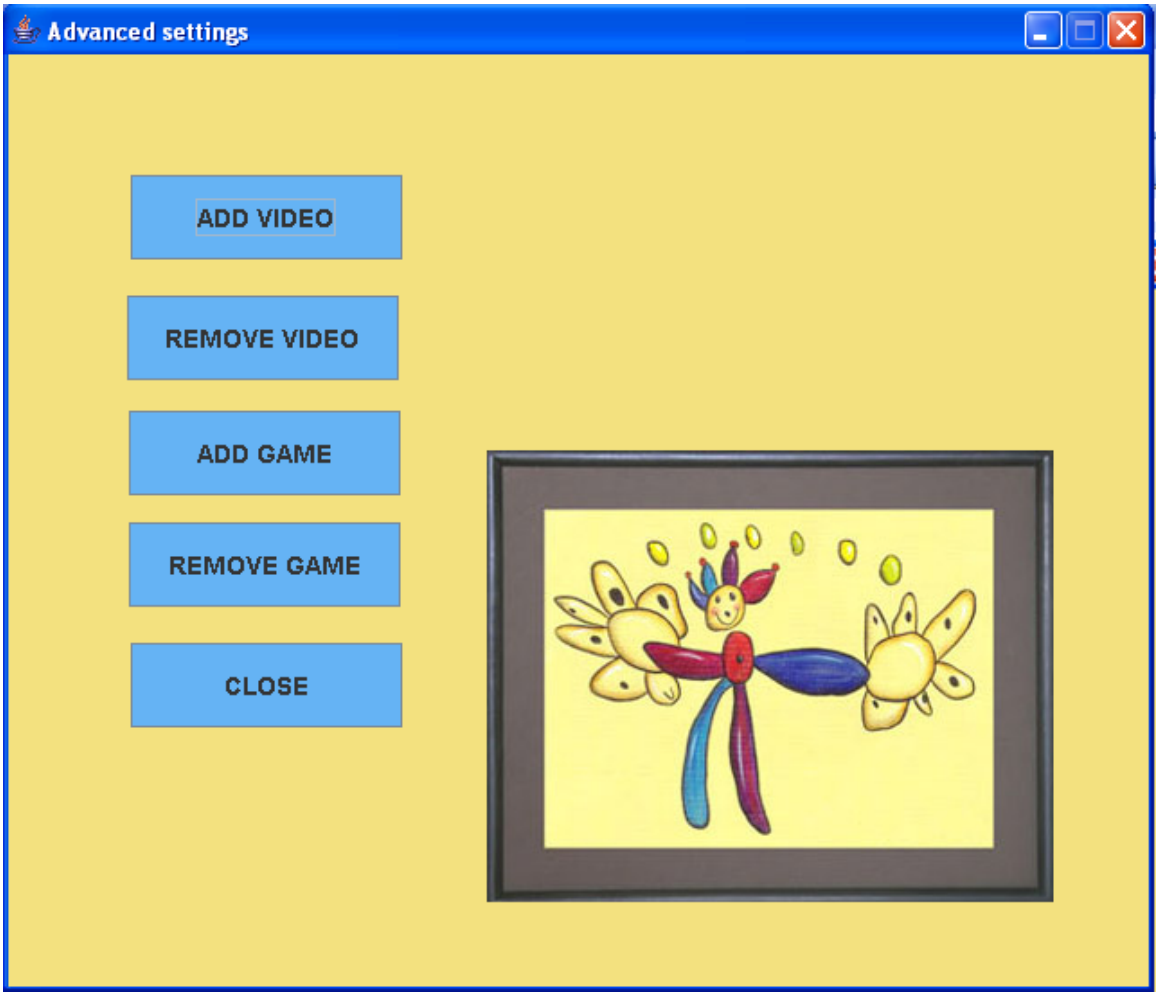


Fig. 6.2.20 Advanced Settings Menu

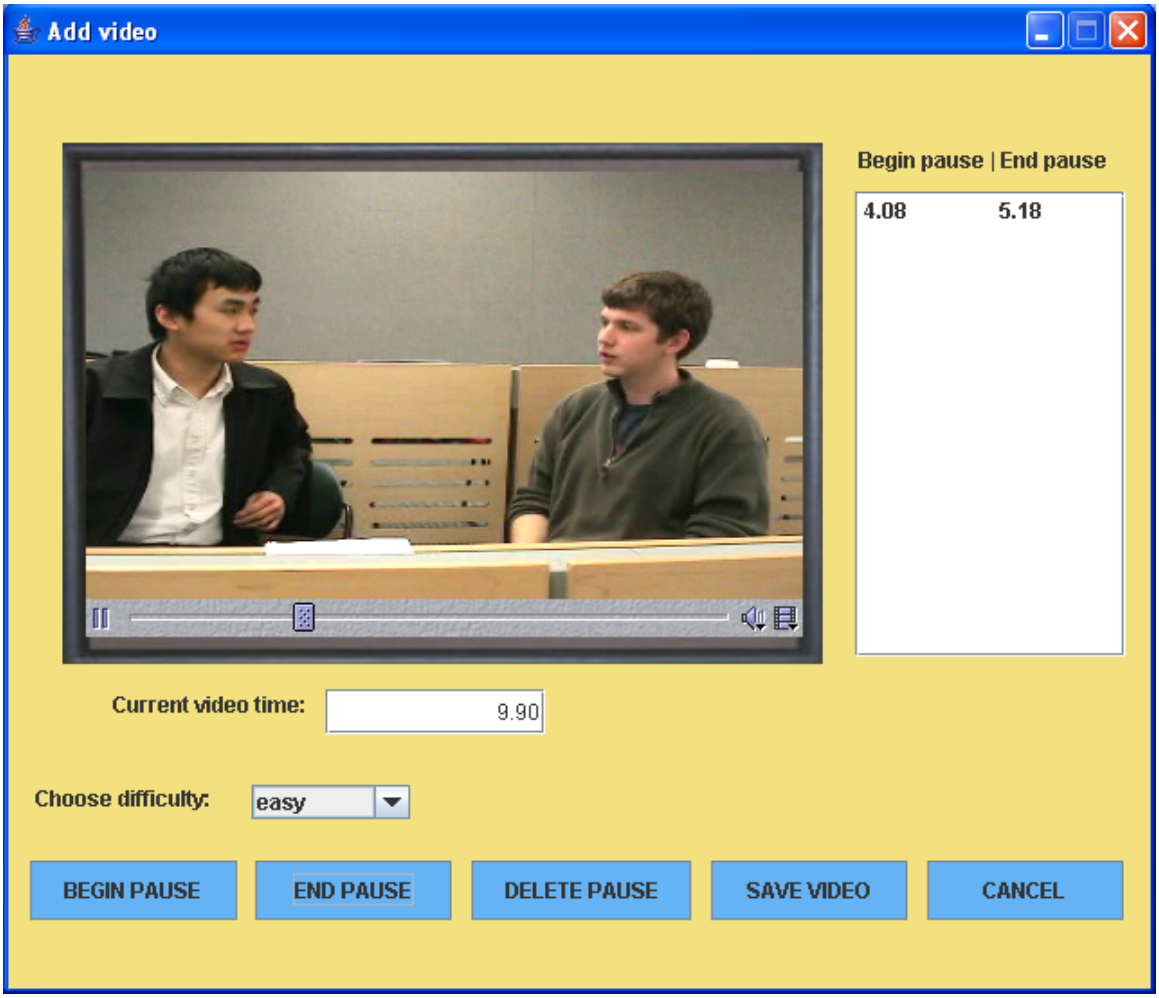


Fig. 6.2.21 Add Video

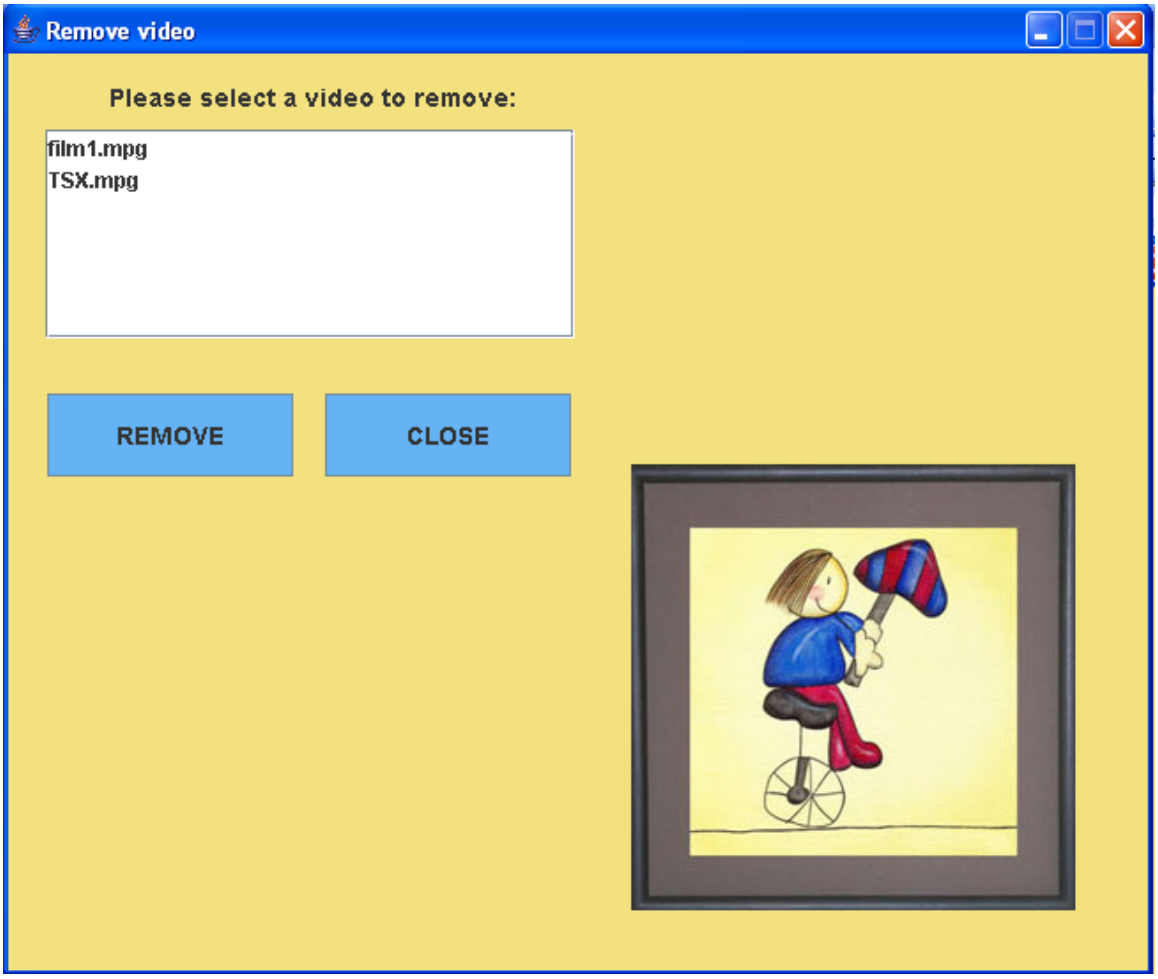


Fig. 6.2.22 Remove Video

Fig. 6.2.23 Add Game

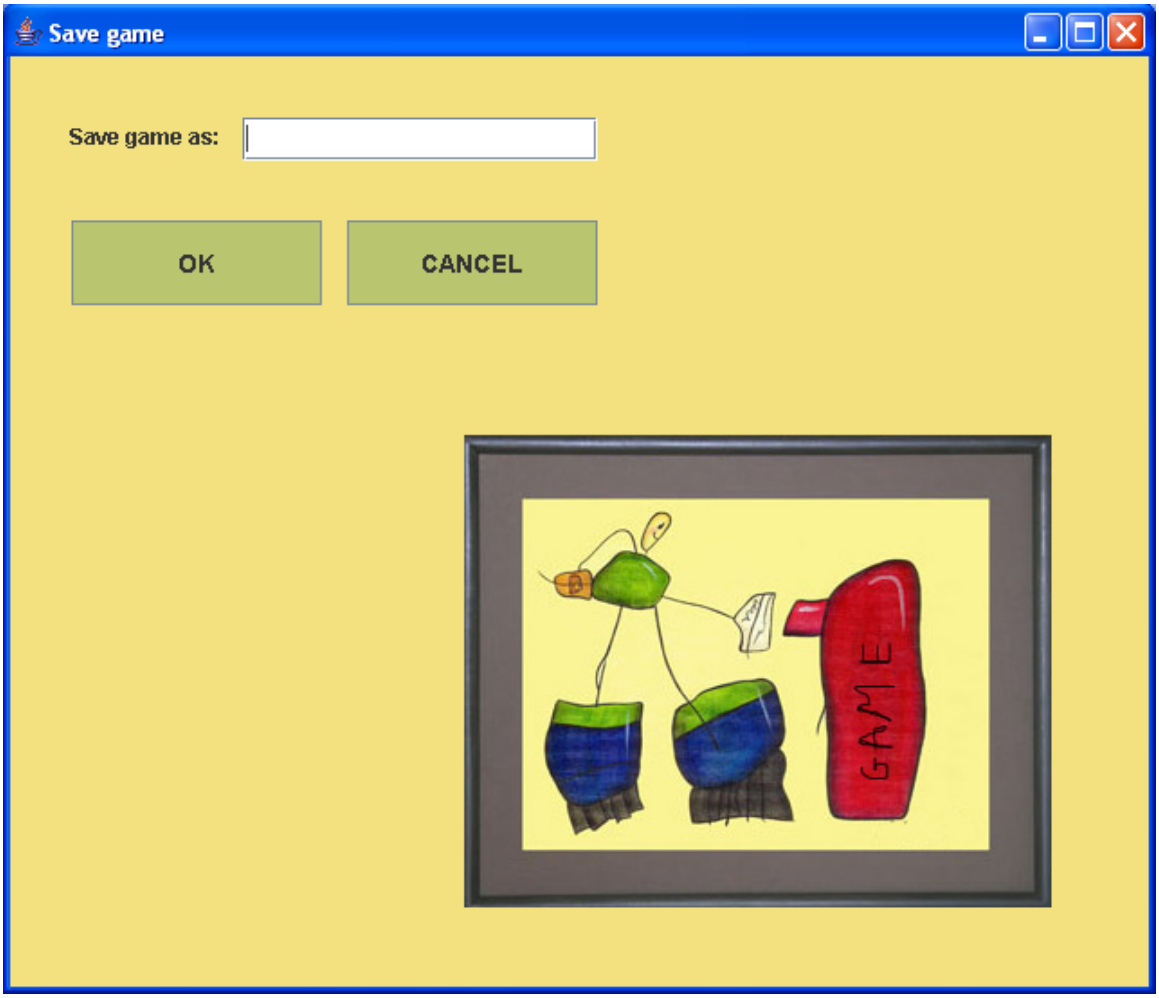


Fig. 6.2.24 Save Game

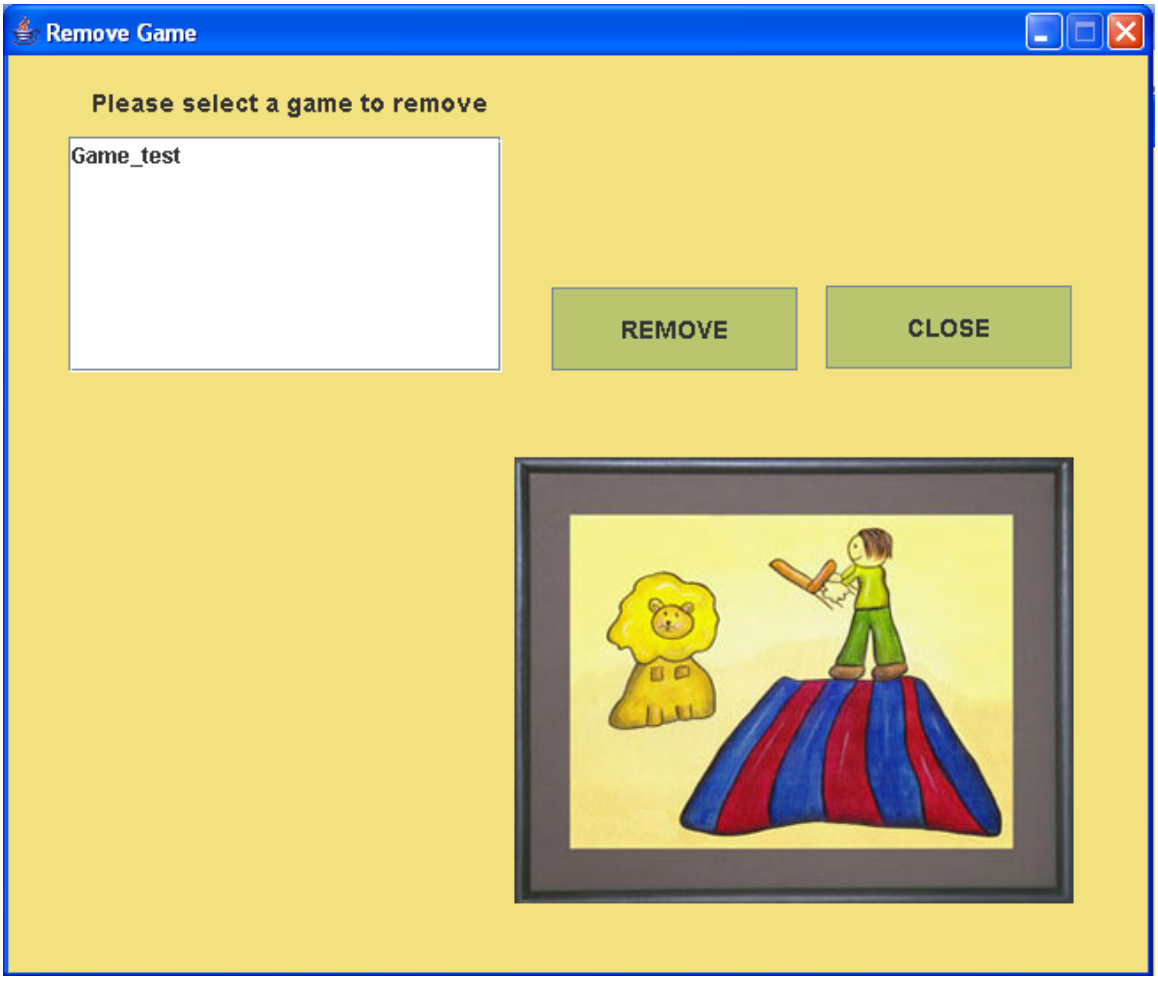


Fig. 6.2.24 Remove Game