

An Online Game-Based Learning System for Pre-school Children

Milad Ebadi

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the Degree of

Master of Science
in
Computer Engineering

Eastern Mediterranean University
February 2015
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Serhan Çiftçiöđlu
Acting Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Computer Engineering.

Prof. Dr. Işık Aybay
Chair, Department of Computer Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Computer Engineering.

Asst. Prof. Dr. Yıldıan Bitirim
Supervisor

Examining Committee

1. Asst. Prof. Dr. Yıldıan Bitirim

2. Asst. Prof. Dr. Cem Ergün

3. Asst. Prof. Dr. Ahmet Ünveren

ABSTRACT

In this thesis, the practical comprehensive system “online game-based learning” system for pre-school children has been developed in order to have a mobile game-based learning system which is able to conduct the children’s skill and/or knowledge development as well as being fun to be played by the children. This system meets all the aspects of mobile game-based learning beside of parent involvement in the procedure of learning. Key features of this system include: children’s playing and activity monitoring; evaluation of the games to detect the usability problems for developers; expert supporting system for parents; and internal messaging system. Our system also contains a dynamic Web service to let the third-party game developers to integrate a game into the system. Therefore, any educational tablet and/or smartphone game for pre-school children will be able to integrate into the system.

Keywords. Web application, Web service, mobile game-based learning.

ÖZ

Bu tezde, okul öncesi çocukların hem beceri ve/veya bilgi gelişimini sağlayacak hem de onlara eğlenceli gelecek bir mobil oyun tabanlı öğrenme sistemi için okul öncesi çocuklar için pratik geniş kapsamlı bir sistem olan “çevrimiçi oyun tabanlı öğrenme” sistemi geliştirilmiştir. Bu sistem mobil oyun tabanlı öğrenmenin tüm yönlerini içermenin yanında öğrenme prosedürüne ebeveyni de dahil etmektedir. Bu sistemin temel özellikleri şunlardır. çocukların oyun ve aktivitelerini izleme; geliştiriciler için kullanılabilirlik sorunların tespit amaçlı oyunların değerlendirilmesi; ebeveynler için uzman destek sistemi; ve iç mesajlaşma sistemi. bizim sistemimiz ayrıca üçüncü parti oyun geliştiricilerinin sistem içerisine oyun entegre edebilmesi için bir dinamik Web Servisi içerir. Bunda dolay, okul öncesi çocuklar için herhangi bir eğitimsel tablet ve/veya akıllı telefon oyun sistem içerisine entegre edilebilecektir.

Keywords. Web uygulaması, Web service, mobil oyun tabanlı öğrenme.

DEDICATION

This thesis work is dedicated to my parents, Majid Ebadi and Mahrokh Monsef, who have always loved me unconditionally and whose good examples have taught me to work hard for the things that I aspire to achieve.

This work is also dedicated to my siblings, Afshin and Noushin, who have been a constant source of support and encouragement during the challenges of the graduate school and life. I am truly thankful for having you in my life.

ACKNOWLEDGMENT

I would like to express my special appreciation and deepest gratitude to my supervisor Asst. Prof. Dr. Yıltan Bitirim; you have been a tremendous mentor for me. I would like to thank you for your excellent guidance, caring, patience, and providing me with an excellent atmosphere for doing research.

I would also like to thank my defense committee members, Asst. Prof. Dr. Ahmet Ünveren, Asst. Prof. Dr. Cem Ergün, Asst. Prof. Dr. Yıltan Bitirim for serving as my committee members even in hardship. I also want to thank you for letting my defense be an enjoyable moment, and for your brilliant comments and suggestions, thanks to you.

Last but not least, I would like to thank my dear parents, my brother, and my sister for their continuous supports in my life.

TABLE OF CONTENTS

ABSTRACT	iii
ÖZ	iv
DEDICATION	v
ACKNOWLEDGMENT	vi
LIST OF TABLES	ix
LIST OF FIGURES	xi
LIST OF ABBREVIATIONS	xvii
1 INTRODUCTION	1
2 BACKGROUND	4
2.1 Mobile Game-Based Learning	4
2.1.1 Evaluation of mGBL.....	4
2.1.2 Evaluation of Player Performance	7
2.2 Model-View-Controller (MVC)	7
2.3 Yii Framework.....	8
2.4 Representational State Transfer (REST) Web Service.....	10
2.5 Unity Game Engine	10
3 DESIGN OF THE SYSTEM	11
3.1 General Architecture of the System	11
3.2 Architecture of OGBL System	12
3.2.1 OGBL System User Types	14
3.2.2 Architecture of the Web Application Module	15
3.2.3 Architecture of the RESTful Web Service Module	29
3.3 Architecture of the Game Module	30

3.4 Database	31
4 IMPLEMENTATION	38
4.1 Web application module.....	38
4.1.1 Administrator sub module	39
4.1.2 Expert sub module	67
4.1.3 Parent sub module.....	74
4.2 Game module.....	79
5 COMPARISON WITH RELATED WORKS	83
6 CONCLUSION	87
REFERENCES.....	90
APPENDICES	95
Appendix A: PHEG Sub Heuristics	96
Appendix B: Controllers.....	98
Appendix C: Views	104
Appendix D: Interactions	111
Appendix E: Models.....	169
Appendix F: Users' Permissions	174
Appendix G: RESTful Web service	179
Appendix H: Game module.....	185

LIST OF TABLES

Table 1. Percentage of heuristic weight according to category	6
Table 2. RESTFul Web services operations	29
Table A.1. List of PHEG sub heuristics	96
Table B.1. Site controller	98
Table B.2. Admin controller	98
Table B.3. Experts controller	99
Table B.4. Parents controller.....	99
Table B.5. Children controller	100
Table B.6. Game controller	100
Table B.7. Lesson controller	100
Table B.8. Category controller.....	101
Table B.9. Tips controller	101
Table B.10. PHEG controller	101
Table B.11. Activity controller	102
Table B.12. Assist controller.....	102
Table B.13. Score controller	102
Table B.14. Session controller	102
Table B.15. Settings controller	103
Table B.16 Mailbox controller	103
Table E.1. tableName() method	169
Table E.2. model() method.....	169
Table E.3. rules () method.....	169
Table E.4. attributeLabels () method	169
Table E.5. relations () method.....	170

Table E.6. search () method	170
Table E.7. beforeValidate () method.....	170
Table E.8. afterValidate () method.....	170
Table E.9. defaultScope () method.....	171
Table E.10. encrypt () method.....	171
Table E.11. getChild () method.....	171
Table E.12. getGame () method	171
Table E.13. getReceiver () method	172
Table E.14. inboxSearch () method	172
Table E.15. outboxSearch () method	172
Table E.16. searchNotEvaluated () method	173
Table F.1. Users' permissions.....	174
Table G.1. Get list of the children.....	179
Table G.2. Get a child's details.....	179
Table G.3. Get settings.....	180
Table G.4. Get list of the games.....	180
Table G.5. Get a game's details	181
Table G.6. Get the list of a game's lessons	181
Table G.7. Get a lesson's details.....	182
Table G.8. Start a session for a child	182
Table G.9. Stop a session for child	183
Table G.10. Sending score	184

LIST OF FIGURES

Figure 1. MVC architecture	8
Figure 2. General work flow of a Yii framework based Web application.....	9
Figure 3. General architecture of the system	11
Figure 4. General architecture of OGBL system	13
Figure 5 . List of controller's classes and their actions.....	17
Figure 6. List of controller's classes and their actions (Cont.)	16
Figure 7. The "Model" component class diagram.....	28
Figure 8. Example of cURL authentication request.....	30
Figure 9. Database entity relationship diagram.....	32
Figure 10. "Login" Page.....	39
Figure 11. "Homepage" page	40
Figure 12. "Manage Parents" page.....	40
Figure 13. "View Parent" page	41
Figure 14. "Update Parent" page.....	42
Figure 15. Confirmation alert for delete a record	43
Figure 16. "Create Parent" page.....	44
Figure 17. "Manage Experts" page	44
Figure 18. "View Expert" page	45
Figure 19. "Update Expert" page	46
Figure 20. "Create Expert" page	47
Figure 21. "Manage Administrators" page.....	50
Figure 22. "View Administrator" page	49
Figure 23. "Update Administrator" page	50
Figure 24. "Create Administrator" page	51

Figure 25. "Manage Games" page	52
Figure 26. "View Game" page	53
Figure 27. "Update Game" page	54
Figure 28. "List Games" page	55
Figure 29. "Create Game" page	56
Figure 30. "Manage Lessons" page.....	57
Figure 31. "Update Lesson" page.....	60
Figure 32. "Create Lesson" page.....	59
Figure 33. "Manage Tips" Page	59
Figure 34. "Update Tips" Page.....	60
Figure 35. "Create Tips" Page.....	61
Figure 36. "Evaluations" page	62
Figure 37. "View Evaluation" page	63
Figure 38. "Inbox" page	64
Figure 39. "Compose" page	65
Figure 40. "View Mail" page	65
Figure 41. "Sent Mail" page.....	66
Figure 42. "Profile Settings" page.....	67
Figure 43. "Evaluations" page	68
Figure 44. "Create Evaluation" page.....	69
Figure 45. "View Evaluation" page	70
Figure 46. "Children Activity" page	70
Figure 47. "View Child Activity" page, "Timeline" section.....	71
Figure 48. "View Child Activity" page, "Playing Hours" section	72
Figure 49. "View Child Activity" page, "Playing Hours by Category" section	72

Figure 50. "Assist Requests" page	73
Figure 51. "View Assist Request" page	74
Figure 52. "Manage Children" page.....	75
Figure 53. "Update Child" page	76
Figure 54. "Add Child" page.....	77
Figure 55. "Request Assist" page.....	78
Figure 56. "Games Settings" page	78
Figure 57. "Login" page	80
Figure 58. "Children" page	80
Figure 59. "Games" page	81
Figure 60. "Playing game" page	82
Figure 61 . "Leo's Pad" application	83
Figure 62. "Fit Brains. Sparky's Adventures!" application	84
Figure 63. "Learn With Homer" application.....	85
Figure C.1. AdminController views.....	104
Figure C.2. ExpertsController views.....	104
Figure C.3. ParentsController views	105
Figure C.4. ChildrenController views.....	105
Figure C.5. GameController views	106
Figure C.6. LessonController views	106
Figure C.7. CategoryController views	107
Figure C.8. TipsController views.....	107
Figure C.9. PHEGController views	108
Figure C.10. ActivityController views.....	108
Figure C.11. AssistController views	109

Figure C.12. SessionController views.....	109
Figure C.13. SettingsController views	109
Figure C.14. MailboxController views	110
Figure D.1. Login dataflow diagram and sequence diagram	111
Figure D.2. Logout dataflow diagram and sequence diagram	112
Figure D.3. Parents registration dataflow diagram and sequence diagram.....	113
Figure D.4. Manage administrators dataflow diagram and sequence diagram	114
Figure D.5. Create administrator dataflow diagram and sequence diagram	115
Figure D.6. View administrator dataflow diagram and sequence diagram.....	116
Figure D.7. Update administrator dataflow diagram and sequence diagram.....	117
Figure D.8. Delete administrator dataflow diagram and sequence diagram	118
Figure D.9. Manage experts dataflow diagram and sequence diagram	119
Figure D.10. Create expert dataflow diagram and sequence diagram	120
Figure D.11. View expert dataflow diagram and sequence diagram	121
Figure D.12. Update expert dataflow diagram and sequence diagram	122
Figure D.13. Delete expert dataflow diagram and sequence diagram	123
Figure D.14. Manage parents dataflow diagram and sequence diagram	124
Figure D.15. Create parent dataflow diagram and sequence diagram	125
Figure D.16. View parent dataflow diagram and sequence diagram	126
Figure D.17. Update parent dataflow diagram and sequence diagram	127
Figure D.18. Delete parent dataflow diagram and sequence diagram	128
Figure D.19. Manage children dataflow diagram and sequence diagram.....	129
Figure D.20. Create child dataflow diagram and sequence diagram	130
Figure D.21. View child dataflow diagram and sequence diagram	131
Figure D.22. Update child dataflow diagram and sequence diagram	132

Figure D.23. Delete child dataflow diagram and sequence diagram	133
Figure D.24. Manage games dataflow diagram and sequence diagram.....	134
Figure D.25. Create game dataflow diagram and sequence diagram.....	135
Figure D.26. View game dataflow diagram and sequence diagram	136
Figure D.27. Update game dataflow diagram and sequence diagram.....	137
Figure D.28. Delete game dataflow diagram and sequence diagram.....	138
Figure D.29. Manage lesson dataflow diagram and sequence diagram.....	139
Figure D.30. Create lesson dataflow diagram and sequence diagram	140
Figure D.31. View lesson dataflow diagram and sequence diagram	141
Figure D.32. Update lesson dataflow diagram and sequence diagram	142
Figure D.33. Delete lesson dataflow diagram and sequence diagram	143
Figure D.34. Manage categories dataflow diagram and sequence diagram.....	144
Figure D.35. Create category dataflow diagram and sequence diagram.....	145
Figure D.36. Update category dataflow diagram and sequence diagram	146
Figure D.37. Delete category dataflow diagram and sequence diagram.....	147
Figure D.38. Delete category dataflow diagram and sequence diagram.....	148
Figure D.39. Create tip dataflow diagram and sequence diagram	149
Figure D.40. View tip dataflow diagram and sequence diagram.....	150
Figure D.41. Update tip dataflow diagram and sequence diagram	151
Figure D.42. Delete tip dataflow diagram and sequence diagram	152
Figure D.43. Manage PHEGs dataflow diagram and sequence diagram.....	153
Figure D.44. Create PHEG dataflow diagram and sequence diagram.....	154
Figure D.45. View PHEG dataflow diagram and sequence diagram.....	155
Figure D.46. Update PHEG dataflow diagram and sequence diagram.....	156
Figure D.47. Delete PHEG dataflow diagram and sequence diagram.....	157

Figure D.48. Manage activities dataflow diagram and sequence diagram	158
Figure D.49. View activity dataflow diagram and sequence diagram	159
Figure D.50. Manage assists dataflow diagram and sequence diagram.....	160
Figure D.51. Create assist dataflow diagram and sequence diagram.....	161
Figure D.52. View assist dataflow diagram and sequence diagram	162
Figure D.53. View assist dataflow diagram and sequence diagram	163
Figure D.54. Update setting dataflow diagram and sequence diagram.....	164
Figure D.55. Manage Inbox/Outbox dataflow diagram and sequence diagram	165
Figure D.56. Compose Mail dataflow diagram and sequence diagram	166
Figure D.57. View E-mail dataflow diagram and sequence diagram	167
Figure D.58. Delete e-mail dataflow diagram and sequence diagram	168
Figure H.1. Sample games' dataflow diagram and sequence diagram	185

LIST OF ABBREVIATIONS

AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
CRUD	Create, Read, Update and Delete
cURL	Client URL Request Library
DAO	Data Access Objects
ER	Entity Relationship
JSON	Java Script Object Notation
HTML	Hypertext Markup Language
HTTP	HyperText Transfer Protocol
mGBL	Mobile Game-Based Learning
MVC	Model-View-Controller
MySQL	My Structured Query Language
OGBL	Online Game-Based Learning
PHEG	Playability Heuristics Evaluation for Educational Computer Game
PHP	Hypertext Preprocessor
REST	Representational State Transfer
UI	User Interface
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
Yii	Yes it is!

Chapter 1

INTRODUCTION

Based on the studies [1-4], speed of the adoption of the consumers to smartphones is phenomenally high. comScore reports show that the number of US mobile users owned a smartphone increased from four percent in 2007 to about seventy-three percent in 2014 [1,2]. In another report, GSMA Intelligence predicted that global smartphone adoption will triple by 2020 [3].

Also, the growth of tablets is equally impressive. As a report, the number of adults that owned a tablet device in the U.S. increased from three percent in 2010 to more than forty-two percent in 2014 [5]. This growth could also be predicted for the entire world.

In other hand, more and more children are using smartphone and tablet devices nowadays. Over seventy-five percent of children less than eight years of age in U.S. have access to a smartphone or tablet device and over seventy-two percent of them have used them [6].

The age between two to six is generally considered as pre-school children. Beside of huge growth of the usage of smartphones and tablets; we believe that education of pre-school children is one of the main usages of these facilities. In 2012, the National Association for the Education of Young Children (NAEYC) reported that “When

used intentionally and appropriately, technology and interactive media are effective tools to support learning and development” for pre-school children [7]. On the other hand, Nielson survey shows that fifty-seven percent of children in tablet-owning households use their tablets for educational games and apps [8]. Also, another analysis on education category of Apple’s app store showed that more than seventy-two percent of best-selling paid apps are targeted towards pre-school children [9].

According to references [10-12], mobile Game-Based Learning (mGBL) is a game played on any handheld devices such as smartphones and tablets which should be attractive and also meet specific educational goals. As a result of these studies, evaluation of mGBL and evaluation of the players’ performance are two major aspects that should be concern in mGBL.

According to above, the importance of work on mGBL for pre-school children arises. In our project, we decided to implement a practical system which meets both mentioned major aspects of mGBL for pre-school children, beside of improvement of the parents’ involvement in the procedure of learning. Our system also provides an expert supporting system and internal messaging system to help parents. Furthermore, we implemented a dynamic Web service to distribute our system with third-party game developers. Third-party game developers are able to integrate their games into our system through the Web service. Additionally, we implemented an online evaluation tool that can be used for helping the developers to detect the usability problems of the games.

In the rest of this report, background information is given in chapter 2, design of the system is described in chapter 3, implementation of our system is given in chapter 4, comparison of our system with related works is described on chapter 5 and finally, thesis is concluded in chapter 6.

Chapter 2

BACKGROUND

This chapter contains a basic introduction to mobile Game-Based Learning (mGBL) and evaluation of major aspects of them. Also, it provides the background knowledge necessary to fully understand the design of the system in chapter 3 of this thesis.

2.1 Mobile Game-Based Learning

Mobile Game-Based Learning (mGBL) also known as serious game is a type of game over mobile devices that have two main goals (i) to be entertaining, and (ii) to be educational. Thus mGBL is designed to be attractive and to meet specific educational goals as well [10]. So, mGBL must consider both aspects of fun and/or enjoyment and educational impact. Therefore, there is also concern about how to evaluate the game outcomes to identify which mGBL is suited for a given goal and, how to develop more effective mGBL. In addition, the evaluation of players' performance is important because mGBL are designed to achieve knowledge and/or skill development. Thus, there are two major concerns of mGBL: (i) evaluation of mGBL, and (ii) evaluation of players' performance.

2.1.1 Evaluation of mGBL

There are plenty of evaluation methods that can be executed on mGBL. But, most of them require sophisticated laboratory equipment to record users interactions, expensive field experiments, or hard-to-process results from widespread interviews and they are time consuming-methods. So that, selecting a usability evaluation

method in order to expose potential usability problems and to be cost-effective is a challenge.

Among the usability inspection methods, heuristic evaluation is the most common and most popular method which meets most of these concerns [13, 14]. In a heuristic evaluation of mGBL, a small set of experts observe the game and evaluate its interface and contents against a list of recognized usability principles—the heuristics. Therefore, there is a need to select a set of efficient heuristics that consider the specific needs of pre-school children as well as the requirements of educational materials to evaluate usability of mGBL. So that we chose Playability Heuristics Evaluation for Educational Computer Game (PHEG) [15] which provides a set of heuristics to evaluate interface in mGBL as well as its educational elements.

According to [15], PHEG includes 43 heuristic attributes in 5 major categories: Interface, Educational, Content, Playability and Multimedia. Each category contains a number of sub-heuristics which could be found in appendix A at the end of this thesis. During an evaluation, experts are asked to rate each sub-heuristic for a game. With the combination of the heuristic weights (table 1) and equation (2.1), degree or level of usability of a game is calculated.

Table 1. Percentage of heuristic weight according to category [15]

Category	Total Heuristics	Heuristic Weight (%)
Interface (I)	10	23
Educational (E)	10	23
Content (C)	8	19
Playability (P)	7	16
Multimedia (M)	8	19
Total Heuristics	43	100

According to [15], the following formula was proposed to evaluate usability of mGBL.

$$USABPHEG = \frac{I}{0.23} + \frac{E}{0.23} + \frac{C}{0.19} + \frac{P}{0.16} + \frac{M}{0.19} \quad (2.1)$$

USABPHEG gives an estimation of the degree or level of usability, which is the value of overall usability of the mGBL. Where *I* represent score of Interface heuristic, *E* is for score of Educational heuristic, *C* for score of Content heuristic, *P* for score of Playability heuristic and *M* for score of Multimedia heuristic. When applying equation (2.2) [17], each of these variables (I, E, C, P and M) obtains the corresponding value:

$$f(x) = \left(\frac{\sum H}{\sum Ht} \right) \times P \quad (2.2)$$

Where, $\sum H$ is the summation of the severity scores (zero to four) for each category sub-heuristic, *P* is the percentage for the current category, and $\sum Ht$ represents the summation of each category sub-heuristic in the worst case (in the event that all

severity ratings were four). We used these heuristics beside these formulas in the evaluation part of the system which the design phase will be described in chapter 3.

2.1.2 Evaluation of Player Performance

In addition to evaluating the mGBL in order to develop more effective mGBL games, evaluation of players' performance is important because mGBLs are designed to achieve knowledge and/or skill development. To achieve this purpose, we included players' performance evaluation methods of [18] to our system. To analyze the child's knowledge and/or skill development, statistical reports are generated from the children activities in the game. User task ranking/rating, average task completion time, learning time, number of errors and completion/non-completion of a task are the factors considered to generate the knowledge and/or skill development reports which the design phase will be described in chapter 3.

2.2 Model-View-Controller (MVC)

In our architectural design, Model-View-Controller (MVC) architecture is used. This architecture is currently considered an architectural pattern used in software engineering. Model-View-Controller pattern provides the separation of application logic layer from User Interface (UI) logic and data-access logic and it provides independent development, testing and maintenance of these elements [19]. The MVC separation also simplifies group development. Different developers can work on the view, the controller logic, and the business logic in parallel.

The control flow of the MVC Architecture (figure 1) is as follows:

1. A user interacts with the "View" (User Interface) by clicking on a link or submitting a form.

2. The "Controller" handles the user input, and converts the request into an appropriate user action, understandable for the "Model".
3. The "Model" receives the information and updates its state (adds data to a database, for example, or calculates today's date).
4. The "View" checks the state of the "Model" and responds accordingly (listing the newly entered data, maybe).
5. The "View" waits for another interaction from the user, which restarts the control flow cycle.

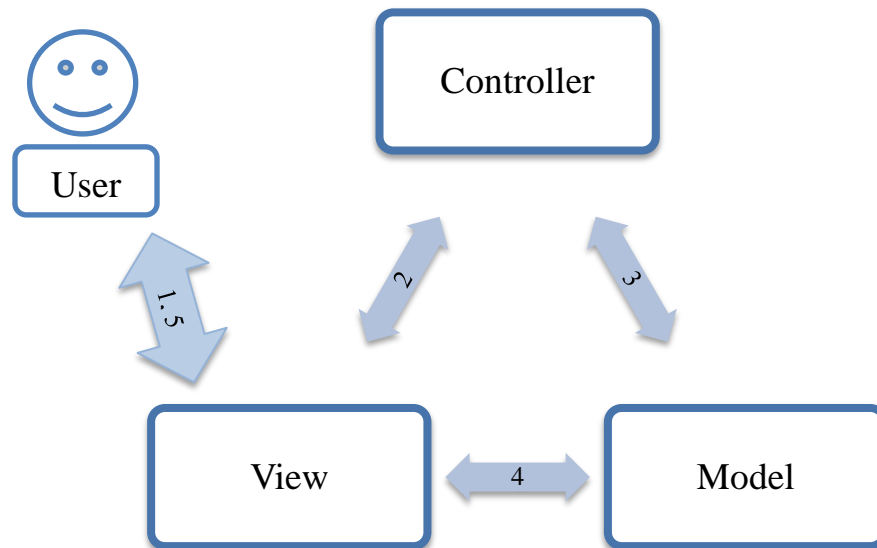


Figure 1. MVC architecture

2.3 Yii Framework

Yii framework [20] is a fully MVC architecture framework written in PHP [21] which enables organized and simple application development of Web applications. Our system is written under Yii framework. Figure 2 shows the general work flow of Yii framework based Web application.

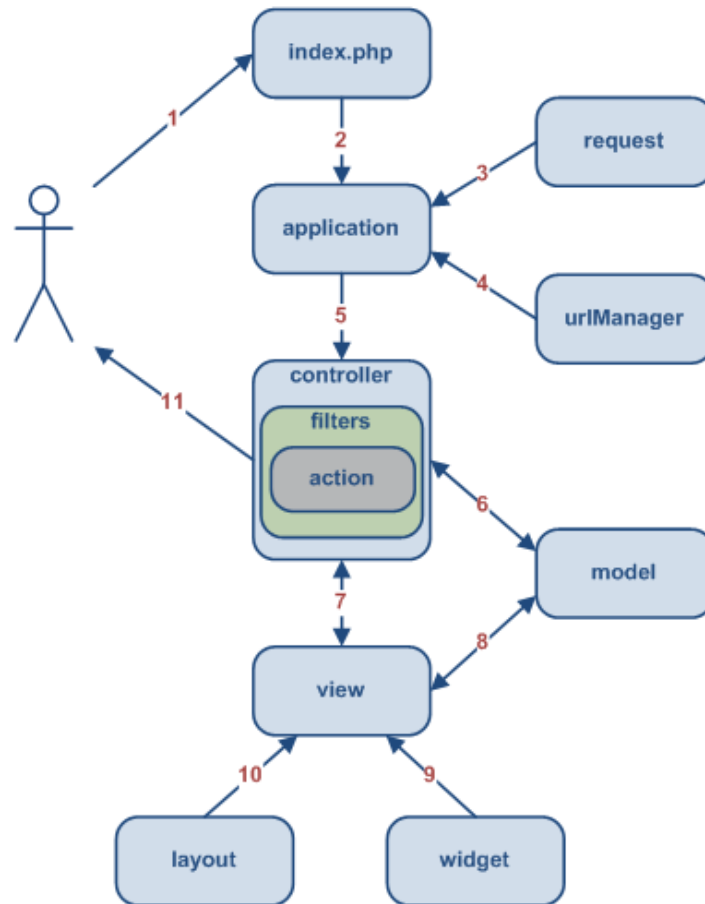


Figure 2. General work flow of a Yii framework based Web application [22]

The following description is extracted from [22] which explain the general work flow of a Yii framework based Web application. First, a user makes a request to the URL of Web application and the Web server handles the request by executing the bootstrap script "index.php" (Process 1). Then, the "index.php" creates an application instance and runs it (Process 2). After that, Web application obtains detailed user request information from an application component named "request" (Process 3) and the application determines the requested controller and action with the help of an application component named "urlManager" (Process 4). Then, the application creates an instance of the requested controller and executes the requested action and filter associated with the action (Process 5). The action then reads database records

(Process 6) and renders a view (Process 7). Next, the view reads and displays the attributes of the model (Process 8) and puts them in an empty layout with embedded widgets (Process 9, 10). Finally, the action completes the view rendering and displays the result to the user (Process 11).

2.4 Representational State Transfer (REST) Web Service

Representational State Transfer (REST) relies on a stateless, client-server, cacheable communications protocol (HTTP protocol) [23]. It is an architecture style for designing network based applications. RESTful applications use HTTP methods (GET, POST, PUT or DELETE) in order to create data, update data, read data or delete data. In summary REST implements all of 4 CRUD operations (Create, Read, Update and Delete) through HTTP protocol to reduce complexity and resource need. Web service module of our system is based on REST structure.

2.5 Unity Game Engine

Unity is a cross-platform game engine developed by Unity Technologies. Unity games can be run on various platforms including Windows, Windows mobile, iOS, Linux, Wii-U and Android [24]. The mentioned operating systems are also used by tablets and smartphones. Therefore, we choose this game engine to develop our games for supporting wide range of devices.

Chapter 3

DESIGN OF THE SYSTEM

This chapter deals with design, and architecture of the system. The initial section of this chapter states the general architecture of the system which will give an overall overview of the system architecture. The following section describes the architecture of the Online Game-Based Learning (OGBL) system and its modules which is the core of the system. Afterwards, the Game module structure will be described in the next section. In addition, database section describes the tables of the database in details.

3.1 General Architecture of the System

This system was developed as a modular Web based application (OGBL system) and cross-platform mGBL games (Game module). The general architecture of the system is as shown in figure 3.

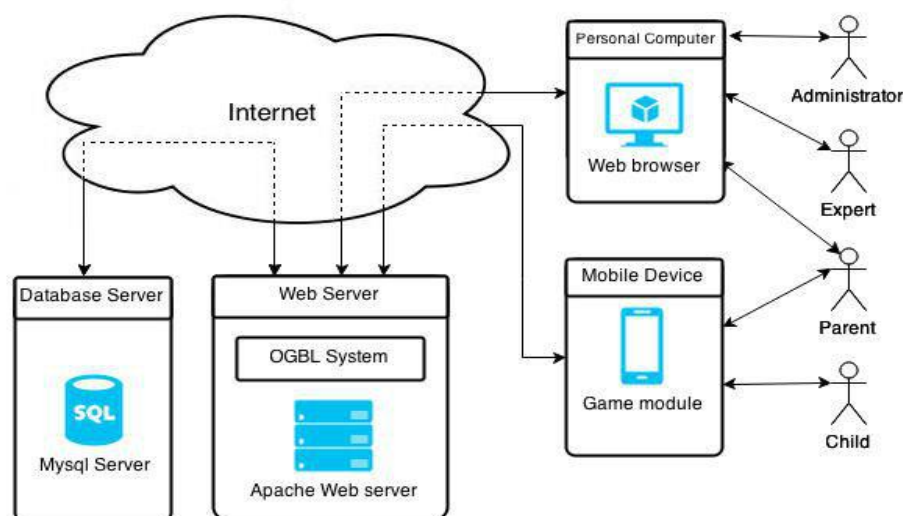


Figure 3. General architecture of the system

The system is divided into 4 components, which are shown in figure 3. The open source Web server, Apache, and the open source database server, MySQL [25], were used for the Web Server and Database Server components. In addition, OGBL system is located on the Web server which is the core of our system. OGBL system was written in PHP language cooperate with Yii framework which runs on the Web server component and stores/retrieves data on/from the Database Server component.

The users (Administrator, Expert, and Parent) can connect to OGBL system by accessing the Web server component's address by using a Web browser from their personal computer (Personal Computer component).

Tablet and smartphone type mobile device is the last component of the system which the game module is located on that. It is a mobile device which runs the Game module over one of these operating systems, iOS, Android, Windows Mobile, Wii U, Windows or Linux. Game module is a collection of mGBL games which is connected to the database server through the OGBL system. Parent or child can use this module to update the games' settings or play the games. All of the activities during the use of game module are reported to the Web server component through the Internet instantly and OGBL system will store them to database server. Architecture of this module is described under subsection 3.3.

3.2 Architecture of OGBL System

OGBL system is separated into two modules which are the Web application module and the RESTful Web service module. Architecture of the OGBL system is shown in figure 4.

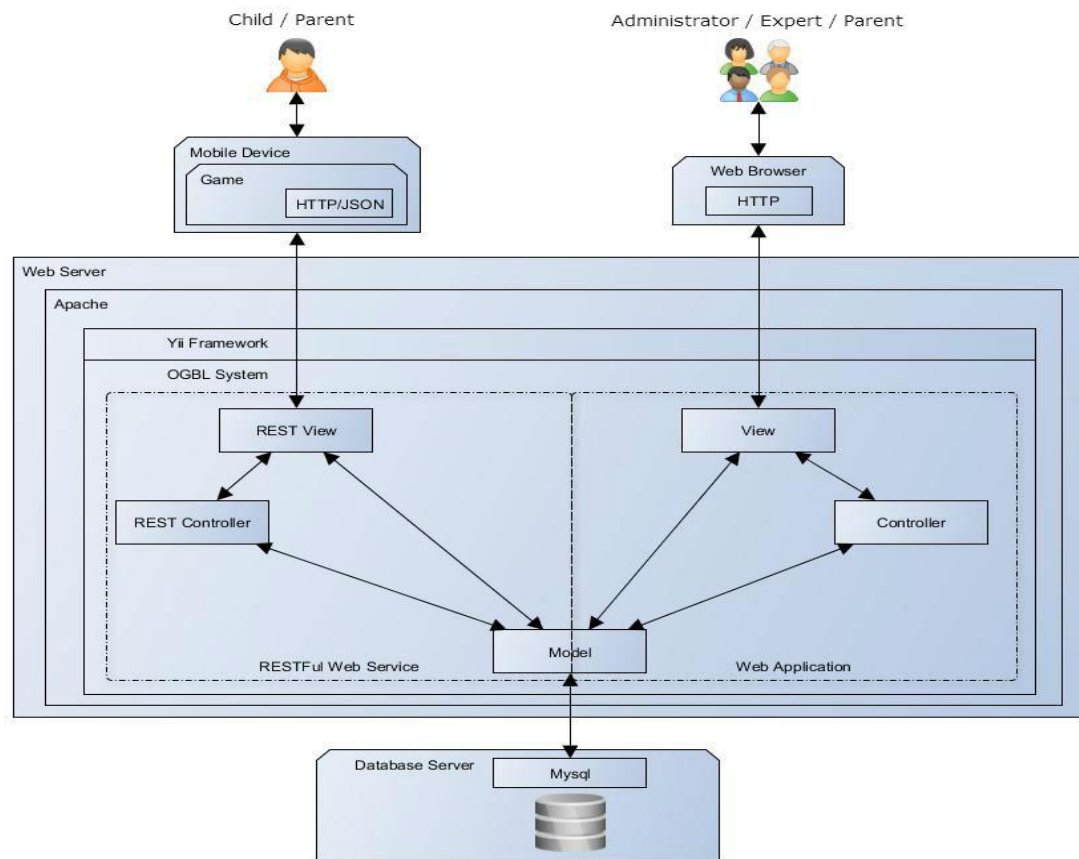


Figure 4. General architecture of OGBL system

The Web application module and the RESTful Web service module are Yii framework based Web applications.

The Web application module is the control panel of the system and the desired operations of the administrator, expert or parent are executed through its Web user interface. This module is described under subsection 3.2.2.

The next module is the RESTful Web service module which doesn't have any user interface and its duty is to provide communication protocol between the game module and the database server. Architecture of this module is described under subsection 3.2.3.

Work flow of the system is as follows. First of all, parents must register and add their children to the system by routing a Web browser to the URL of the Web application module. After the parent registered to the system through the sign up procedure, they have to download and install the game module into a mobile device component. After running the game on the mobile device, the parents must log in to their account by their credentials for the first time run. These communications between the game module and the system are established by calling the RESTful Web service module through the Internet. After a successful log in to the system, parents must choose between their registered children. Then, a session will be opened for the chosen child and the game will be started. All the activities and scores during the play will be stored in to the database server by calling the RESTful Web service module. From here onwards, the parents can manage and track their children activity by browsing to the Web application module and logging in to the system. Also, experts and admins by accessing to the URL of the Web application module through Web browser can log in to the system and manage and execute their duties.

Before we describe the details of these modules, we will define user types and their duties in the next subsection and in the other subsections we will describe the architecture of the modules and database of the OGBL system.

3.2.1 OGBL System User Types

We defined 4 types of user accounts in OGBL system. These user types and their defined duties are mentioned below:

- Administrator:

Administrator tasks are using system statistics and game's evaluation results that came from expert's heuristic evaluations to improve the system usability

in future. Also administrator can manage (insert, delete, or edit) user's (administrator, expert, parent, and child) accounts, games, lessons, and tips.

- Expert:

Expert user is responsible for providing assistance to the parents in order to develop children's knowledge and/or skills through the observation of the children activities and giving feedback to them. Experts are also responsible to evaluate the games through a heuristic evaluation method for developers.

- Parent:

Parent user role is client of our system. This user role can track the children skill development during playing of the games. Timeline and interactive statistics charts, in addition to useful hints, are designed to help parents to track children activity and observe children interests in each game category. Furthermore, parents can request for analyses of their children learning development by asking experts and can set the settings of games for their children. In addition, management (insert, delete, and edit) of child account is another utility of this user role.

- Child:

This user role doesn't have any responsibilities apart from using the system by playing mGBL game.

3.2.2 Architecture of the Web Application Module

This module is the control panel of the OGBL system. Users (Administrators, Experts or Parents) use this module through their Web browsers in order to connect to control panel of the system.

Because of using the MVC structure for the implementation of this module, we need to describe the "Controller", the "View" and the "Model" to explain architecture of this module clearly. Also, to be capable of understanding the architecture of this module, the next subsections illustrate these components beside of interactions and user rights.

3.2.2.1 Controllers

This is the backbone of the Web application module. The "Controller" takes care of collecting requests from the client through a Web browser, and decides what business logic to execute depending on the request. Afterwards, it redirects each request to a specific action class based on the URI of the incoming request. In the action class, the "Model" of the application is gathered or modified, then "Controller" decides what "View" component must be presented. List of all the controllers and their defined actions is shown in figure 5 and 6. Descriptions of the actions of each controller are also given in tables from table B.1 to B.16 of appendix B.

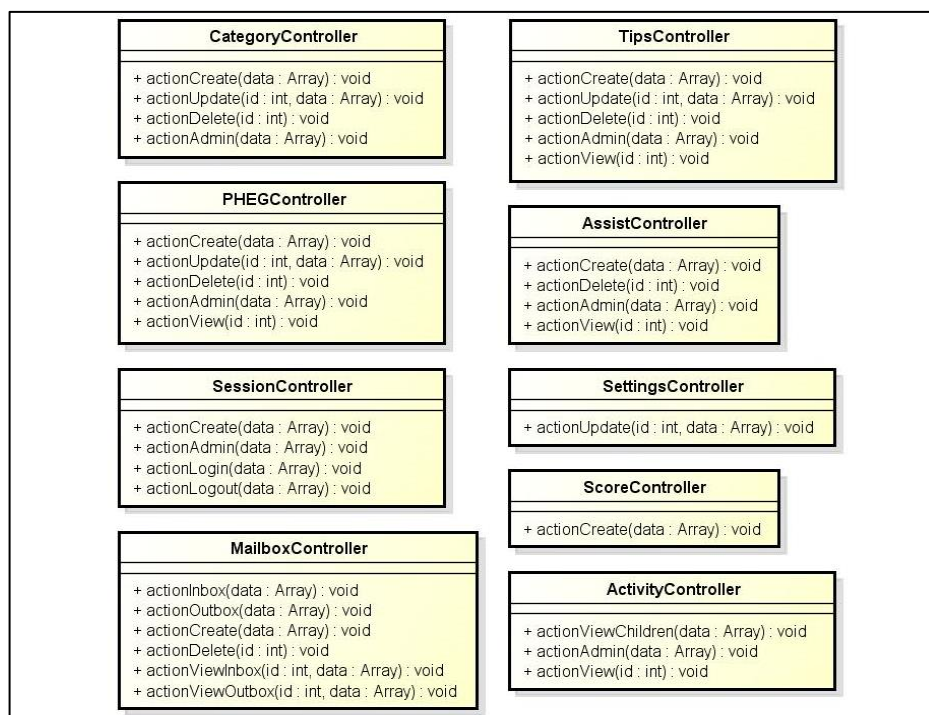


Figure 5. List of controller's classes and their actions

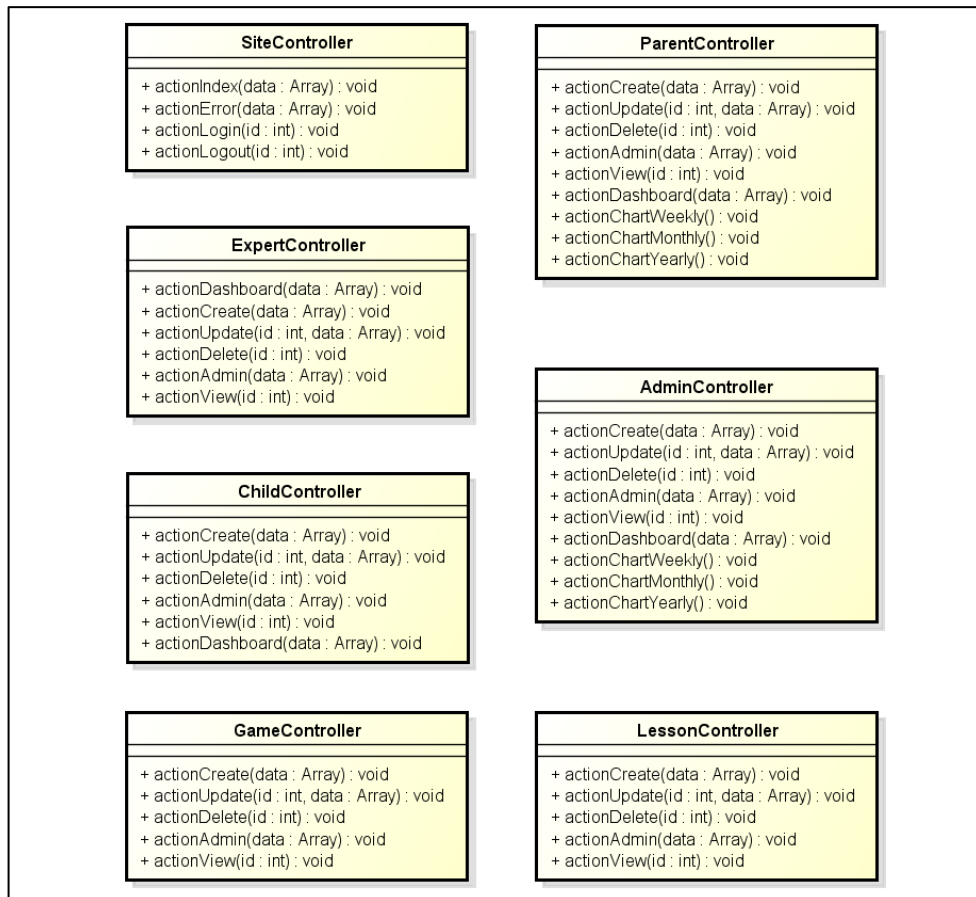


Figure 6. List of controller's classes and their actions (Cont.)

3.2.2.2 Views

Each "View" component in the Web application module can contain any combination of static HTML and dynamic content that changes depending on the interpretation of special actions. These actions are controller actions which are called through AJAX technique.

"View" component carries out two main roles in the Web application module. The first one is to work as a presentation layer for the "Controller" component. The second one is to collect information from the user in order to execute a controller action. List of views of each controller are shown in figures from figure C.1 to C.14 of appendix C.

3.2.2.3 Interactions

In this subsection interactions of the system will be given. In order to describe these interactions, after an introduction to the interaction, dataflow diagram in addition to sequence diagram of the interaction will be given as follows. Note that, the dataflow diagrams are marked by "pkg" and sequence diagrams are marked by "sd" in figures from figure D.1 to D.58 of appendix D.

Login

Users (administrator, expert or parent) can log in to the system through the login form. The dataflow and sequence diagrams of this interaction are shown in figure D.1 of appendix D.

Logout

Through this interaction, users (administrator, expert or parent) can log out from the system. The dataflow and sequence diagrams of this interaction are shown in figure D.2 of appendix D.

Parents registration

Parents can register in the system through this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.3 of appendix D.

Manage administrators

Administrators use this interaction to access the management page of the administrator user accounts. The dataflow and sequence diagrams of this interaction are shown in figure D.4 of appendix D.

Create administrator

In order to add a new administrator to the system, existing administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.5 of appendix D.

View administrator

Administrators use this interaction to view an administrator user account details. The dataflow and sequence diagrams of this interaction are shown in figure D.6 of appendix D.

Update administrator

Administrators can update an administrator user account through this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.7 of appendix D.

Delete administrator

In order to delete an administrator user account from the system, administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.8 of appendix D.

Manage experts

Administrators use this interaction to access the management page of the expert user accounts. The dataflow and sequence diagrams of this interaction are shown in figure D.9 of appendix D.

Create expert

In order to add a new expert to the system, existing administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.10 of appendix D.

View expert

Administrators use this interaction to view an expert user account details. The dataflow and sequence diagrams of this interaction are shown in figure D.11 of appendix D.

Update expert

Administrators can update an expert user account through this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.12 of appendix D.

Delete expert

In order to delete an expert user account from the system, administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.13 of appendix D.

Manage parents

Administrators use this interaction to access the management page of the parent user accounts. The dataflow and sequence diagrams of this interaction are shown in figure D.14 of appendix D.

Create parent

In order to add a new parent to the system, existing administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.15 of appendix D.

View parent

Administrators use this interaction to view a parent user account details. The dataflow and sequence diagrams of this interaction are shown figure D.16 of appendix D.

Update parent

Administrators can update an administrator user account through this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.17 of appendix D.

Delete parent

In order to delete a parent user account from the system, administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.18 of appendix D.

Manage children

Parents use this interaction to access the management page of the children accounts. The dataflow and sequence diagrams of this interaction are shown in figure D.19 of appendix D.

Create child

In order to add a new child to the system, parents use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.20 of appendix D.

View child

Parents use this interaction in order to view a child activities and timeline. The dataflow and sequence diagrams of this interaction are shown in figure D.21 of appendix D.

Update child

Administrators can update a child user account through this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.22 of appendix D.

Delete child

In order to delete a child account from the system, parents use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.23 of appendix D.

Manage games

Administrators use this interaction to access the management page of the games. The dataflow and sequence diagrams of this interaction are shown in figure D.24 of appendix D.

Create game

In order to add a game to the system, existing administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.25 of appendix D.

View game

Administrators use this interaction to view a game detail. The dataflow and sequence diagrams of this interaction are shown in figure D.26 of appendix D.

Update game

Administrators can update a game through this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.27 of appendix D.

Delete game

In order to delete a game from the system, administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.28 of appendix D.

Manage lesson

Administrators use this interaction to access the management page of the lessons. The dataflow and sequence diagrams of this interaction are shown in figure D.29 of appendix D.

Create lesson

In order to add a lesson to the system, existing administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.30 of appendix D.

View lesson

Administrators in order to view a lesson details use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.31 of appendix D.

Update lesson

Administrators can update a lesson through this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.32 of appendix D.

Delete lesson

In order to delete a lesson from the system, administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.33 of appendix D.

Manage categories

Administrators use this interaction to access the management page of the categories. The dataflow and sequence diagrams of this interaction are shown in figure D.34 of appendix D.

Create category

In order to add a new game category to the system, existing administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.35 of appendix D.

Update category

In order to update game categories, administrators through this interaction use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.36 of appendix D.

Delete category

In order to delete a game category from the system, administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.37 of appendix D.

Manage tips

Administrators use this interaction to access the management page of the tips. The dataflow and sequence diagrams of this interaction are shown in figure D.38 of appendix D.

Create tip

In order to assigning tips to the games, existing administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.39 of appendix D.

View tip

Administrators in order to view the details of a tip use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.40 of appendix D.

Update tip

Administrators can update tips through this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.41 of appendix D.

Delete tip

In order to delete a tip from the system, administrators use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.42 of appendix D.

Manage PHEGs

Experts or administrators use this interaction to access the management page of the PHEG (heuristic evaluation of the games). The dataflow and sequence diagrams of this interaction are shown in figure D.43 of appendix D.

Create PHEG

In order to evaluate a game, existing experts use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.44 of appendix D.

View PHEG

Administrators and experts use this interaction to view an evaluation of a game. The dataflow and sequence diagrams of this interaction are shown in figure D.45 of appendix D.

Update PHEG

Experts can update a game evaluation through this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.46 of appendix D.

Delete PHEG

In order to delete an evaluation from the system, administrators and experts use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.47 of appendix D.

Manage activities

Experts use this interaction to access the management page of the children activities. The dataflow and sequence diagrams of this interaction are shown in figure D.48 of appendix D.

View activity

Experts in order to view the activity of a child use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.49 of appendix D.

Manage assists

Experts use this interaction to access the management page of the assist requests. The dataflow and sequence diagrams of this interaction are shown in figure D.50 of appendix D.

Create assist

Parents use this interaction to send an assist request to the experts. The dataflow and sequence diagrams of this interaction are shown in figure D.51 of appendix D.

View assist

Experts in order to view an assist request from parents use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.52 of appendix D.

Delete assist

In order to delete an assist request from the system, experts use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.53 of appendix D.

Update setting

Parents use this interaction in order to update the games settings. The dataflow and sequence diagrams of this interaction are shown in figure D.54 of appendix D.

Manage Inbox/Outbox

Parents, experts or administrators of the system use this interaction to access their mailbox. The dataflow and sequence diagrams of this interaction are shown in figure D.55 of appendix D.

Compose Mail

In order to compose an e-mail, users (administrators, experts or parents) use this interaction to send e-mails to other users. The dataflow and sequence diagrams of this interaction are shown in figure D.56 of appendix D.

View E-mail

Users (administrators, experts, parents) use this interaction to view an e-mail. The dataflow and sequence diagrams of this interaction are shown in figure 57.

Delete E-mail

In order to delete an e-mail from the mailbox, users (administrators, experts or parents) use this interaction. The dataflow and sequence diagrams of this interaction are shown in figure D.58 of appendix D.

3.2.2.4 Models

The "Model" component used to keep data and their relevant business rules. It represents the database objects of the application which will be explained in the database section. Both the Web application controller and REST controller can call the "Model" component methods in order to get data in the form of the Data Access Objects (DAO). The "Model" component populates the Data Access Object, and passes it back to the Web application controller or REST controller classes.

In order to get required data in the form of DAO, a number of methods were implemented. The class diagram of the "Model" component is shown in figure 7 in

order to see the relation between the models and their methods. Each method is described in tables from table E1 to E16 of appendix E.

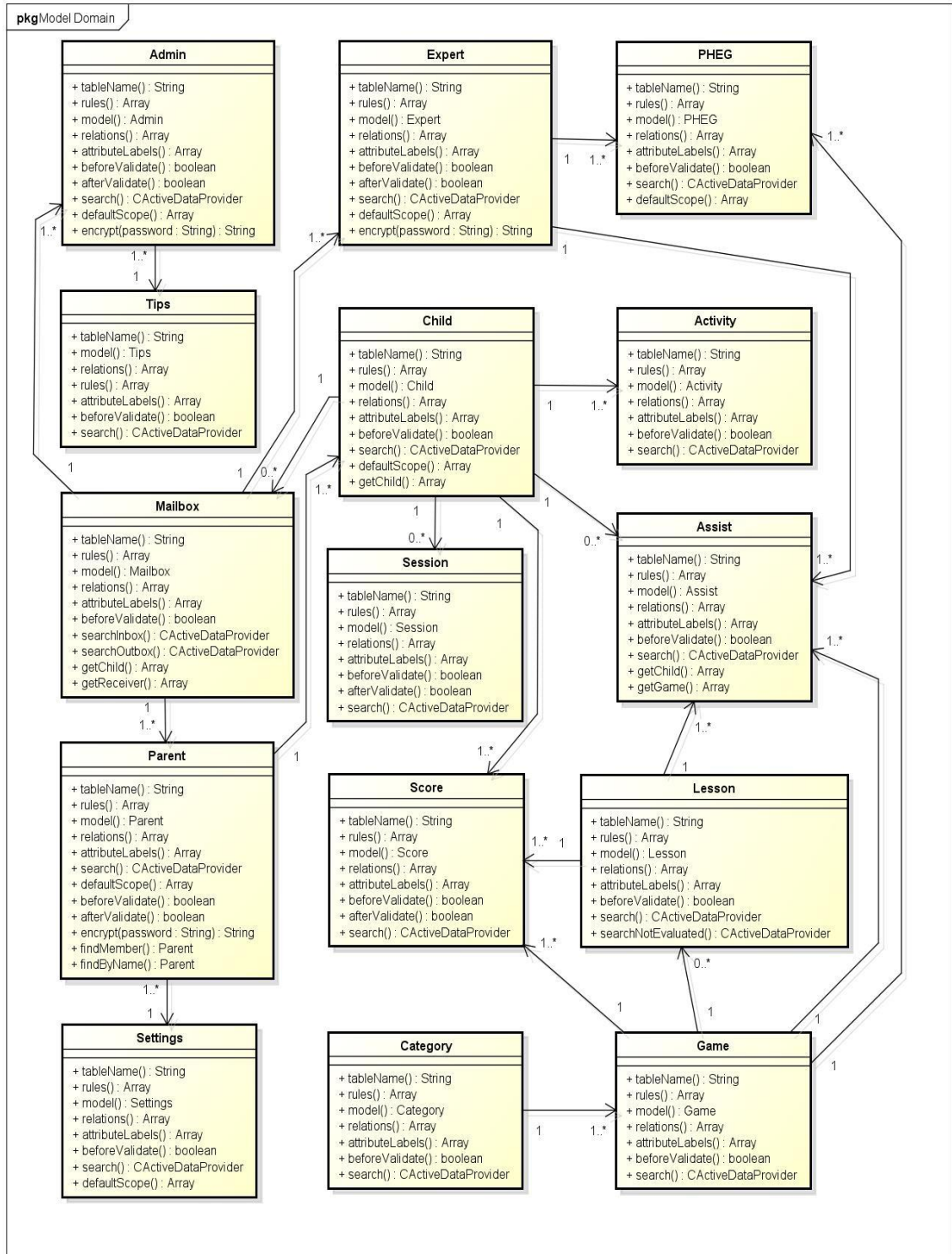


Figure 7. The "Model" component class diagram

3.2.2.5 Users' Permissions

Users' authorization checks in Yii framework based applications are performed by enabling CaccessControlFilter and setting the security rules. The user will be able to access to an action when s/he is not denied by one of the security rules or allowed by a rule explicitly in the CaccessControlFilter class. Table F.1 of appendix F shows the users' permissions which can be used to set these security rules. Each row of the table gives an action beside of the users permissions. User types with 'X' symbol have access to the corresponding action and empty box shows the user does not have access to it.

3.2.3 Architecture of the RESTful Web Service Module

We have used RESTFullyii extension [26] which provides Yii framework based application compatibility of act as RESTful Web service. RESTFullyii provides full HTTP verb support (GET, PUT, POST, and DELETE) (table 2) for accessing to models, as well as the ability to offset, limit, sort and filter the models. The list of http verbs and their operations are as follows.

Table 2. RESTFul Web services operations

Operation	HTTP verb
Retrieving of existing records	GET
Creating new records	POST
Removing records	DELETE
Updating existing records	PUT

In the following subsections, authentication and the list of the RESTful Web service module actions will be described.

3.2.3.1 Authentication

Authentications in the RESTful Web service module is provided by setting X_REST_USERNAME equal to username and X_REST_PASSWORD equal to password as HTTP headers and send it with JSON request to the RESTful Web service module URL. An Example of authentication request is given in figure 8 in order to better understanding. cURL is a command-line tool for transferring data using various protocols. It can be used to test RESTful Web service.

```
Curl -i -H "Accept: application/json" -H "X_REST_USERNAME: admin" -H "X_REST_PASSWORD: admin" http://OGBLSystem/api/child
```

Figure 8. Example of cURL authentication request

3.2.3.2 The RESTful Web Service Module Actions

In this section, the list of operations that we need to perform during the playing of the game and details of their Web service calling requests will be described.

Third-party game developers can also use these operations to integrate their games to our system. The list of the RESTful Web service module actions and structure of requests and responses are given in tables form table G.1 to G.10 of appendix G.

3.3 Architecture of the Game Module

The game module includes set of mGBL games which are available to play by pre-school children. We have developed two sample games to demonstrate our system. These games are developed over Unity game engine [27] which is one of the widely used cross-platform game-engines.

To run these games, you just have to modify the URL in the source code of the games and compile them by unity.

The names of these games are "Choose the correct color" and "Choose the correct animal" with 5 levels in each one. Dataflow diagram in addition to the sequence diagram of our sample games are given in figure H.1 of appendix H.

3.4 Database

The database of our system is described in this section. All of the database tables with their relationships are shown as an Entity Relationship (ER) diagram in figure 9. In the following, structure of each table and its relations to other tables will be described.

"Admin" table

"Admin" table stores the administrator users data. The 'id' field is the primary key of table and shows the record number. 'username' field stores usernames and 'password' field stores MD5 encrypted passwords of the administrator users. 'e-mail' field is used in order to store e-mail address of the administrators. 'createdate' is also used in order to store the exact time and date when administrator user is created. The field 'mailbox_id' of "admin" table is the foreign key for 'Id' field of "mailbox" table. The relationship between these fields is one-to-one.

"Expert" table

"Expert" table stores the expert users data. The 'id' field is the primary key of table and shows the record number. 'username' field stores usernames and 'password' field stores MD5 encrypted passwords of the expert users. 'e-mail' field is used in order to store e-mail address of the experts. 'createdate' is also used in order to store the exact time and date when expert user is created. The field 'mailbox_id' of "expert" table is

the foreign key for 'Id' field of "mailbox" table. The relationship between these fields is one-to-one.

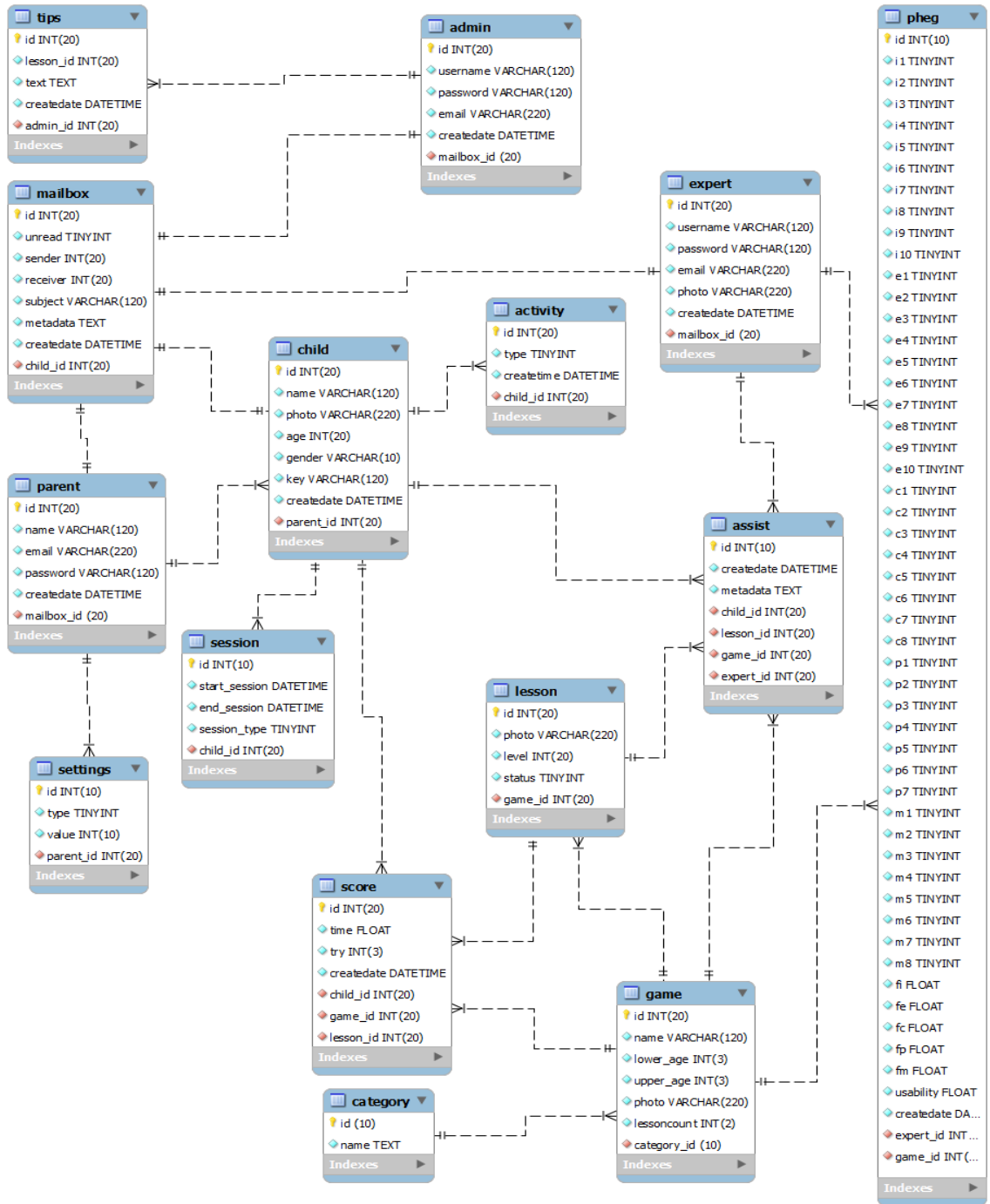


Figure 9. Database entity relationship diagram

"Parent" table

"Parent" table stores the parent users data. The 'id' field is the primary key of table and shows the record number. 'name' field stores names and 'password' field stores MD5 encrypted passwords of the parent users. 'e-mail' field is used in order to store e-mail address of the parents. 'createdate' is also used in order to store the exact time and date when parent user is created. The field 'mailbox_id' of "parent" table is the foreign key for 'Id' field of "mailbox" table. The relationship between these fields is one-to-one.

"Child" table

"Child" table stores the children data. The 'id' field is the primary key of table and shows the record number. 'name' field stores names and 'photo' field stores the location of uploaded photo of the children on the server as a varchar data type. 'age' field is used to store the age of the children. 'gender' field is used in order to store the gender of the children. The value of 0 shows the gender of children is male while the value of 1 show the gender of children is female. 'createdate' is also used in order to store the exact time and date when parent user is created. The field 'parent_id' of "child" table is the foreign key for 'Id' field of "parent" table. The relationship between these fields is many-to-one.

"Game" table

"Game" table stores the games data. The 'id' field is the primary key of table and shows the record number. 'name' field stores names and 'photo' field stores the location of uploaded photo of the game on the server as a varchar data type. 'lower_age' and 'upper_age' fields are integer values which are used to specify lower age and upper age of the game. The value of these fields shows the count of the months in order to specify the age of the allowed children. 'gender' field is used in

order to store the gender of the children. The value of 0 shows the gender of children is male while the value of 1 show the gender of children is female. 'lessoncount' is also used in order to store the number of the game's levels. The field 'category_id' of "game" table is the foreign key for 'Id' field of "category" table. The relationship between these fields is one-to-many.

"Lesson" table

"Lesson" table stores the lessons data. The 'id' field is the primary key of table and shows the record number. 'photo' field stores the location of uploaded photo of the lesson on the server as a varchar data type. 'level' field is integer value which is used to specify the level of the lesson. The field 'game_id' of "Lesson" table is the foreign key for 'Id' field of "game" table. The relationship between these fields is many-to-one.

"Category" table

"Category" table stores the category names of the games. The 'id' field is the primary key of table and shows the record number. 'name' field stores the category name as a varchar data type.

"Tips" table

"Tips" table stores the tips data. The 'id' field is the primary key of table and shows the record number. 'text' field stores the content of tips as a varchar data type. 'createdate' is also used in order to store the exact time and date when the tip is created by an admin user. The field 'lesson_id' of "tips" table is the foreign key for 'Id' field of "lesson" table. The relationship between these fields is many-to-one. The field 'admin_id' of "tips" table is the foreign key for 'Id' field of "admin" table. the relationship between these fields is one-to-one.

"Activity" table

"Activity" table stores the children activities data. The 'id' field is the primary key of table and shows the record number. 'type' field stores the type of activity as a TINYINT data type. The value of 0 shows the type of activity is logging in while the value of 1 shows the type of activity is logging off, and also the value of 2 shows the type of this activity is finishing a level and 3 is finishing a game. 'createdate' is also used in order to store the exact time and date when the activity is created. The field 'child_id' of "activity" table is the foreign key for 'Id' field of "child" table. The relationship between these fields is many-to-one.

"Assist" table

"Assist" table stores the assist request data. The 'id' field is the primary key of table and shows the record number. 'metadata' field is the body of message for assist request. 'createdate' is also used in order to store the exact time and date when the assist is requested. The field 'child_id' of "assist" table is the foreign key for 'Id' field of "child" table. The relationship between these fields is many-to-one. The field 'lesson_id' of "assist" table is the foreign key for 'Id' field of "lesson" table. The relationship between these fields is many-to-one. The field 'game_id' of "assist" table is the foreign key for 'Id' field of "game" table. The relationship between these fields is many-to-one. The field 'expert_id' of "assist" table is the foreign key for 'Id' field of "expert" table. The relationship between these fields is one-to-one.

"Session" table

"Session" table stores the sessions data. The 'id' field is the primary key of table and shows the record number. 'start_session' field stores the date and time of the beginning of the session and 'end_session' field stores the date and time of ending a

session. The field 'child_id' of "session" table is the foreign key for 'Id' field of "child" table. The relationship between these fields is many-to-one.

"Settings" table

"Settings" table stores the settings data. The 'id' field is the primary key of table and shows the record number. 'type' field stores the type of setting. The value of 1 show the type of setting is limiting play time per day while the value of 1 shows the type of setting is play new games by parent approval. The 'value' field stores the value of the setting which the value 1 shows that it is enabled and value 0 shows it is disabled. The field 'child_id' of "session" table is the foreign key for 'Id' field of "child" table. The relationship between these fields is many-to-one.

"Score" table

"Score" table stores the scores data. The 'id' field is the primary key of table and shows the record number. 'time' field stores the amount of time which is spent to finish a lesson (In milliseconds) and 'try' field stores the number of wrong answer until finishing the. The field 'child_id' of "session" table is the foreign key for 'Id' field of "child" table. The relationship between these fields is many-to-one. The field 'game_id' of "score" table is the foreign key for 'Id' field of "game" table. The relationship between these fields is many-to-one. The field 'lesson_id' of "score" table is the foreign key for 'Id' field of "lesson" table. The relationship between these fields is many-to-many.

"PHEG" table

"PHEG" table stores the results of PHEG evaluations. The 'id' field is the primary key of table and shows the record number. 'i1' to 'i10' fields store the interface related questions' answers. 'e1' to 'e10' fields store the educational related questions' answers. 'c1' to 'c8' fields store the content related questions' answers. 'p1' to 'p7'

fields store the playability related questions' answers. 'm1' to 'm8' fields store the multimedia related questions' answers. 'fi', 'fe', 'fc', 'fp' and 'fm' fields store the results of heuristic weight function. The 'usability' field is used in order to store the result of usability function. 'createdate' is also used in order to store the exact time and date of evaluation. The field 'expert_id' of "PHEG" table is the foreign key for 'Id' field of "expert" table. The relationship between these fields is many-to-one. The field 'game_id' of "PHEG" table is the foreign key for 'Id' field of "game" table. The relationship between these fields is many-to-one.

Chapter 4

IMPLEMENTATION

OGBL system is separated into three modules which are the Web application module, the RESTful Web service module and the Game module. The RESTful Web service module does not have any user interface and we refer the reader to the explanation of this module under subsection 3.2.3. In this chapter, we will describe the implementation phase of Web application and Game modules.

4.1 Web application module

This module is separated into three sub modules which are the administrator, expert and parent. The Web application module runs each sub module according to the user type of the logged-in user. The main page of Web application module is "Login" page (figure 10) through which the users can log in to the system by filling the "Username" and "Password" fields and clicking on the "Login" button. After clicking on the "Login" button, if the username or password was not correct, an error message will be shown as. "Username or password is incorrect"; otherwise, the user will be logged in to the system and according to the type of the logged-in user, one of the mentioned sub modules (administrator, expert or parent) will be executed. These modules are given in the following subsections in detail.

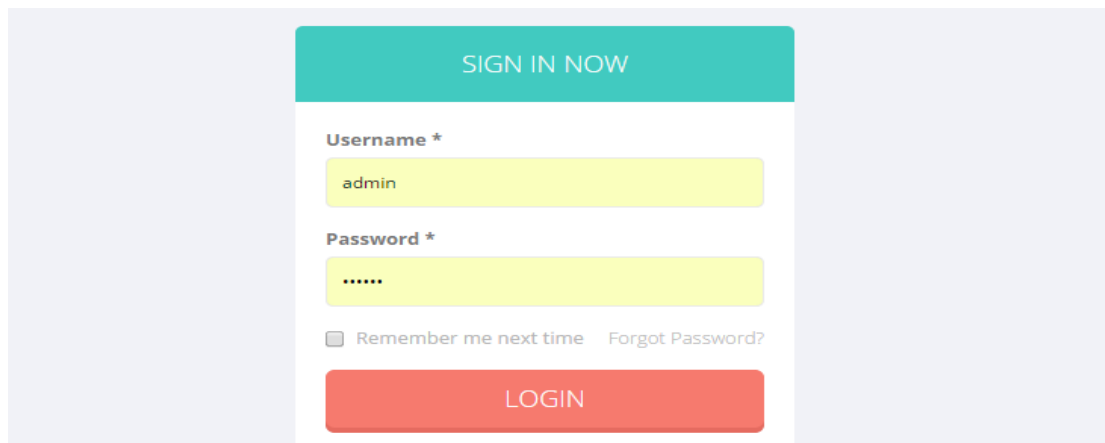


Figure 10. "Login" Page

4.1.1 Administrator sub module

When an administrator user logs in to the system, this sub module will be executed. The "Homepage" page of administrator sub module is shown in figure 11.

"Homepage" page of administrator sub module consists of a sidebar on left which is the navigation of this sub module and the statistical report of OGBL system on right. This report includes the total number of registered children, parent, game and lesson and daily, monthly, yearly charts of the number of registration of children and parent accounts. The administrator sub module navigation items are "Homepage", "Manage Parents", "Manage Experts", "Manage Admins", "Manage Games", "Manage Lessons", "Manage Tips", "View Evaluations", "Mailbox", "Profile Settings" and "Logout".

"Homepage" navigation item was described above. The other navigation items will be described in the following subsections.

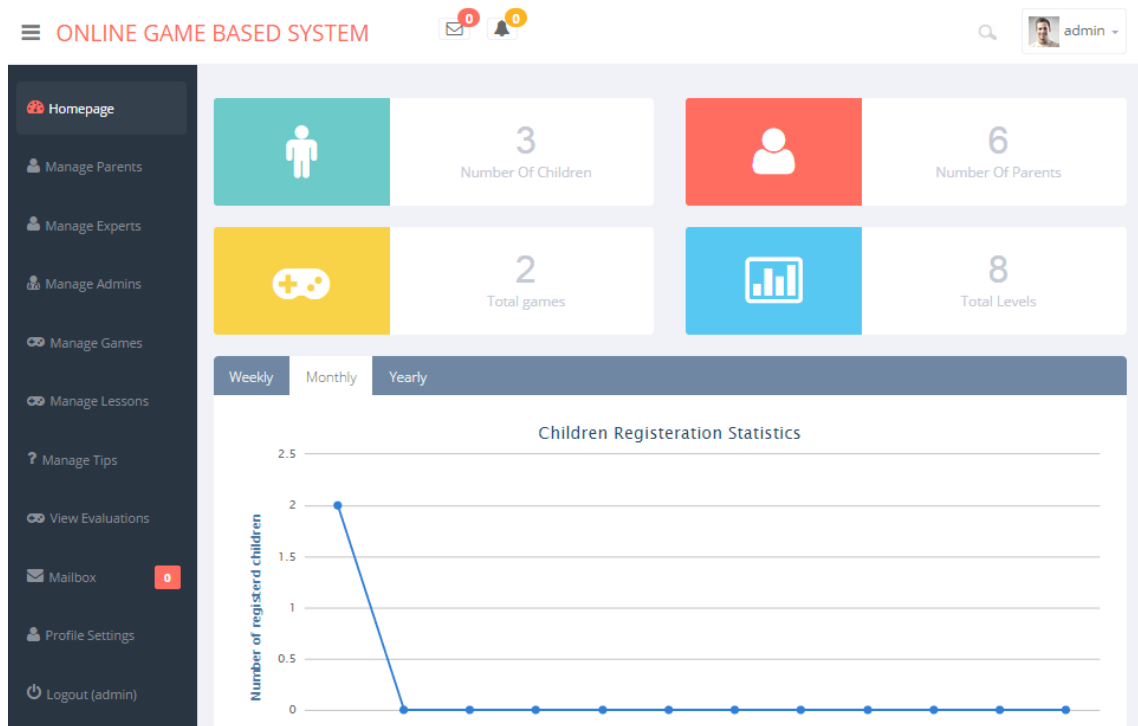


Figure 11. "Homepage" page

4.1.1.1 "Mange Parents" navigation item

"Mange Parents" navigation item redirects administrators to the "Manage Parents" page. This page contains the table of the registered parents in addition to operations section which are shown in figure 12.

The screenshot shows the 'Manage Parents' page in the 'ONLINE GAME BAsED SYSTEM'. The page header includes the breadcrumb 'Home » Parents » Manage'. The main content area displays a table of registered parents with the following data:

ID	Parent	Name	Email	Password	Createdate
1	10000	ali	mili@mili.com	202cb962ac59075b964b07152d234b70	2013-10-21 09:13:38
2	10001	Milad	m@m.com	4495414f08a5ddb58d03f8896ab73e4d	2013-11-09 12:54:23
3	10002	hasan	mm@m.com	c4ca4238a0b923820dcc509a6f75849b	2013-11-09 13:25:33
4	10003	kaveh	knot@me.com	c4ca4238a0b923820dcc509a6f75849b	2013-11-12 09:05:26
5	10004	milad ebadi	mil2k@yahoo.com	202cb962ac59075b964b07152d234b70	2013-11-21 13:21:18
6	10005	sina	s@s.com	c4ca4238a0b923820dcc509a6f75849b	2013-12-04 17:30:18

Below the table, there is an 'Operations' section with a 'Create Parent' button.

Figure 12. "Manage Parents" page

Each row of the table contains "ID" number, "Parent" unique id, "Name", "E-mail", encrypted "Password" and "Created Date" of each parent user in addition to the available actions. These actions are view (🔍), update (✎) and delete (✖) over a parent user. "Create Parent" is the only available operation in this page which is located at the operations section.

When the administrator clicks on view (🔍) action, s/he will be redirected to "View Parent" page as shown in figure 13. In this page, details of the selected parent user are given in a table. Also, operations section of this page contains "Update Parent" and "Delete Parent".

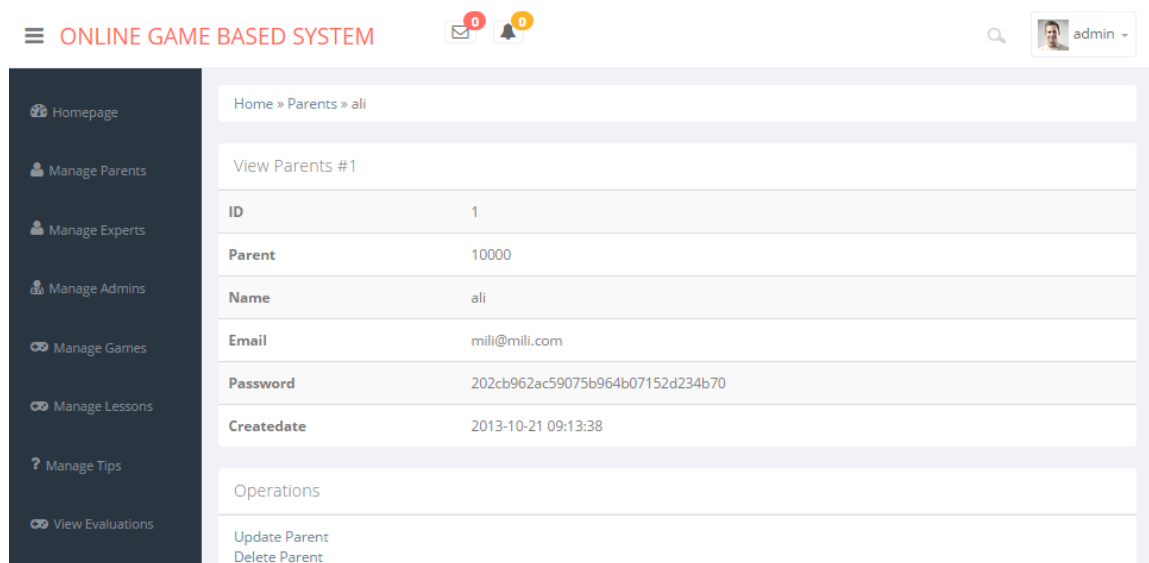


Figure 13. "View Parent" page

When the administrator clicks on "Update Parent" operation, s/he will be redirected to "Update Parent" page as shown in figure 14. It has a form section and operations section. The form section contains the current details of the selected parent and can be updated by changing the values of the fields and clicking on the "Save" button. After clicking on the "Save" button, if all the fields are filled, the administrator will

be redirected to updated "View Parent" page; otherwise, the system will show a message that says the administrator should fill all the fields and then submit the form again.

Also, the operations section includes "View Parent" which redirects the user to "View Parent" page and "Manage Parents" which will redirect the administrator to "Manage Parents" page.

The screenshot displays the "Update Parent" page within an "ONLINE GAME BASED SYSTEM". The page features a dark sidebar on the left with navigation items: Homepage, Manage Parents, Manage Experts, Manage Admins, Manage Games, Manage Lessons, Manage Tips, View Evaluations, Mailbox (with a red notification badge), and Profile Settings. The main content area has a breadcrumb trail "Home » Parents » ali » Update" and a title "Update Parents ali". Below the title, a note states "Fields with * are required." The form contains three input fields: "Name *" with the value "ali", "Email *" with the value "mili@mili.com", and "Password *" which is masked with dots. A blue "Save" button is positioned below the password field. At the bottom of the page, there is an "Operations" section with links for "View Parent" and "Manage Parents". The top right corner shows a search icon and a user profile for "admin".

Figure 14. "Update Parent" page

When the administrator clicks on "Delete Parent" operation in "View Parent" page, an alert will be appeared for the confirmation of deleting selected parent user. By clicking "Yes" on the confirmation alert, the parent user will be deleted and the page will be refreshed. This alert is shown in figure 15.

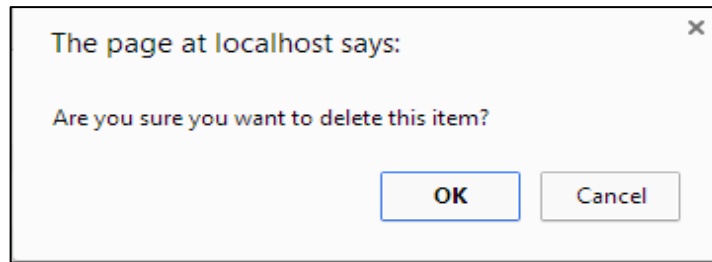


Figure 15. Confirmation alert for delete a record

Within the "Manage Parents" page, when the administrator clicks on update (🔄), s/he will be redirected to "Update Parent" page. Also if the administrator clicks on delete (✖) action, the delete will be done in the same manner as described above.

Finally, when the user clicks on "Create Parent" operation in the "Manage Parents" page, s/he will be redirected to "Create Parent" page as shown in figure 16. It is divided into a form section and operations section. By filling this form and clicking on "Create" button, if all the fields are filled, the new parent will be added to the system and the administrator will be redirected to "View Parent" page; otherwise, the system will show a message that says the administrator should fill all the fields and then submit the form again.

Also, the operations section includes "Manage Parents" which will redirect the administrator to "Manage Parents" page.

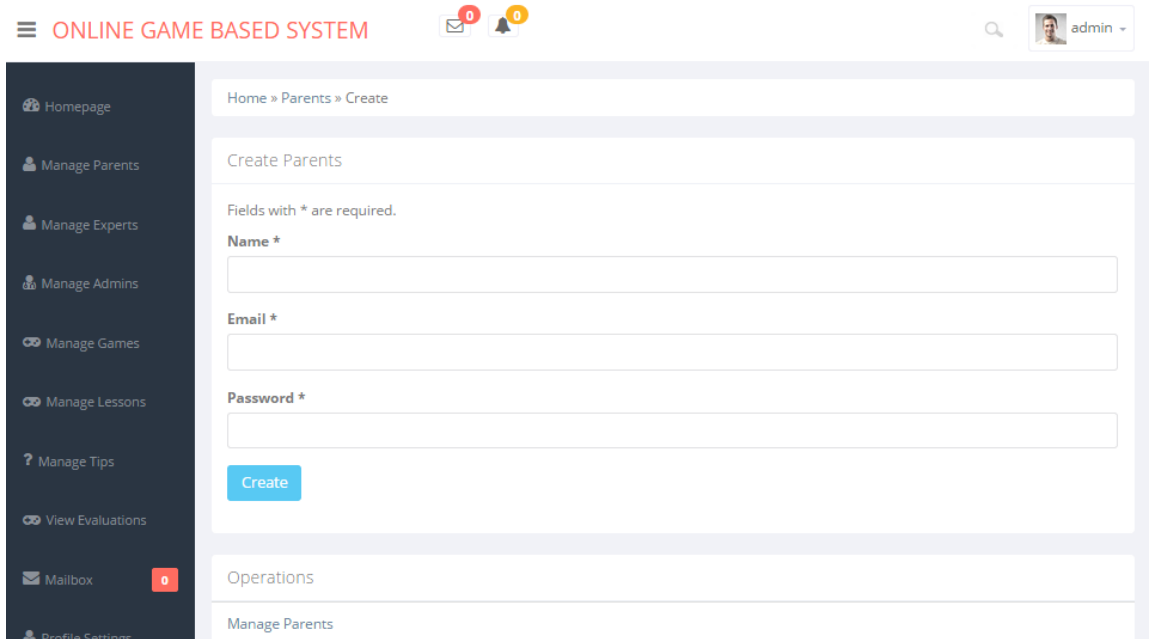


Figure 16. "Create Parent" page

4.1.1.2 "Manage experts" navigation item

"Manage experts" navigation item redirects administrators to "Manage Experts" page. This page contains the table of the registered experts in addition to operations section which are shown in figure 17.

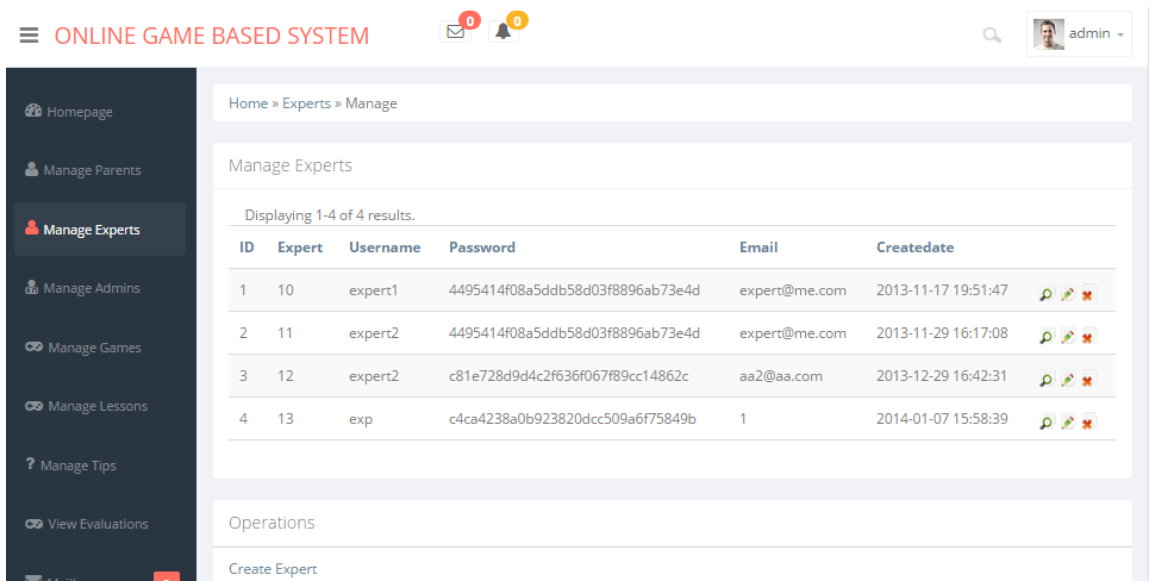


Figure 17. "Manage Experts" page

Each row of the table contains "ID" number, "Expert" unique id, "Username", encrypted "Password", "E-mail" and "Created Date" of each expert user in addition to the available actions. These actions are view (🔍), update (✏️) and delete (🗑️) over an expert user. "Create Expert" is the only available operation in this page which is located at the operations section.

When the administrator clicks on view (🔍) action, s/he will be redirected to "View Expert" page as shown in figure 18. In this page, details of the selected expert user are given in a table. Also, operations section of this page contains "Update Expert" and "Delete Expert".

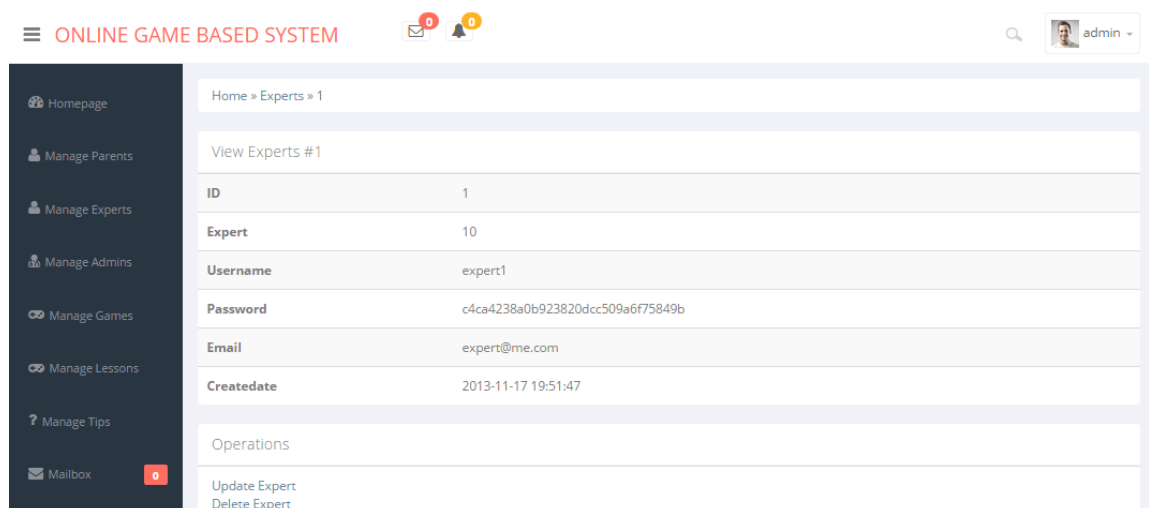


Figure 18. "View Expert" page

When the administrator clicks on "Update Expert" operation, s/he will be redirected to "Update Expert" page as shown in figure 19. It has a form section and operations section. The form section contains the current details of the selected expert user and can be updated by changing the values of the fields and clicking on the "Save" button. After clicking on the "Save" button, if all the fields are filled, the administrator will be redirected to updated "View Expert" page; otherwise, the

system will show a message that says the administrator should fill all the fields and then submit the form again.

Also, the operations section includes "View Expert" which redirects the user to "View Expert" page and "Manage Experts" which will redirect the administrator to "Manage Experts" page.

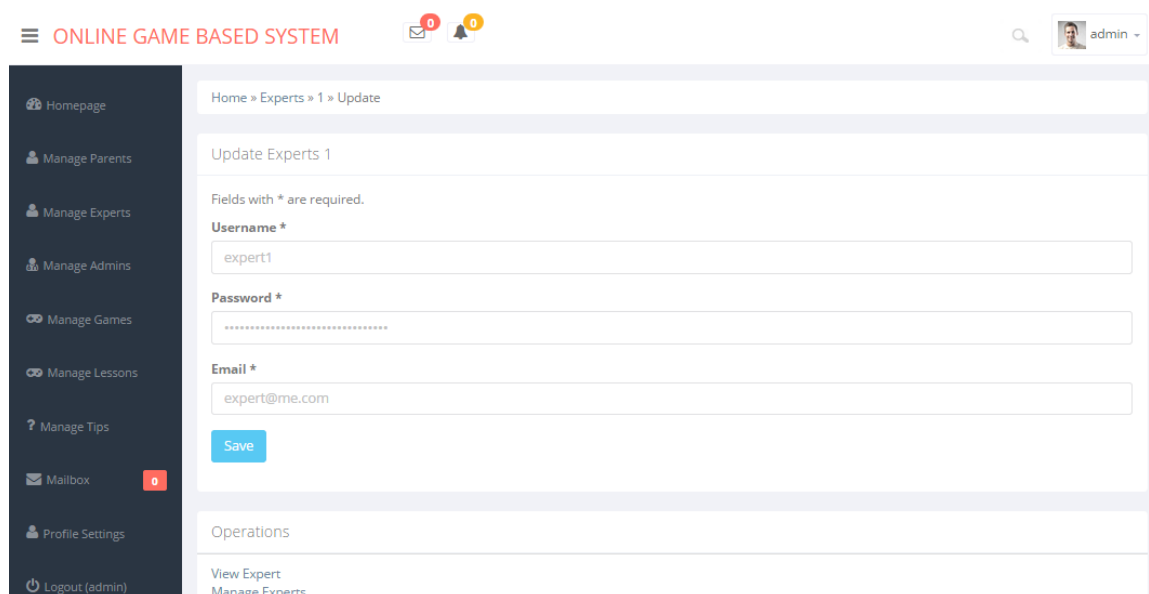


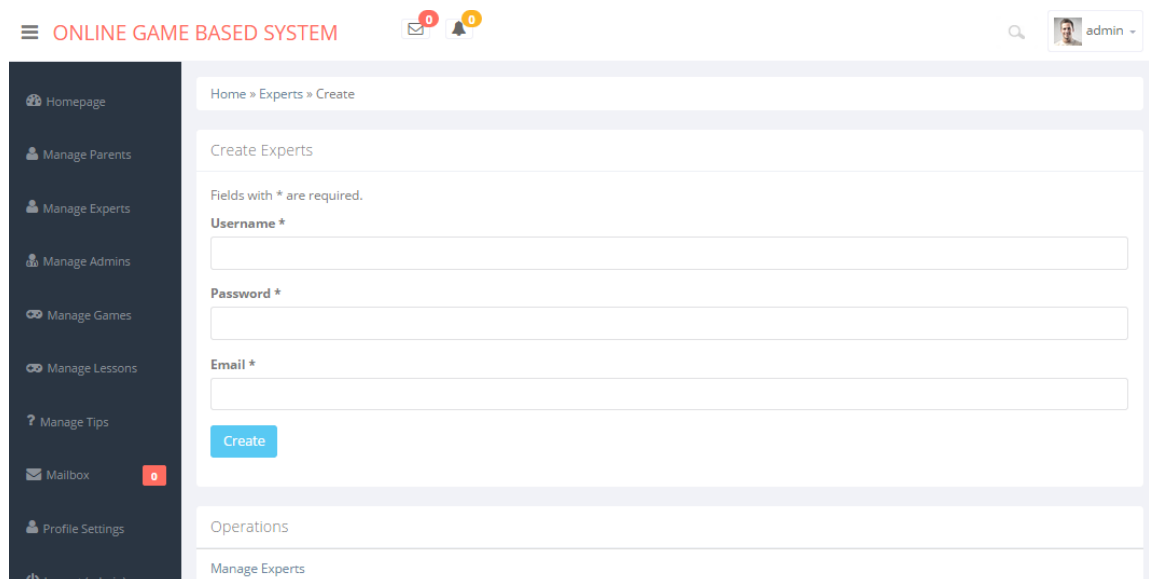
Figure 19. "Update Expert" page

When the administrator clicks on "Delete Expert" operation in "View Expert" page, an alert will be appeared for the confirmation of this action. By clicking "Yes" on the confirmation alert, the expert user will be deleted and the page will be refreshed. This alert is shown in figure 15.

Within the "Manage Experts" page, when the administrator clicks on update (🍷), s/he will be redirected to "Update Expert" page. Also if the administrator clicks on delete (✖) action, the delete will be done in the same manner as described above.

Finally, when the user clicks on "Create Expert" operation within the "Manage Experts" page, s/he will be redirected to "Create Expert" page as shown in figure 20. It is divided into a form and operations section. By filling this form and clicking on "Create" button, if all the fields are filled, the new expert user will be added to the system and the administrator will be redirected to "View Expert" page; otherwise, the system will show a message that says the administrator should fill all the fields and then submit the form again.

Also, the operations section includes "Manage Experts" which will redirect the administrator to "Manage Experts" page.



The screenshot shows a web application interface for an "ONLINE GAME BASED SYSTEM". The top navigation bar includes the system name, notification icons, a search icon, and a user profile for "admin". A dark sidebar on the left contains navigation items: Homepage, Manage Parents, Manage Experts, Manage Admins, Manage Games, Manage Lessons, Manage Tips, Mailbox, Profile Settings, and Logout (admin). The main content area is titled "Home » Experts » Create" and "Create Experts". It features a form with three required fields: "Username *", "Password *", and "Email *", each with an input box. A blue "Create" button is positioned below the form. Below the form, there is an "Operations" section with a link for "Manage Experts".

Figure 20. "Create Expert" page

4.1.1.3 "Manage Admins" navigation item

"Mange Admins" navigation item redirects administrators to the "Manage Administrators" page. This page contains the table of the registered administrators in addition to operations section which are shown in figure 21.

ONLINE GAME BASED SYSTEM

Home » Admins » Manage

Manage Admins

Displaying 1-3 of 3 results.

ID	Username	Password	Email	Createdate	
1	admin	21232f297a57a5a743894a0e4a801fc3	admin@admin.com	2013-10-21 09:12:09	
2	admin2	c4ca4238a0b923820dcc509a6f75849b	admin@admin.com	2013-11-17 17:42:40	
3	admin3	c4ca4238a0b923820dcc509a6f75849b	admin@admin.com	2013-11-17 17:45:10	

Operations

Create Administrator

Figure 21. "Manage Administrators" page

Each row of the table contains "ID" number, "Username", encrypted "Password", "E-mail" and "Created Date" of each administrator user in addition to the available actions. These actions are view () , update () and delete () over an administrator user. "Create Administrator" is the only available operation in this page which is located at the operations section.

When the administrator clicks on view () action, s/he will be redirected to "View Administrator" page as shown in figure 22. In this page, details of the selected administrator user are given in a table. Also, operations section of this page contains "Update Administrator" and "Delete Administrator".

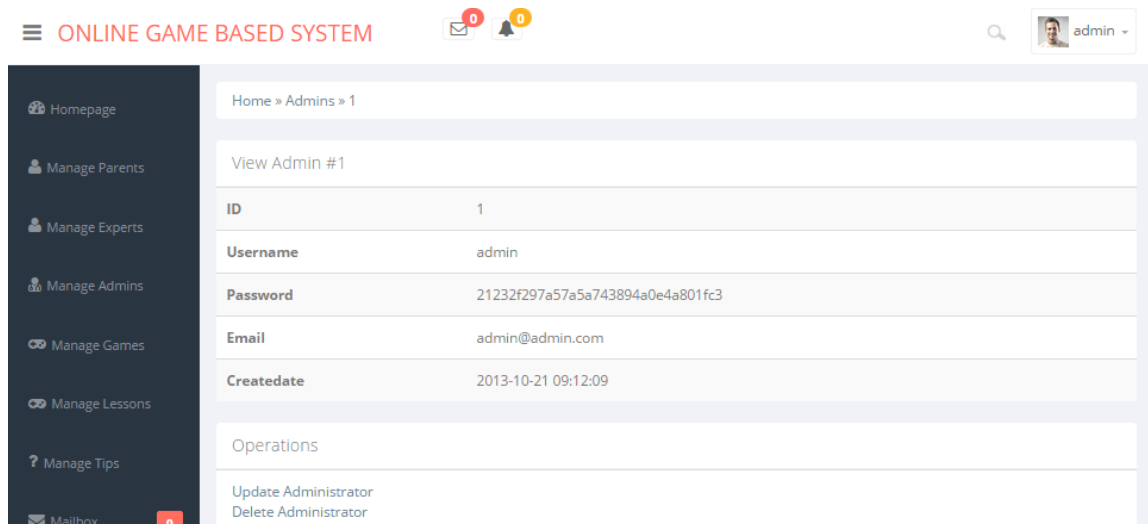


Figure 22. "View Administrator" page

When the administrator clicks on "Update Administrator" operation, s/he will be redirected to "Update Administrator" page as shown in figure 23. It has a form section and operations section. The form section contains the current details of the selected administrator and can be updated by changing the values of the fields and clicking on the "Save" button. After clicking on the "Save" button, if all the fields are filled, the administrator will be redirected to updated "View Administrator" page; otherwise, the system will show a message that says the administrator should fill all the fields and then submit the form again.

Also, the operations section includes "View Administrator" which redirects the user to "View Administrator" page and "Manage Administrators" which will redirect the administrator to the "Manage Administrators" page.

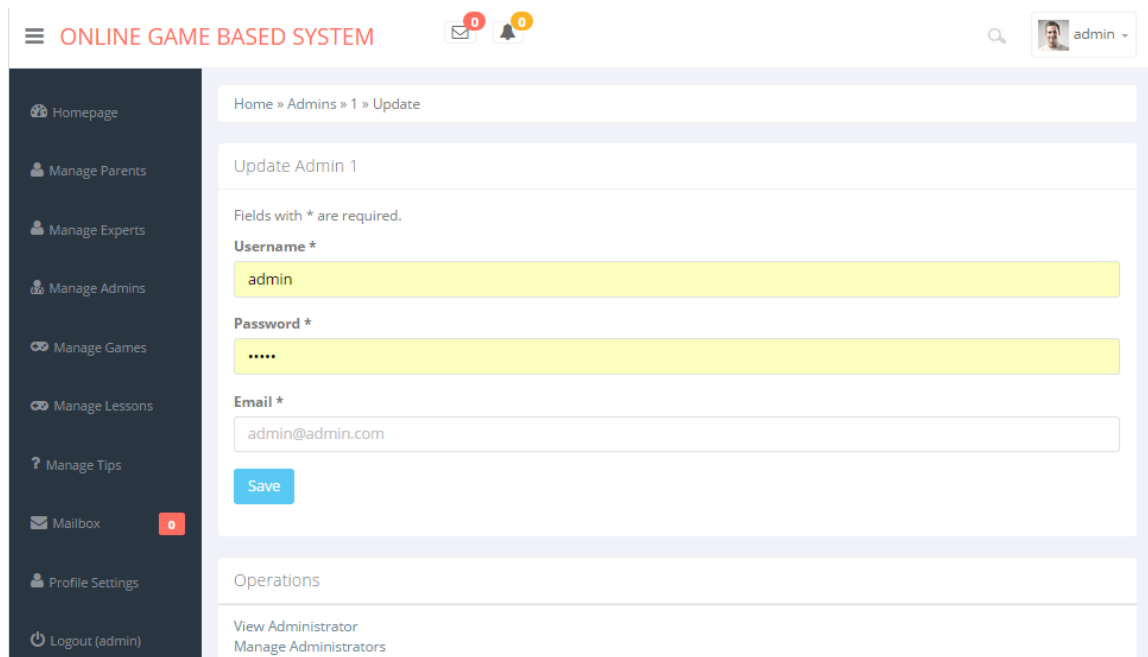


Figure 23. "Update Administrator" page

When the administrator clicks on "Delete Administrator" operation in "View Administrator" page, an alert will be appeared for the confirmation of this action. By clicking "Yes" on the confirmation alert, the administrator user will be deleted and the page will be refreshed. This alert is shown in figure 15.

Within the "Manage Administrators" page, when the administrator clicks on update (🎨), s/he will be redirected to "Update Administrator" page. Also if the administrator clicks on delete (✖) action, the delete will be done in the same manner as described above.

Finally, when the user clicks on "Create Administrator" operation within the "Manage Administrators" page, s/he will be redirected to "Create Administrator" page as shown in figure 24. It is divided into a form and operations section. By filling this form and clicking on "Create" button, if all the fields are filled, the new administrator user will be added to the system and the administrator will be

redirected to "View Administrator" page; otherwise, the system will show a message that says the administrator should fill all the fields and then submit the form again.

Also, the operations section includes "Manage Administrator" which will redirect the administrator to "Manage Administrators" page.

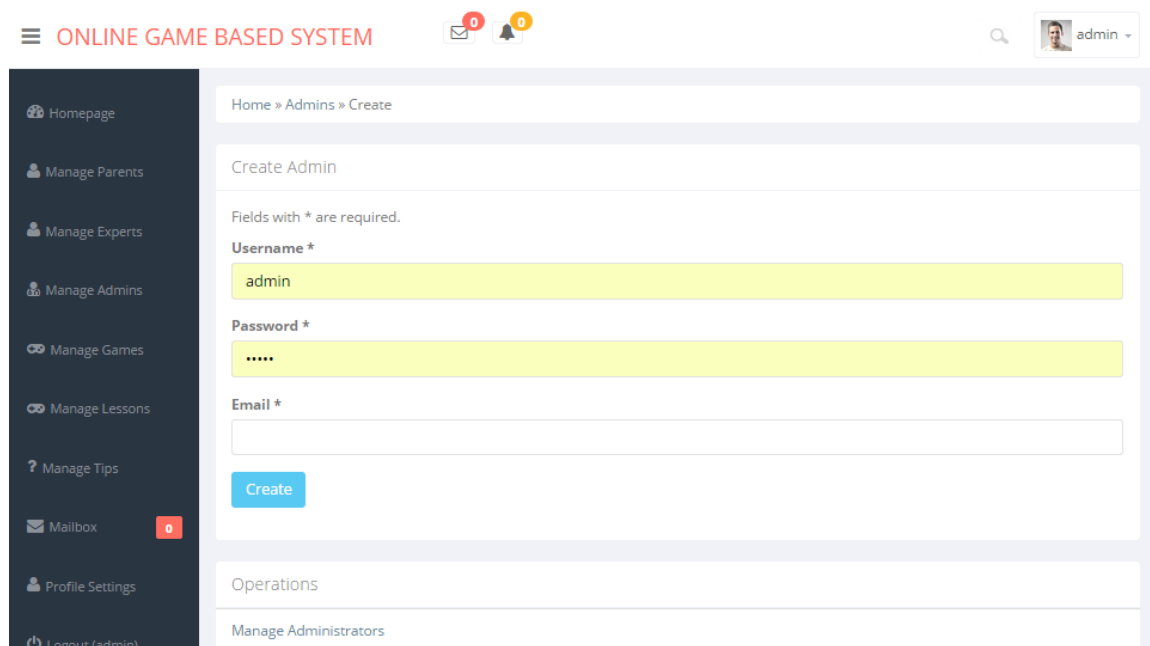


Figure 24. "Create Administrator" page

4.1.1.4 "Manage Games" navigation item

"Mange Games" navigation item redirects administrators to the "Manage Games" page. This page contains the table of the games in addition to operations section which are shown in figure 25.

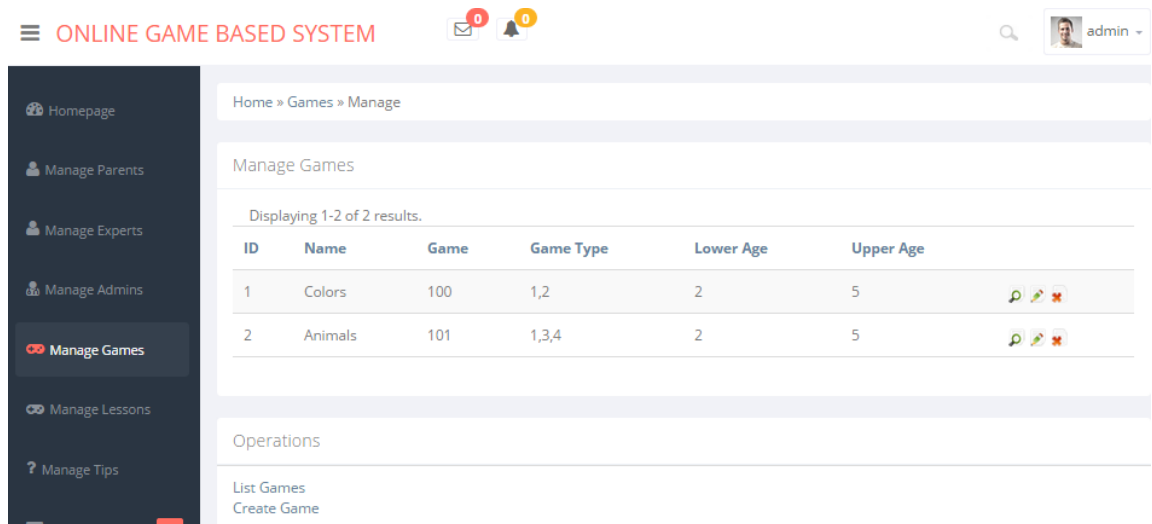


Figure 25. "Manage Games" page

Each row of the table contains "ID" number, "Name", "Game" unique id, "Game Type", "Lower Age" and "Upper Age" of each game in addition to the available actions. These actions are view () , update () and delete () over a game. "List Games" and "Create Game" are the available operations in this page which is located at the operations section.

When the administrator clicks on view () action, s/he will be redirected to "View Game" page as shown in figure 26. In this page, details of the selected game are given in a table. Also, operations section of this page contains "Update Game" and "Delete Game".

The screenshot shows the 'View Game' page for a game named 'Colors'. The table below contains the following data:

ID	1
Name	Colors
Game	100
Game Type	1,2
Lower Age	2
Upper Age	5
Photo	colors.jpeg

The 'Operations' section includes the following links:

- Update Game
- Delete Game

Figure 26. "View Game" page

When the administrator clicks on "Update Game" operation, s/he will be redirected to "Update Game" page as shown in figure 27. It has a form section and operations section. The form section contains the current details of the selected game and can be updated by changing the values of the fields and clicking on the "Save" button. After clicking on the "Save" button, if all the fields are filled, the administrator will be redirected to updated "View Game" page; otherwise, the system will show a message that says the administrator should fill all the fields and then submit the form again.

Also, the operations section includes "View Game" which redirects the user to "View Game" page and "Manage Games" which will redirect the administrator to the "Manage Games".

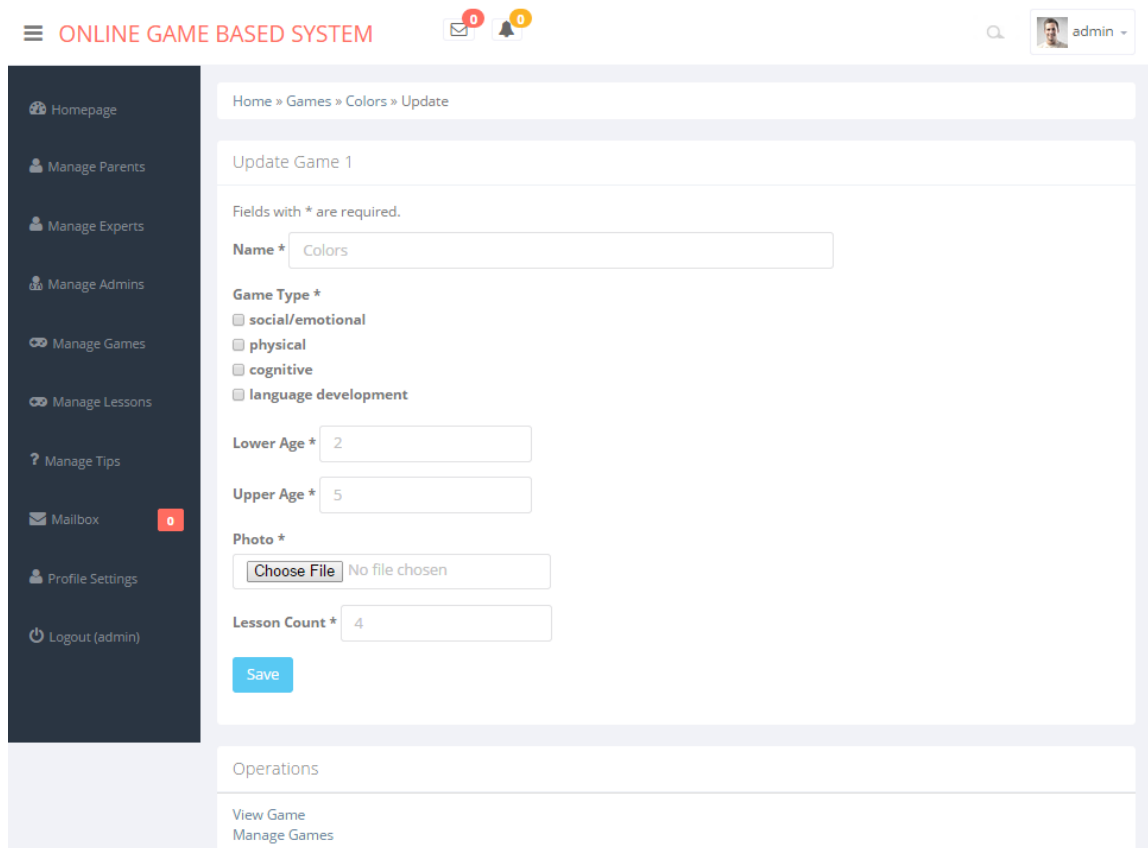


Figure 27. "Update Game" page

When the administrator clicks on "Delete Game" operation in "View Game" page, an alert will be appeared for the confirmation of this action. By clicking "Yes" on the confirmation alert, the expert user will be deleted and the page will be refreshed. This alert is shown in figure 15.

Within the "Manage Games" page, when the administrator clicks on update (🖌️), s/he will be redirected to "Update Game" page. Also if the administrator clicks on delete (🗑️) action, the delete will be done in the same manner as described above.

Within "Manage Games" page, if the user clicks on "List Games" operation, s/he will be redirected to "List Games" page as shown in figure 28. This page contains the grid

view of the games in order to give an overview of all games to the administrators. Administrators by clicking on each game will be redirected to "Update Game" page.

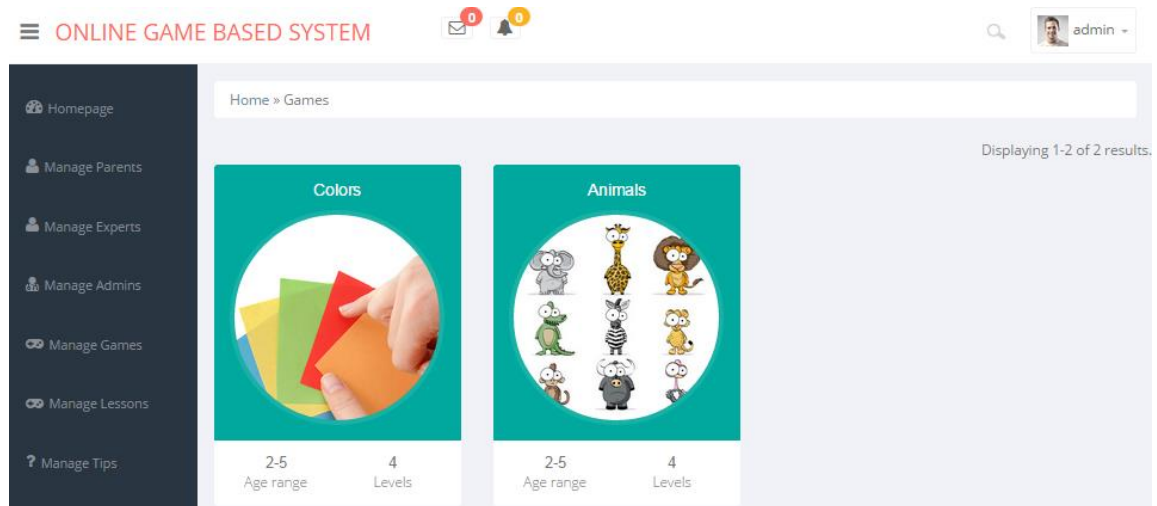


Figure 28. "List Games" page

Finally, when the user clicks on "Create Game" operation within the "Manage Games" page, s/he will be redirected to "Create Game" page as shown in figure 29. It is divided into a form and operations section. By filling this form and clicking on "Create" button, if all the fields are filled, the new game will be added to the system and the administrator will be redirected to "View Game" page; otherwise, the system will show a message that says the administrator should fill all the fields and then submit the form again.

Also, the operations section includes "Manage Games" which will redirect the administrator to "Manage Games" page.

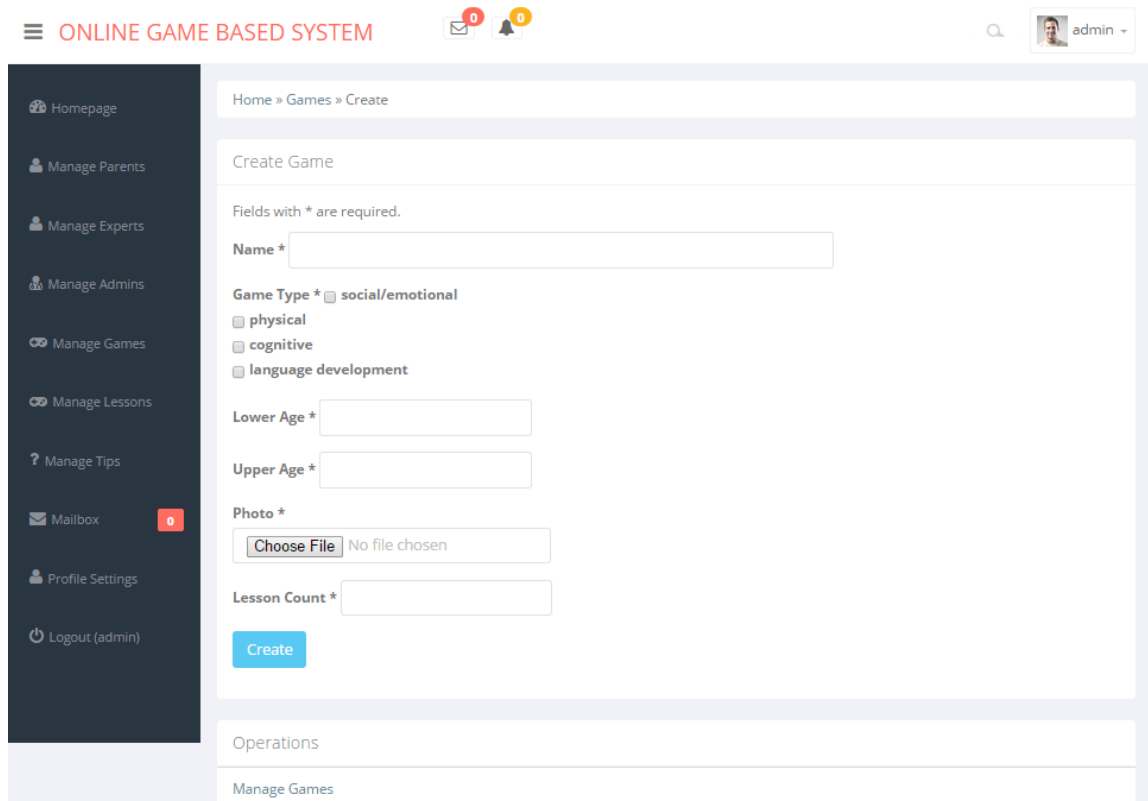


Figure 29. "Create Game" page

4.1.1.5 "Manage Lessons" navigation item

















"Mange Lessons" navigation item redirects administrators to the "Manage Lessons" page. This page contains the table of the lessons in addition to operations section which are shown in figure 30.

ONLINE GAME BASED SYSTEM

admin

Manage Lessons



Displaying 1-8 of 8 results.


ID	Game	Lesson	Level	
1	Colors	100	1	 
2	Colors	101	2	 
3	Colors	102	3	 
4	Colors	103	4	 
5	Animals	104	1	 
6	Animals	105	2	 
7	Animals	106	3	 
8	Animals	107	4	 

Operations

Create Lesson

Figure 30. "Manage Lessons" page

Each row of the table contains "ID" number, "Game" name, "Lesson" unique id and "Level" of each lesson in addition to the available actions. These actions are update () and delete () a lesson. "Create Lesson" is the only available operation in this page which is located at the operations section.

When the administrator clicks on update () action, s/he will be redirected to "Update Lesson" page as shown in figure 31. It has a form section and operations section. The form section contains the current details of the selected lesson and can be updated by changing the values of the fields and clicking on the "Save" button. After clicking on the "Save" button, if all the fields are filled, the administrator will be redirected to "Manage Lessons" page; otherwise, the system will show a message that says the administrator should fill all the fields and then submit the form again.

Also, the operations section includes "Manage Lessons" which will redirect the administrator to "Manage Lessons" page.

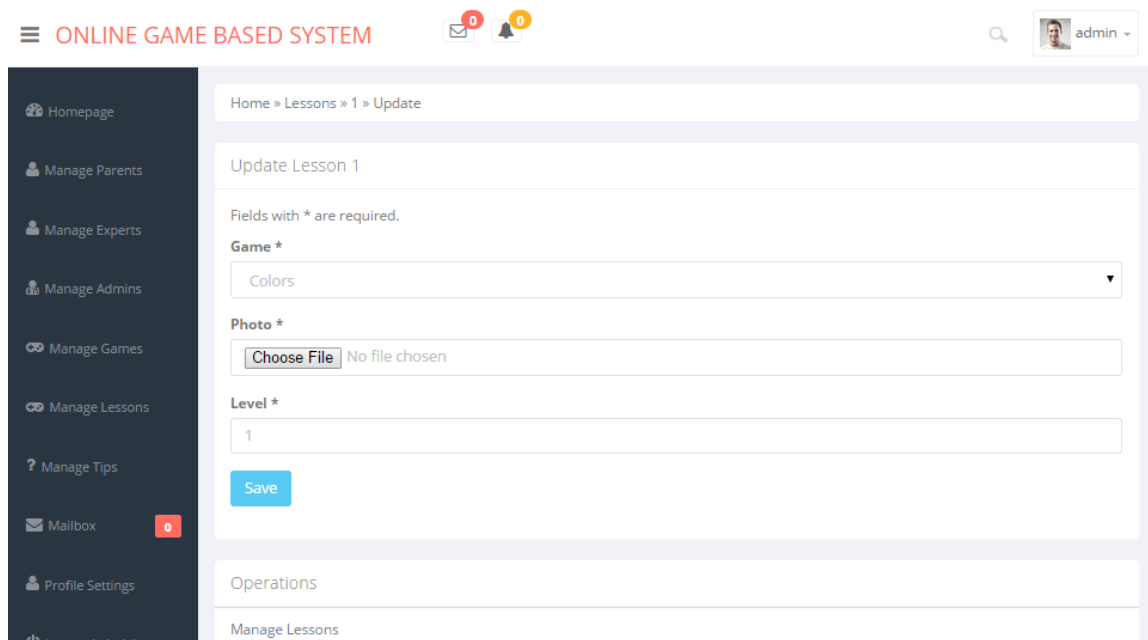


Figure 31. "Update Lesson" page

Within the "Manage Lessons" page, when the administrator clicks on delete (✖) action, an alert will be appeared for the confirmation of this action. By clicking "Yes" on the confirmation alert, the lesson will be deleted from the system. This alert is shown in figure 15.

Finally, when the user clicks on "Create Lesson" operation within the "Manage Lessons" page, s/he will be redirected to "Create Lesson" page as shown in figure 32. It is divided into a form and operations section. By filling this form and clicking on "Create" button, if all the fields are filled, the new lesson will be added to the system and the administrator will be redirected to "Manage Lessons" page; otherwise, the system will show a message that says the administrator should fill all the fields and then submit the form again.

Also, the operations section includes "Manage Lessons" which will redirect the administrator to "Manage Lessons" page.

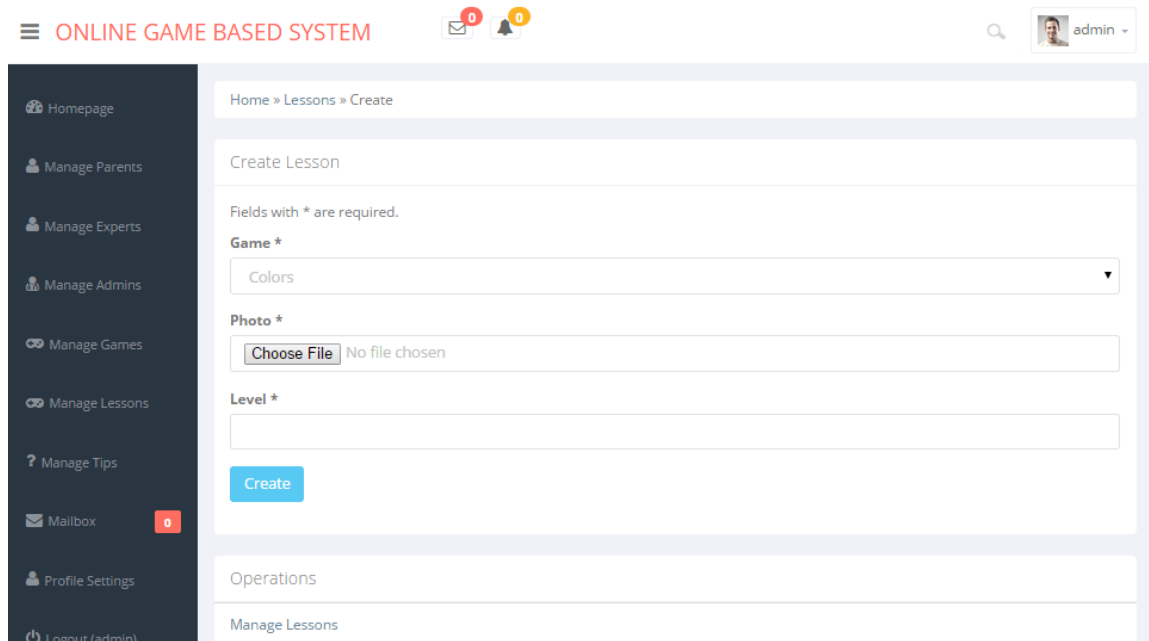


Figure 32. "Create Lesson" page

4.1.1.5 "Manage Tips" navigation item

"Mange Tips" navigation item redirects administrators to the "Manage Tips" page. This page contains the table of the tips in addition to operations section which are shown in figure 33.

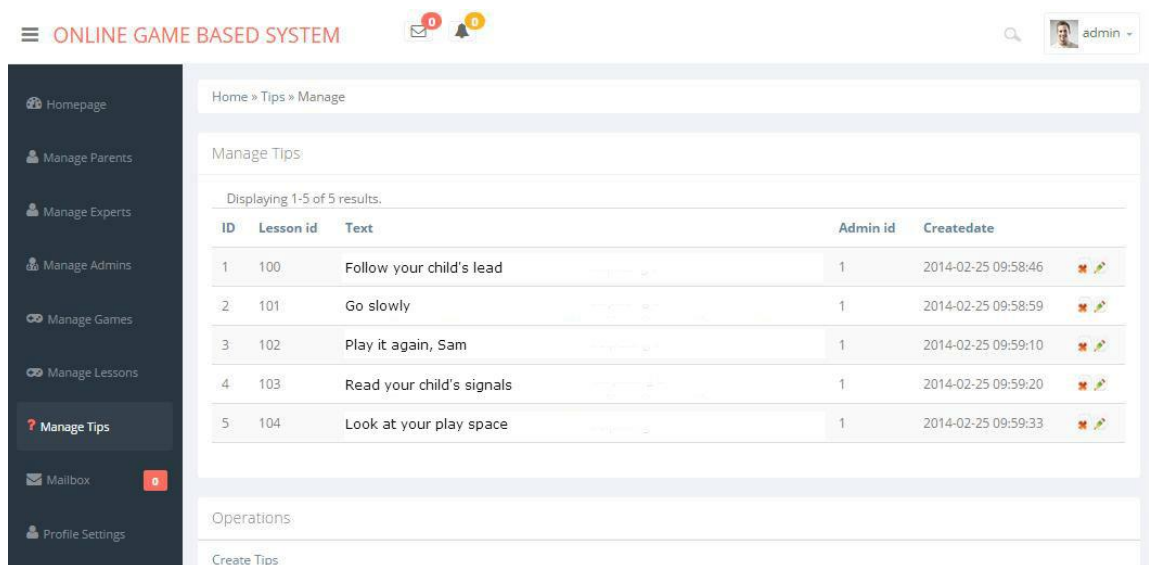
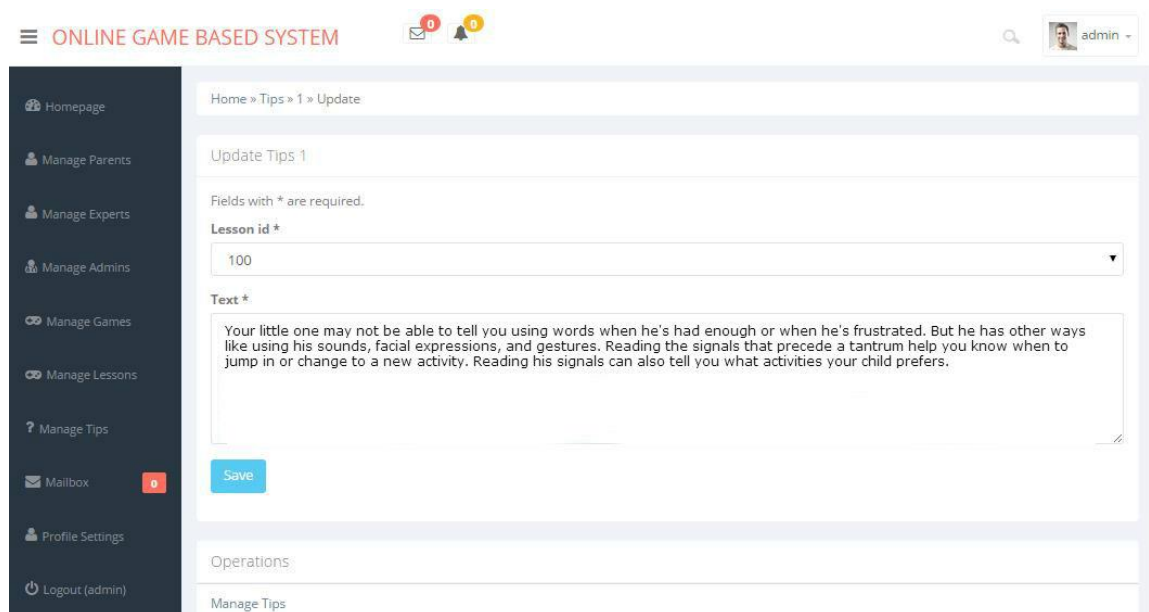


Figure 33. "Manage Tips" Page

Each row of the table contains "ID" number, "Lesson id", "Text", "Admin id" and "Created Date" of each tip in addition to the available actions. These actions are update (✎) and delete (✖) a tip. "Create Tips" is the only available operation in this page which is located at the operations section.

When the administrator clicks on update (✎) action, s/he will be redirected to "Update Tips" page as shown in figure 34. It has a form section and operations section. The form section contains the current details of the selected tip and can be updated by changing the values of the fields and clicking on the "Save" button. After clicking on the "Save" button, if all the fields are filled, the administrator will be redirected to "Manage Tips" page; otherwise, the system will show a message that says the administrator should fill all the fields and then submit the form again.

Also, the operations section includes "Manage Tips" which will redirect the administrator to "Manage Tips" page.



The screenshot displays the "Update Tips" page within an "ONLINE GAME BASED SYSTEM". The page has a dark sidebar on the left with navigation options: Homepage, Manage Parents, Manage Experts, Manage Admins, Manage Games, Manage Lessons, Manage Tips, Mailbox, Profile Settings, and Logout (admin). The main content area shows the breadcrumb "Home » Tips » 1 » Update" and the title "Update Tips 1". A message states "Fields with * are required." Below this, there are two form fields: "Lesson id *" with a dropdown menu showing "100", and "Text *" with a text area containing the text: "Your little one may not be able to tell you using words when he's had enough or when he's frustrated. But he has other ways like using his sounds, facial expressions, and gestures. Reading the signals that precede a tantrum help you know when to jump in or change to a new activity. Reading his signals can also tell you what activities your child prefers." A blue "Save" button is located below the text area. At the bottom of the page, there is an "Operations" section with a link for "Manage Tips".

Figure 34. "Update Tips" Page

Within the "Manage Tips" page, when the administrator clicks on delete (✖) action, an alert will be appeared for the confirmation of this action. By clicking "Yes" on the confirmation alert, the tip will be deleted from the system. This alert is shown in figure 15.

Finally, when the user clicks on "Create Tips" operation within the "Manage Tips" page, s/he will be redirected to "Create Tips" page as shown in figure 35. It is divided into a form and operations section. By filling this form and clicking on "Create" button, if all the fields are filled, the new tips will be added to the system and the administrator will be redirected to "Manage Tips" page; otherwise, the system will show a message that says the administrator should fill all the fields and then submit the form again.

Also, the operations section includes "Manage Tips" which will redirect the administrator to "Manage Tips" page.

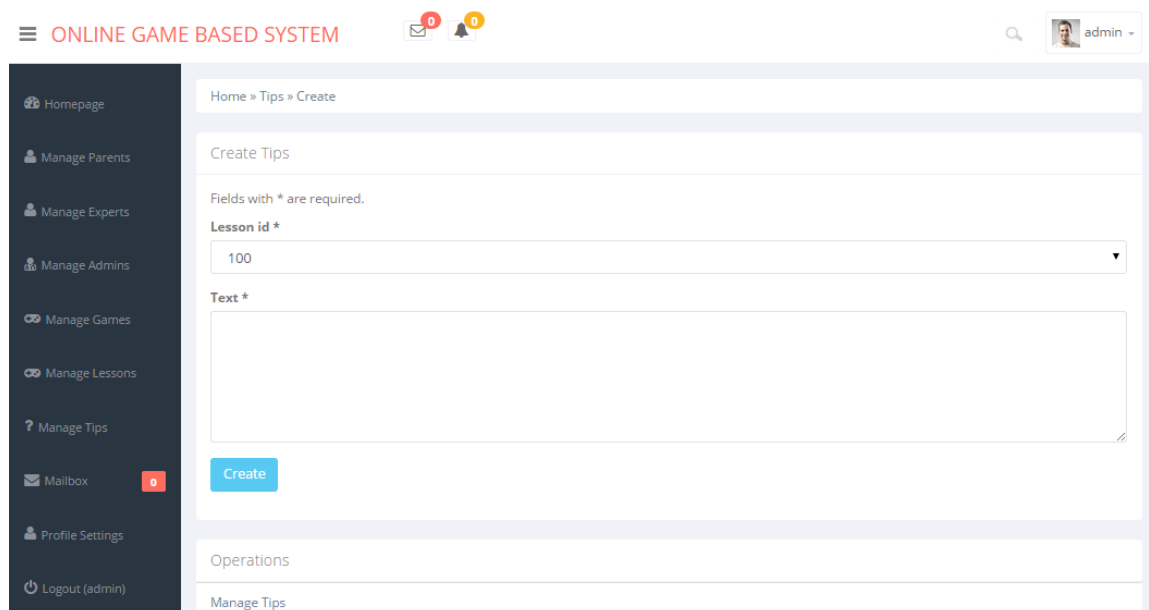


Figure 35. "Create Tips" Page

4.1.1.6 "View Evaluations" navigation item

"View Evaluations" navigation item redirects administrators to the "Evaluations" page. This page is shown in figure 36 and contains grid view of all games and shows the administrator that does each game evaluated or not. In addition date and time of the last evaluation of a game will be shown. Also there is a table of all evaluations in this page.

Each row of the table contains "Game" unique id, "Name", "Evaluations count" and average of evaluation function for interface, education, content, playability, multimedia and average of usability mean for each game in addition to the available actions. The only action on this page is view (🔍) an evaluation of a game.

The screenshot displays the 'ONLINE GAME BASED SYSTEM' interface. The top navigation bar includes the system name, notification icons, and a user profile for 'admin'. A sidebar on the left contains navigation options, with 'View Evaluations' highlighted. The main content area shows a grid of two game cards: 'Colors' (evaluated 2014-12-08 at 12:36:22) and 'Animals' (evaluated 2014-12-08 at 15:28:00). Below the grid is a table titled 'PHEG Evaluations table' with the following data:

Game	Name	Evaluation Count	Avg F(i)%	Avg F(e)%	Avg F(c)%	Avg F(p)%	Avg F(m)%	Avg Mean %	
100	Colors	2	54	51	20	24	29	35	🔍
101	Animals	3	38	20	27	19	8	22	🔍

Figure 36. "Evaluations" page

By clicking on each game in grid view or clicking on view (🔍) action, the administrator will be redirected to "View Evaluation" page of selected game as shown in figure 37. This page is divided into a chart and operations section. This chart shows the average percentage of all evaluations for each evaluation category (Interface, Educational, Content, Playability and Multimedia) and usability mean over a specific game. Also, the operations section includes "List Games" which will redirect the administrator to "Evaluations" page.

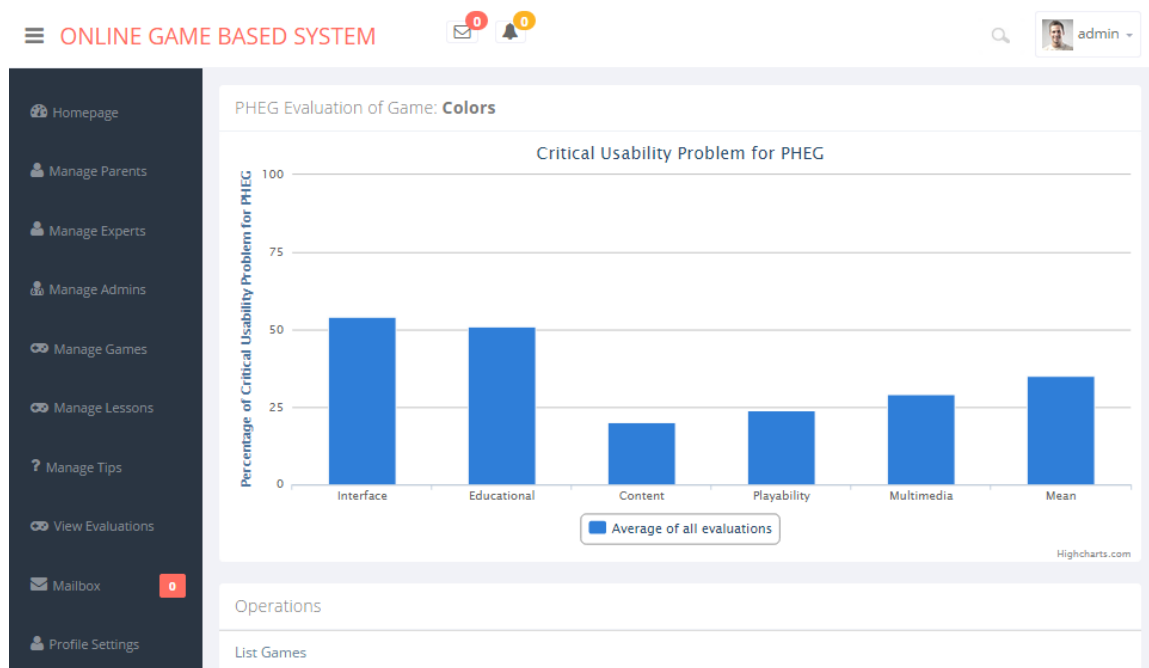


Figure 37. "View Evaluation" page

4.1.1.6 "Mailbox" navigation item

When the user clicks on "Mailbox" navigation item, s/he will be redirected to the "Inbox" page. This page is shown in figure 38.

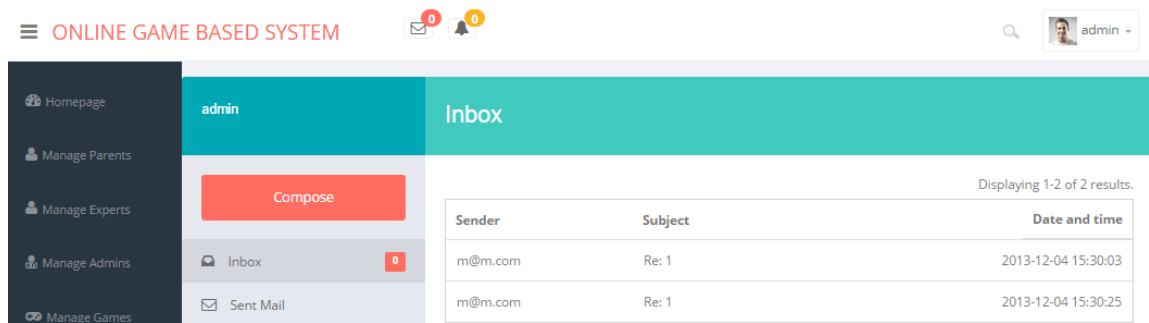


Figure 38. "Inbox" page

This page contains a sidebar on left and a container on right. Left sidebar contains "Compose" button, "Inbox" and "Sent Mail" sub navigation. Also, there is a red badge beside of the "Inbox" sub navigation item which shows the number of unread messages. The container of this page is a table of received mails. This table's columns are "Sender", "Subject" and "Date and time" of mails. In order to separate read mails and unread mails, the color of read mail table rows is shown in white and the color of unread mail table rows is shown in gray.

By clicking on "Compose" button within "Inbox" page, the user will be redirected to "Compose" page as shown in figure 39. It contains a form in which the user should select a receiver user from a dropdown list and fill other fields of the form and click on "Send" button in order to send a mail. After clicking on the "Send" button, if all the fields are filled, the user will be redirected to "Inbox" page; otherwise, the system will show a message that says the user should fill all the fields and then submit the form again.

The image shows a 'Compose' form with the following elements:

- Header:** A teal bar containing the text 'Compose' and a small 'x' icon for closing the window.
- To:** A dropdown menu with the text 'Choose One' and a downward arrow.
- Subject:** A standard text input field.
- Message:** A larger text area for composing the email body.
- Send:** A teal button located at the bottom left of the form.

Figure 39. "Compose" page

Within "Inbox" page, if the user clicks on any row of the inbox table, s/he will be redirected to "View Mail" page of selected mail as shown in figure 40.

The image shows the 'View Mail' page with the following layout:

- Header:** A teal bar with the text 'View Mail' and a 'Reply' button.
- Left Sidebar:** A dark sidebar with navigation links: Homepage, Manage Parents, Manage Experts, Manage Admins, Manage Games, Manage Lessons, Manage Tips, Mailbox (with a red notification badge), and Profile Settings.
- Main Content Area:**
 - From:** m@m.com
 - To:** admin
 - Subject:** Re: Child user problem
 - Message:** Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit

Figure 40. "View Mail" page

This page contains a sidebar on left and a container on right. Left sidebar is same as "Inbox" page sidebar and the container of this page shows the contents of the mail. Mail contents includes "From" field which is the mail sender's e-mail or username,

"To" field which is the mail receiver's e-mail or username, "Subject" field is the subject of the mail and "Message" is the body of this mail. Also, there is a "Reply" button on top of this page which the user by clicking on this button will be redirected to "Compose" page.

"Sent Mail" sub navigation item redirects the user to the sent box of the mailbox. This page is shown in figure 41.



Figure 41. "Sent Mail" page

This page contains a sidebar on left and a container on right. Left sidebar is same as "Inbox" page's left sidebar and the container of this page is a table of sent mails. This table's columns are "Sender", "Subject" and "Date and time" of mails. "Receiver" is the mail receiver's e-mail or username, "Subject" is the subject of the mail and "Date and time" is the time of creating this mail. By clicking on each row of the table, the user will be redirected to "View Mail" page of selected mail.

4.1.1.7 "Profile settings" navigation item

This navigation item will redirect the administrator user to his/her "Profile Settings" page. This page contains a form in order to give administrator, the ability of changing his/her username, e-mail address or password. This page is shown in figure 42.

Figure 42. "Profile Settings" page

4.1.1.8 "Log out" navigation item

Administrators can exit from the system by clicking on this navigation item. After successful logout, administrator users will be redirected to "Login" page.

4.1.2 Expert sub module

When an expert user logs in to the system, this sub module will be executed. Expert users after logging to the system will be redirected to the "Evaluations" page as shown in figure 43.

Expert sub module consists of a sidebar on left which is the navigation of this sub module and a content section on right. The content section will show contents of each navigation item page. The expert sub module navigation items are "Game Evaluation", "Children Activity", "Assist Requests", "Mailbox", "Profile Settings" and "Logout". Each navigation item will be described in the following subsections.

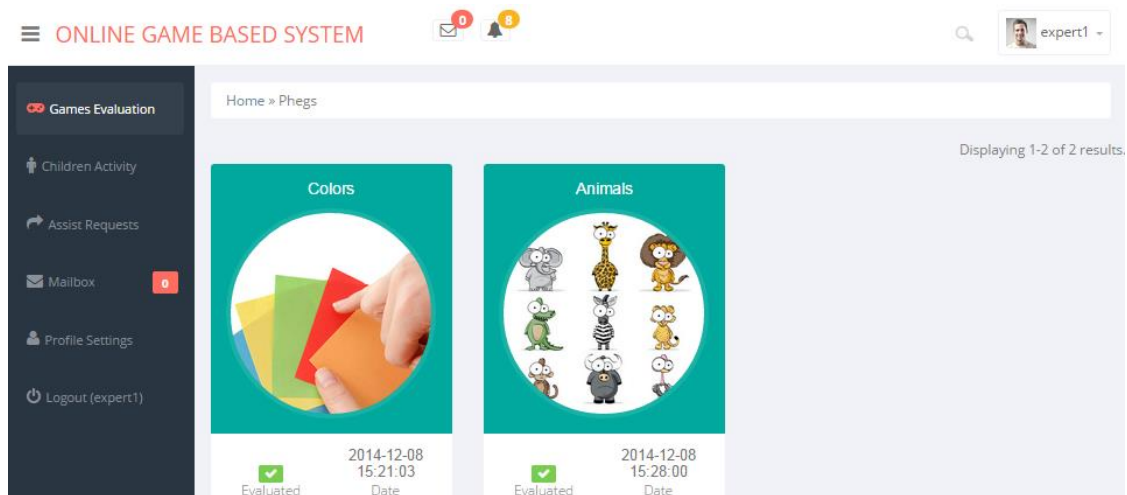


Figure 43. "Evaluations" page

4.1.2.1 "Games Evaluation" navigation item

"Games Evaluation" navigation item redirects experts to the "Evaluations" page as shown in figure 44. This page contains the grid view of all games and shows the expert that the game is evaluated or not. In addition, date and time of evaluation of a game will be shown to the expert. By clicking on each game, if the game was not evaluated before, the experts will be redirected to "Create Evaluation" page; otherwise they will be redirected to "Update Evaluation" page.

In a same way, "Create Evaluation" page and "Update Evaluation" page contain a form which includes all the heuristics questions, which are available in appendix A, as shown in figure 45. Also, answer fields of this form are a dropdown list whose the options are listed below:

- "I don't agree that this is a usability problem at all"
- "Cosmetic problem only. need not be fixed unless extra time is available on project"
- "Minor usability problem. fixing this should be given low priority"
- "Major usability problem. important to fix, so should be given high priority"
- "Usability catastrophe. imperative to fix this before product can be released"

Expert should select answers of questions and then click on "Save" button in order to submit an evaluation to the system. After clicking on the "Save" button, if all the fields are filled, the expert will be redirected to "View Evaluation" page; otherwise, the system will show a message that says the expert should fill all the fields and then submit the form again.

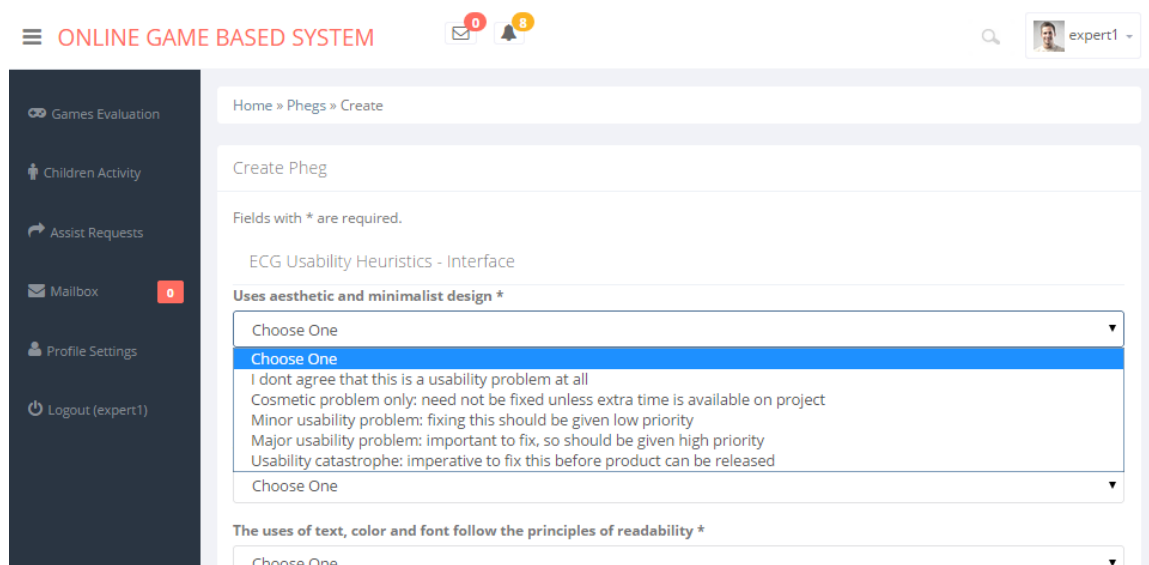


Figure 44. "Create Evaluation" page

"View Evaluation" page is divided into a chart and operations section which is shown in figure 45. This chart shows the average percentage of all evaluations beside of the percentage of the evaluation of the logged in expert for each evaluation category (Interface, Educational, Content, Playability and Multimedia) and usability mean. Also, the operations section includes "List Games" which will redirect the expert to "Evaluations" page.

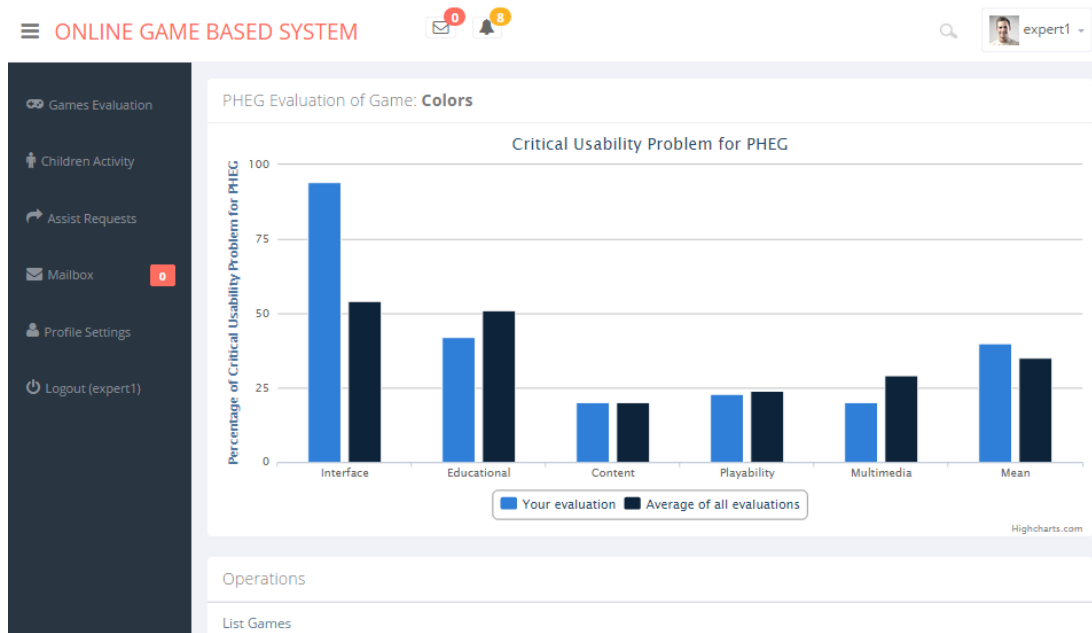


Figure 45. "View Evaluation" page

4.1.2.2 "Children Activity" navigation item

This navigation item redirects the expert user to "Children Activity" page. This page as shown in figure 46 contains grid view of all registered children; the expert user by clicking on each child will be redirected to "View Child Activity" page.

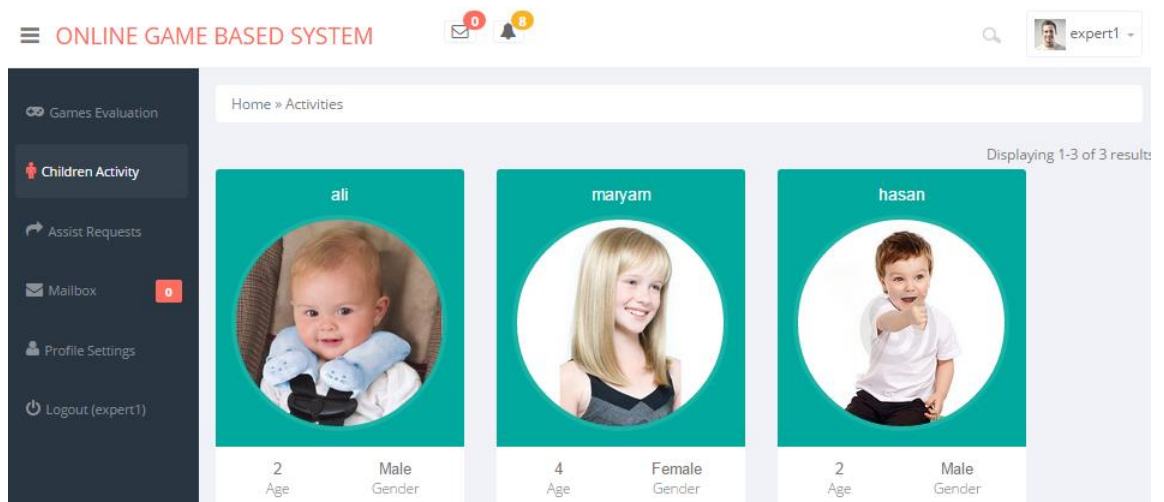


Figure 46. "Children Activity" page

"View Child Activity" page is divided into 3 sections: "Timeline", "Playing Hours charts" and "Playing Hours by Category charts".

Figure 47 shows the "Timeline" section of this page. There are three types of timeline activity which are "Started Playing" (Green Color), "Stopped Playing" (Red Color) and "Level Completion" (Blue Color). "Started Playing" and "Stopped Playing" are the time and date of start and stop the playing. Also, "Level Completion" timeline activity gives the playing game data which are the name of the game, level, duration of completion and number of wrong answers. In addition, expert advice and tips section are included to give the parents tips and advices about a game level.

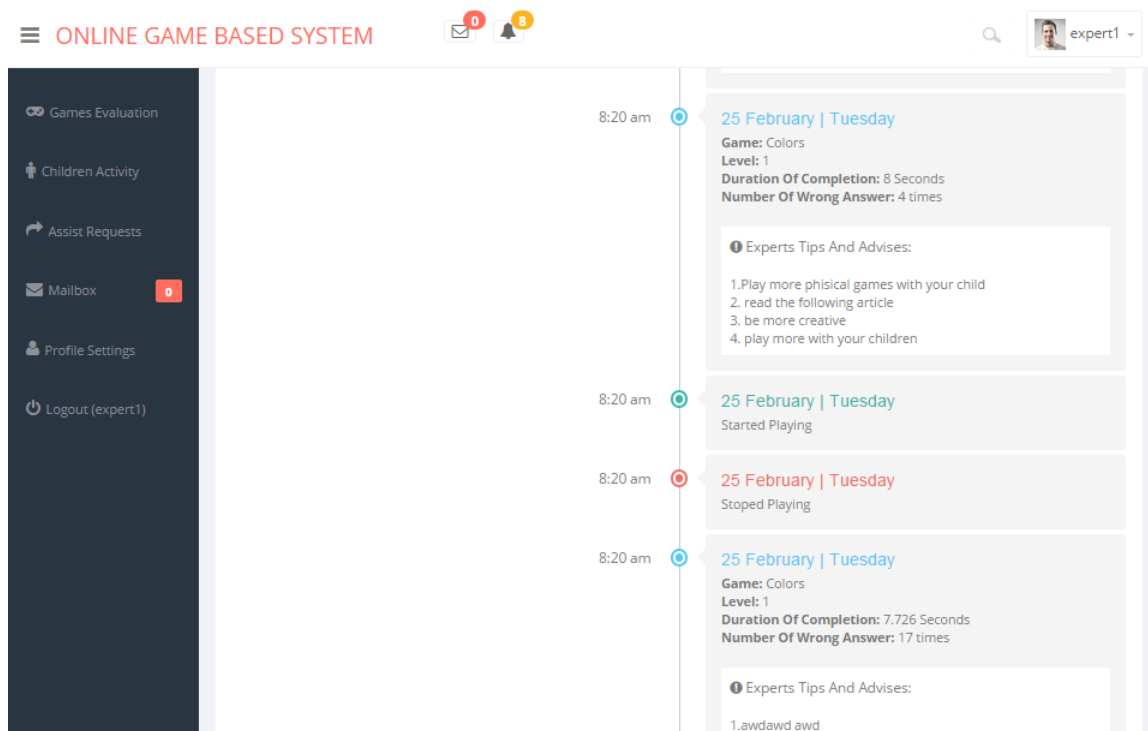


Figure 47. "View Child Activity" page, "Timeline" section

"Playing Hours charts" section includes duration of children playing time (in seconds) as weekly, monthly and yearly charts as shown in figure 48.

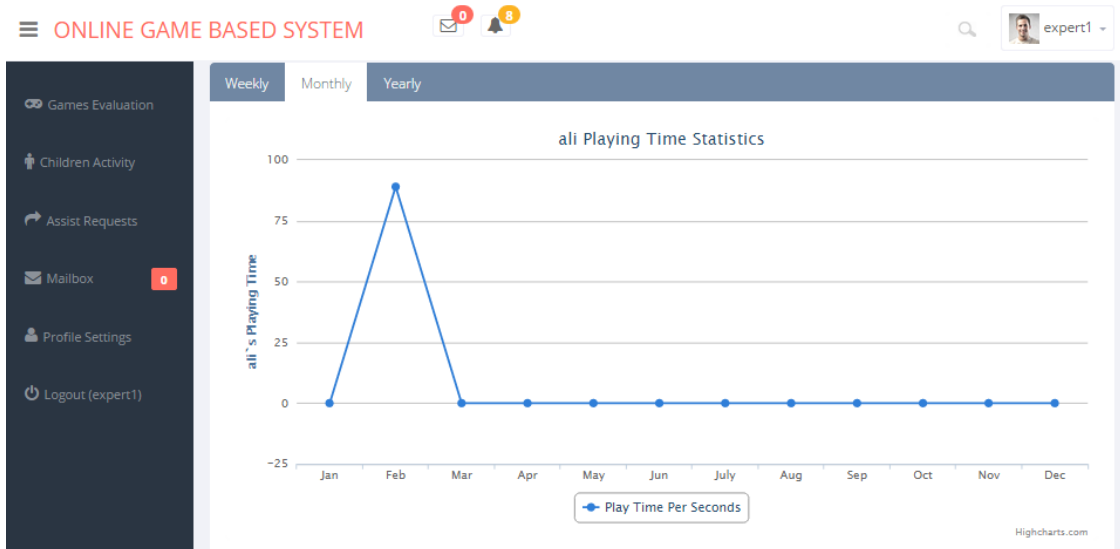


Figure 48. "View Child Activity" page, "Playing Hours" section

Also duration of playing time per game category was given as weekly, monthly and yearly charts in "Playing Hours by Category charts" section which are shown in figure 49.

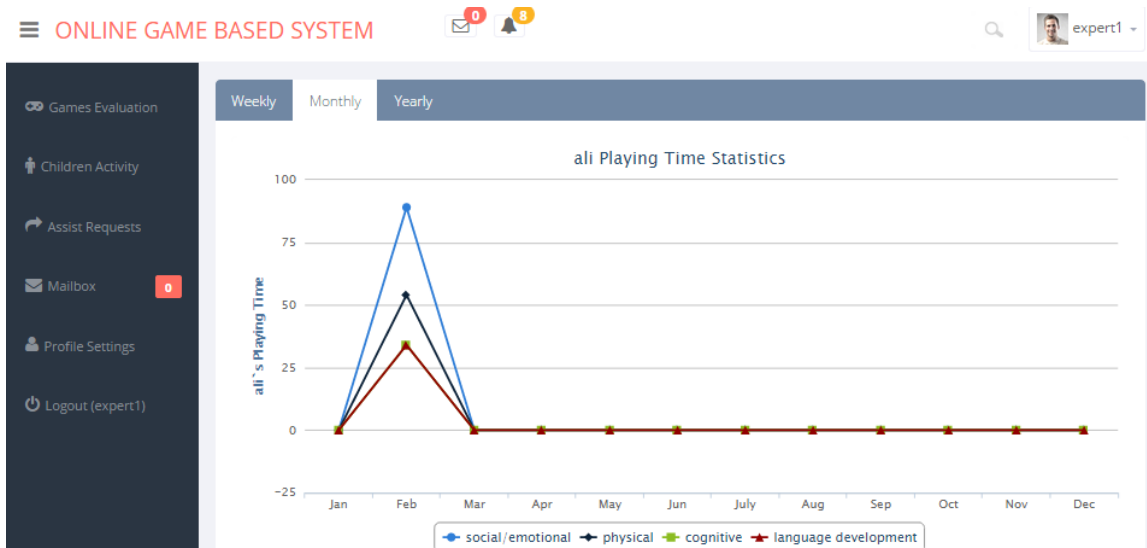


Figure 49. "View Child Activity" page, "Playing Hours by Category" section

4.1.2.3 "Assist Requests" navigation item

"Assist Requests" navigation item, redirects the experts to "Assist Requests" page as shown in figure 50. When parents request for an assist, new requests will appear in this page. This page contains the table of all the parents requested assists.

Each row of this table contains "ID", "Parent id", "Child" unique id, game's "Name" and "Created Date" of each assist request in addition to the available actions. The only action on this page is the view (👁️) of an assist request.

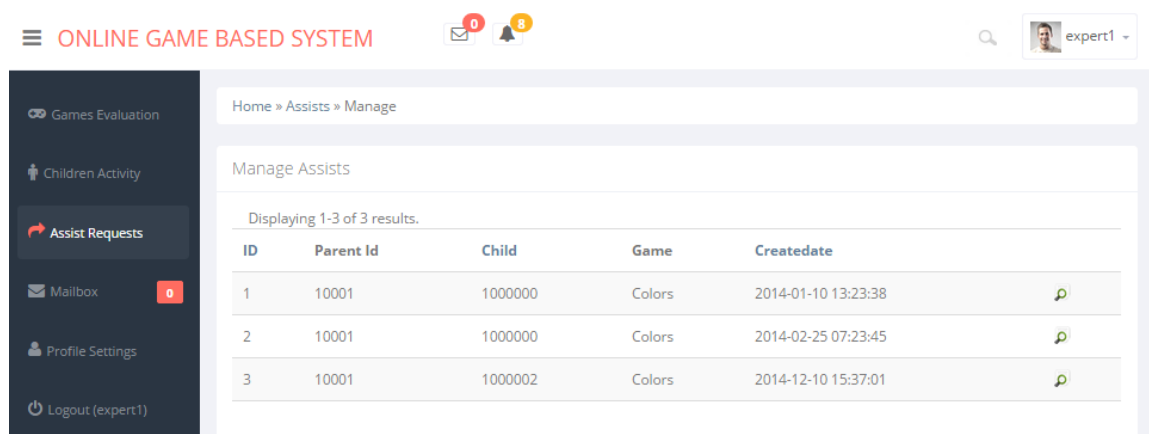


Figure 50. "Assist Requests" page

The expert user by clicking on view (👁️) action of each assist request will be redirected to "View Assist Request" page of the selected request assist as shown in figure 51. This page contains the details of selected assist request in a table beside of "Reply to sender" button. By clicking on "Reply to sender" button, expert user redirects to "Compose" page which is described on section 4.1.1.6 in order to answer the requested assist.

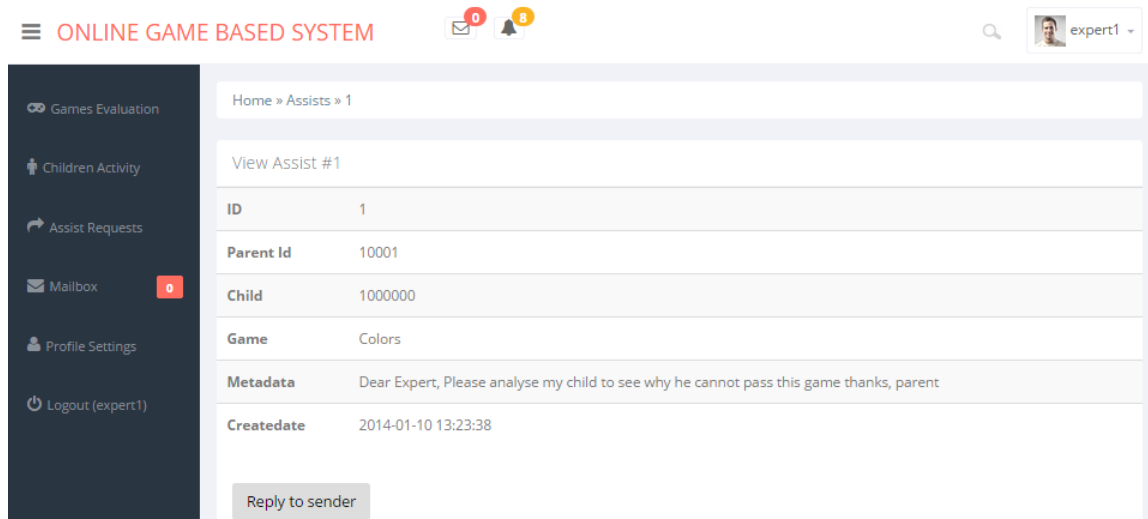


Figure 51. "View Assist Request" page

4.1.2.4 "Mailbox" navigation item

This navigation item is the same as "Mailbox" navigation item which is described under subsection 4.1.1.6.

4.1.2.5 "Profile Settings" navigation item

This navigation item is the same as "Profile Settings" navigation item which is described under subsection 4.1.1.7.

4.1.2.6 "Logout" navigation item

This navigation item is the same as "Logout" navigation item which is described under subsection 4.1.1.8.

4.1.3 Parent sub module

When a parent user logs in to the system, this sub module will be executed. Parent users after logging to the system will be redirected to the "Manage Children" page as shown in figure 52.

Parent sub module consists of a sidebar on left which is the navigation of this sub module and a content section on right. The content section will show contents of each navigation item page. The parent sub module navigation items are "Manage

Children", "Children Activity", "Request Assist", "Games Settings", "Mailbox", "Profile Settings" and "Logout". Each navigation item will be described in the following subsections.

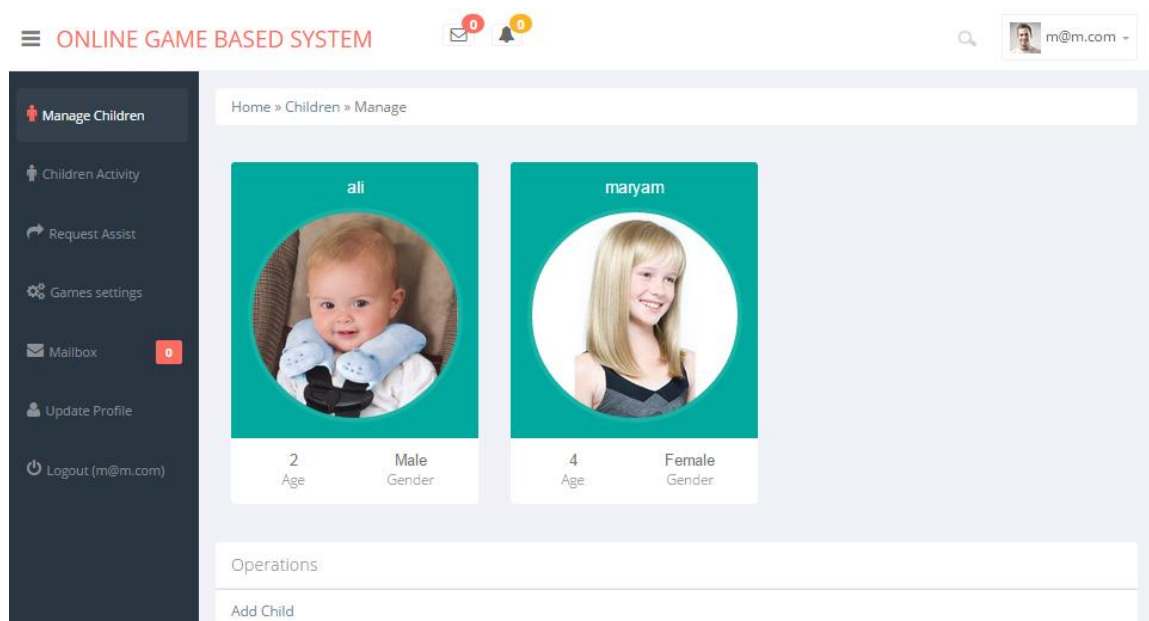


Figure 52. "Manage Children" page

4.1.3.1 "Manage Children" navigation item

"Manage Children" navigation item, redirects parents to "Manage Children" page. This page contains the grid view of registered children in addition to operations section which are shown in figure 52. Grid view contains the photo, age and gender of the child. Also, "Add Child" is the only available operation in this page which is located at the operations section.

By clicking on each child, parent user is redirected to "Update Child" page as shown in figure 53. This page contains a form section which the current details of the selected child is shown and can be updated by changing the values of the fields and clicking on the "Save" button. After clicking on the "Save" button, if all the fields are

filled, the parent will be redirected to "Manage Children" page; otherwise, the system will show a message that says the parent should fill all the fields and then submit the form again.

The screenshot displays the "Update Child" page within an "ONLINE GAME BASED SYSTEM". The page features a dark sidebar on the left with navigation options: "Manage Children", "Children Activity", "Request Assist", "Games settings", "Mailbox" (with a red notification badge), "Update Profile", and "Logout (m@m.com)". The main content area shows a breadcrumb trail "Home » Children » ali » Update" and the title "Update Children 1". A note states "Fields with * are required." The form contains the following fields: "Name *" with the value "ali"; "Photo *" with a "Choose File" button and "No file chosen" text; "Age *" with the value "2"; and "Gender" with radio buttons for "Male" (selected) and "Female". A blue "Save" button is positioned at the bottom of the form.

Figure 53. "Update Child" page

Finally, when the parent clicks on "Add Child" operation within the "Manage Children" page, s/he will be redirected to "Add Child" page as shown in figure 54. It is divided into a form and operations section. By filling this form and clicking on "Create" button, if all the fields are filled, the new child will be added to the system and the parent will be redirected to "Manage Children" page; otherwise, the system will show a message that says the parent should fill all the fields and then submit the form again.

Figure 54. "Add Child" page

4.1.3.2 "Children Activity" navigation item

This navigation item is the same as "Children Activity" navigation item which is described under subsection 4.1.2.2.

4.1.3.3 "Request Assist" navigation item

This navigation item will redirect the parents to the "Request Assist" page as shown in figure 55. This page contains a form which the parent after selecting a child and a game from the dropdown list and filling the message text area and clicking on "Create" Button can request for an assist request from experts. If all the fields were filled, a new assist request will be added to the system and the page will be refreshed; otherwise, the system will show a message that says the user should fill all the fields and then submit the form again.

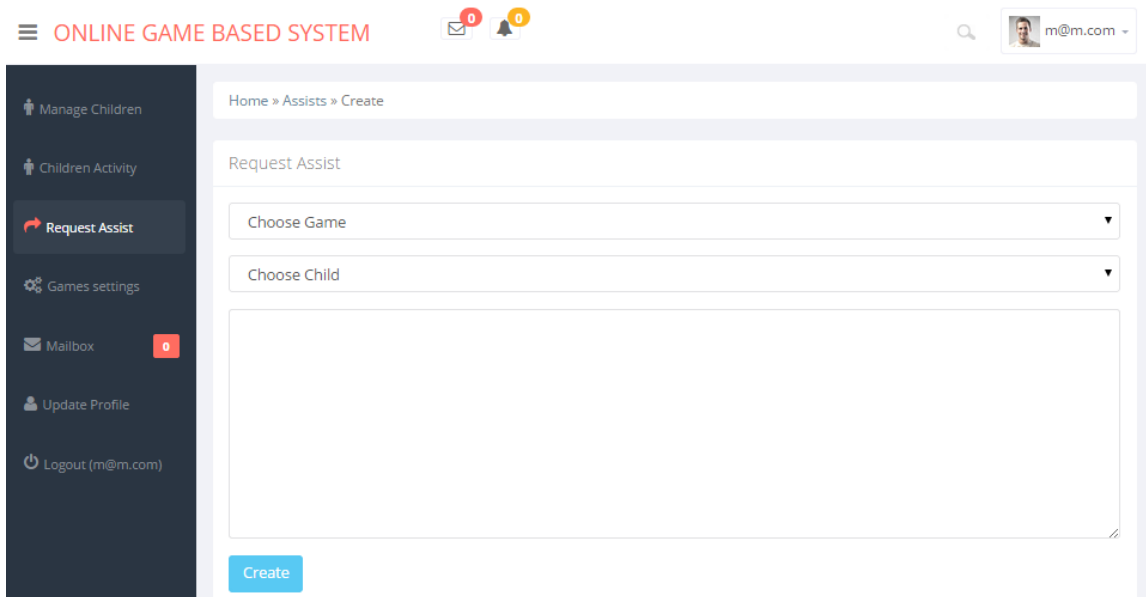


Figure 55. "Request Assist" page

4.1.3.4 "Games Settings" navigation item

"Games Settings" navigation item will redirect the parent to the "Games Settings" page as shown in figure 56. The parent after filling the "Limit playing" field and the checkbox should click on "Save" button in order to set a limit of playing time and accessing the games of children. If the parent leaves the "Limit playing" field blank and clicks on "Save" button, the system will consider as there is no limit of playing for the kids. After successful submission of the form, the page will be reloaded with the updated values.

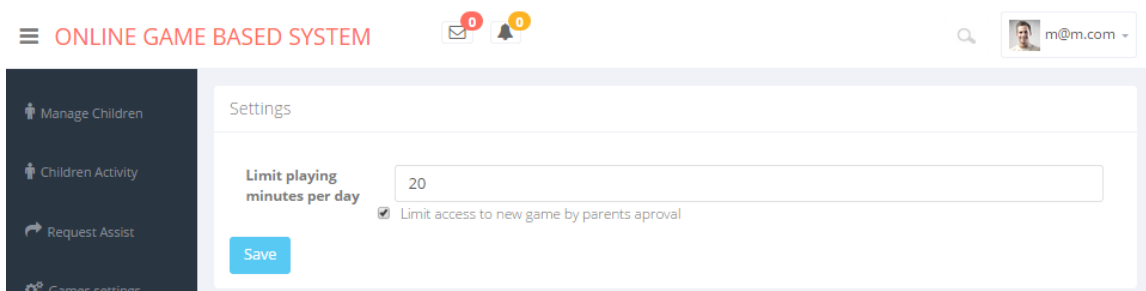


Figure 56. "Games Settings" page

4.1.3.5 "Mailbox" navigation item

This navigation item is the same as "Mailbox" navigation item which is described under subsection 4.1.1.6.

4.1.3.6 "Logout" navigation item

This navigation item is the same as "Logout" navigation item which is described under subsection 4.1.1.8.

4.2 Game module

Game module is a set of mGBL games which are developed by third-party developers. In the middleware layer, there is our RESTful Web service which developers can integrate it to their games. Set of the games which are integrated to our system are considered as game module.

So that, there is not a particular implementation phase for these games and each game has its own implementation phase. As an example, we had implemented two games in order to demonstrate our system. Below, the implementation phases of these demo games are given in details.

Third-party developers can use the following details beside of the description we gave on subsection 3.3 in order to integrate their games with our system.

Main screen of our sample games is the "Login" page that the parents can log in to the system by typing their username and password and clicking on login button as shown in figure 57. The login procedure is required just for the first time and after one successful login, username and password of the parent will be saved on the system and upon next run of the game, the parent will be directly redirected to "Children" page.

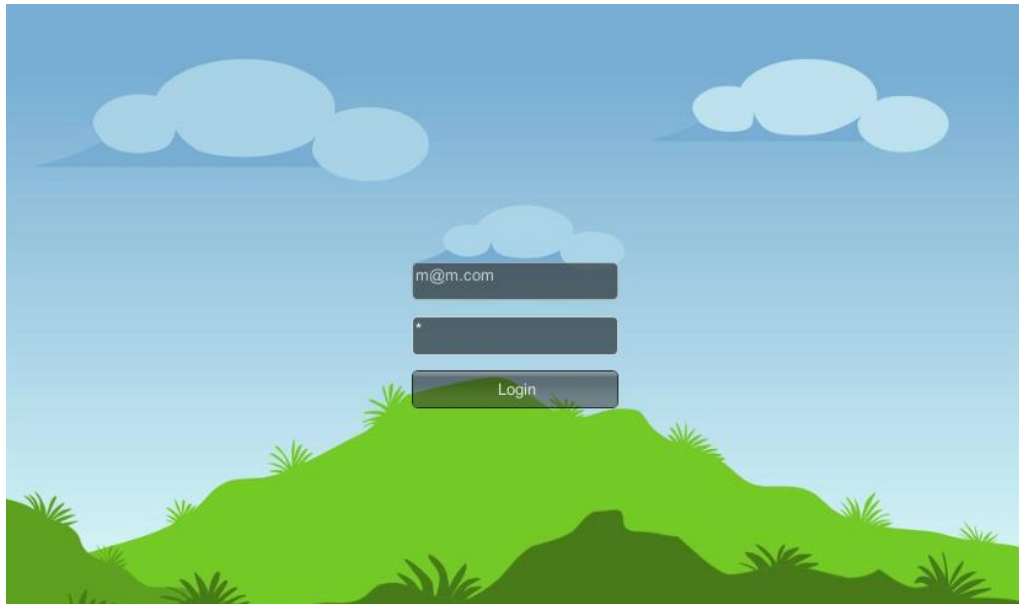


Figure 57. "Login" page

As we mentioned, after successful login, the parents will be redirected to "Children" page as shown in figure 58. This page contains the photos of each registered child of the logged-in parent. From this point, child will be redirected to "Games" page by clicking on his/her photo as shown in figure 59.

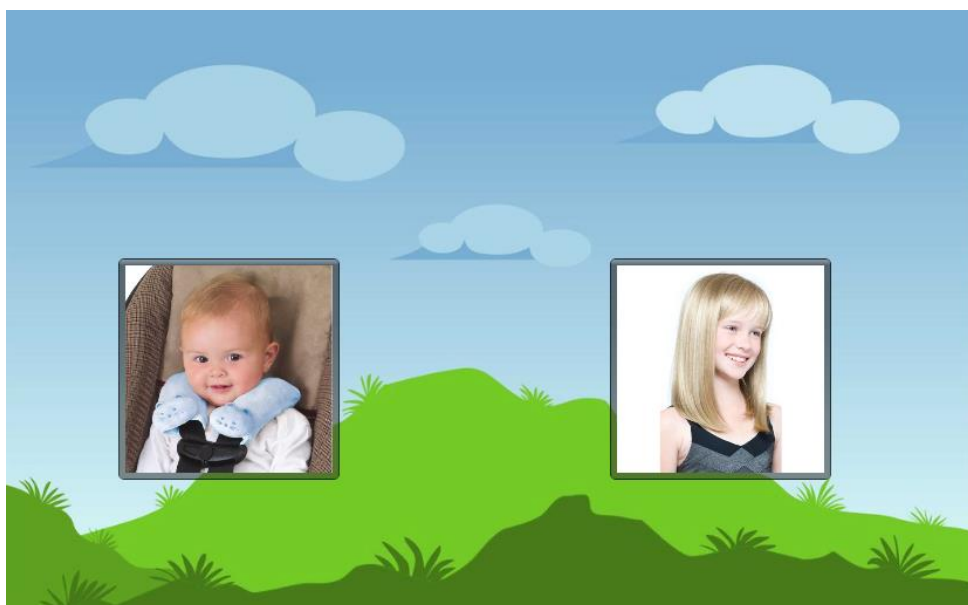


Figure 58. "Children" page

On the "Games" page, children by clicking on each game will be redirected to the "Playing game" page in order to start playing a game which is shown in figure 59.



Figure 59. "Games" page

From this point on, all the activities of the children will be sent to the system database through the Internet.

One of our sample games is "choose the correct animal". "Game playing" page of this game is shown in figure 60. This game is teaching the names of the animals to children. First, a random animal name sound will be played and children should choose the correct animal. By tapping on a wrong animal, the game will play a sound which shows the selected answer is incorrect and until the player does not tap on the correct answer, s/he will not be able to go to the next level. When the child taps on the correct answer, s/he will be redirected to next level of the game and so on. This game has 5 levels and each level is harder than the previous one. All the answers and activities of the children will be monitored by the system through the Internet.

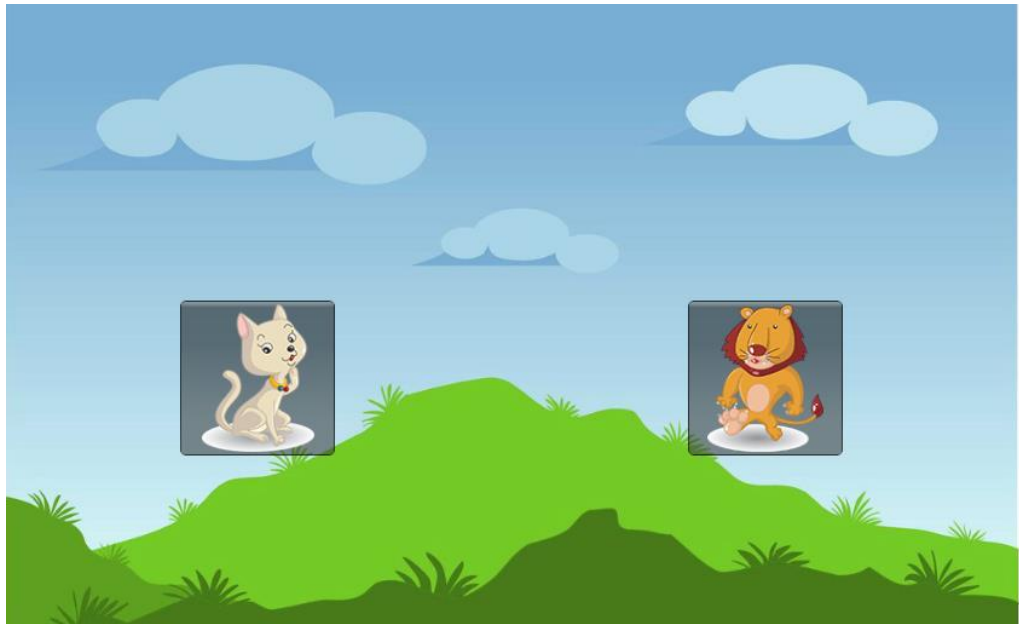


Figure 60. "Playing game" page

Our other sample game is "choose the correct color". This game is teaching the names of the colors to children. First, a random color name sound will be played and children should choose the correct color. By tapping on a wrong color, the game will play a sound which shows the selected answer is incorrect and until the player does not tap on the correct answer, s/he will not be able to go to the next level. When the child taps on the correct answer, s/he will be redirected to next level of the game and so on. This game has 5 levels and each level is harder than the previous one.

Chapter 5

COMPARISON WITH RELATED WORKS

When our system is considered, there are no other academic based practical systems encountered. However, there are some commercial systems as game-based learning systems for pre-school children. Some of the most popular ones of these are discussed below.

The first one is "Leo's Pad" which is an iPad application made by Kidaptive, Inc. [28], which consists of a series of interactive games aimed at supporting metacognitive development in pre-school learners. Additionally, the application uses embedded assessments to customize learning. This app also provides progress report and skill development report of the children to parents. "Leo's Pad" application is shown in figure 61.



Figure 61 . "Leo's Pad" application [28]

The second one is "Fit Brains. Sparky's Adventures!" application made by Rosetta Stone Inc. [29] which is a timed game that helps improve the children concentration, problem solving, memory, and visuals. There are mini-games in each of the four areas mentioned. Additionally, it keeps track of their children progress. "Fit Brains. Sparky's Adventures! " Application is shown in figure 62.



Figure 62. "Fit Brains. Sparky's Adventures!" application [29]

The third application called "Learn With Homer" [30] is a learn-to-read app for children ages 3 to 6 that incorporates drawing, voice recording, stories, songs, and more, along with more traditional phonics exercises. Parents can also make

recordings of their children during the process and measuring their progress as they move through learning. This application is shown in figure 63.



Figure 63. "Learn With Homer" application [30]

As we compare our system with the mentioned applications above, unlike the others, the dynamicity of our system that allows the third-party developers to create and integrate their games to our system is the first advantage of our system.

Also all of the mentioned applications were just focused on evaluation of the player performance. Our system not only focused on the evaluation of the players' performance but also focused on the evaluation of the games. Generated reports from

these evaluations aim us to develop games with more quality in contents and learning materials in order to improve the knowledge and/or skill development of pre-school children.

In addition, unlike the other mentioned applications, our system includes an internal messaging system which connects the parents, experts and administrators with each other.

Chapter 6

CONCLUSION

In our study, we faced many general challenges in order to improve the mGBL systems for pre-school children. Some of these challenges are given as list below:

- Design and implementation of attractive content-rich educational games which motivate the children to play;
- Selection of an appropriate evaluation method to use in games evaluation process;
- Selection of an appropriate evaluation method to use in children evaluation process;
- Selection of an appropriate way of experts' involvement in the processes of evaluation of games and children;
- Improvement of the parents' involvement in children playing process;
- Improvement of the communication between parents and experts;

In our study, an online game-based learning system is developed as an mGBL system for pre-school children by also covering these challenges. Some of the main benefits of our system are as follows:

- The system contains an online evaluation tool that can be used for evaluating the usability of the mGBL games to develop more effective mGBL games. Evaluation can be done over Internet. This will help experts and game developers to shorten the evaluation time because gathering experts to

conduct an evaluation process are not an easy task. Also, the evaluation reports generate automatically by the system in order to reduce time spend for identification of usability problems,

- The system contains expert support service which provides expert's assistance to the parents through observation of the children activities by experts and giving feedback to parents,
- The system not only offers detailed tools and charts to track children skill development and their interests, but also provides recommendations and guidelines to parents to develop the children knowledge/skill,
- In order to overcome the difficulties associated with communication between parents and experts, an internal messaging system is included. Parents can directly communicate with the experts and experts can communicate the requirements and instructions clearly to the parents,
- Evaluation methods of the system are derived from academic-based researches which improves the reliability of the study,
- The system provides the ability of limiting daily playing time and limiting the access of children to new games by parent's approval to control the children mobile game playing,
- Our system provides a Web service which other third-party developers can use it in order to integrate their games into our system. Having third-party game development facility allows our system to expand the variety of games in itself,
- The unity game engine is used as our game engine. The unity game engine supports many operating systems which are not only the operating systems

that used by tablets and smartphones. Therefore, other mobile devices and desktop computers can also be used in our system.

We would like to bring your notice that we did not encounter any other system which covers all the features covered by ours.

A number of future considerations can be suggested for this study. The experimental study could be done on this project for testing the system and making fine-tuning over it; Pre-production phase evaluation and production phase evaluation of mGBL also required to be focused on [16, 31, 32]; and other mGBL evaluation methods based on user success rate [33] and/or observation method [34] could be included in the system.

REFERENCES

- [1] comScore (2012). 5 Years Later. A Look Back at the Rise of the iPhone. [Online]. Available: <http://www.comscore.com/ita/Insights/Blog/5-Years-Later-A-Look-Back-at-the-Rise-of-the-iPhone>. [Accessed January 2015].
- [2] comScore (2014). Reports October 2014 U.S. Smartphone Subscriber Market Share. [Online]. Available: <http://www.comscore.com/Insights/Market-Rankings/comScore-Reports-October-2014-US-Smartphone-Subscriber-Market-Share> [Accessed January 2015].
- [3] GSMA Intelligence (2014). Smartphone forecasts and assumptions. 2007-2020. [Online]. Available: <http://www.gsma.com/newsroom/press-release/smartphones-account-two-thirds-worlds-mobile-market-2020/> [Accessed January 2015].
- [4] Dediu, Horace (2014). When will smartphones saturate? [Online]. Available: <http://www.asymco.com/2014/01/07/when-will-smartphones-saturate/> [Accessed January 2015].
- [5] Pew Internet Report (2014). E-Reading Rises as Device Ownership Jumps. [Online]. Available: <http://www.pewInternet.org/2014/01/16/e-reading-rises-as-device-ownership-jumps/> [Accessed January 2015].
- [6] Common Sense Media (2013). Zero to eight. Children's media use in America. [Online]. Available: <https://www.commonsensemedia.org/research/zero-to-eight-childrens-media-use-in-america-2013> [Accessed January 2015].

[7] National Associations for the Education of Young Children (2012). Technology and Interactive Media as Tools in Early Childhood Programs Serving Children from Birth Through Age 8. [Online]. Available:

http://www.naeyc.org/files/naeyc/file/positions/PS_technology_WEB2.pdf

[Accessed January 2015].

[8] Nielsen (2012). American Families See Tablets as Playmate, Teacher and Babysitter.

[Online]. Available: <http://www.nielsen.com/us/en/insights/news/2012/american-families-see-tablets-as-playmate-teacher-and-babysitter.html> [Accessed January 2015].

[9] C. Shuler (2012). iLearn II. An analysis of the education category of Apple's Appstore: The Joan Ganz Cooney Center at Sesame Workshop.

[10] F. Bellotti, B. Kapralos, K. Lee, P. Moreno-Ger & R. Berta (2013). Assessment in and of Serious Games. An Overview. *Hindawi Advances in Human-Computer Interaction*.

[11] S. Chen & D. Michael (2006). Serious Games. Games that Educate, Train and Inform. *Boston. Thomson Course Technology PTR*.

[12] T. Susi, M. Johannesson, & P. Backlund (2005). Serious Games - An Overview. *Technical Report*, School of Humanities and Informatics, University of Skovde, Sweden.

- [13] J. Nielsen (1993). Usability Engineering. *Academic Press*, Cambridge, MA.
- [14] J. Nielsen (1994). Heuristic evaluation. *J. Nielsen, R.L. Mack (Eds.)*, Usability Inspection Methods, Wiley, New York, pp. 25–62.
- [15] H. Mohamed & A. Jaafar (2010). Development and potential analysis of Heuristic Evaluation for Educational Computer Game (PHEG). *Computer Sciences and Convergence Information Technology (ICCIT)*, 2010 5th International Conference on, 2010, pp. 222-227.
- [16] L. Shoukry & C. Sturm (2015). Pre-MEGa. A Proposed Framework for the Design and Evaluation of Preschoolers' Mobile Educational Games. *Innovations and Advances in Computing, Informatics, Systems Sciences, Networking and Engineering*, Springer International Publishing.
- [17] M. González (2009). Quantitative analysis in a heuristic evaluation experiment, *Advances in Engineering Software* vol. 40, pp. 1271-1278.
- [18] T. Connolly, M. Stansfield & T. Hainey (2008). Development of a general framework for evaluating games-based learning, *2nd European conference on games-based learning*, Universitat Oberta de Catalunya, Barcelona, Spain.
- [19] R. Oshana & M. Kraeling (2013). Software engineering for embedded systems. Methods, practical techniques, and applications. Newnes, an imprint of Elsevier Science and Technology Books.

- [20] Yii Software LLC (2014). Yii Framework. [Online]. Available:
<http://www.yiiframework.com/> [Accessed January 2015].
- [21] w3schools (2013), PHP MySQL Introduction. [Online]. Available:
http://www.w3schools.com/php/php_mysql_intro.asp [Accessed January 2015].
- [22] Yii Software LLC (2014). Model-View-Controller (MVC). [Online]. Available:
<http://www.yiiframework.com/doc/guide/1.1/en/basics.mvc> [Accessed January 2015].
- [23] Elkstein (2012). Learn Rest. A tutorial. [Online]. Available:
<http://rest.elkstein.org/> [Accessed January 2015].
- [24] S. Blackman (2011). Beginning 3D Game Development with Unity. All-in-one, multiplatform game development. Apress; 1 edition.
- [25] Oracle Corporation (2014). MySQL. [Online]. Available:
<http://www.mysql.com/> [Accessed January 2015].
- [26] Yii Software LLC (2014). RESTFullyii extension. [Online]. Available.:
<http://www.yiiframework.com/extension/restfullyii/> [Accessed January 2015].
- [27] Unity Technologies (2014). Unity 3d game engine.. [Online]. Available:
<http://unity3d.com/> [Accessed January 2015].

- [28] Kidaptive, Inc. (2014). Leo's Pad. [Online]. Available: <http://kidaptive.com/leos-pad/> [Accessed January 2015].
- [29] Rosetta Stone Inc. (2014). Fit Brains. Sparky's Adventures!. [Online]. Available: <http://www.fitbrains.com/blog/fit-brains-kids/> [Accessed January 2015].
- [30] HomerLearning Inc. (2014). Learn With Homer. [Online]. Available: <https://learnwithhomer.com/> [Accessed January 2015].
- [31] S. Zabon & S. Shiratuddin (2010). Mobile game-based learning (mGBL) engineering model as a systematic development approach. *Proceedings of Global Learn Asia Pacific 2010* , pp. 1862-1871.
- [32] S. Zabon (2011). Mobile game-based learning (mGBL) engineering model. *PhD Thesis*, Universiti Utara Malaysia.
- [33] M. Ismail, N. Diah, S. Ahmad, N. Kamal & M. Dahari (2011). Measuring Usability of Educational Computer Games Based on The User Success Rate. *Humanities, Science & Engineering Research (SHUSER)*, 2011 International Symposium on, pp. 56-60.
- [34] N. Diah, M. Ismail, S. Ahmad & M. Dahari (2010). Usability testing for educational computer game using observation method. In *2010 international conference on information retrieval & knowledge management (CAMP)*, pp. 157–161.

APPENDICES

Appendix A: PHEG Sub Heuristics

Table A.1. List of PHEG sub heuristics

Usability	Usability Heuristics – Interface
UI1	Uses aesthetic and minimalist design
UI2	Maximizes consistency and matches standards
UI3	The uses of space, color and text are according to the principles of screen design
UI4	The uses of text, color and font follow the principles of readability
UI5	Ensure appropriate use of standard and propriety control
UI6	The interactivity of the game is suitable to learners level
UI7	The integration of presentation means is well coordinated
UI8	Quality of user interface is acceptable
UI9	Provide specific and self- identified key for specific task (exit, glossary, main, objective)
UI10	Overall interface of the game is appealing
Usability	Usability Heuristics – Educational
ED1	Clear goal and learning objectives
ED2	The activities are interesting and engaging
ED3	The design and the contents are reliable and proven
ED4	Can be used as self- directed learning tools
ED5	Support for self- learning skills
ED6	Medium for learning by doing
ED7	Considers the individual differences
ED8	Performance should be an outcome-based
ED9	Offers the ability to select the level of difficulty in games
ED10	Overall interface of the game is appealing

Usability	Usability Heuristics – Content
CO1	Reliable content with correct flow
CO2	Clear and understandable structure of contents
CO3	Navigation of content is easy and accurate
CO4	Supporting materials are sufficient and relevant (quiz)
CO5	Materials are interesting and engaging
CO6	Players able to understand the learning goal
CO7	The content is chunk based on topic and subtopic
CO8	Major and minor topic is differentiate clearly
Usability	Usability Heuristics – Multimedia
MM1	Usage of multimedia elements are acceptable
MM2	Combination of multimedia elements are adequate
MM3	The presentation of multimedia elements are well manage
MM4	Suitability of multimedia elements for specific use
MM5	Not too many multimedia element in one screen
MM6	The use of multimedia elements support meaningfully the text provided

Appendix B. Controllers

Table B.1. Site controller

Action Name	Description
Index	Represents base HTML contents
Error	Represents error messages
Login	Represents login Web page
Logout	Handles logout procedure

Table B.2. Admin controller

Action Name	Description
Dashboard	Represents admin account homepage
View	Displays a particular model
Create	Creates a new model
Update	Updates a particular model
Admin	Manages all models
Delete	Deletes a particular model
ChartWeekly	Generates statistical report charts (weekly)
ChartMonthly	Generates statistical report charts (monthly)
ChartYearly	Generates statistical report charts (yearly)

Table B.3. Experts controller

Action Name	Description
Dashboard	Represents expert account homepage
View	Displays a particular model
Create	Creates a new model
Update	Updates a particular model
Admin	Manages all models
Delete	Deletes a particular model

Table B.4. Parents controller

Action Name	Description
Dashboard	Represents parent account homepage
View	Displays a particular model
Create	Creates a new model
Update	Updates a particular model
Admin	Manages all models
Delete	Deletes a particular model
ChartWeekly	Generates statistical report charts (weekly)
ChartMonthly	Generates statistical report charts (monthly)
ChartYearly	Generates statistical report charts (yearly)

Table B.5. Children controller

Action Name	Description
View	Displays a particular model
Create	Creates a new model
Update	Updates a particular model
Admin	Manages all models
Delete	Deletes a particular model
Dashboard	Represents child score and statistical reports homepage

Table B.6. Game controller

Action Name	Description
View	Displays a particular model
Create	Creates a new model
Update	Updates a particular model
Admin	Manages all models
Delete	Deletes a particular model
Category	Lists games categories

Table B.7. Lesson controller

Action Name	Description
View	Displays a particular model
Create	Creates a new model
Update	Updates a particular model
Admin	Manages all models
Delete	Deletes a particular model

Table B.8. Category controller

Action Name	Description
Create	Creates a new model
Update	Updates a particular model
Admin	Manages all models
Delete	Deletes a particular model

Table B.9. Tips controller

Action Name	Description
View	Displays a particular model
Create	Creates a new model
Update	Updates a particular model
Admin	Manages all models
Delete	Deletes a particular model

Table B.10. PHEG controller

Action Name	Description
View	Displays a particular model
Create	Creates a new model
Update	Updates a particular model
Admin	Manages all models
Delete	Deletes a particular model

Table B.11. Activity controller

Action Name	Description
View	Displays a particular model
Admin	Manages all models
ViewChildren	Returns list of children

Table B.12. Assist controller

Action Name	Description
Create	Creates a new model
Update	Updates a particular model
Admin	Manages all models
Delete	Deletes a particular model

Table B.13. Score controller

Action Name	Description
Create	Creates a new model

Table B.14. Session controller

Action Name	Description
Create	Creates a new model
Admin	Manages all models
Login	opens a session
Logout	closes a session

Table B.15. Settings controller

Action Name	Description
Update	Updates a particular model

Table B.16. Mailbox controller

Action Name	Description
Inbox	Lists incoming messages
Outbox	Lists outgoing messages
ViewInbox	View a particular incoming message
ViewOutbox	View a particular outgoing message
Create	Creates a new model
Delete	Deletes a particular model

Appendix C. Views

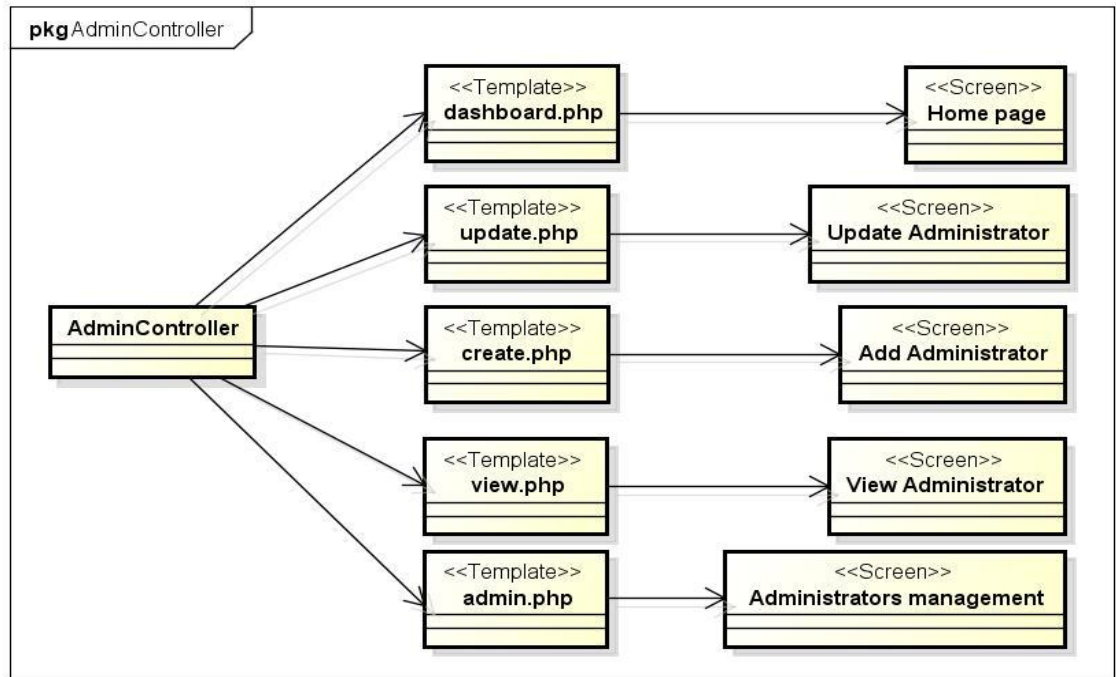


Figure C.1. AdminController views

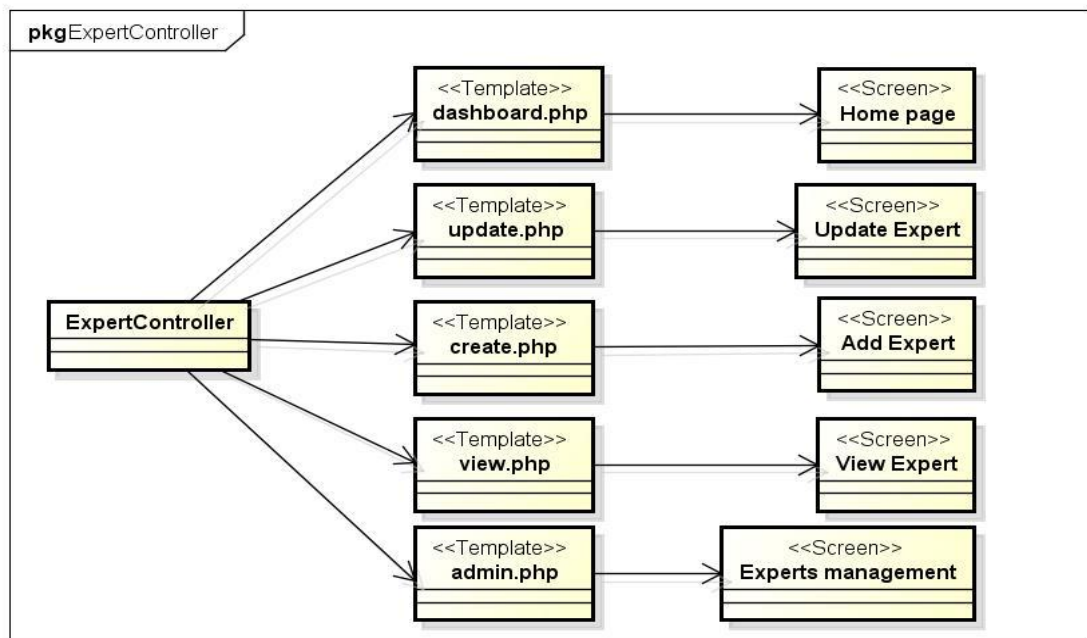


Figure C.2. ExpertsController views

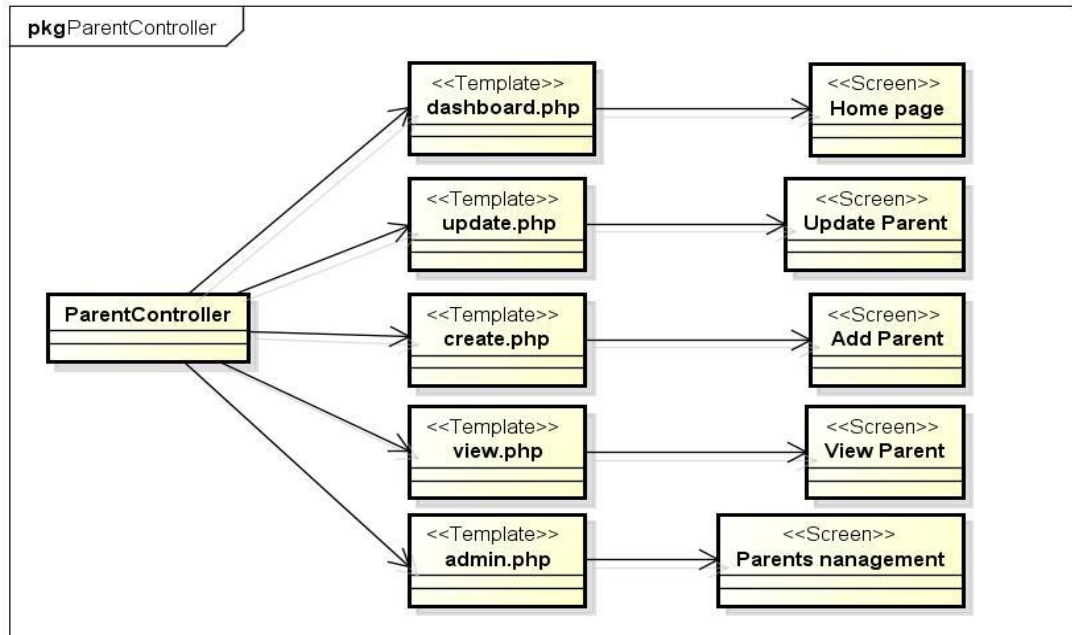


Figure C.3. ParentsController views

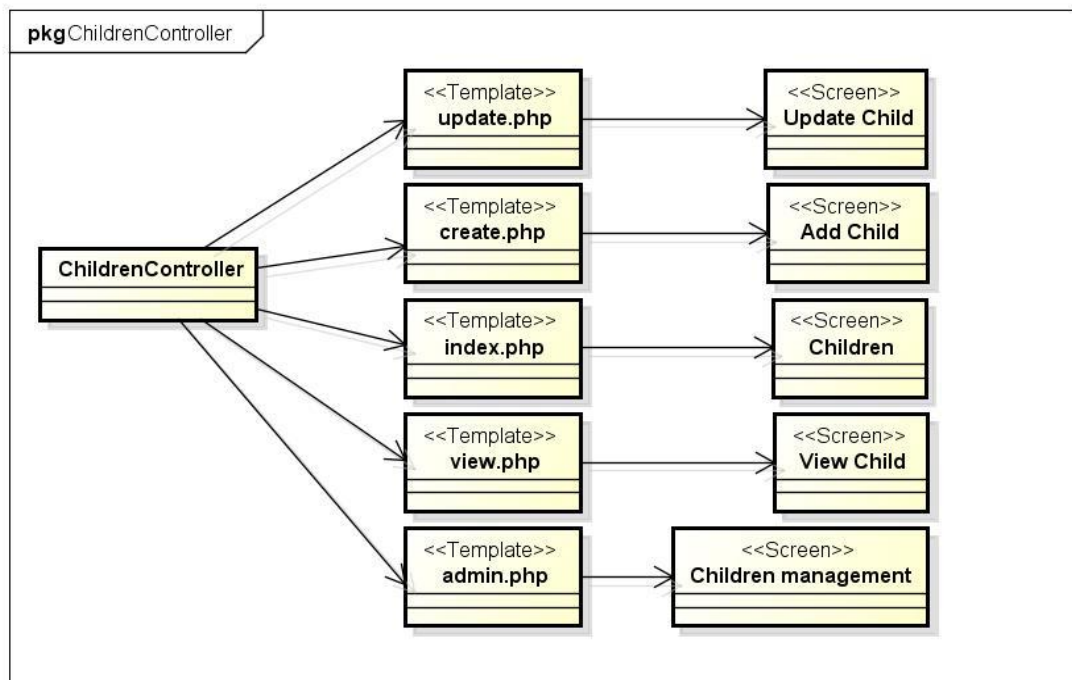


Figure C.4. ChildrenController views

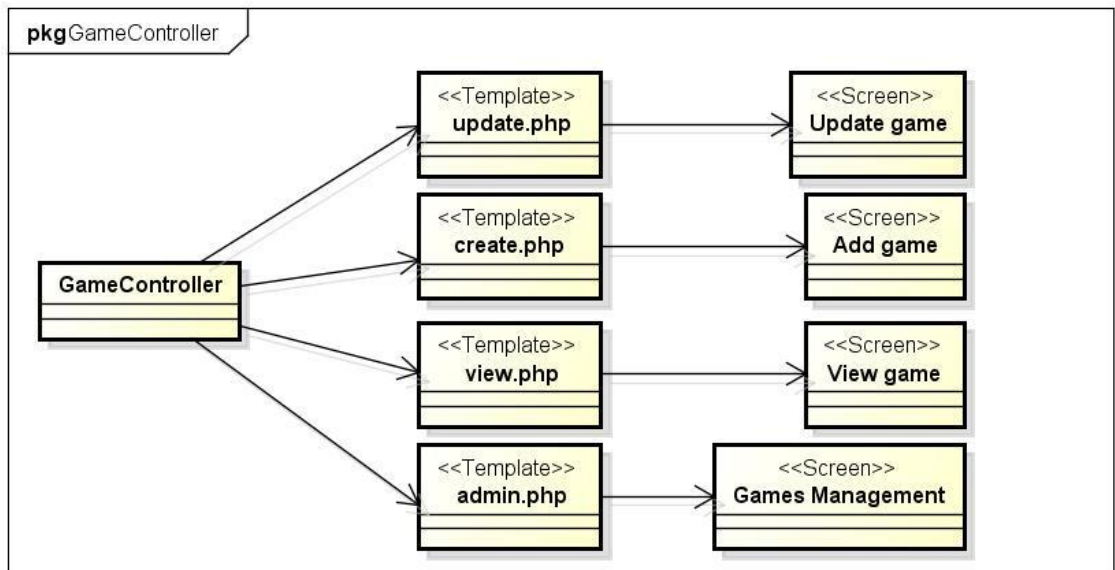


Figure C.5. GameController views

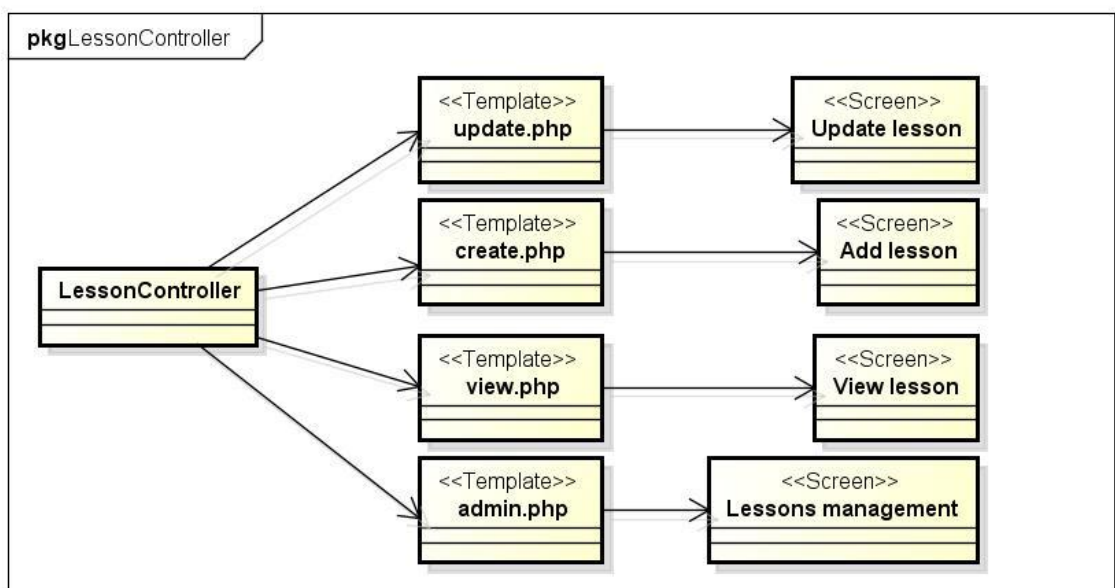


Figure C.6. LessonController views

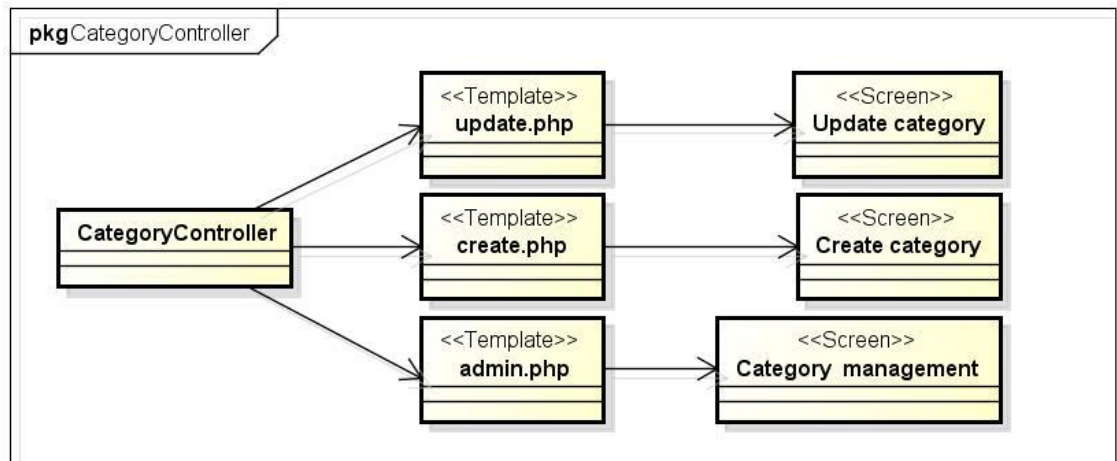


Figure C.7. CategoryController views

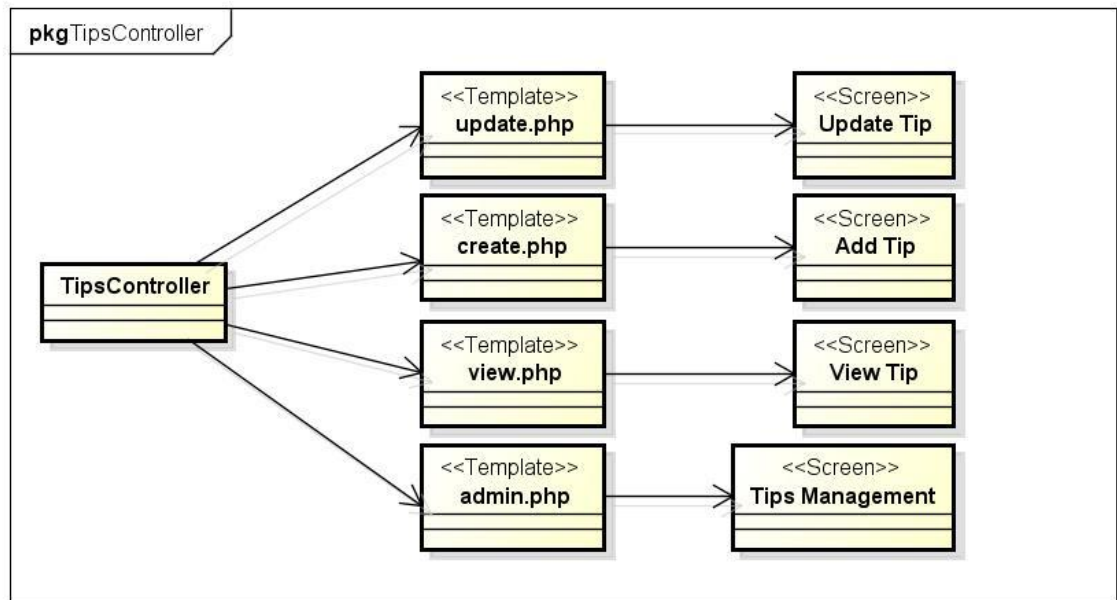


Figure C.8. TipsController views

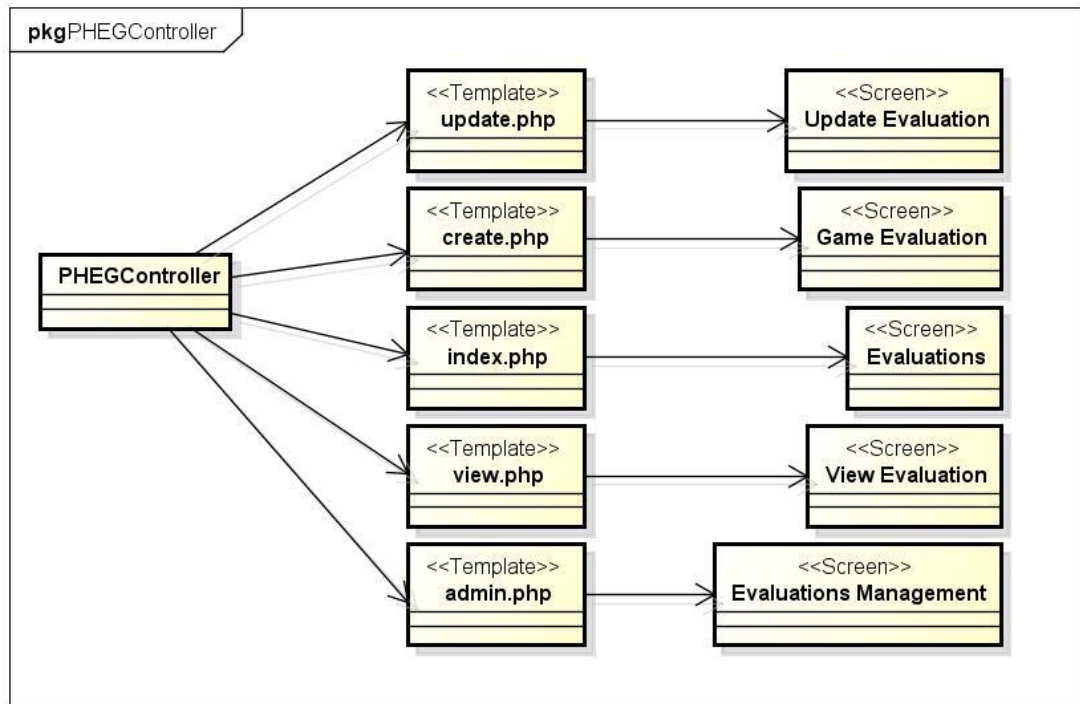


Figure C.9. PHEGController views

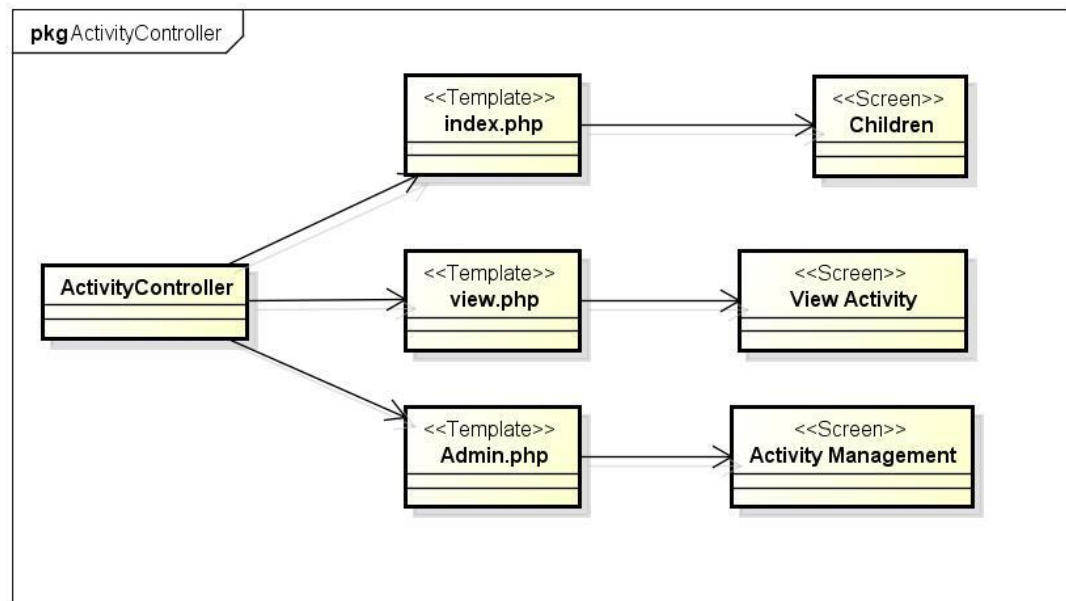


Figure C.10. ActivityController views

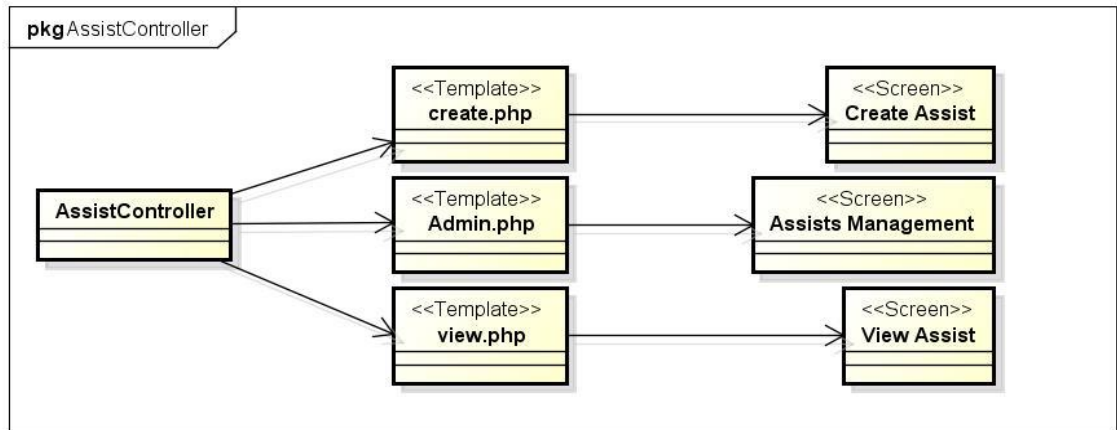


Figure C.11. AssistController views

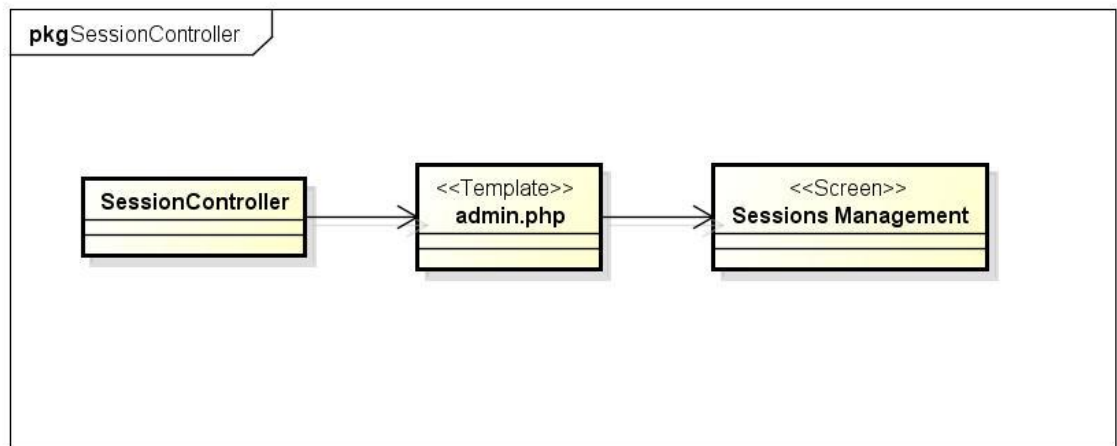


Figure C.12. SessionController views

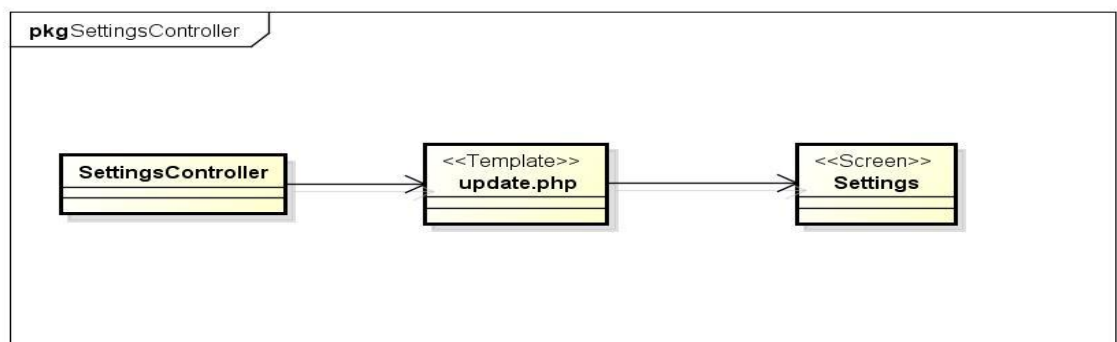


Figure C.13. SettingsController views

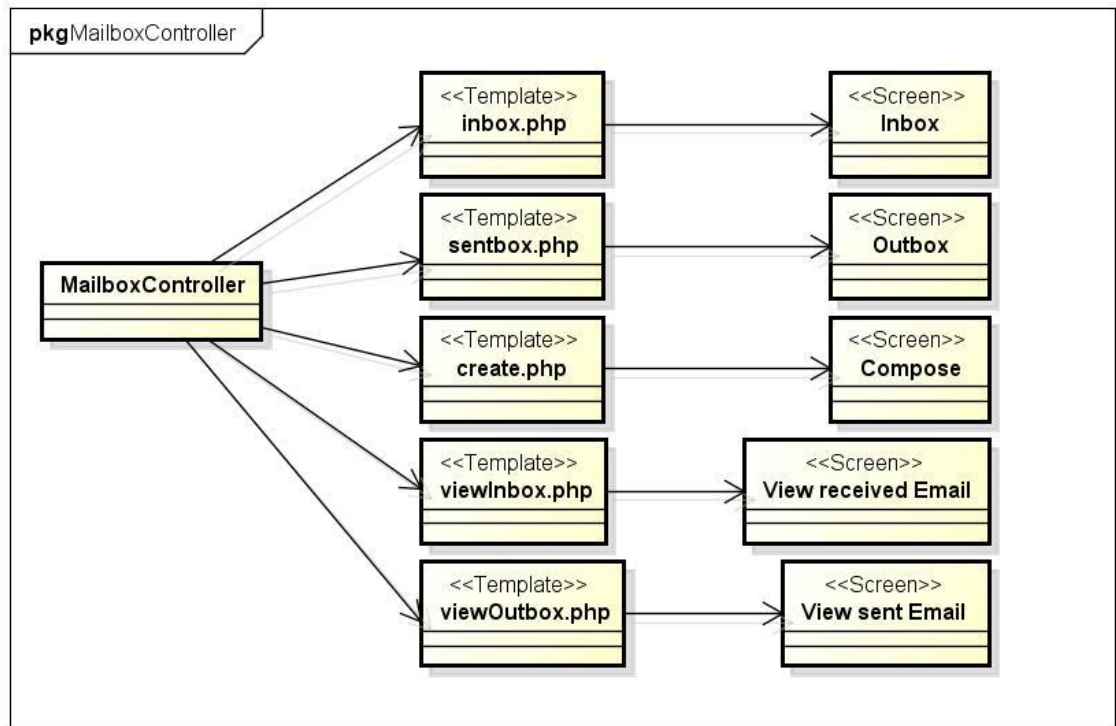


Figure C.14. MailboxController views

Appendix D. Interactions

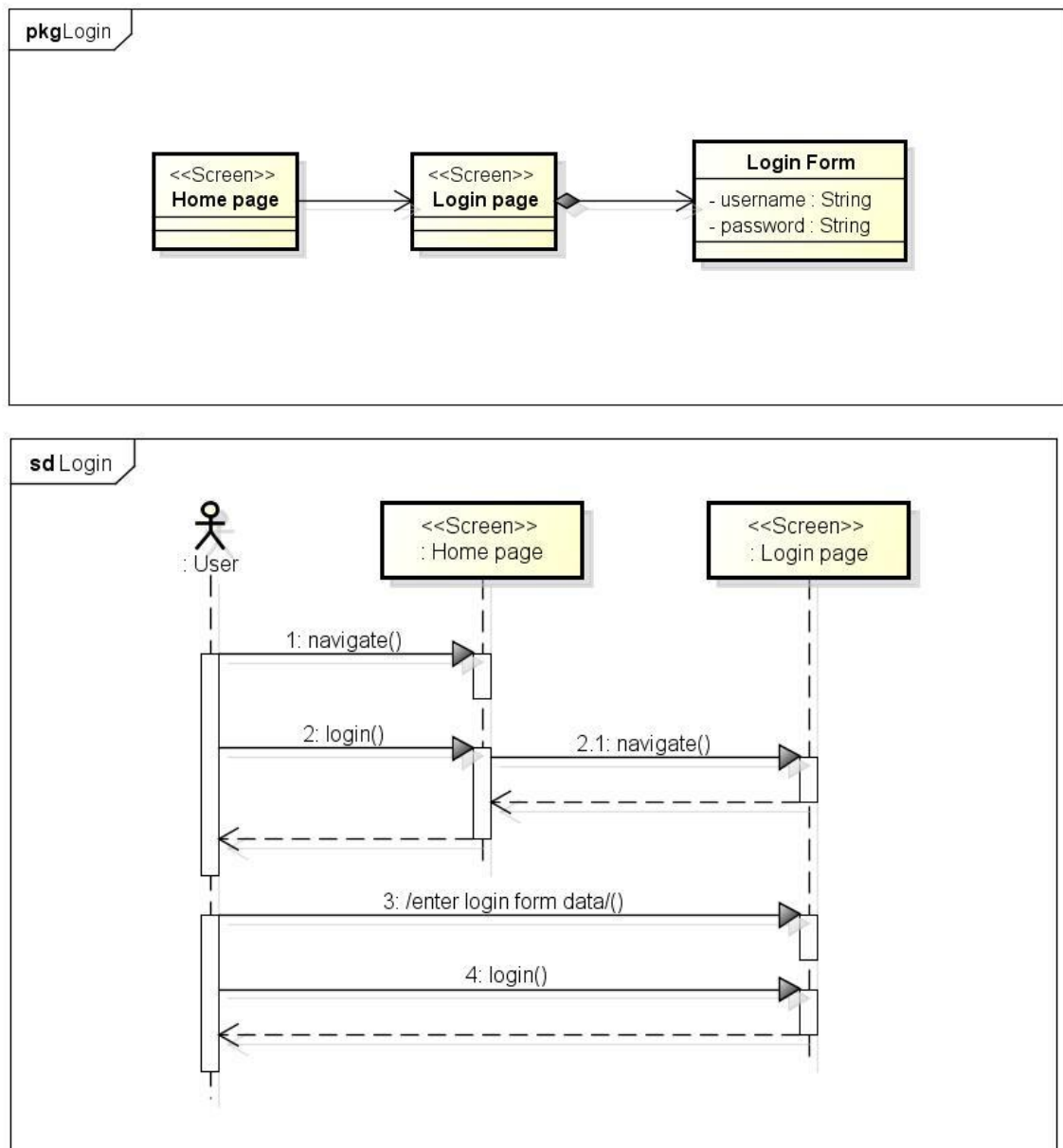


Figure D.1. Login dataflow diagram and sequence diagram

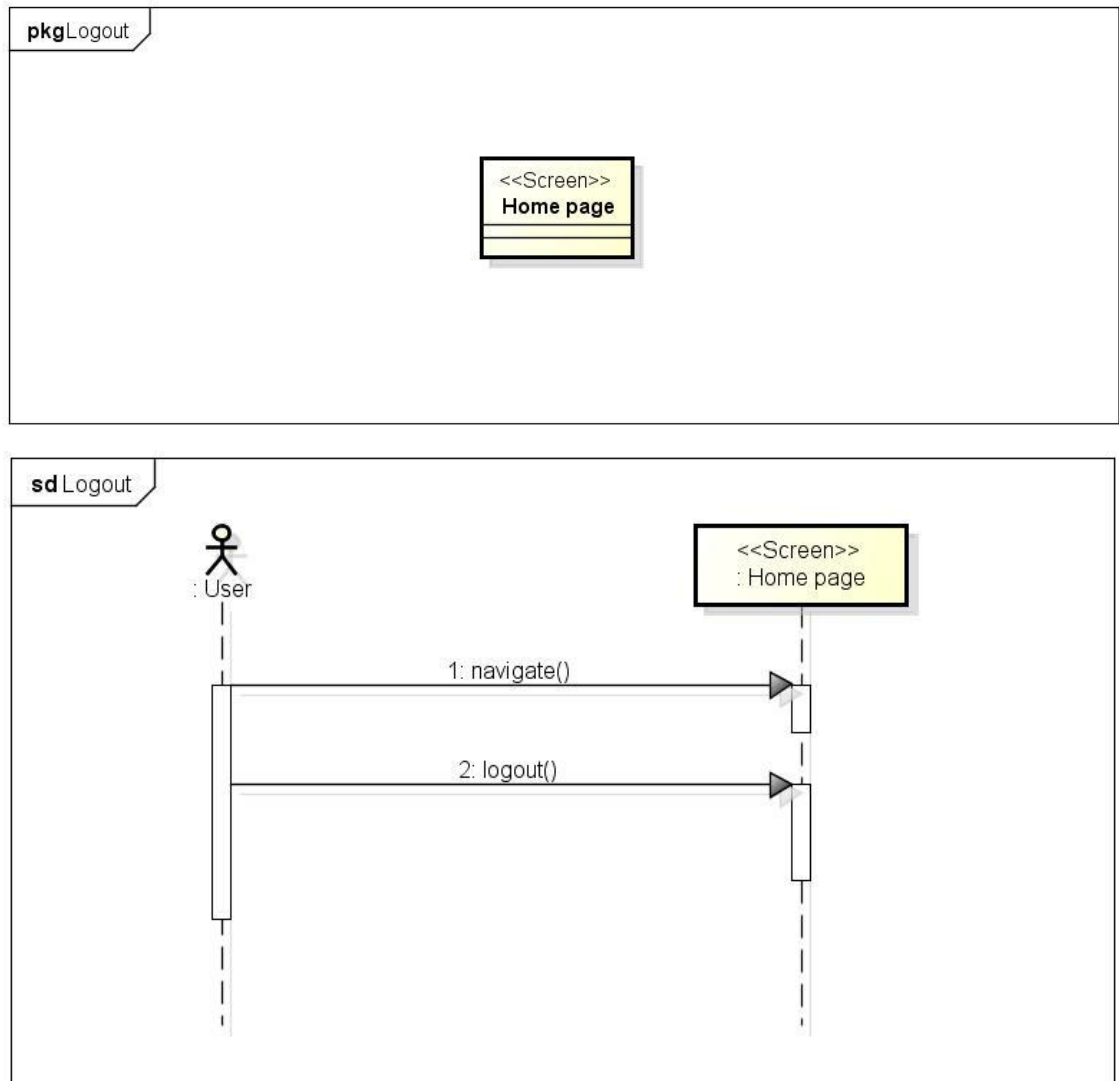


Figure D.2. Logout dataflow diagram and sequence diagram

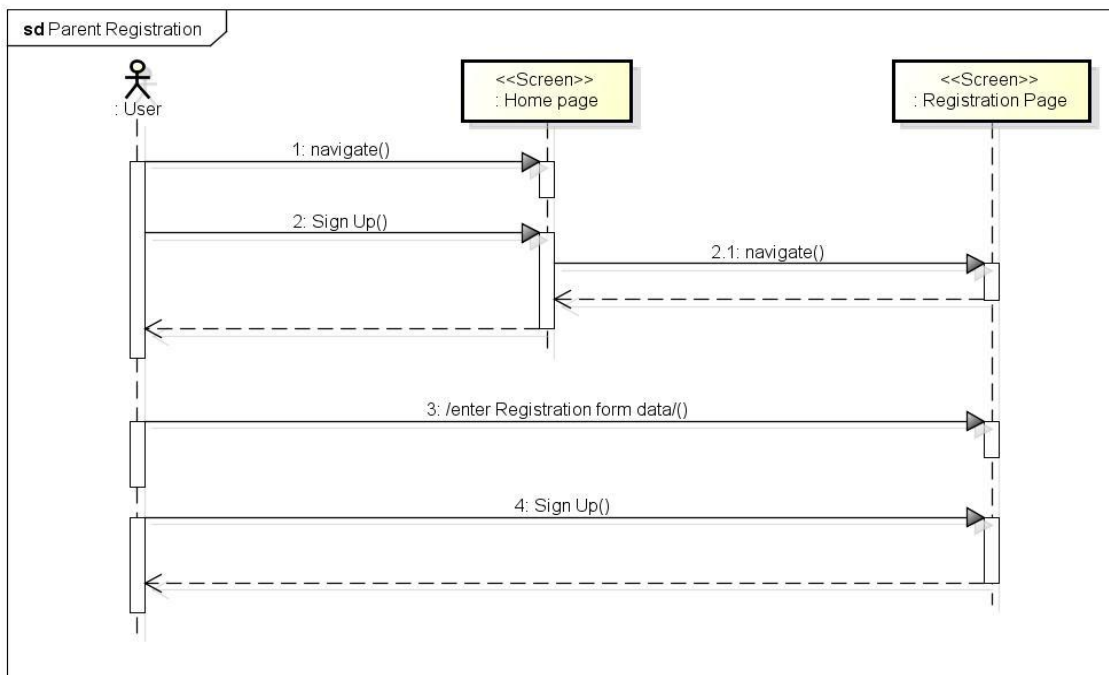
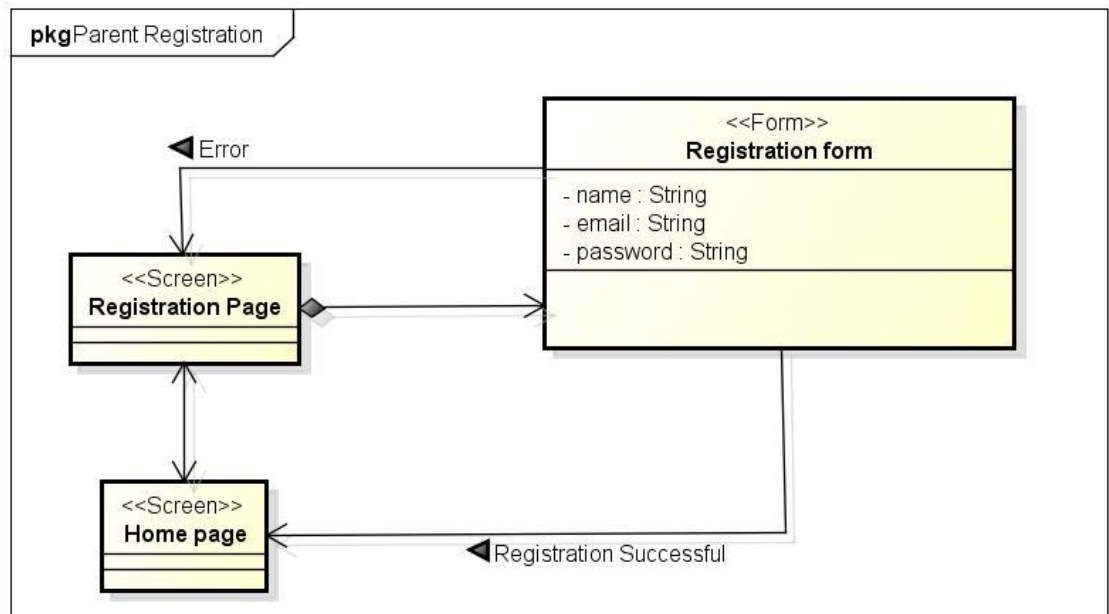


Figure D.3. Parents registration dataflow diagram and sequence diagram

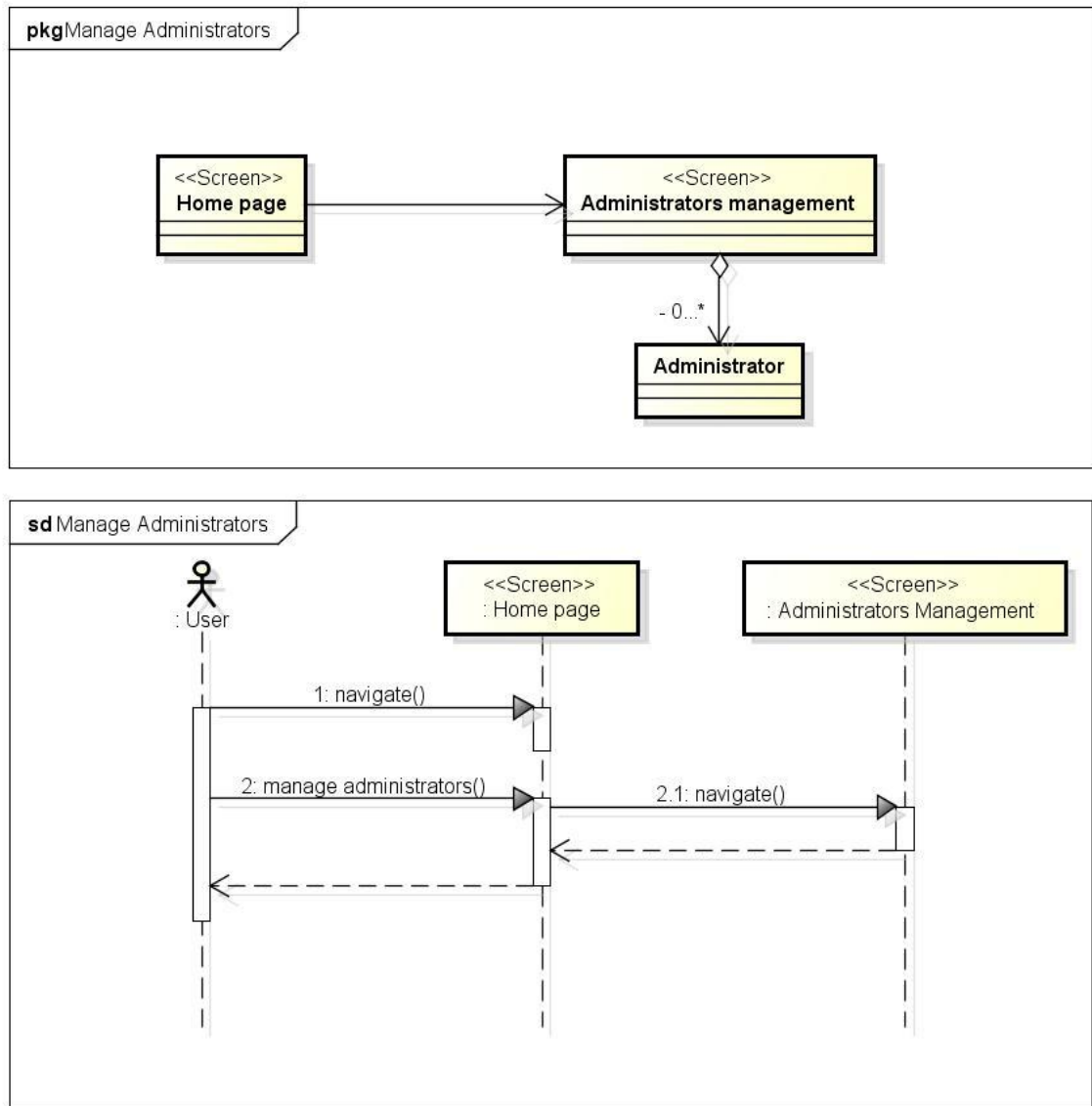


Figure C.4. Manage administrators dataflow diagram and sequence diagram

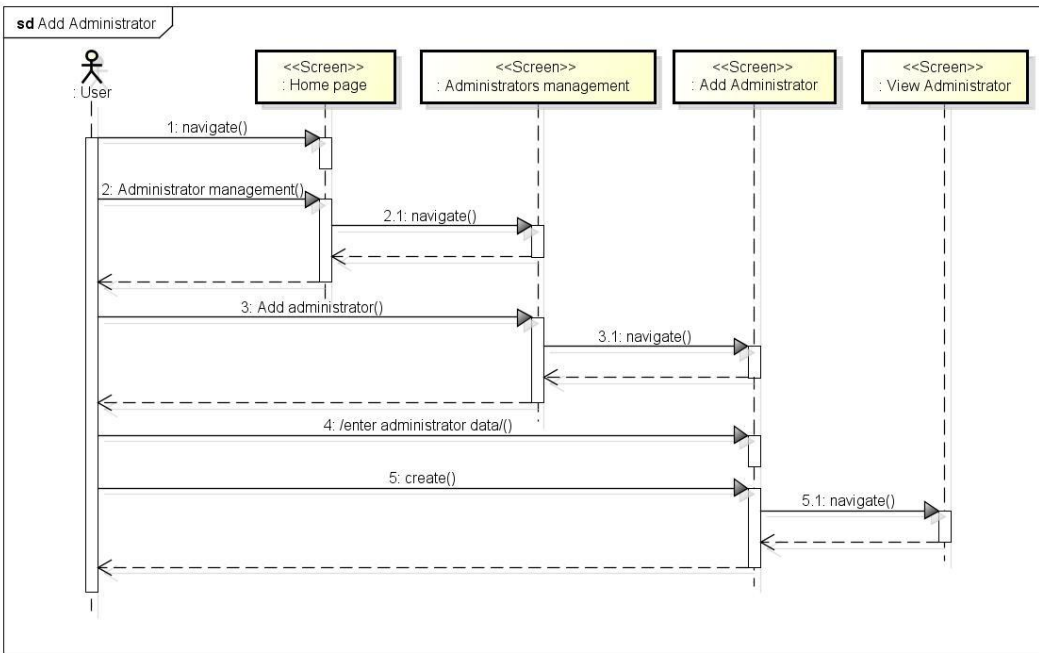
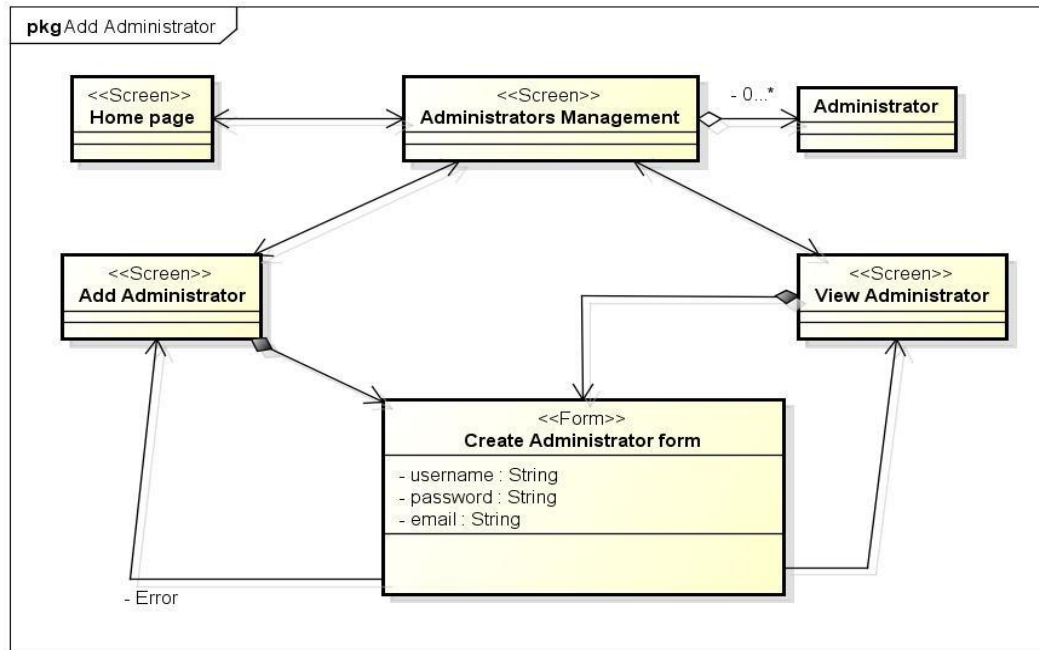


Figure D.5. Create administrator dataflow diagram and sequence diagram

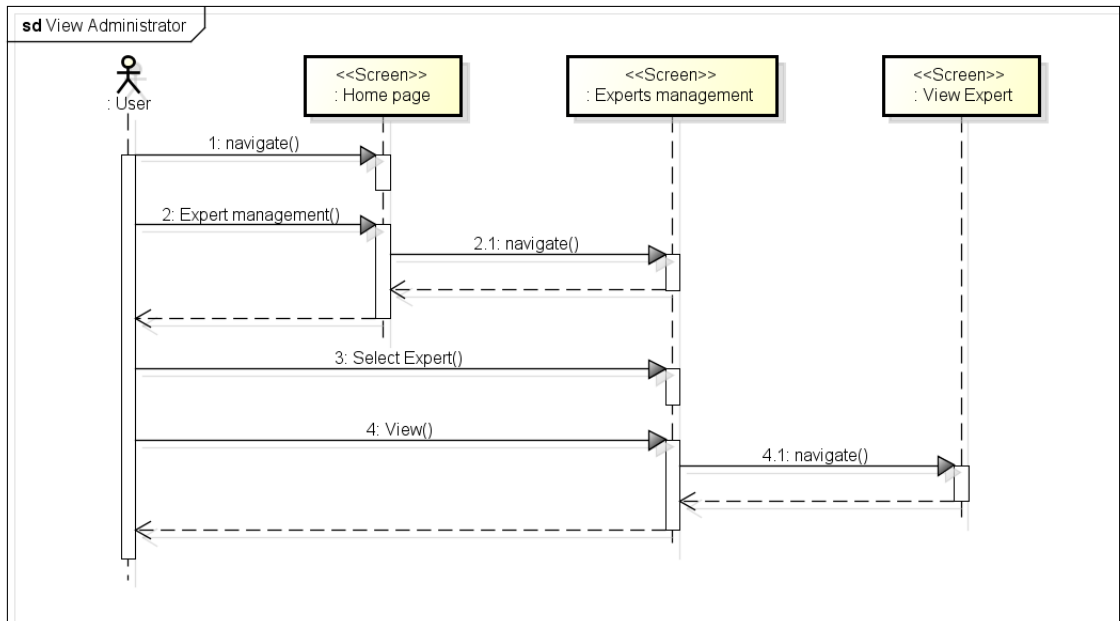
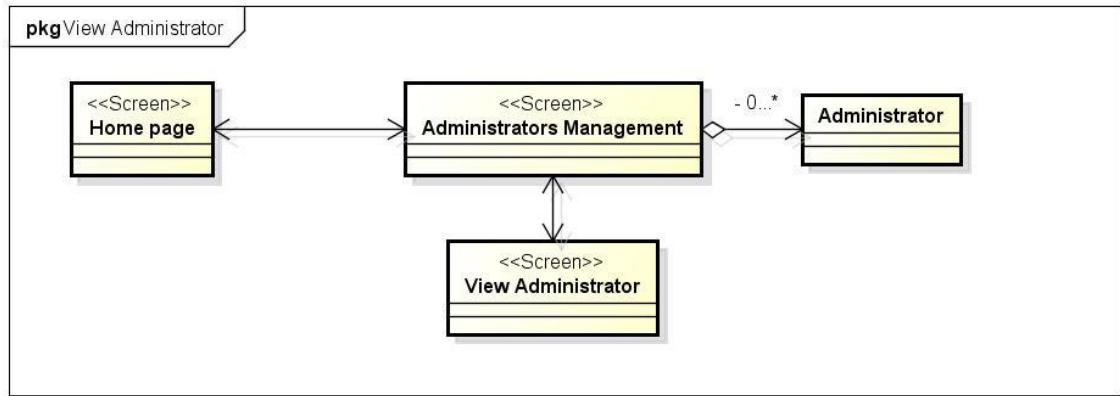


Figure D.6. View administrator dataflow diagram and sequence diagram

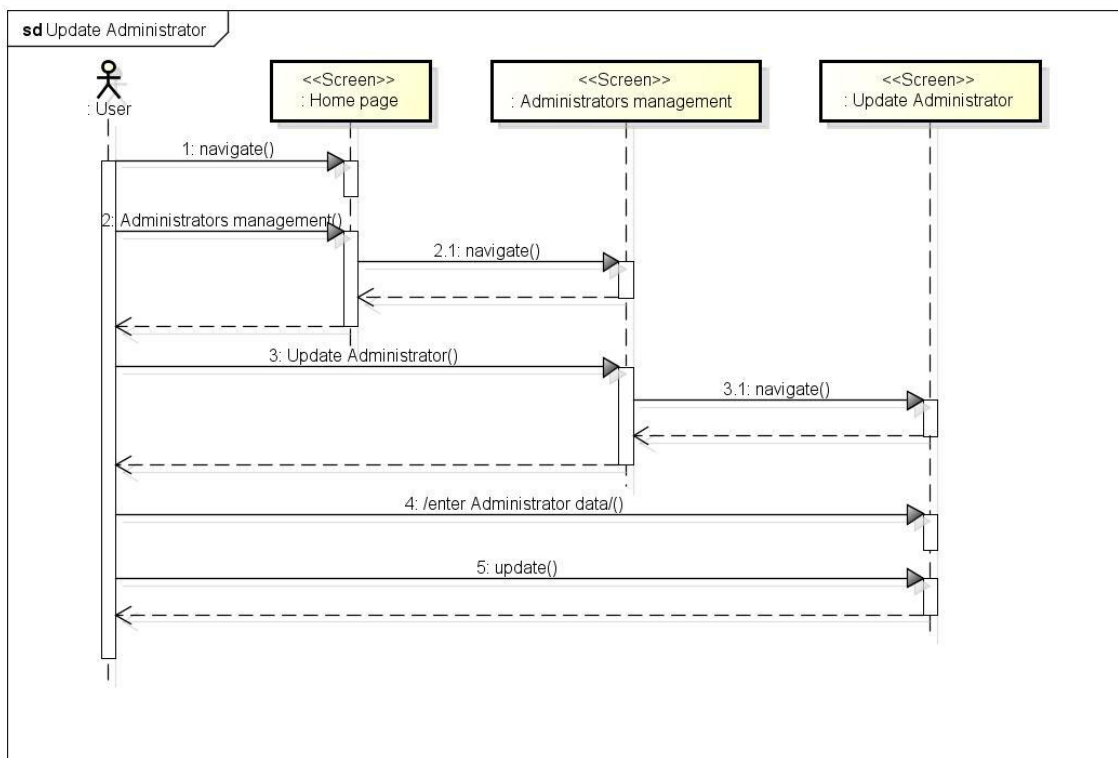
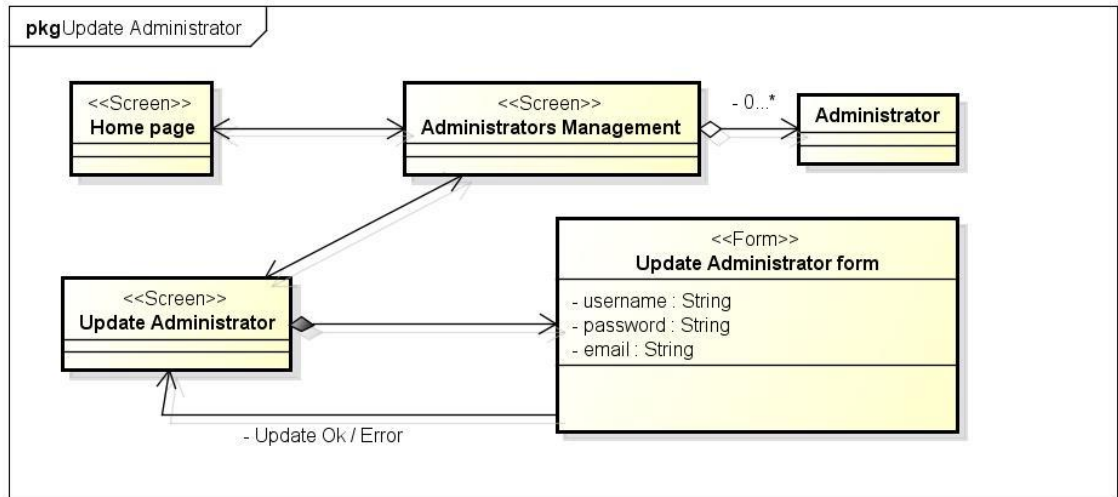


Figure D.7. Update administrator dataflow diagram and sequence diagram

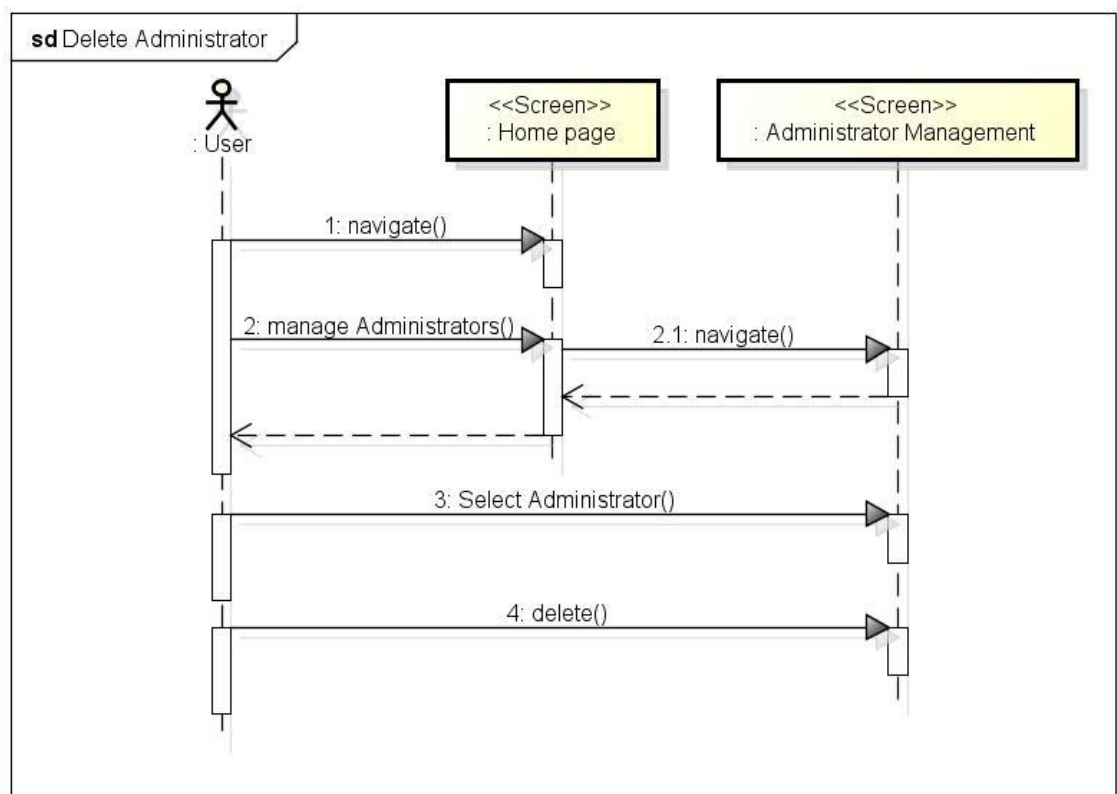
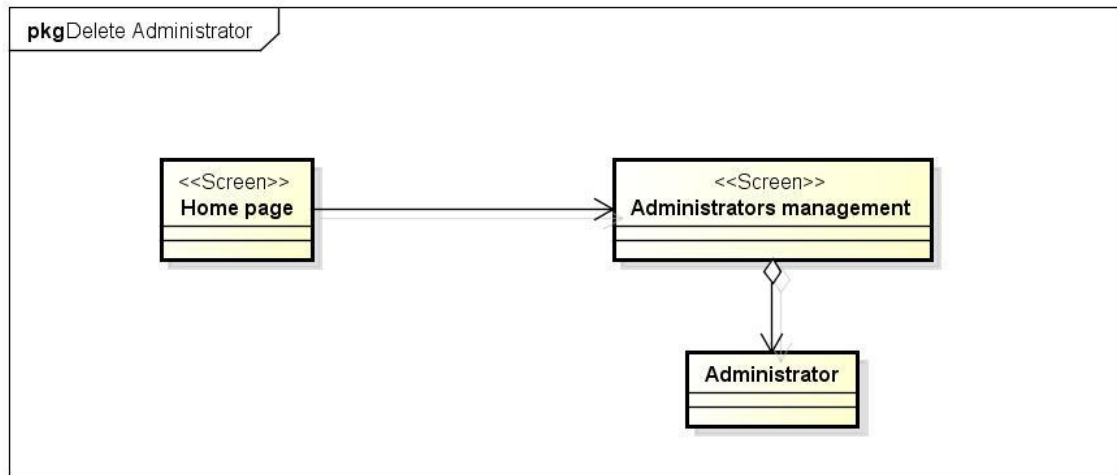


Figure D.8. Delete administrator dataflow diagram and sequence diagram

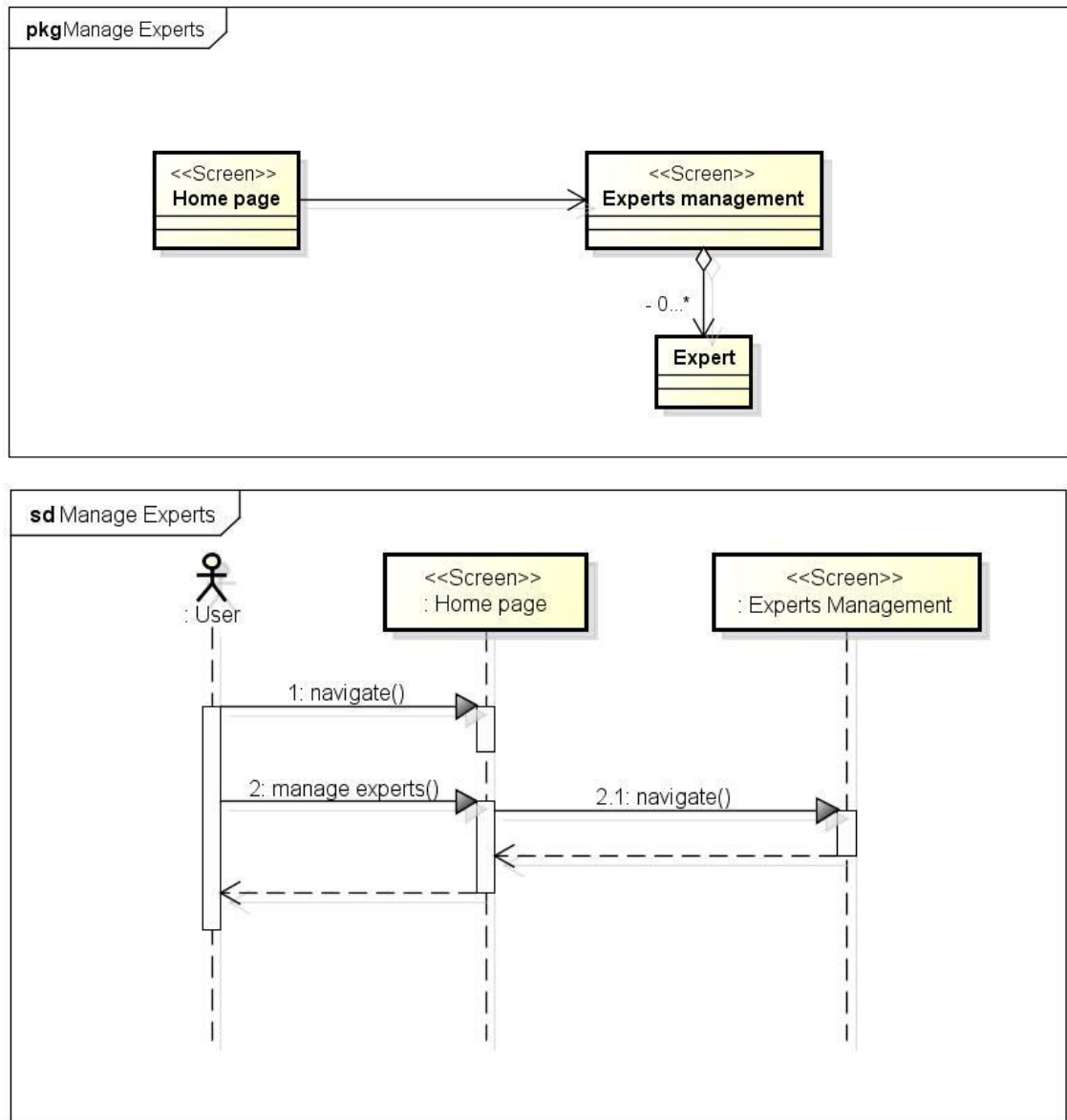


Figure D.9. Manage experts dataflow diagram and sequence diagram

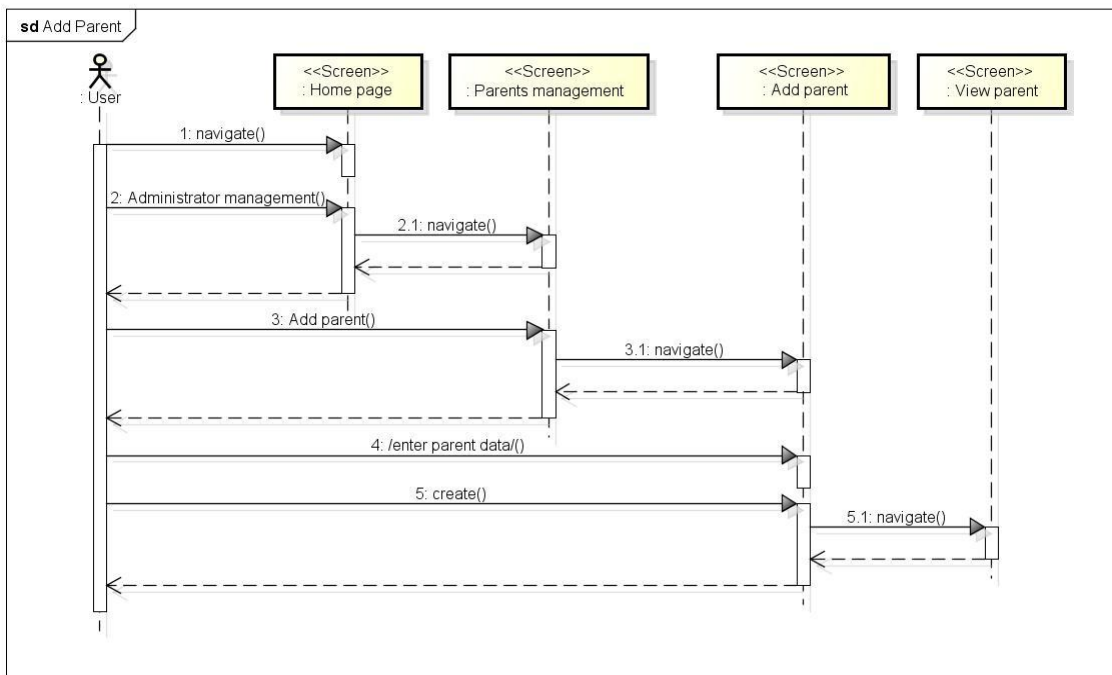
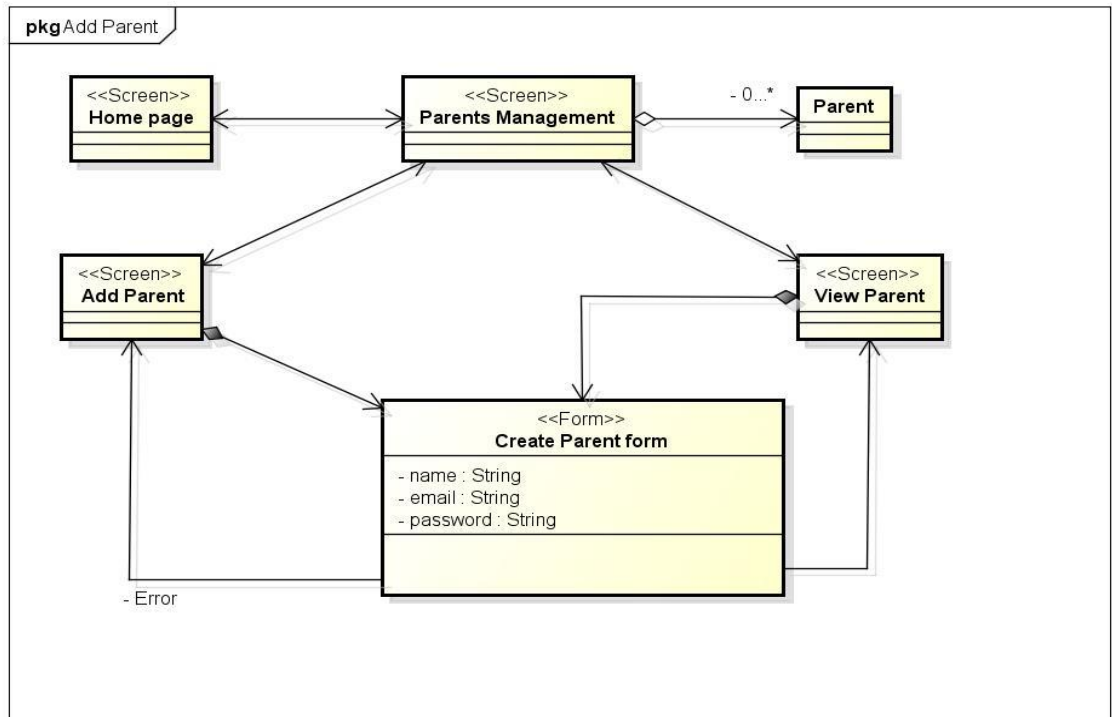


Figure D.10. Create expert dataflow diagram and sequence diagram

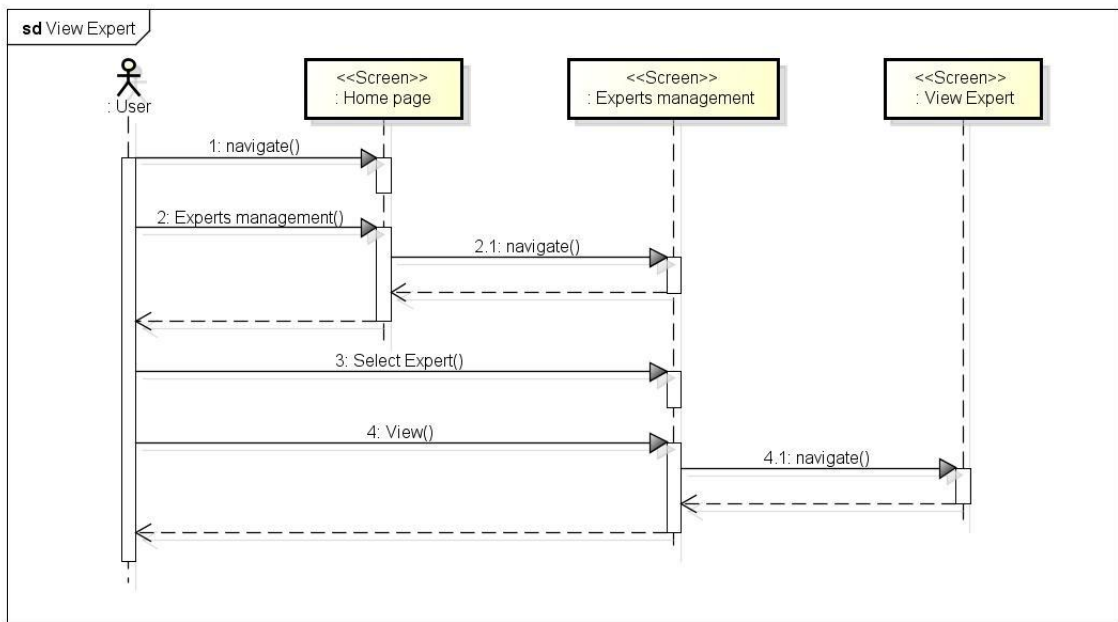
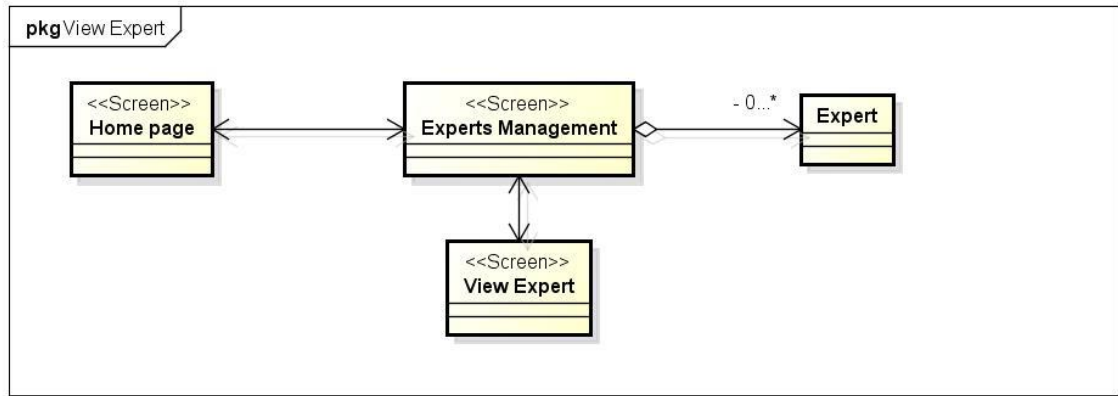


Figure D.11. View expert dataflow diagram and sequence diagram

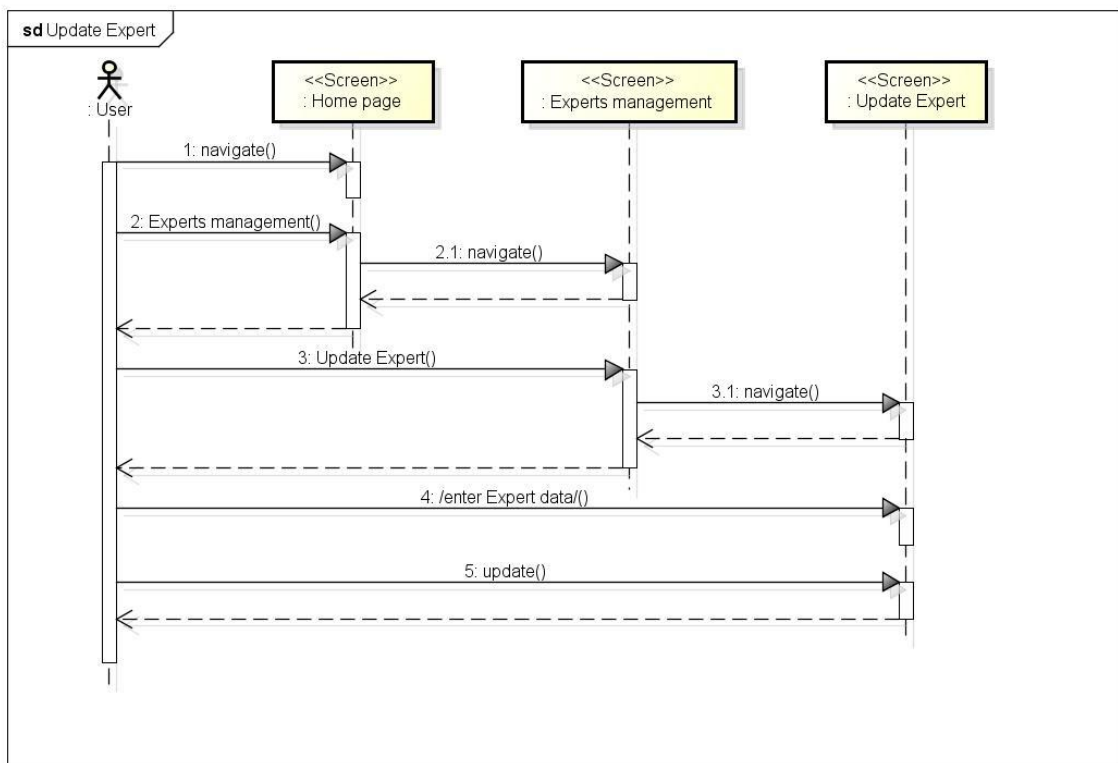
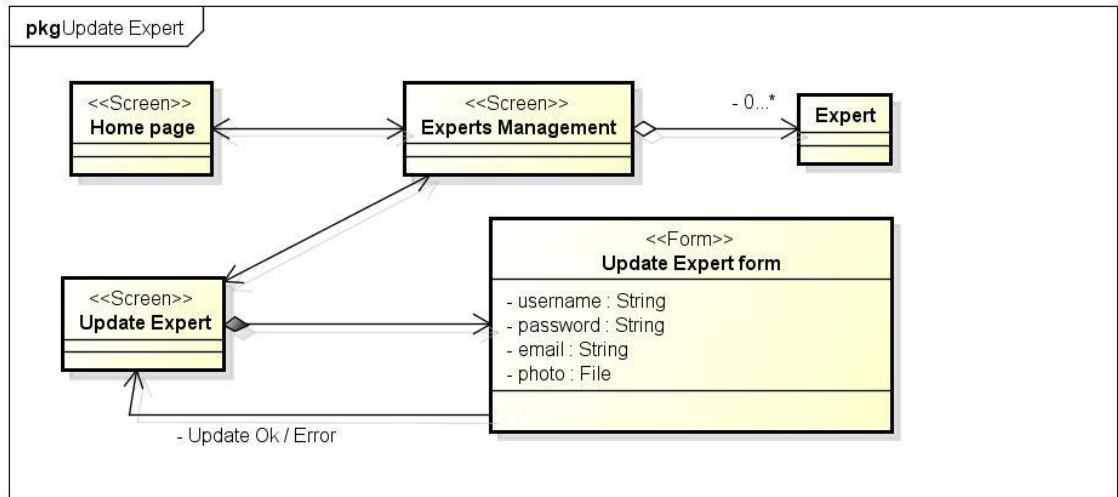


Figure D.12. Update expert dataflow diagram and sequence diagram

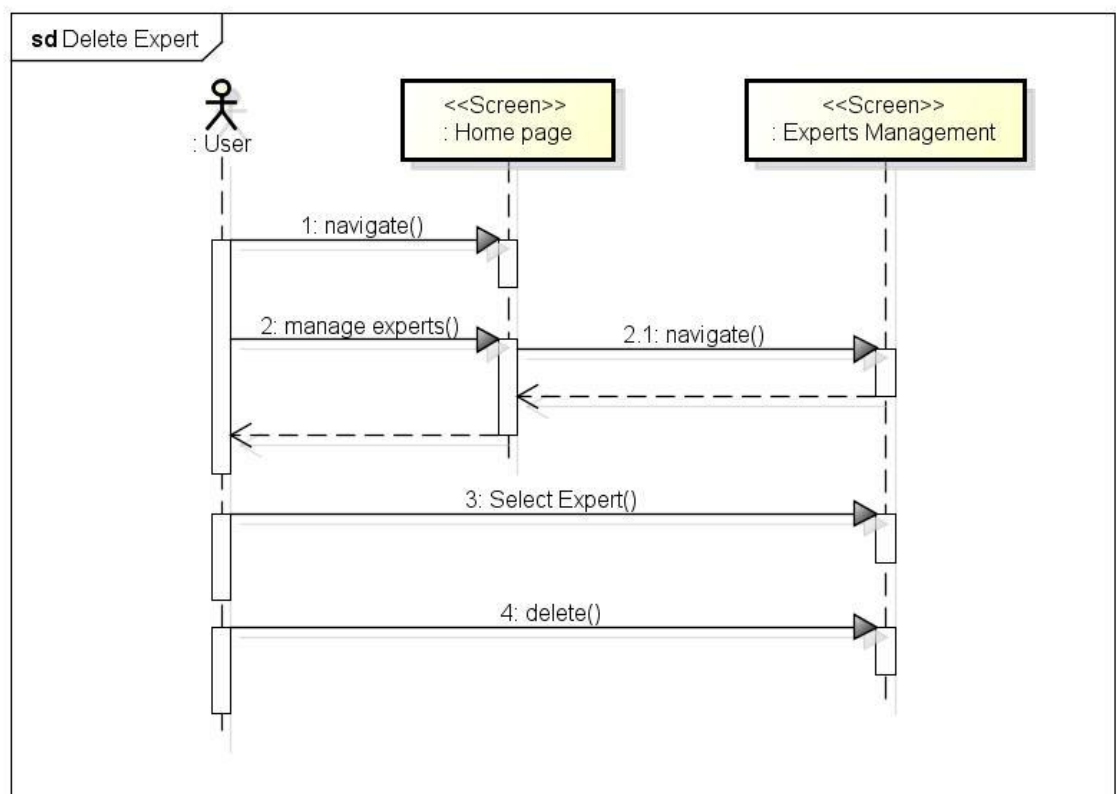
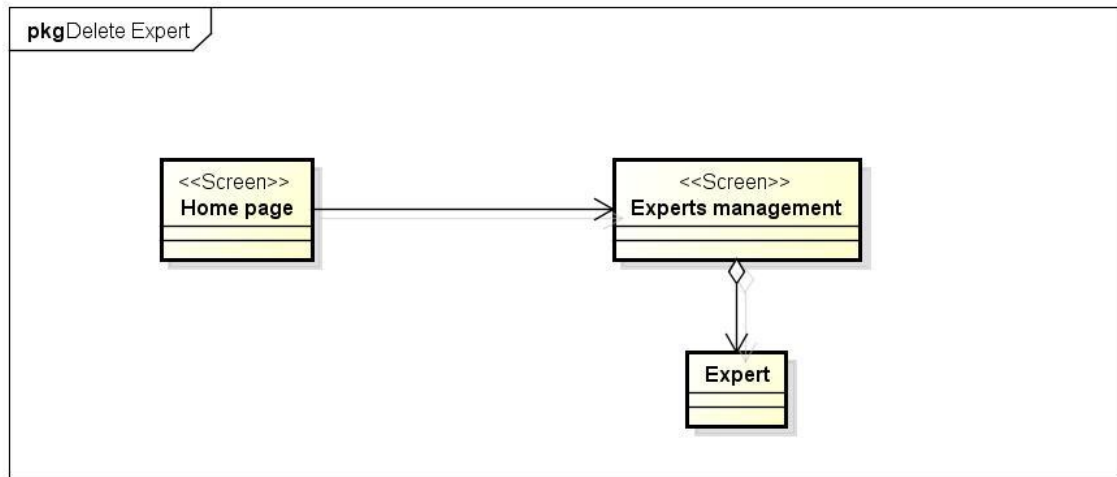


Figure D.13. Delete expert dataflow diagram and sequence diagram

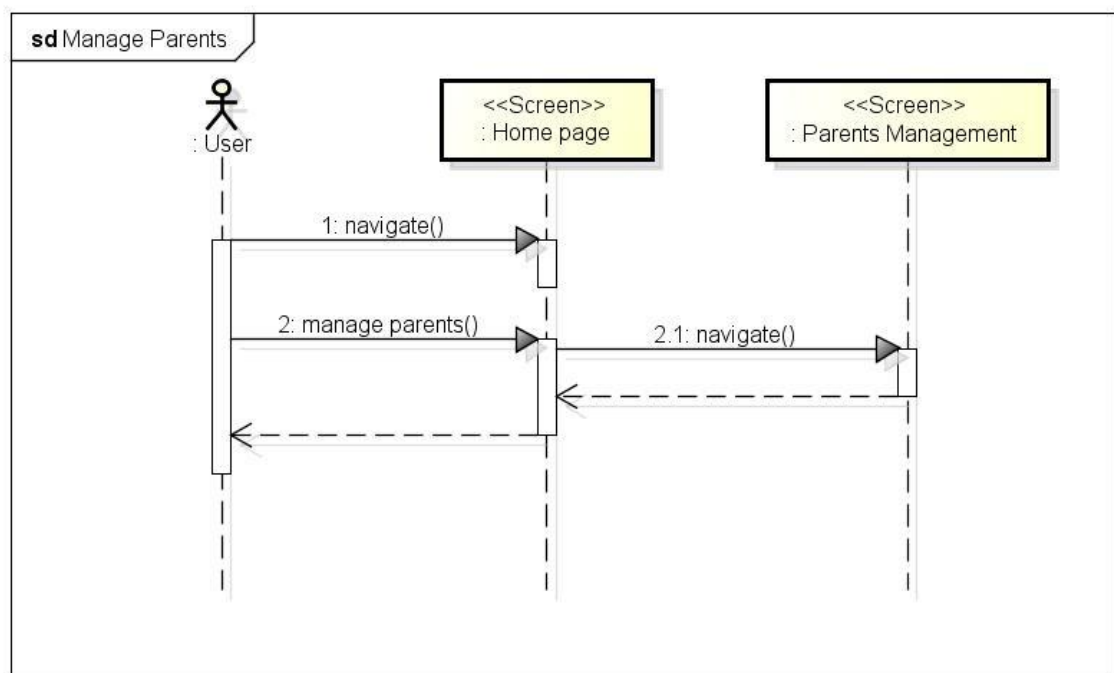
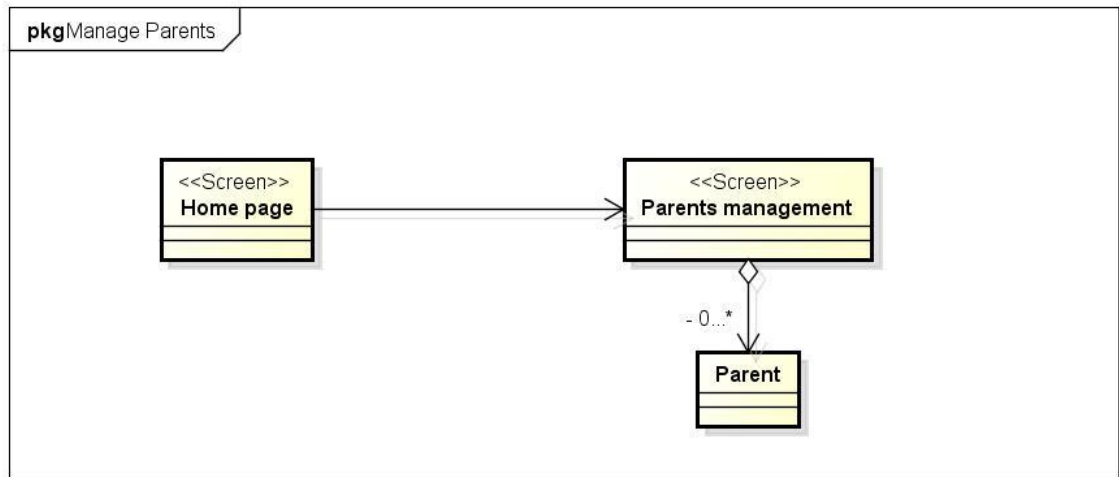


Figure D.14. Manage parents dataflow diagram and sequence diagram

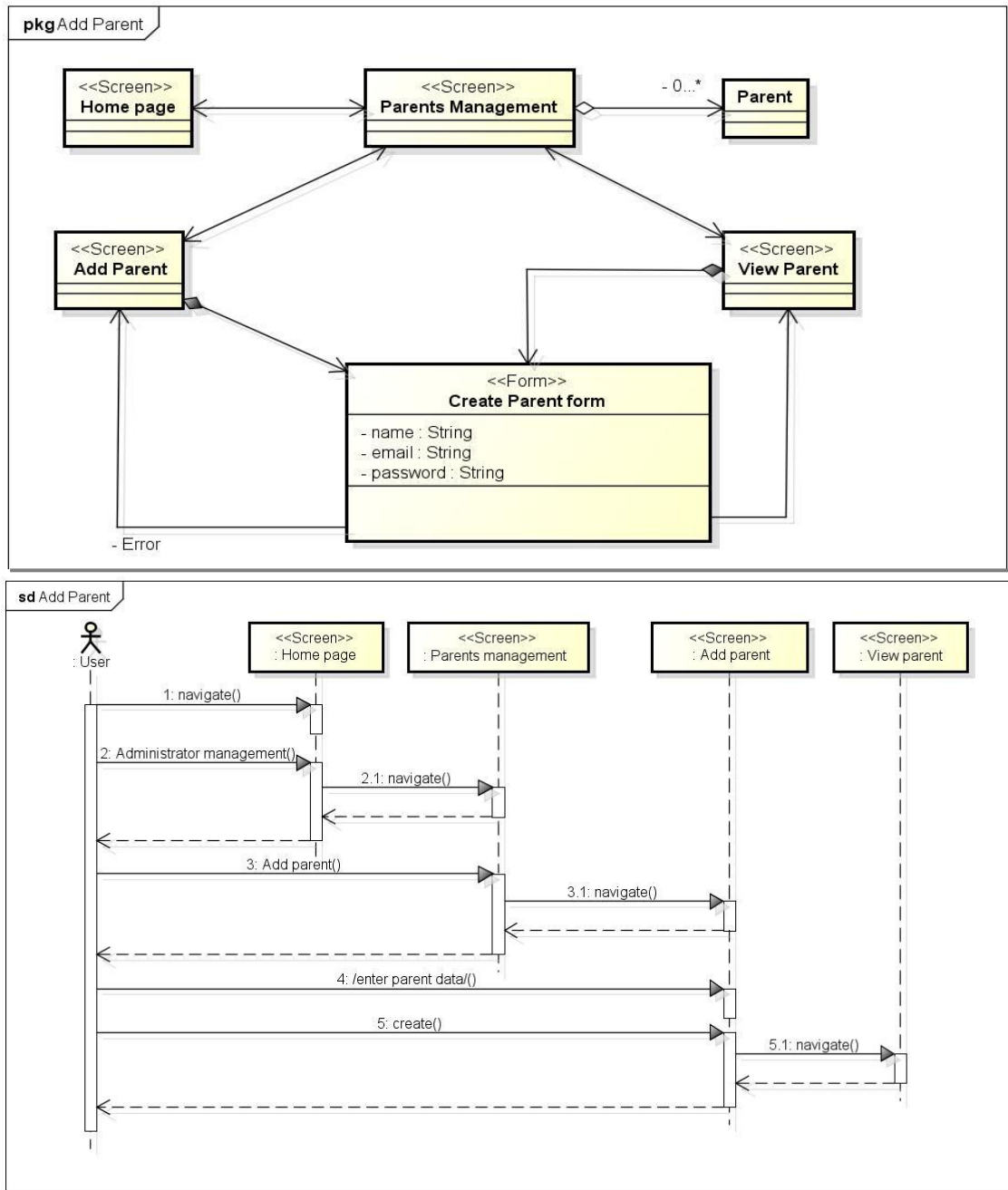


Figure D.15. Create parent dataflow diagram and sequence diagram

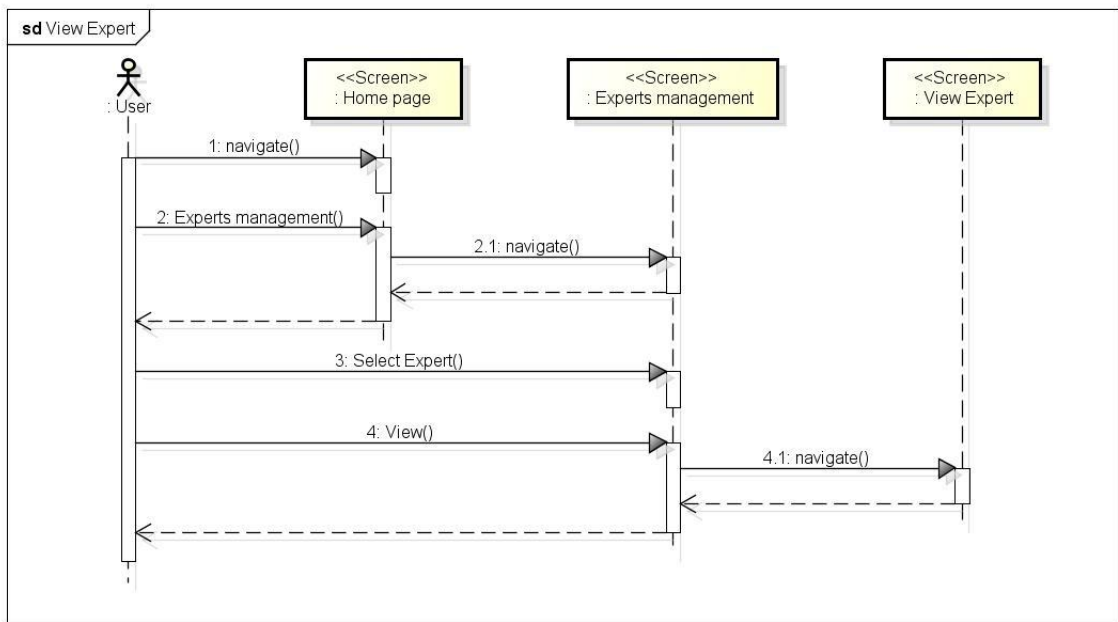
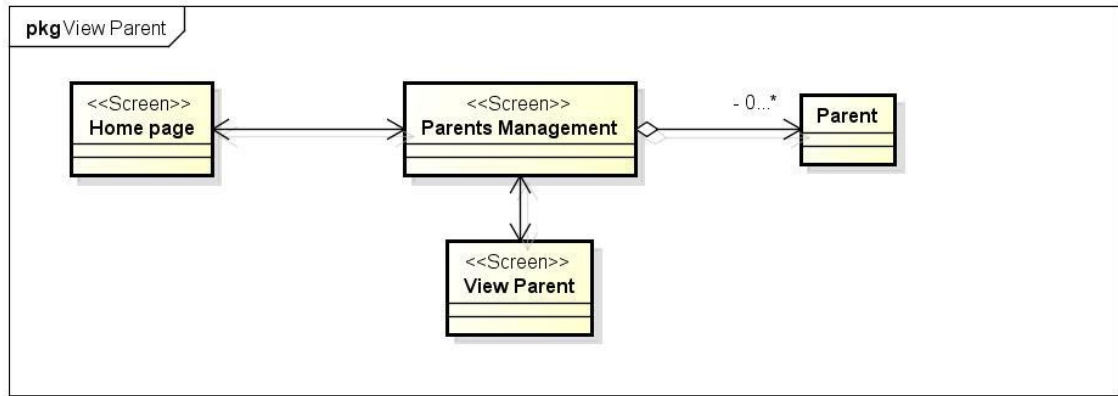


Figure D.16. View parent dataflow diagram and sequence diagram

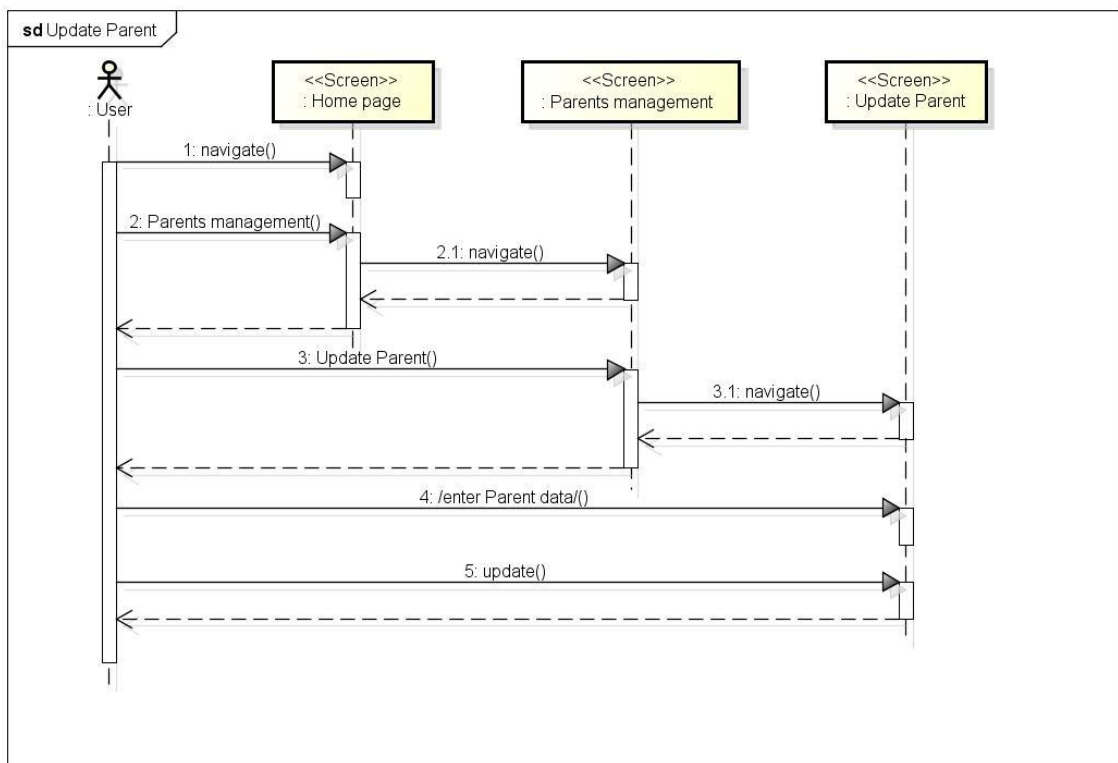
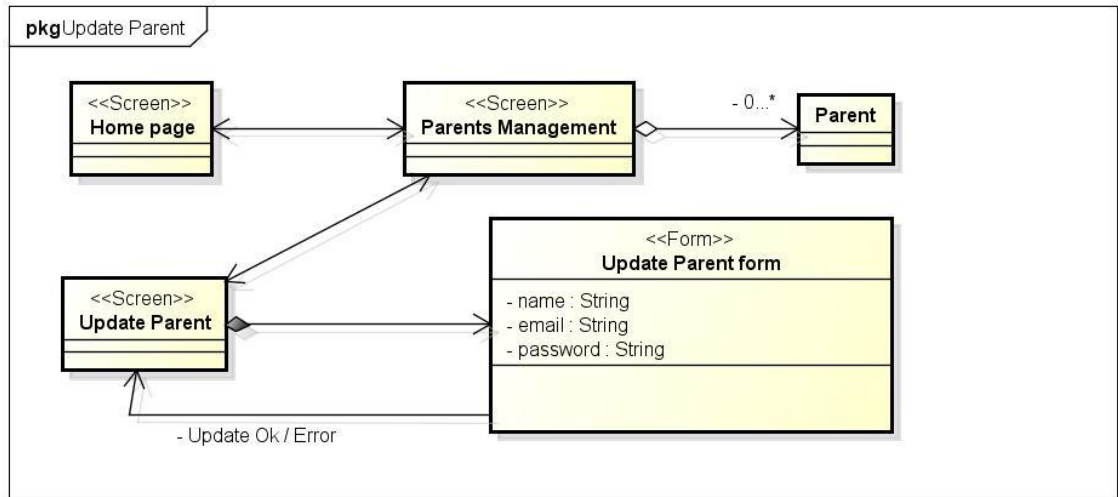


Figure D.17. Update parent dataflow diagram and sequence diagram

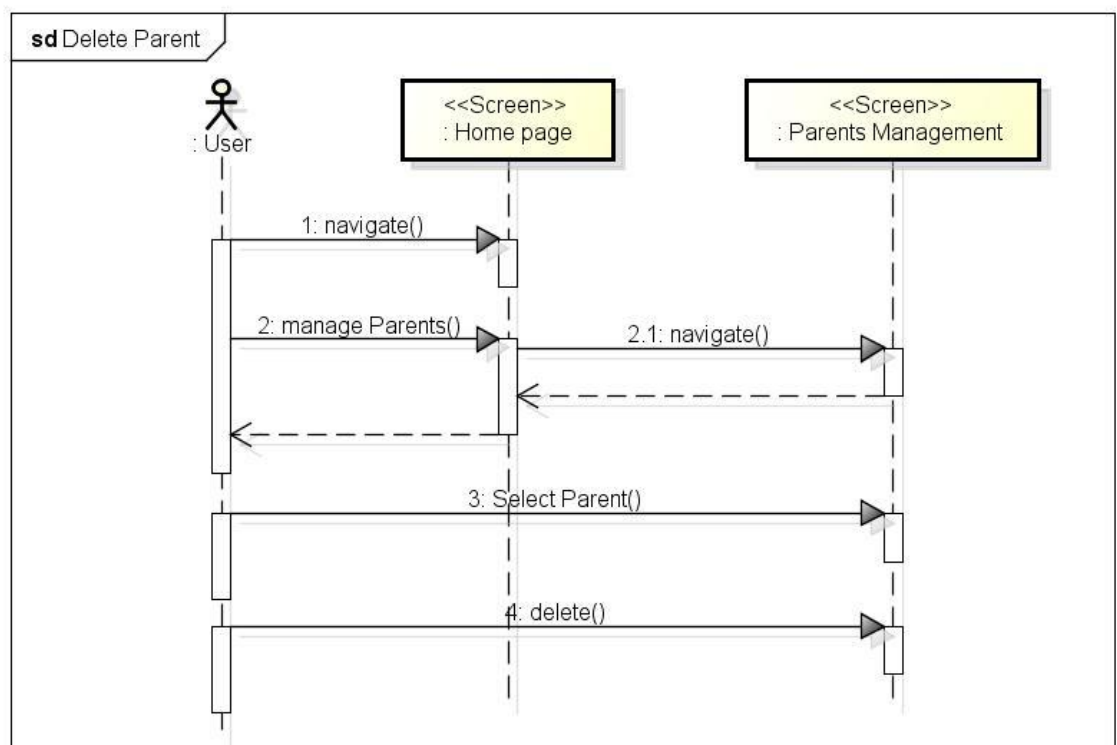
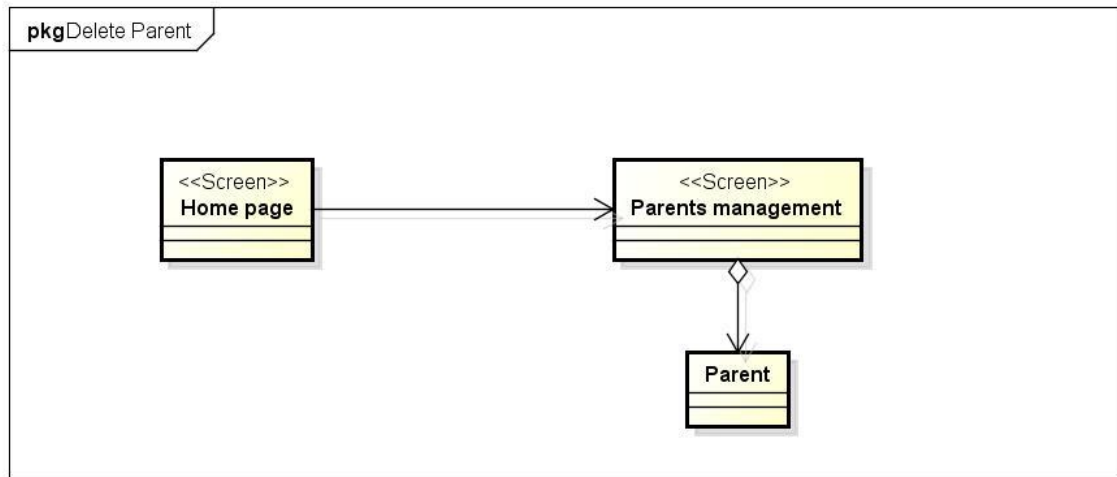


Figure D.18. Delete parent dataflow diagram and sequence diagram

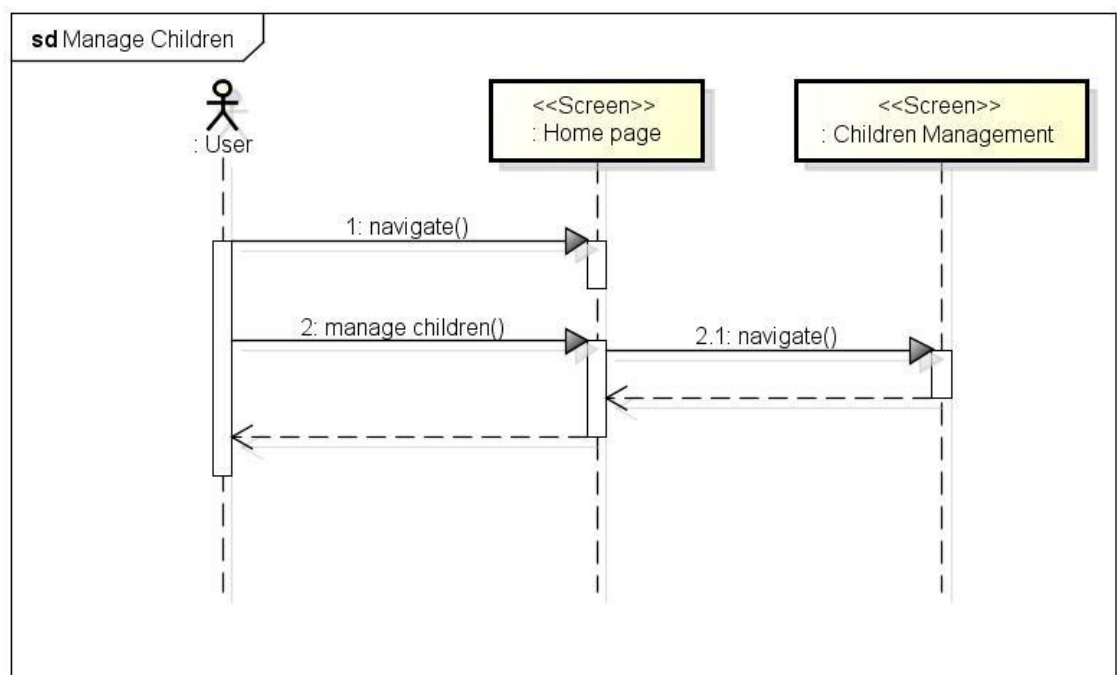
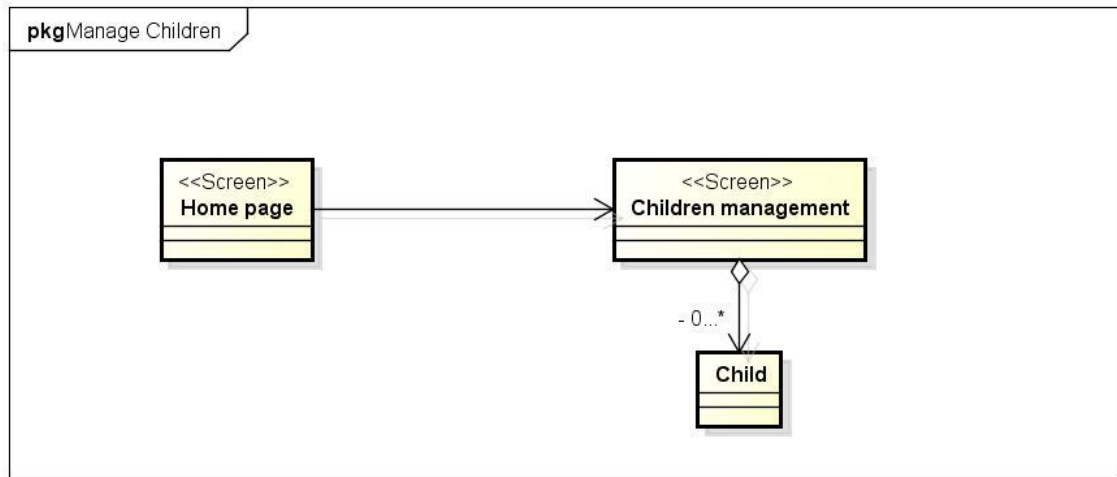


Figure D.19. Manage children dataflow diagram and sequence diagram

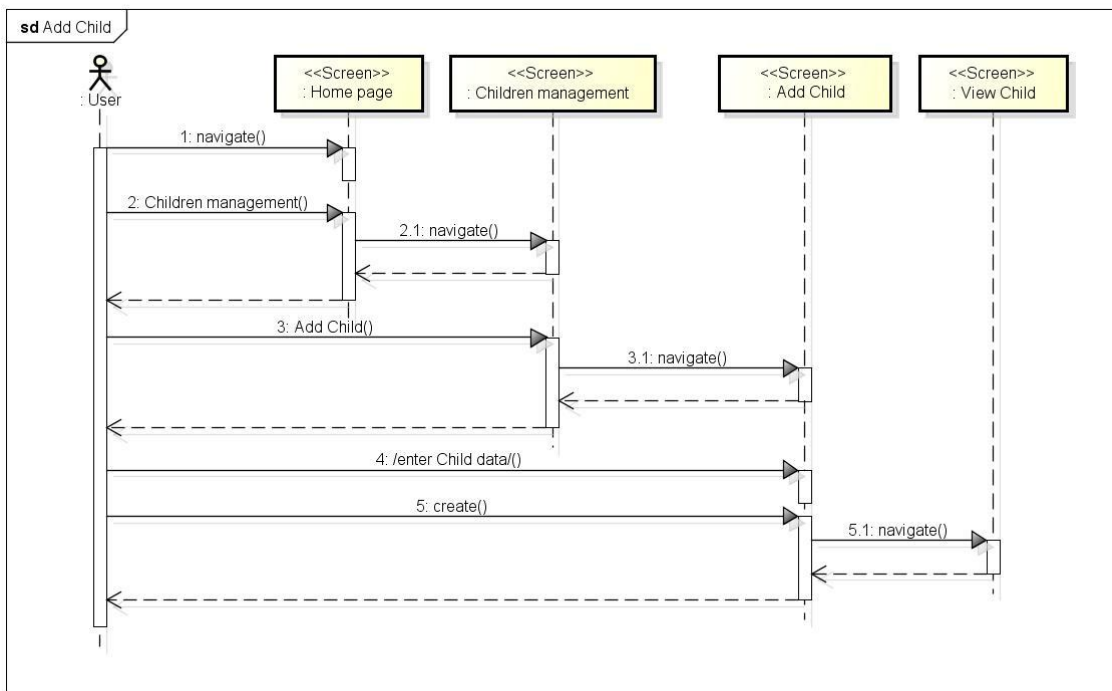
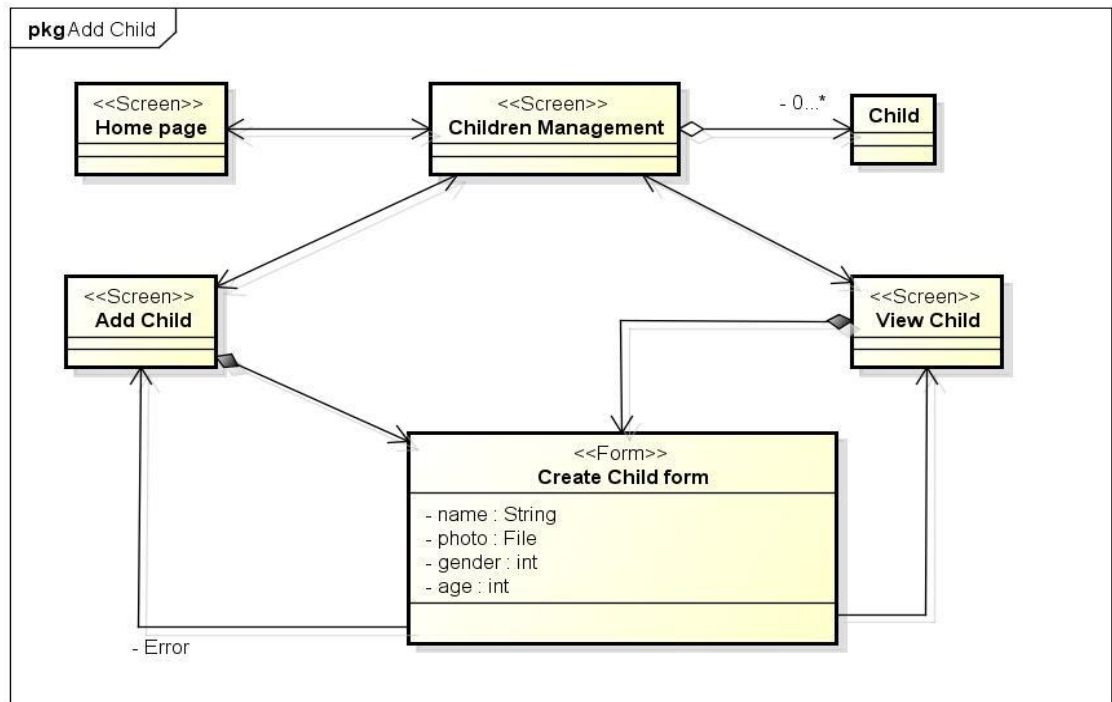


Figure D.20. Create child dataflow diagram and sequence diagram

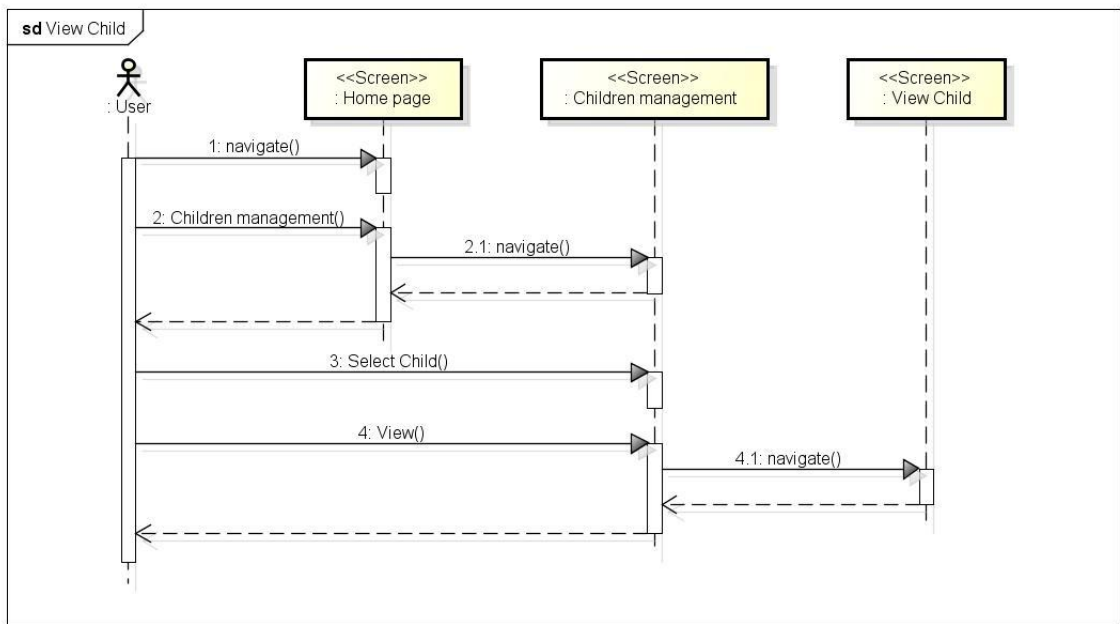
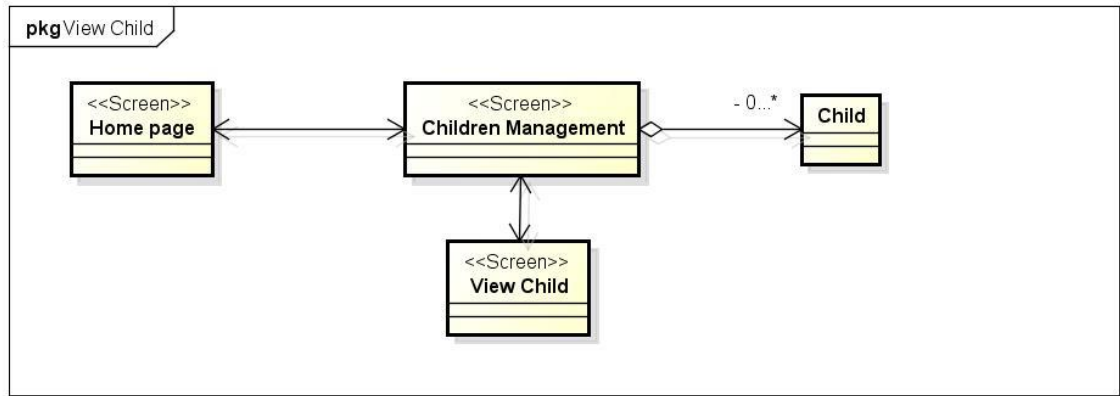


Figure D.21. View child dataflow diagram and sequence diagram

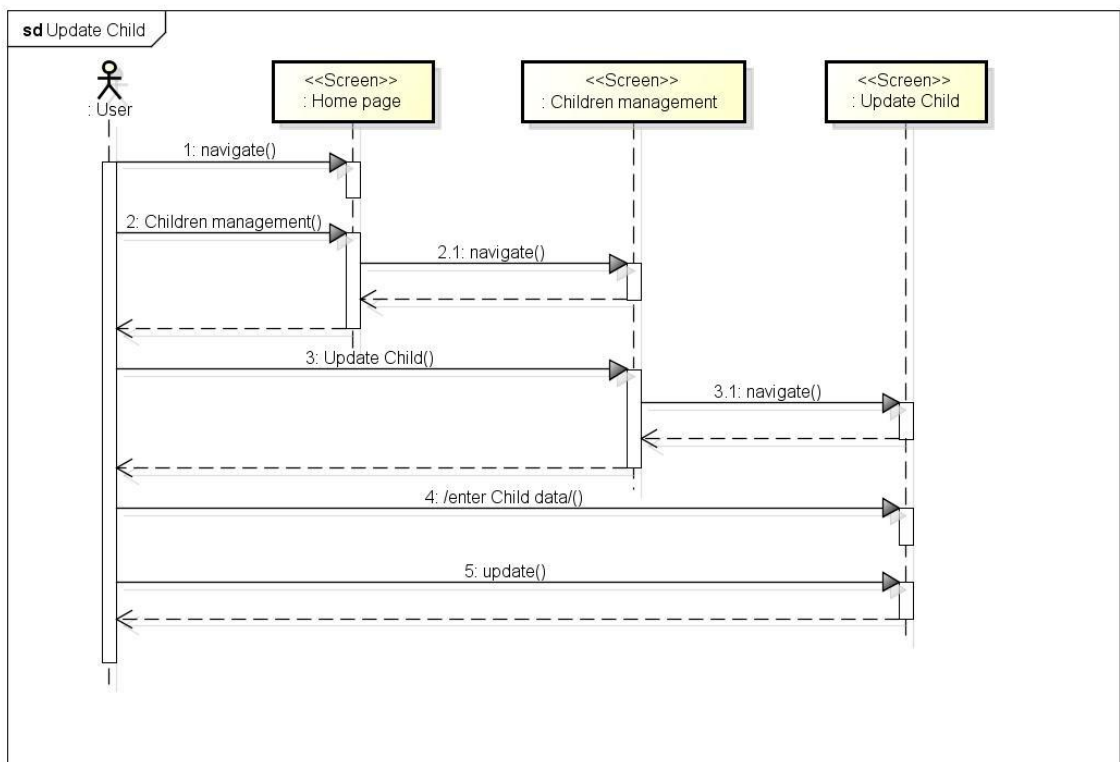
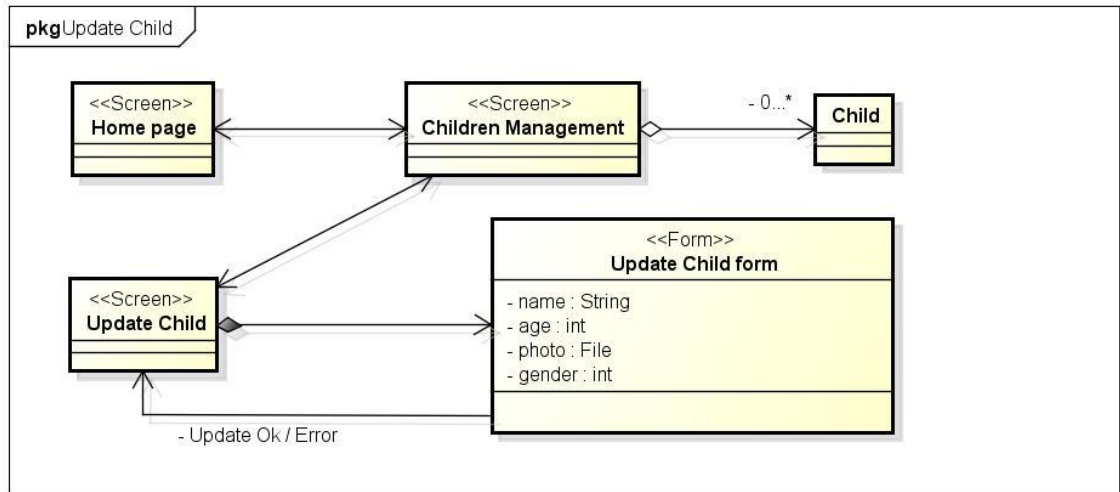


Figure D.22. Update child dataflow diagram and sequence diagram

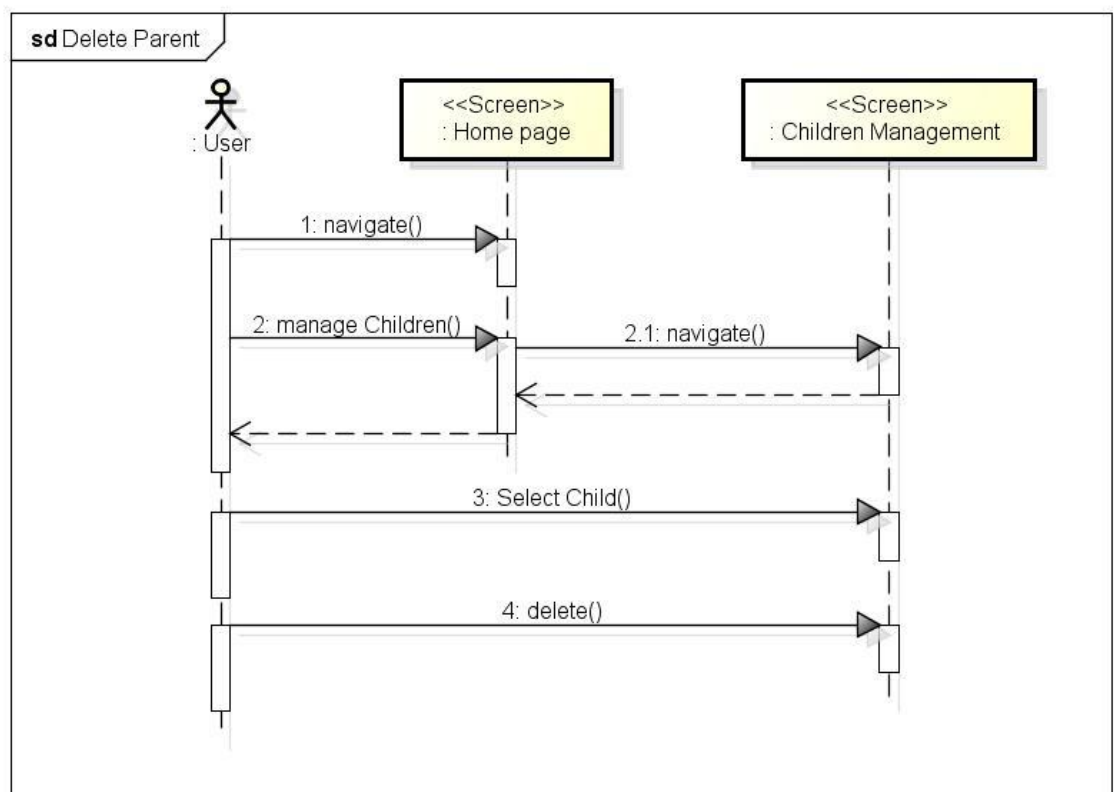
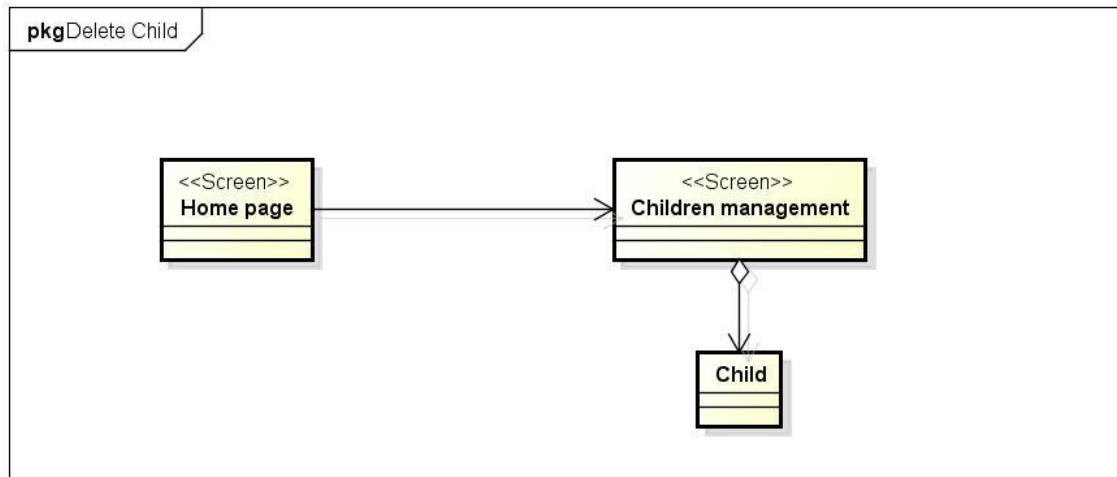


Figure D.23. Delete child dataflow diagram and sequence diagram

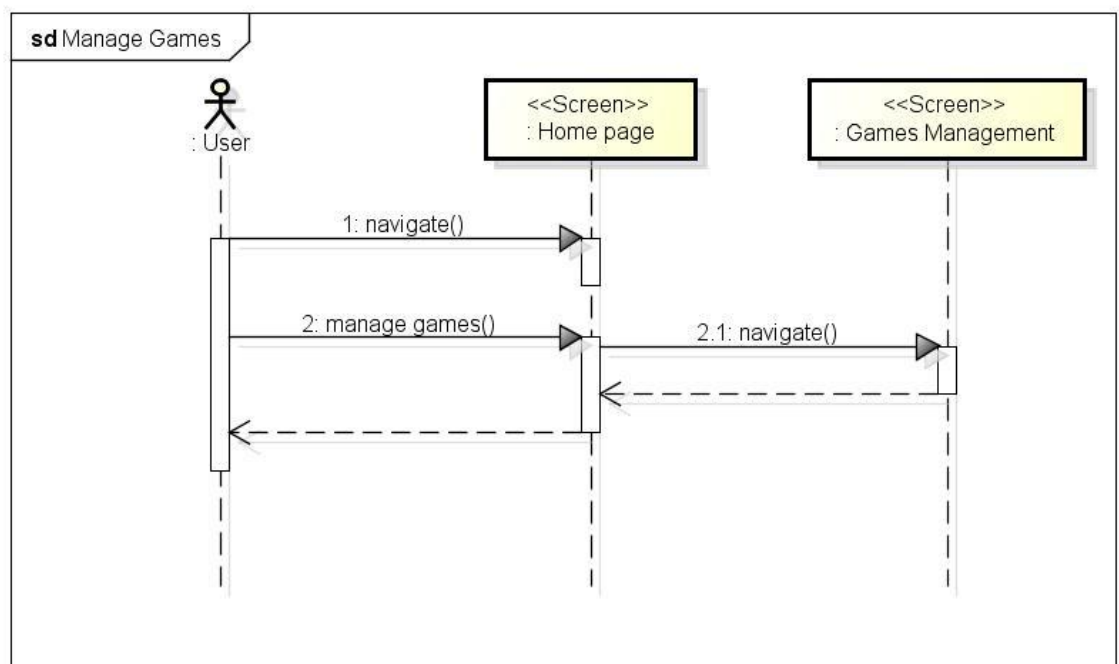
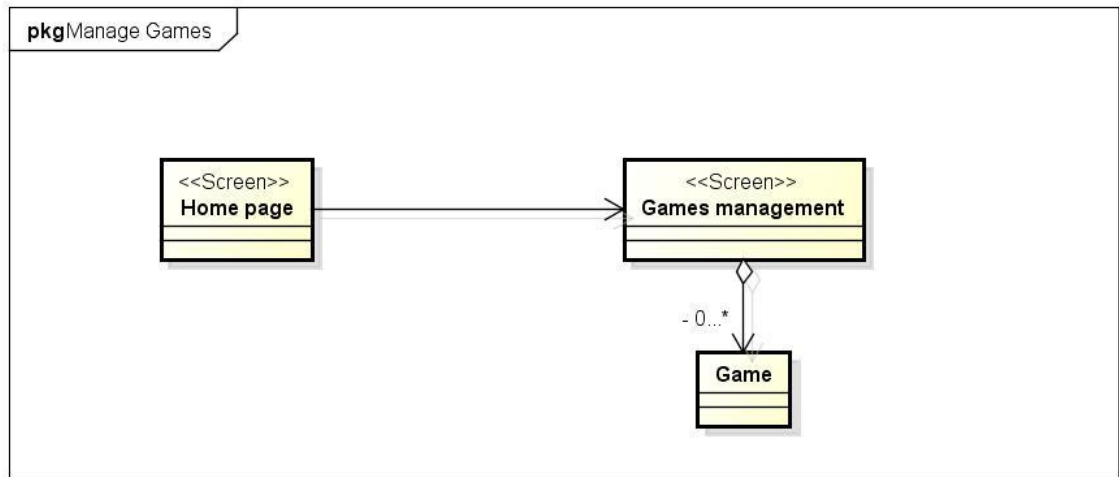


Figure D.24. Manage games dataflow diagram and sequenyutjce diagram

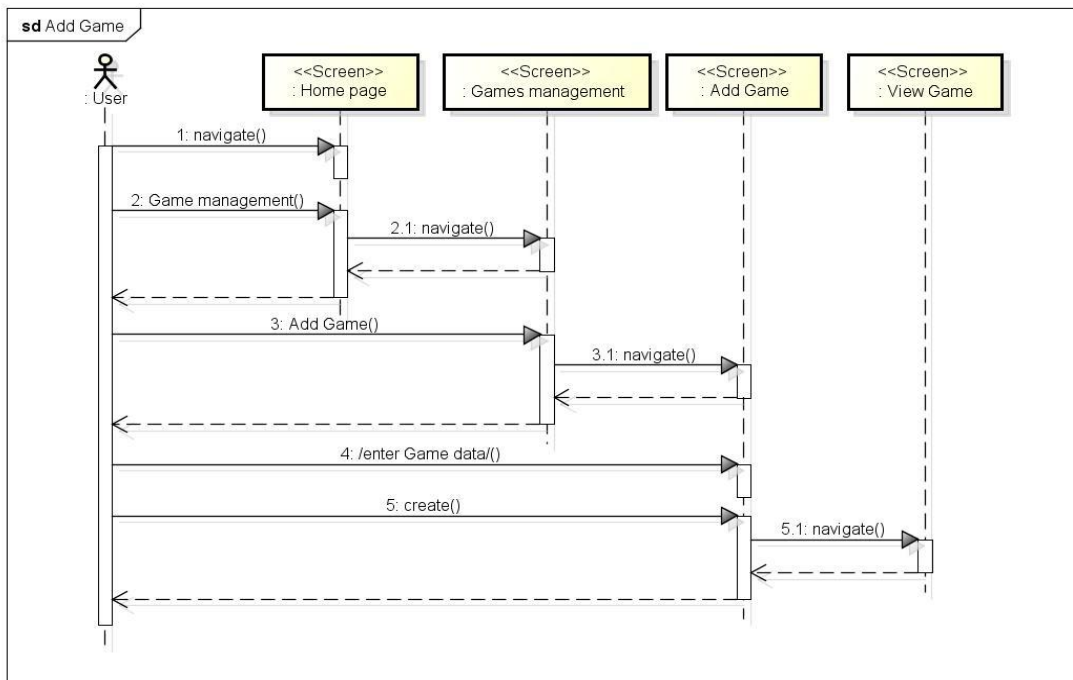
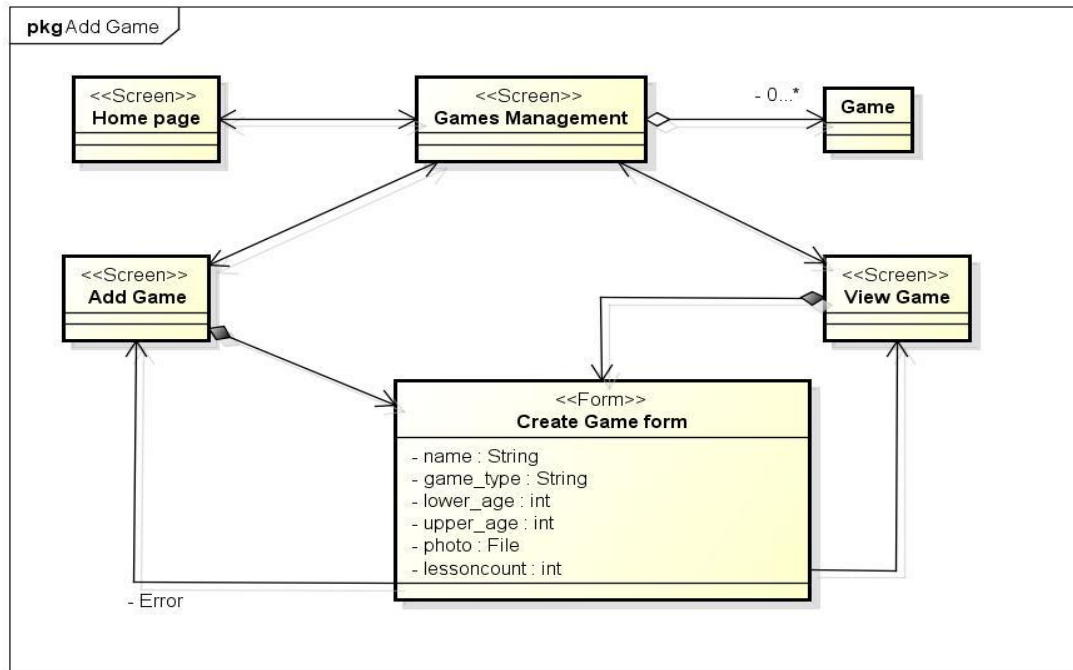


Figure D.25. Create game dataflow diagram and sequence diagram

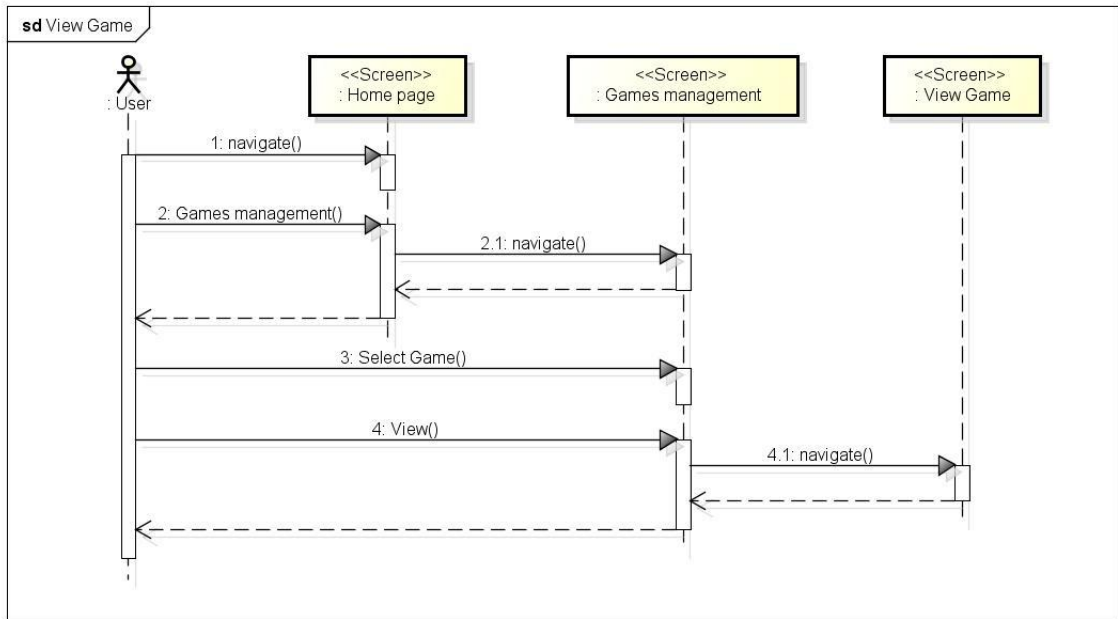
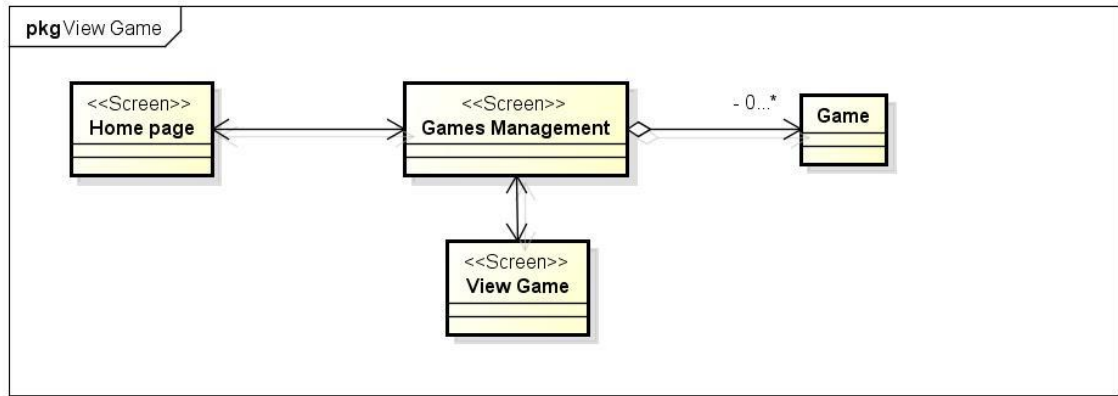


Figure D.26. View game dataflow diagram and sequence diagram

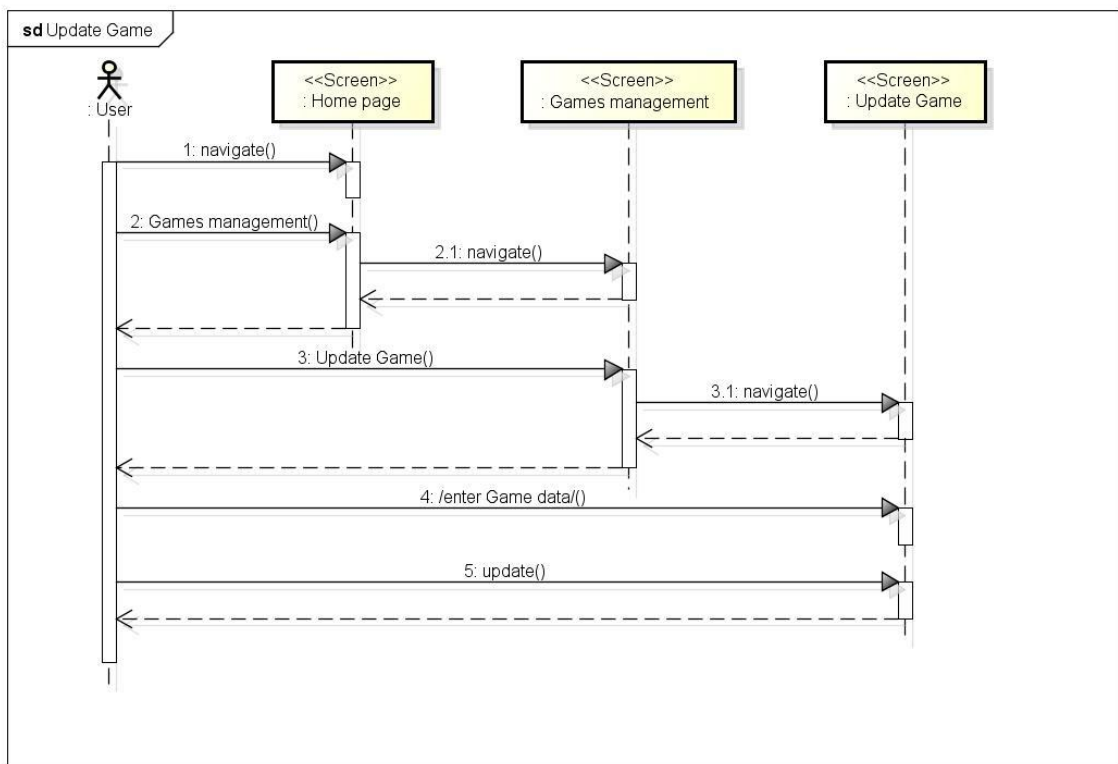
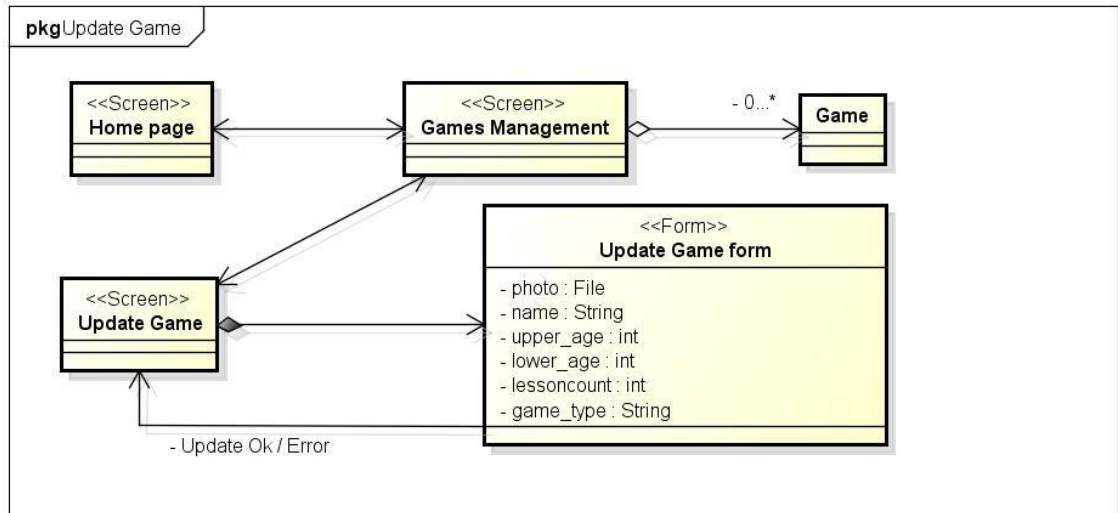


Figure D.27. Update game dataflow diagram and sequence diagram

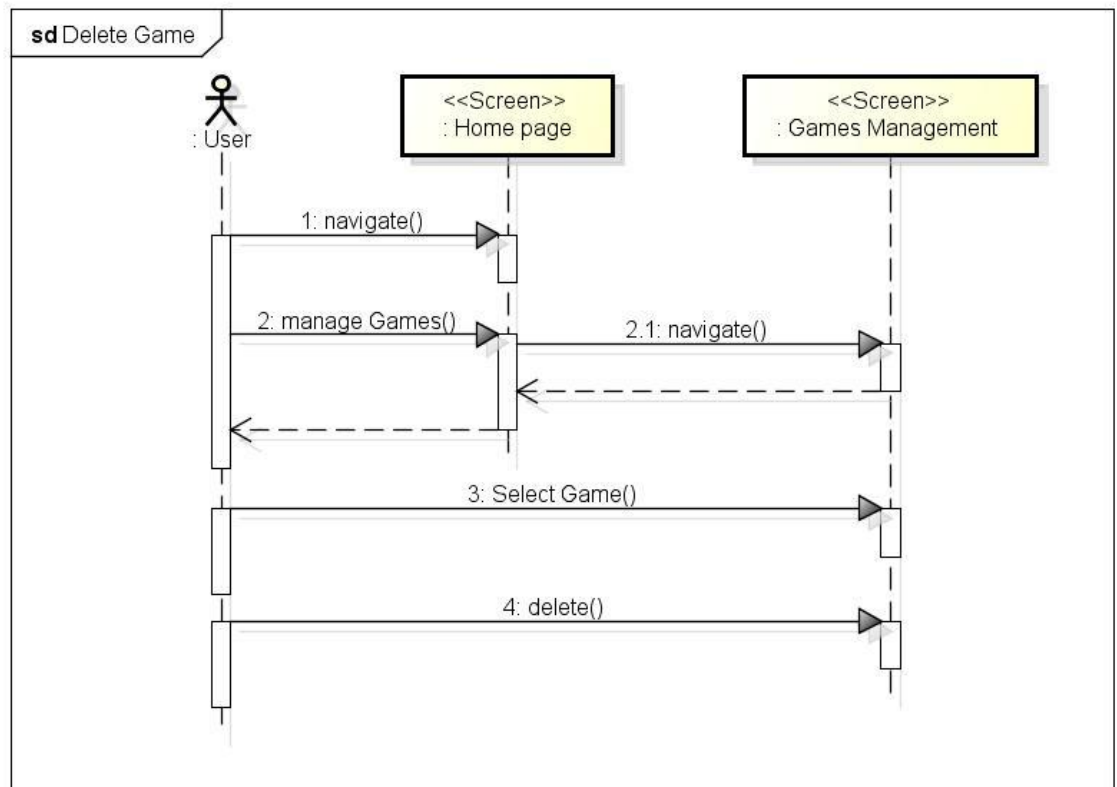
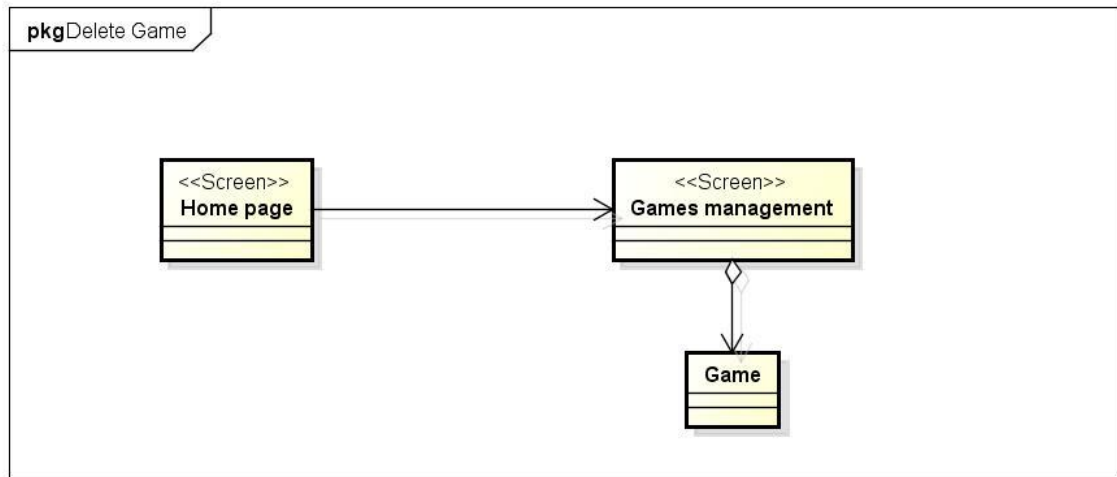


Figure D.28. Delete game dataflow diagram and sequence diagram

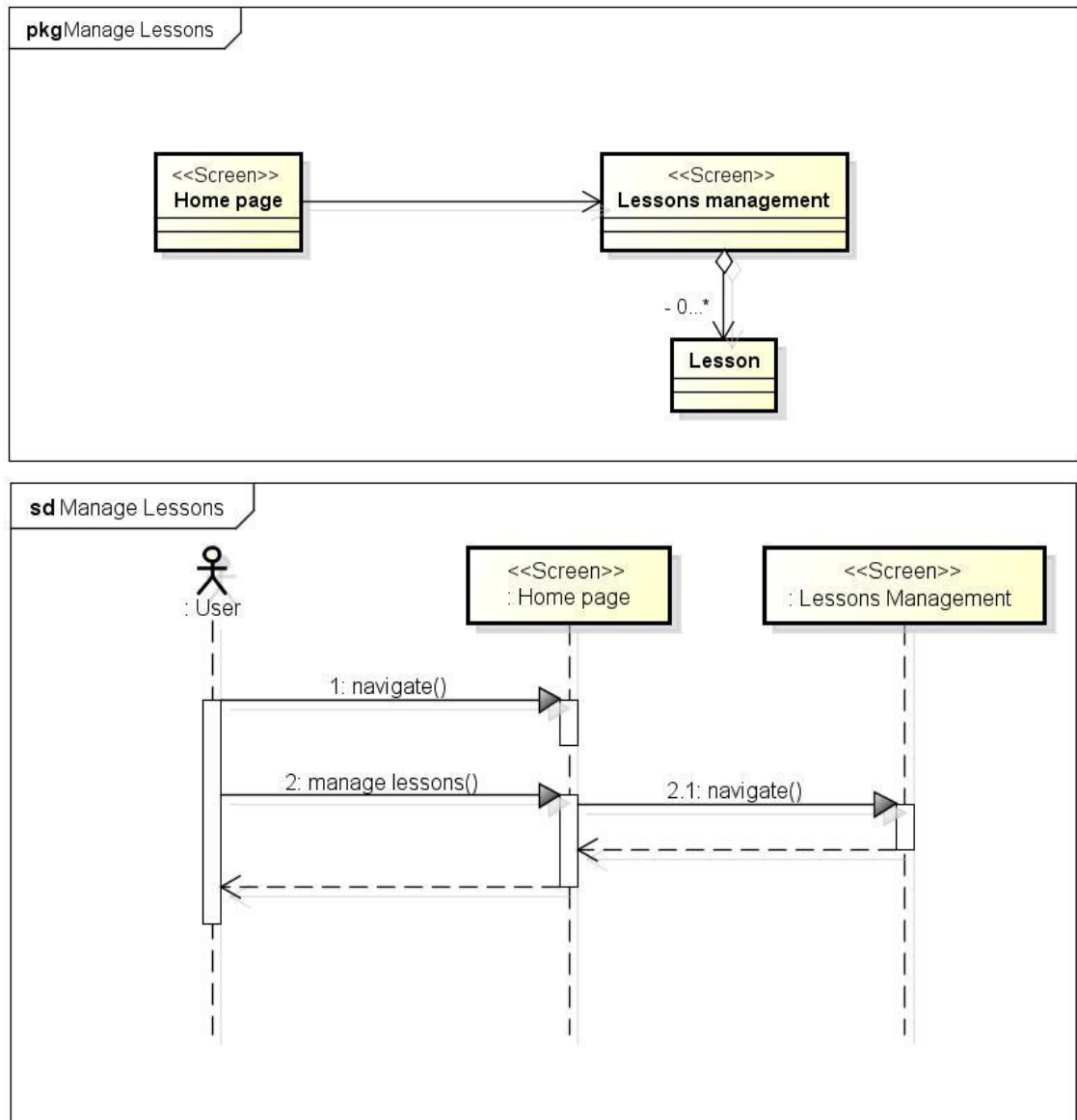


Figure D.29. Manage lesson dataflow diagram and sequence diagram

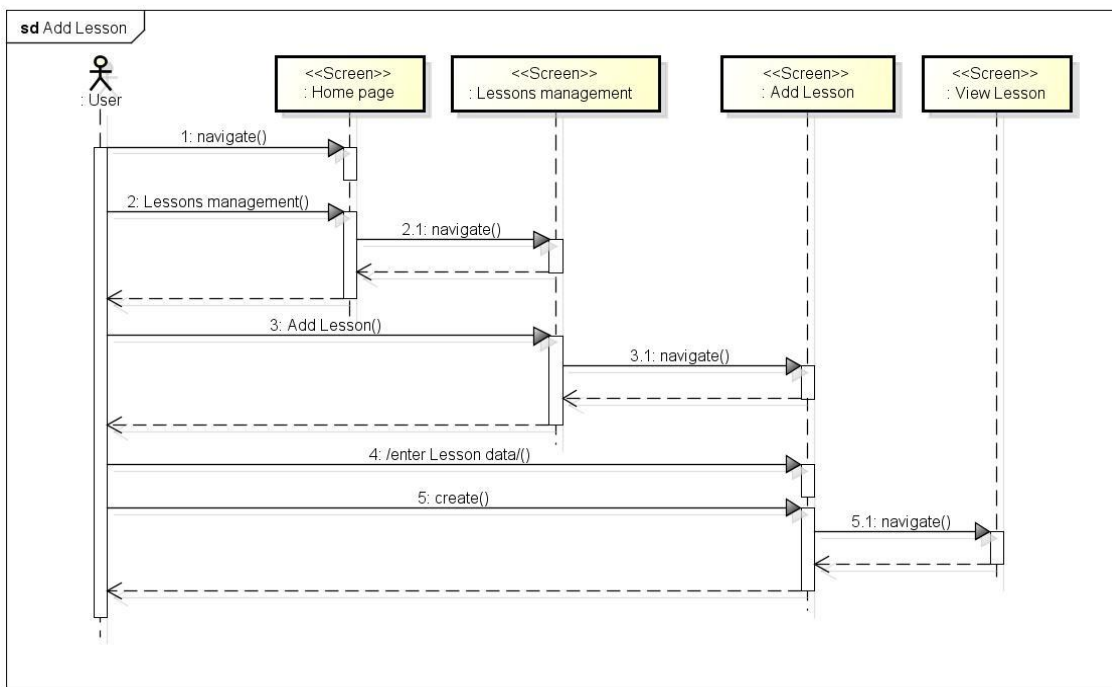
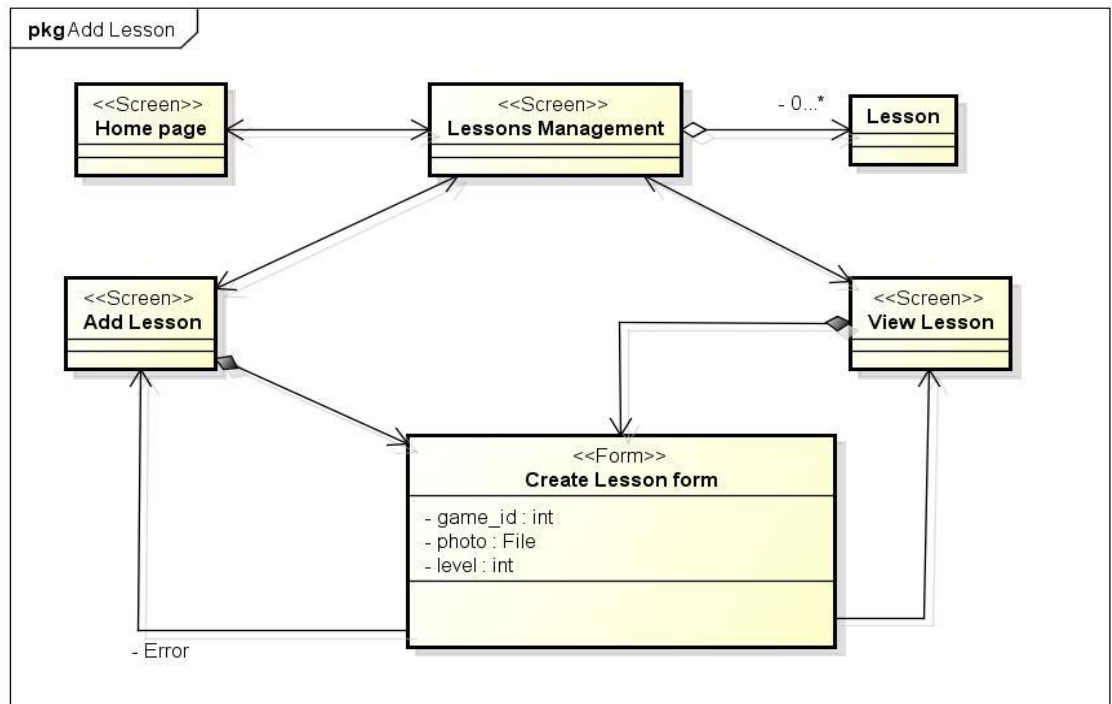


Figure D.30. Create lesson dataflow diagram and sequence diagram

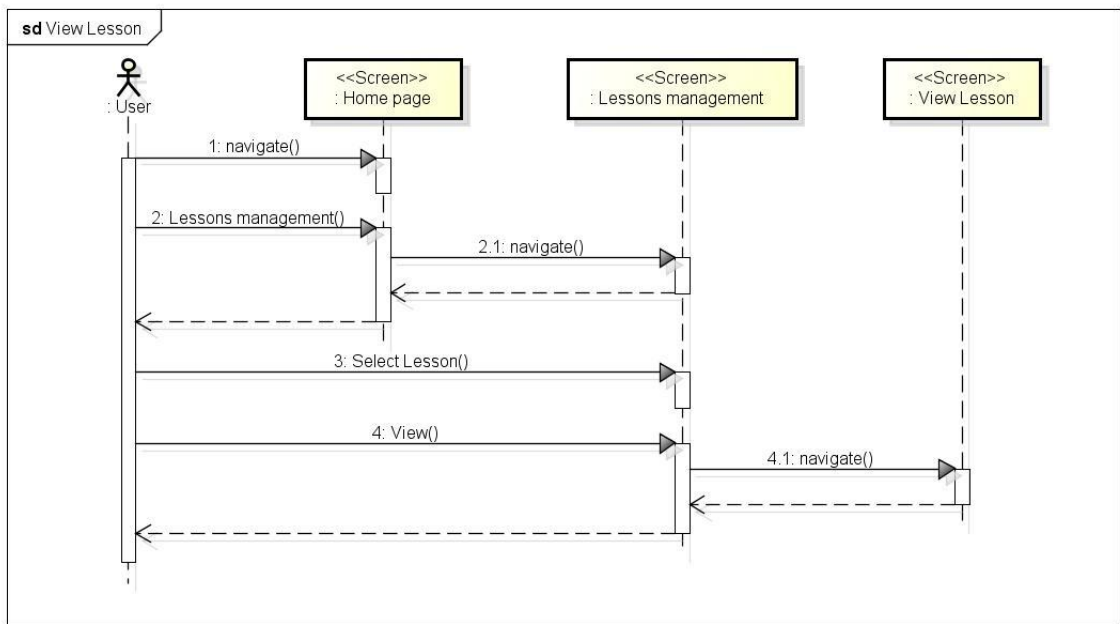
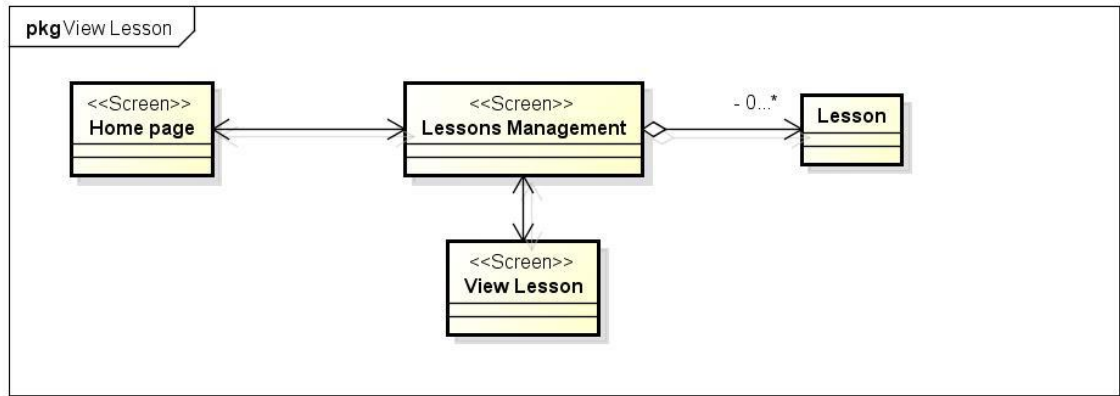


Figure D.31. View lesson dataflow diagram and sequence diagram

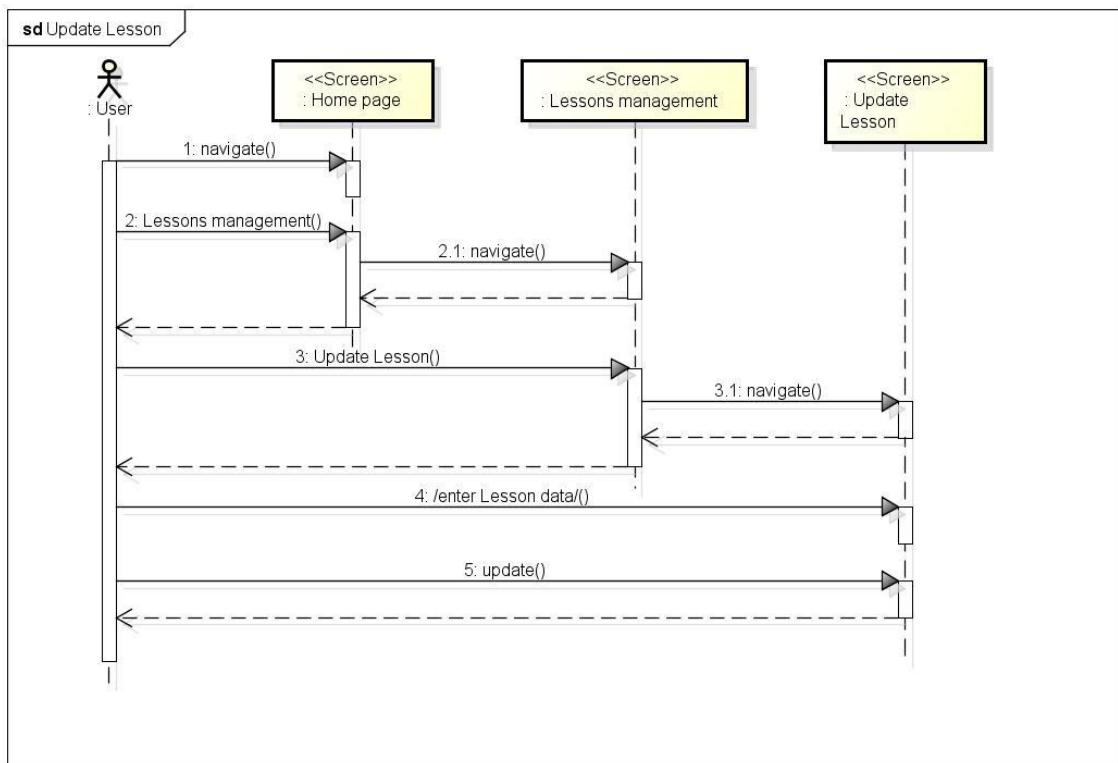
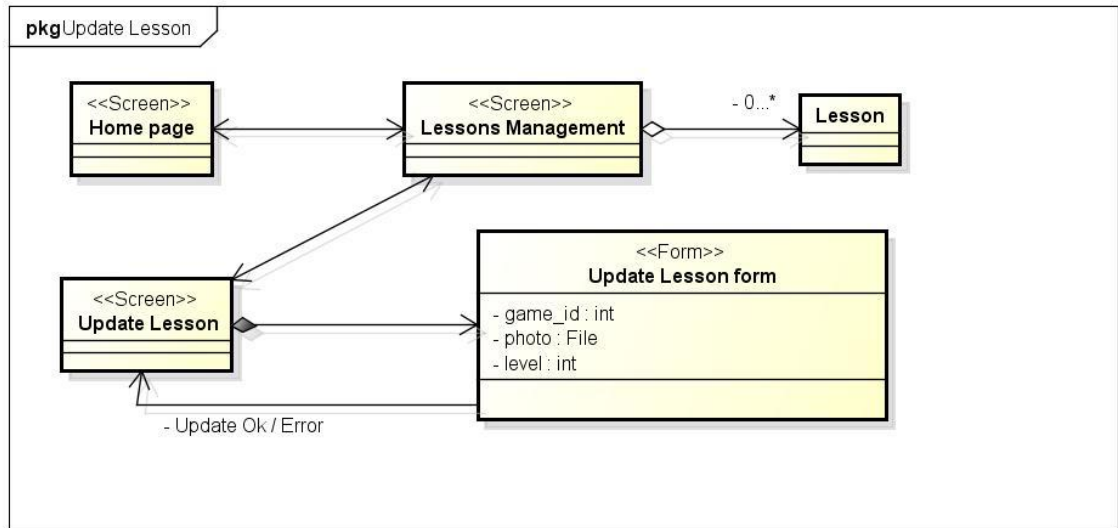


Figure D.32. Update lesson dataflow diagram and sequence diagram

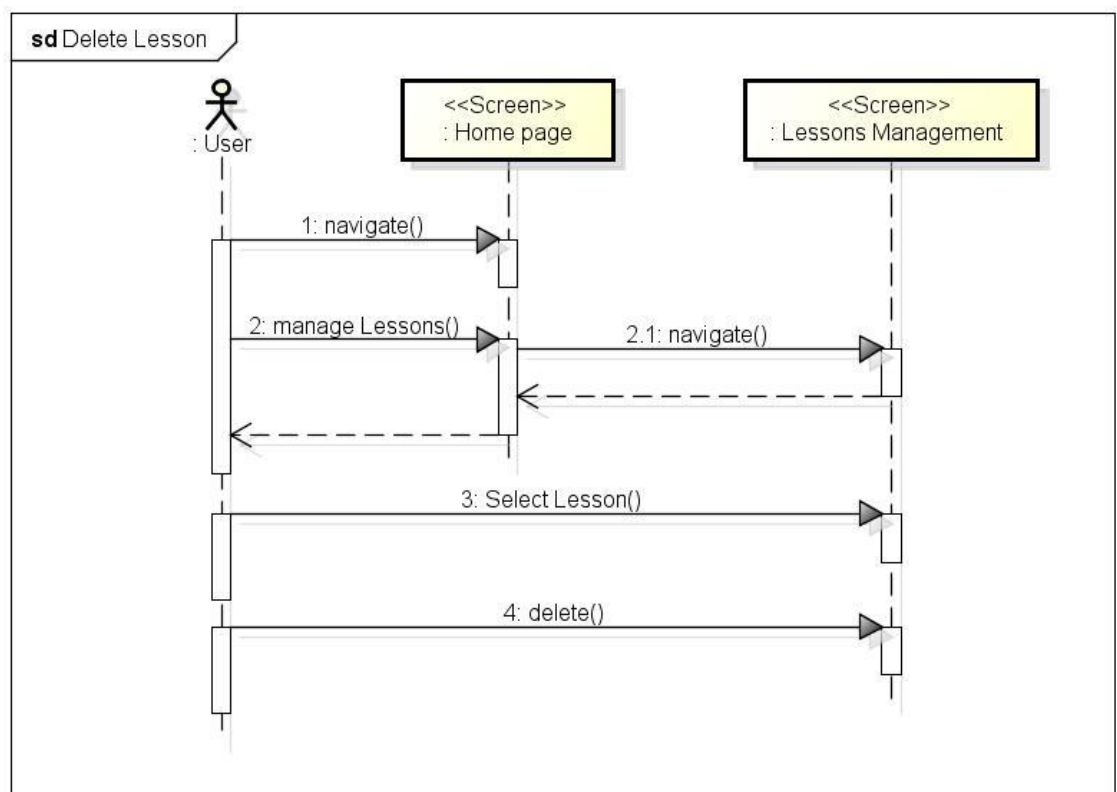
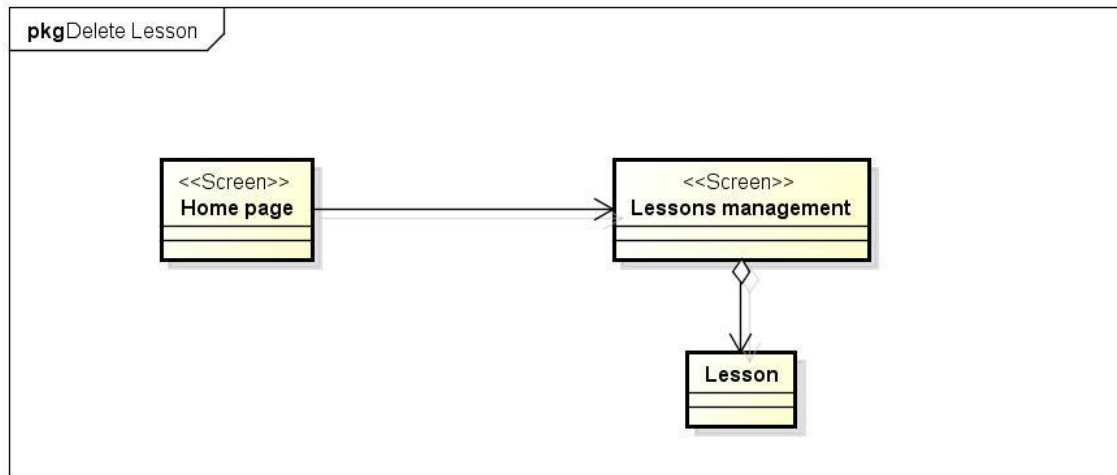


Figure D.33. Delete lesson dataflow diagram and sequence diagram

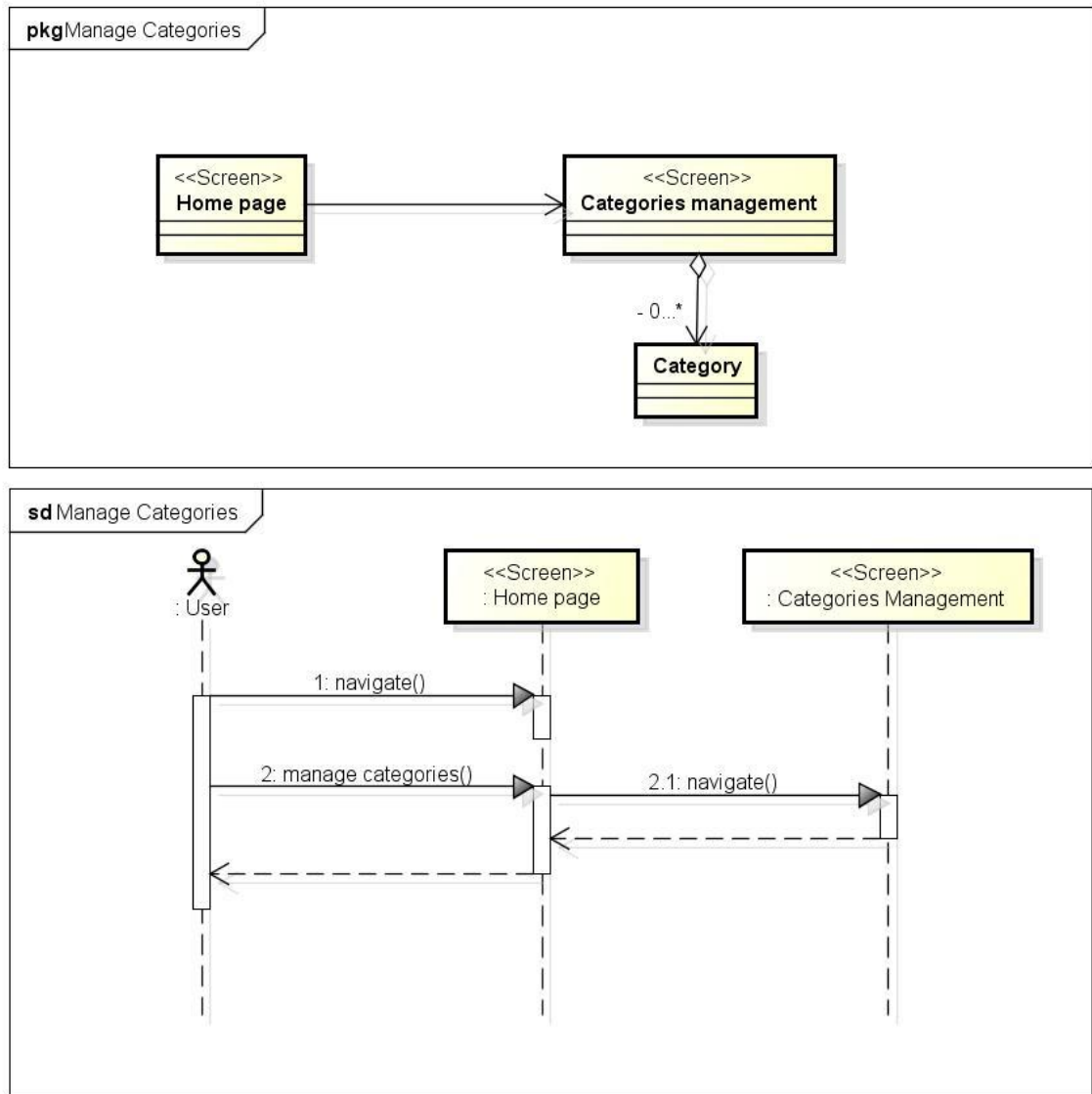


Figure D.34. Manage categories dataflow diagram and sequence diagram

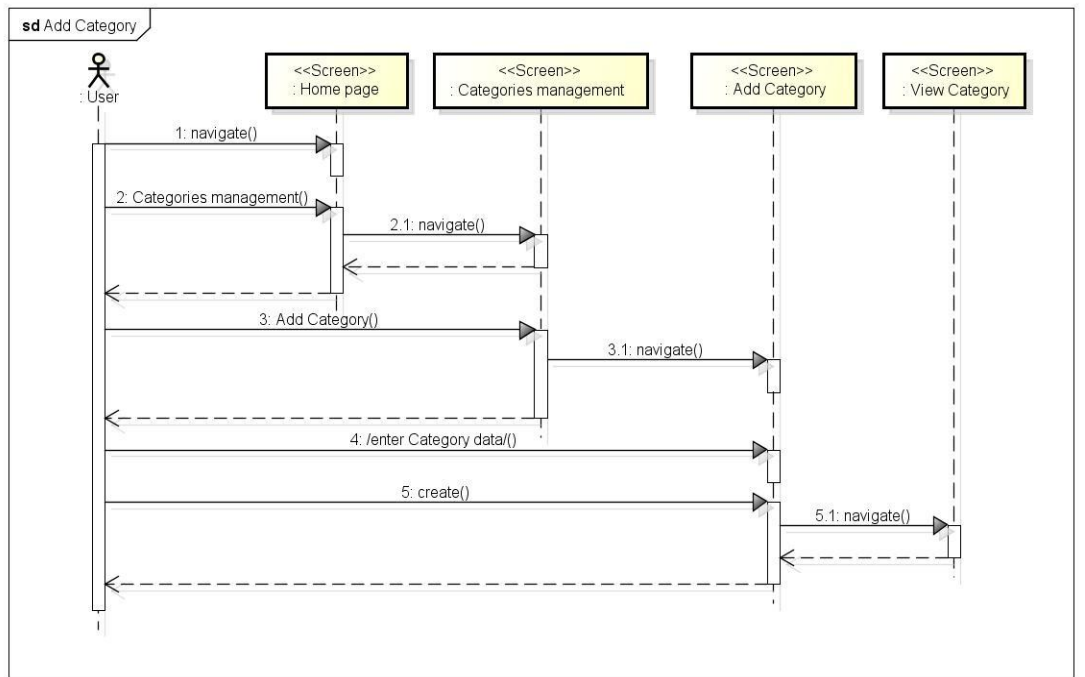
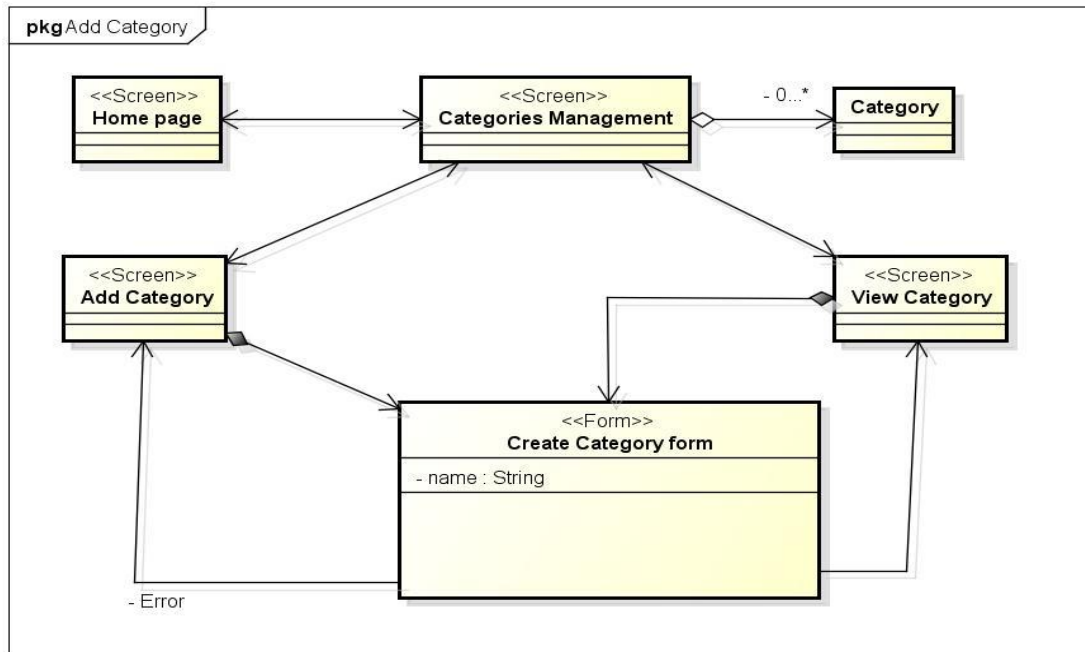


Figure D.35. Create category dataflow diagram and sequence diagram

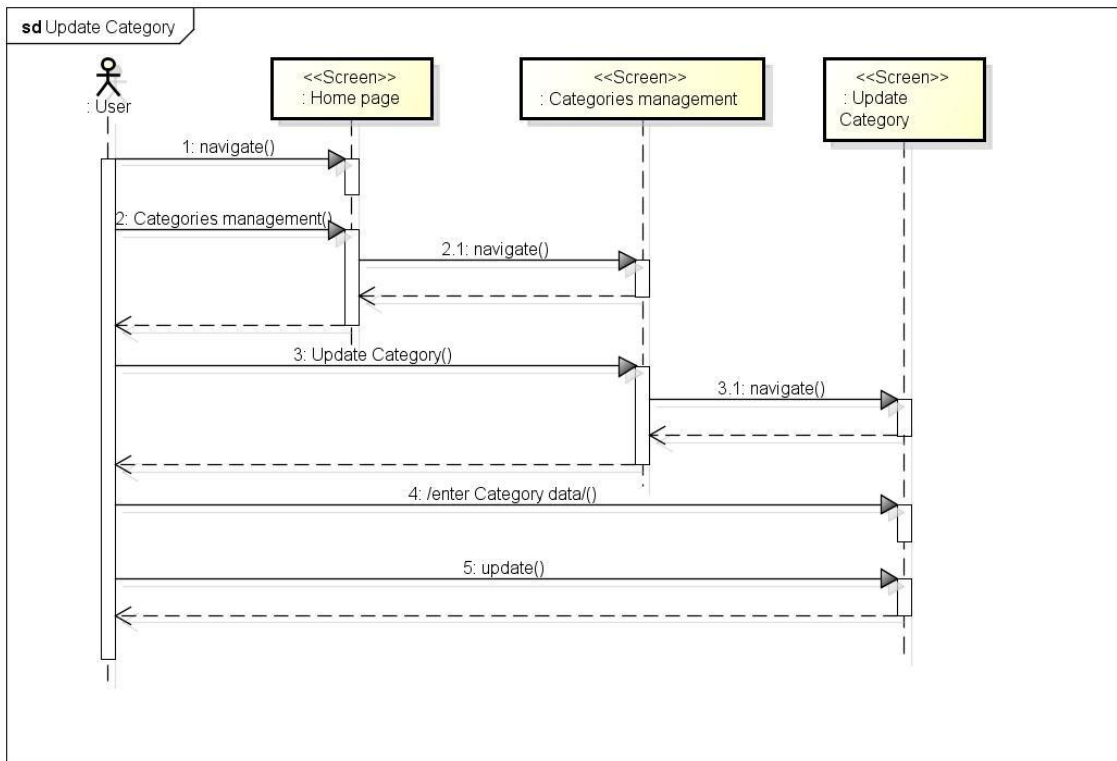
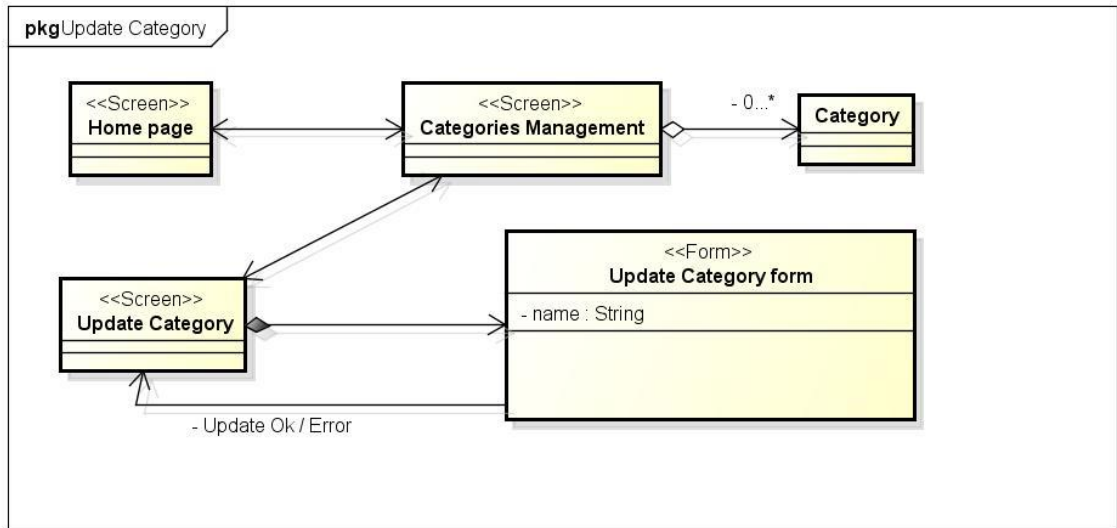


Figure D.36. Update category dataflow diagram and sequence diagram

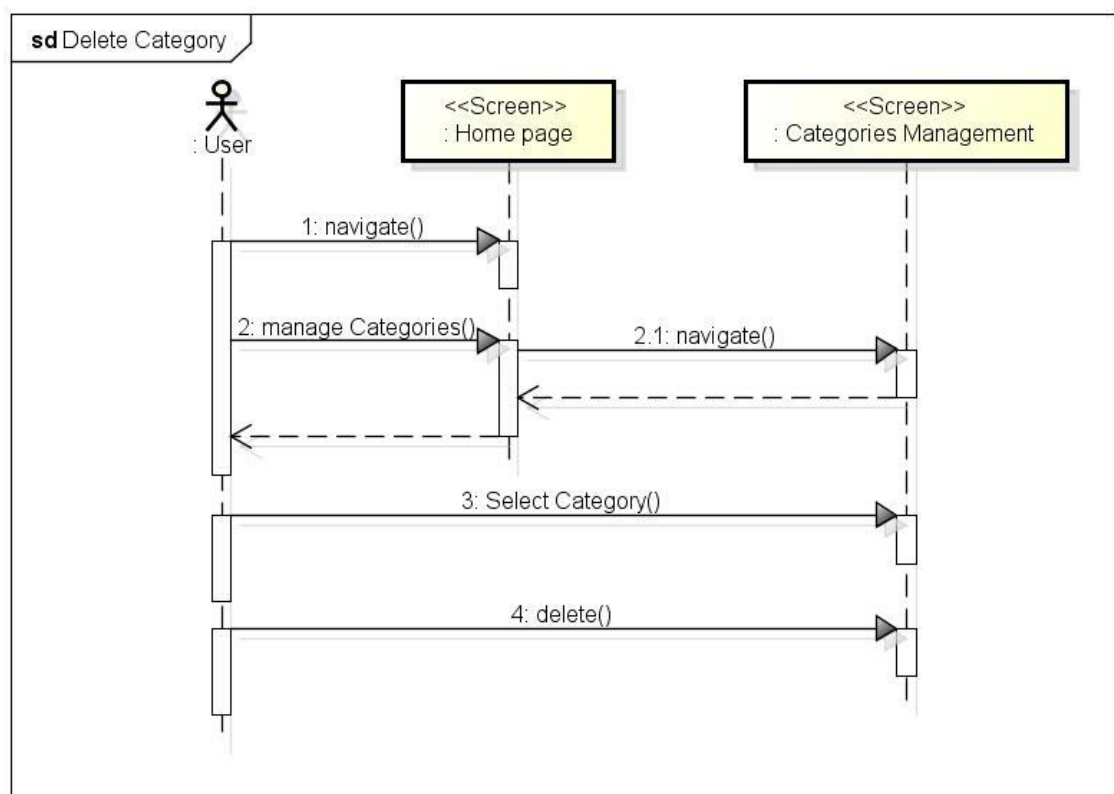
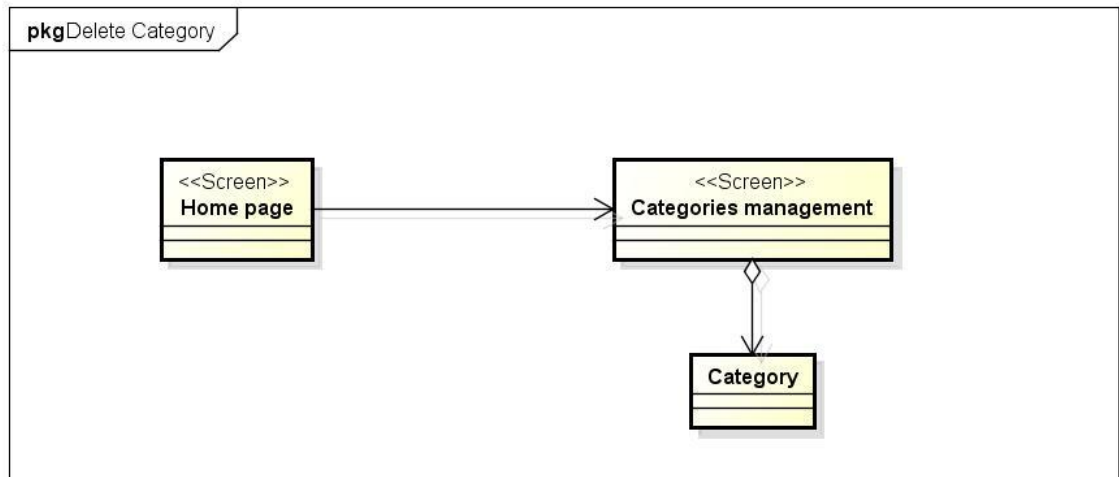


Figure D.37. Delete category dataflow diagram and sequence diagram

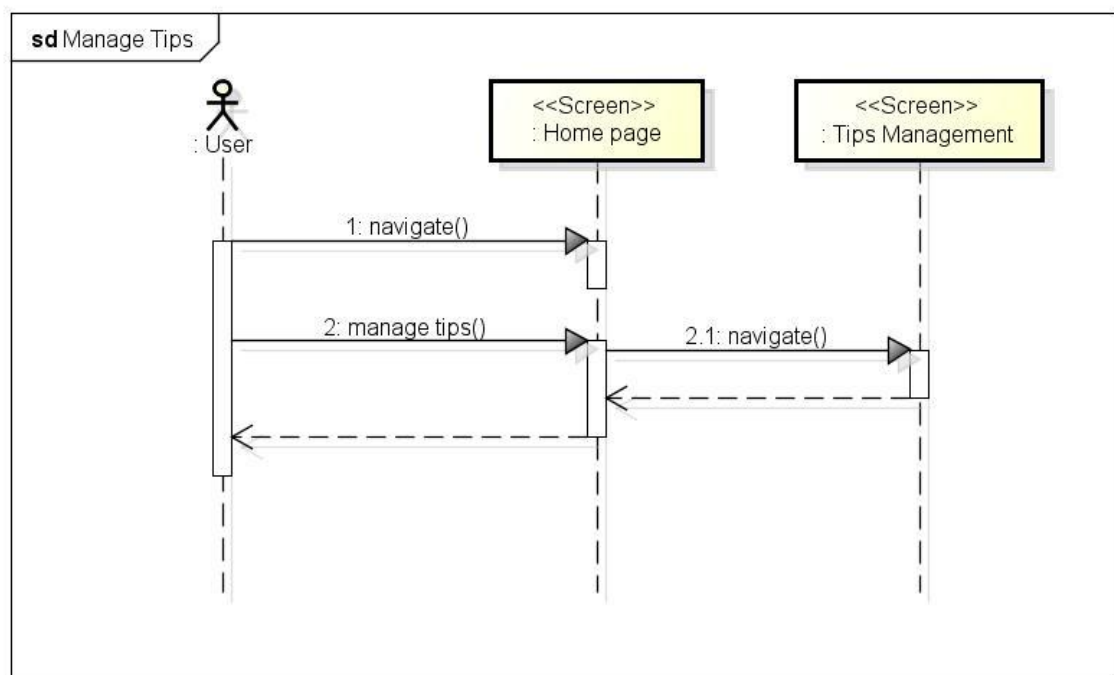
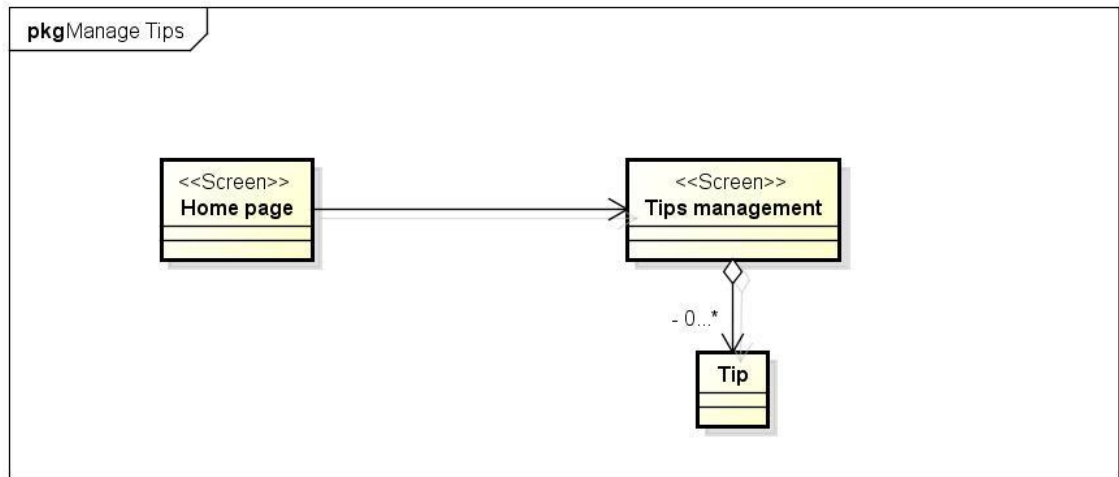


Figure D.38. Delete category dataflow diagram and sequence diagram

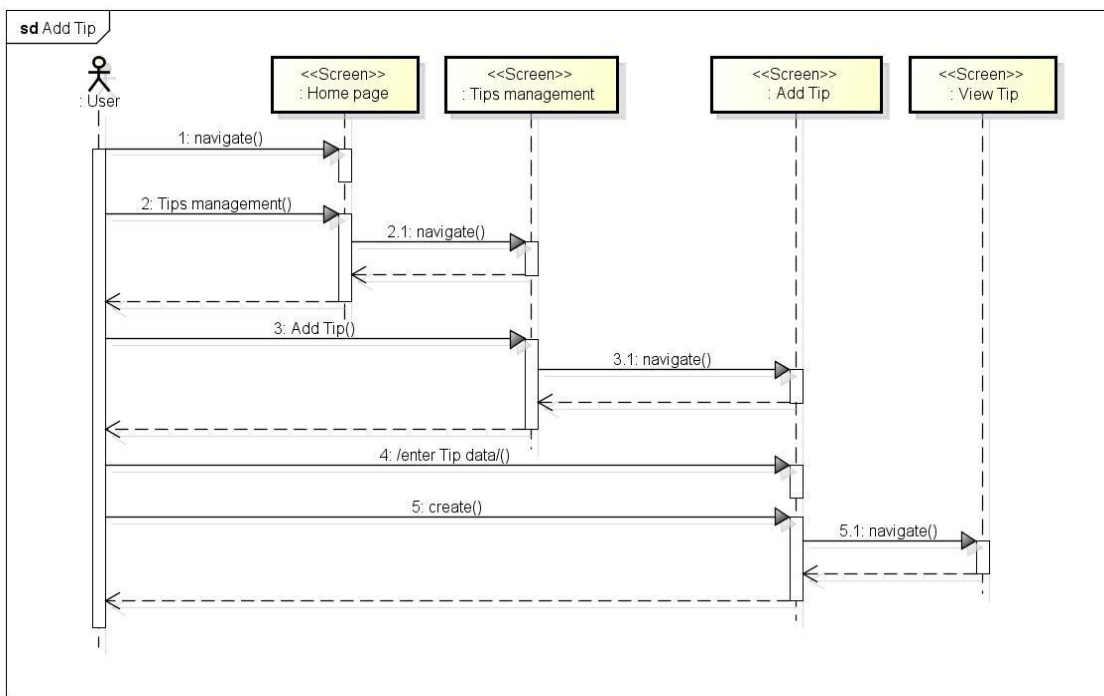
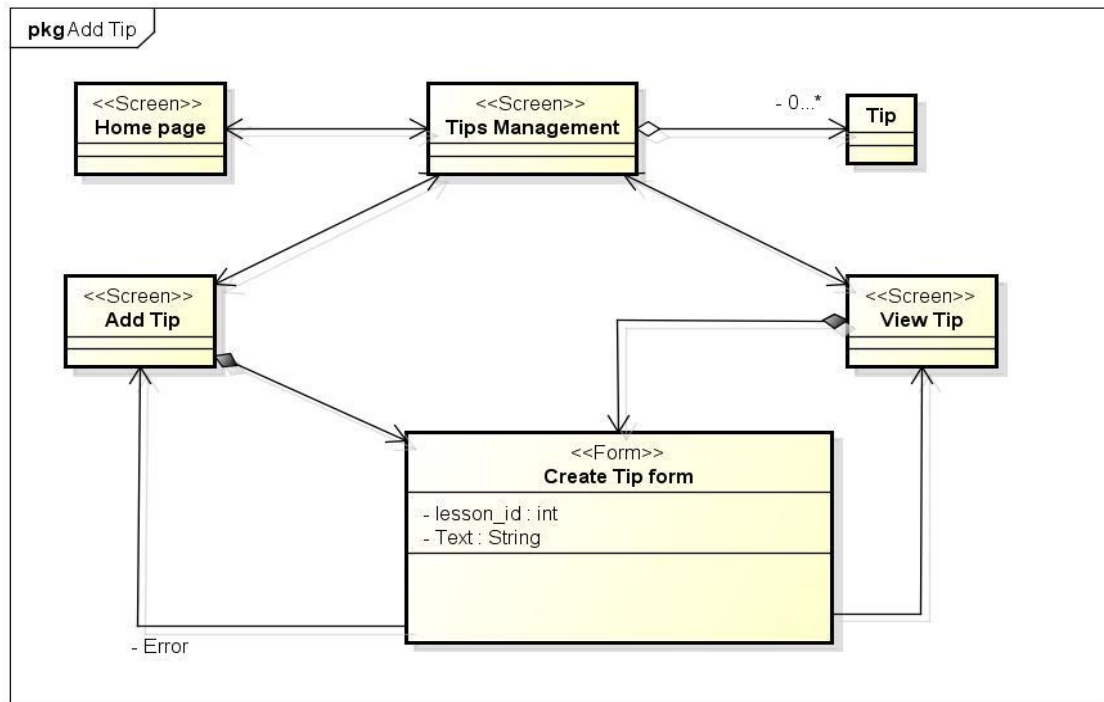


Figure D.39. Create tip dataflow diagram and sequence diagram

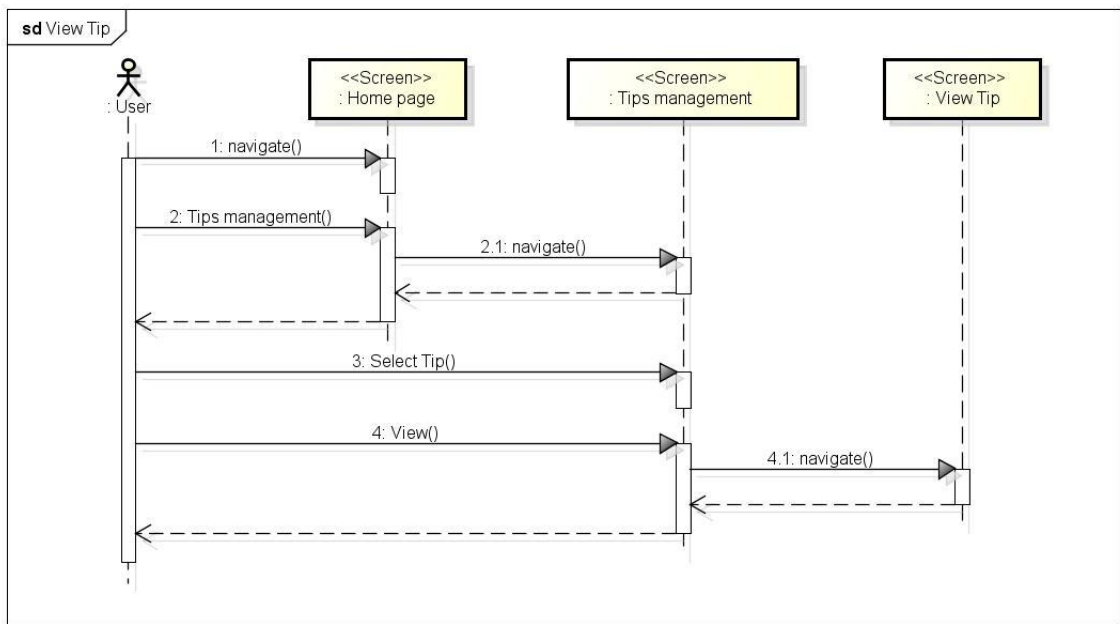
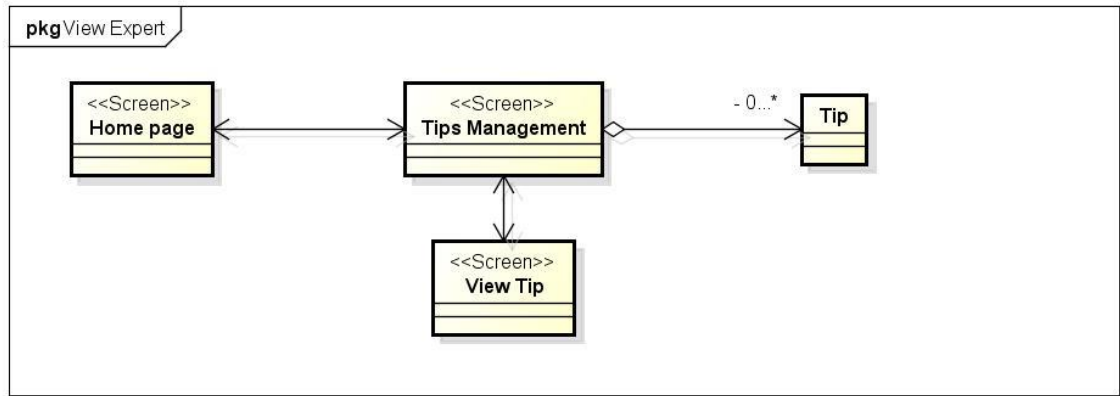


Figure D.40. View tip dataflow diagram and sequence diagram

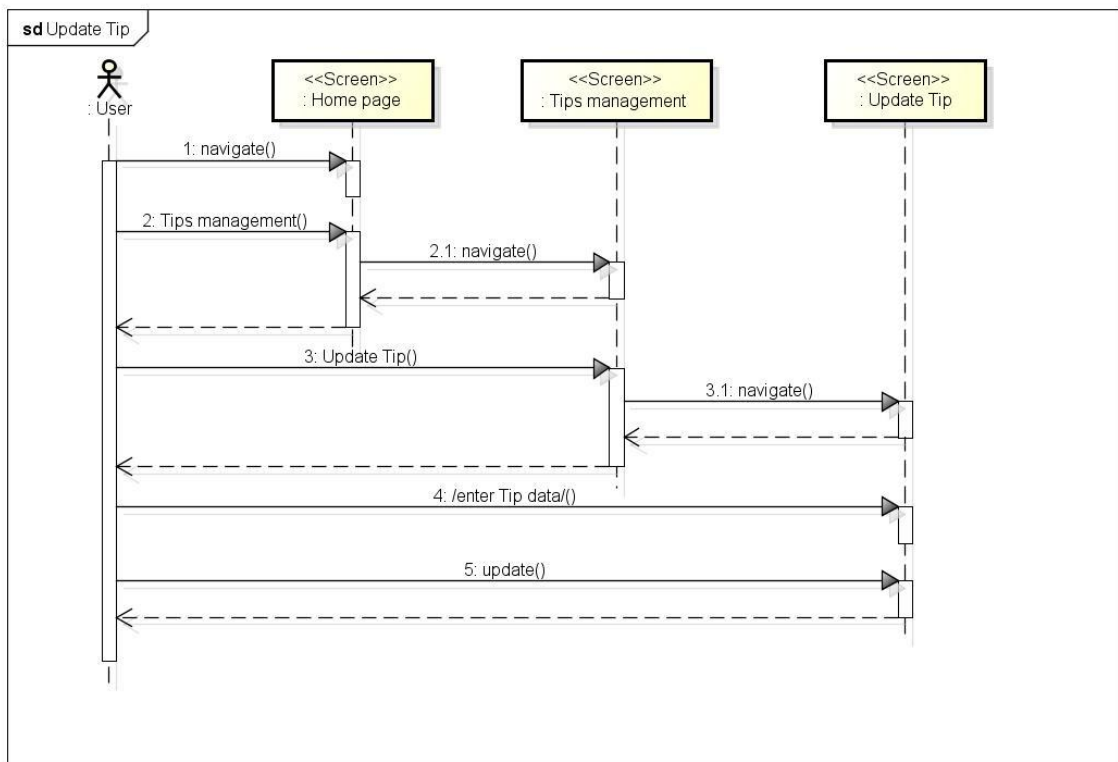
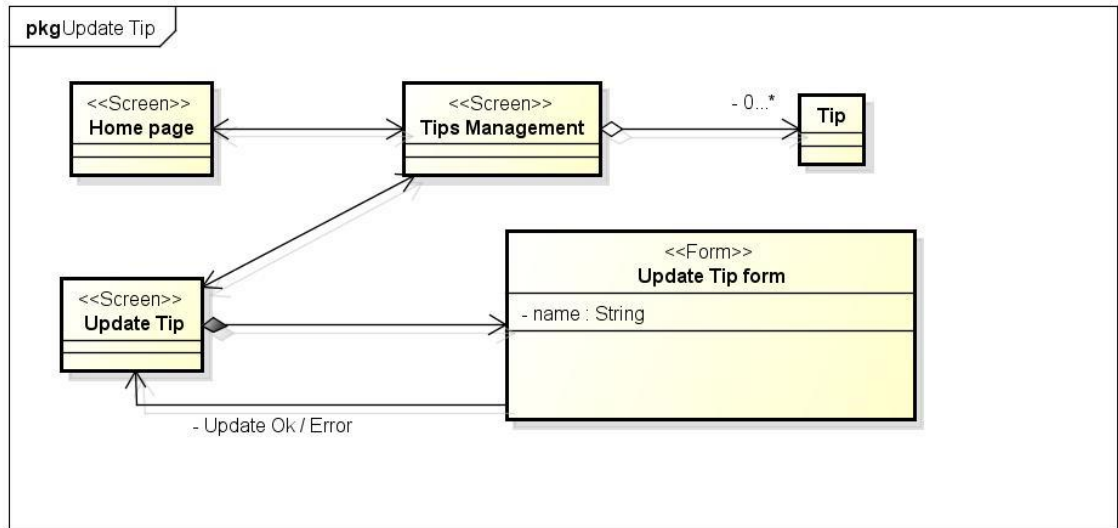


Figure D.41. Update tip dataflow diagram and sequence diagram

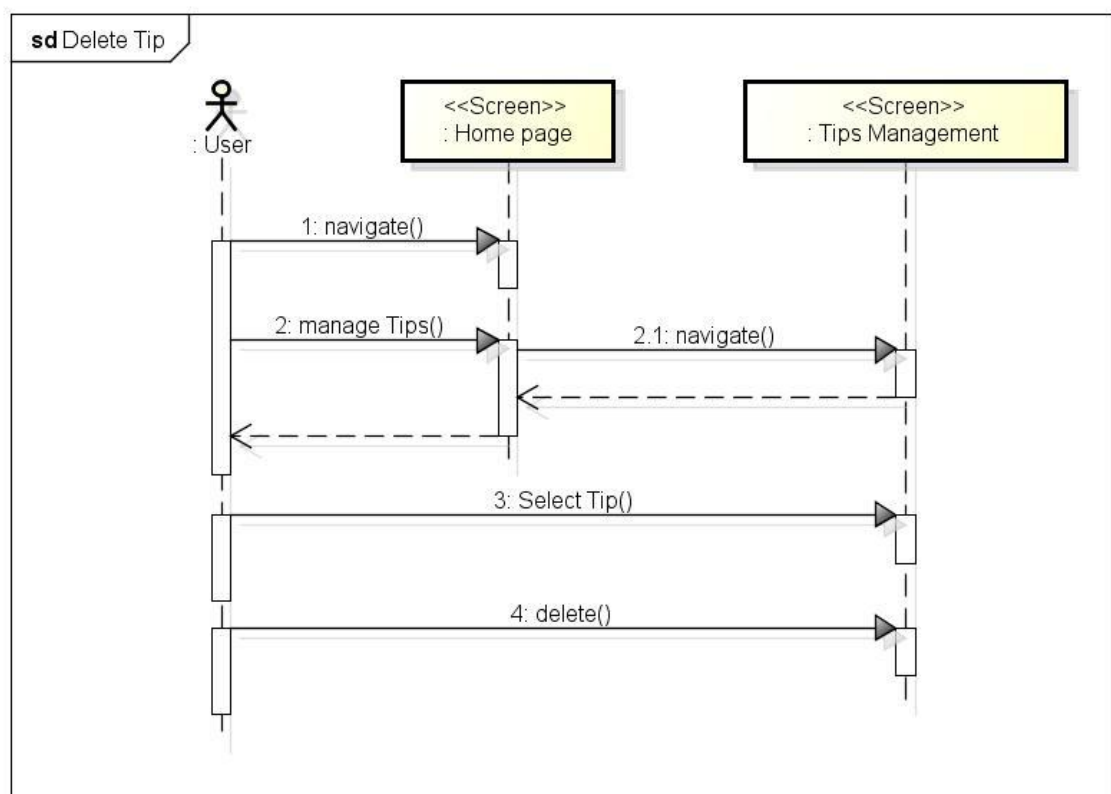
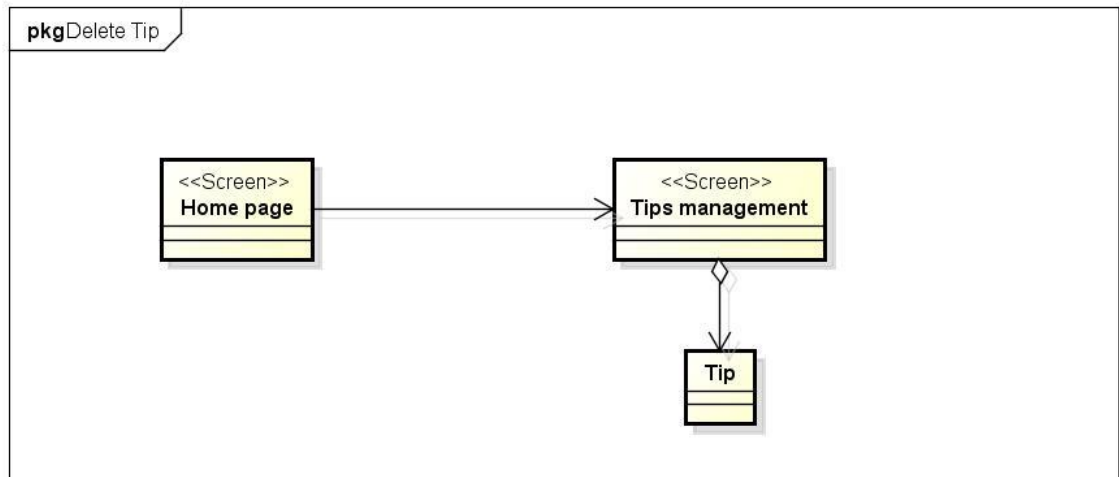


Figure D.42. Delete tip dataflow diagram and sequence diagram

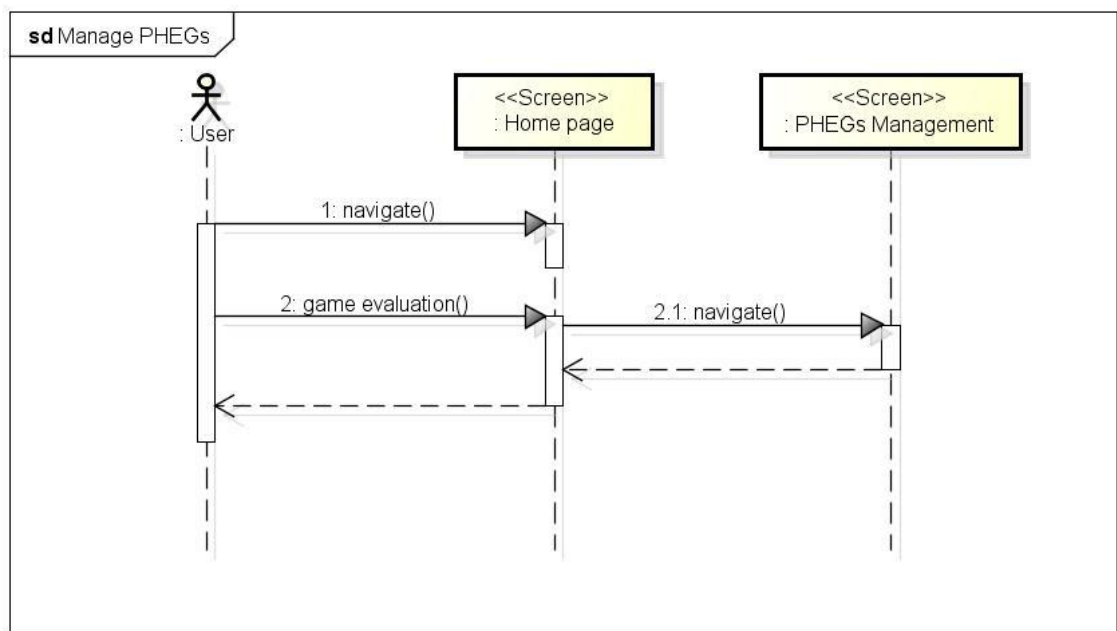
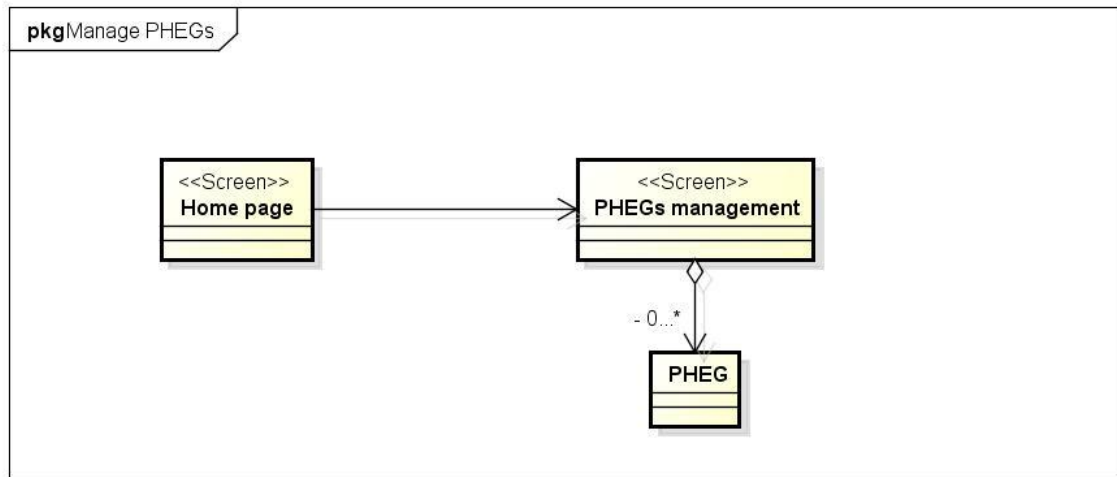


Figure D.43. Manage PHEGs dataflow diagram and sequence diagram

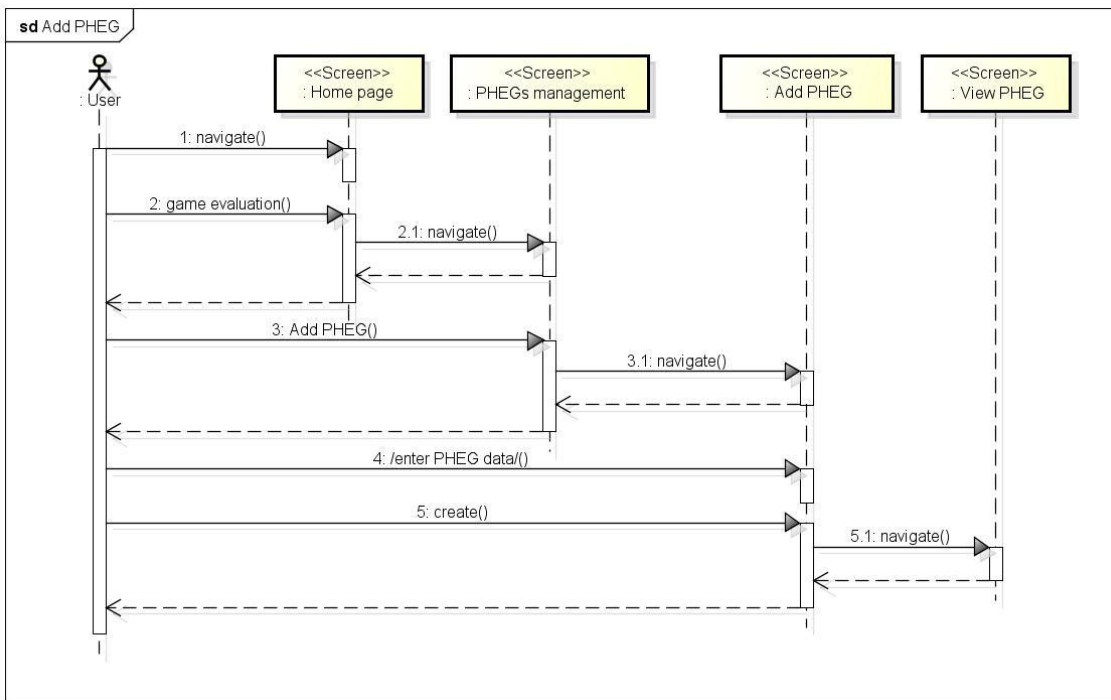
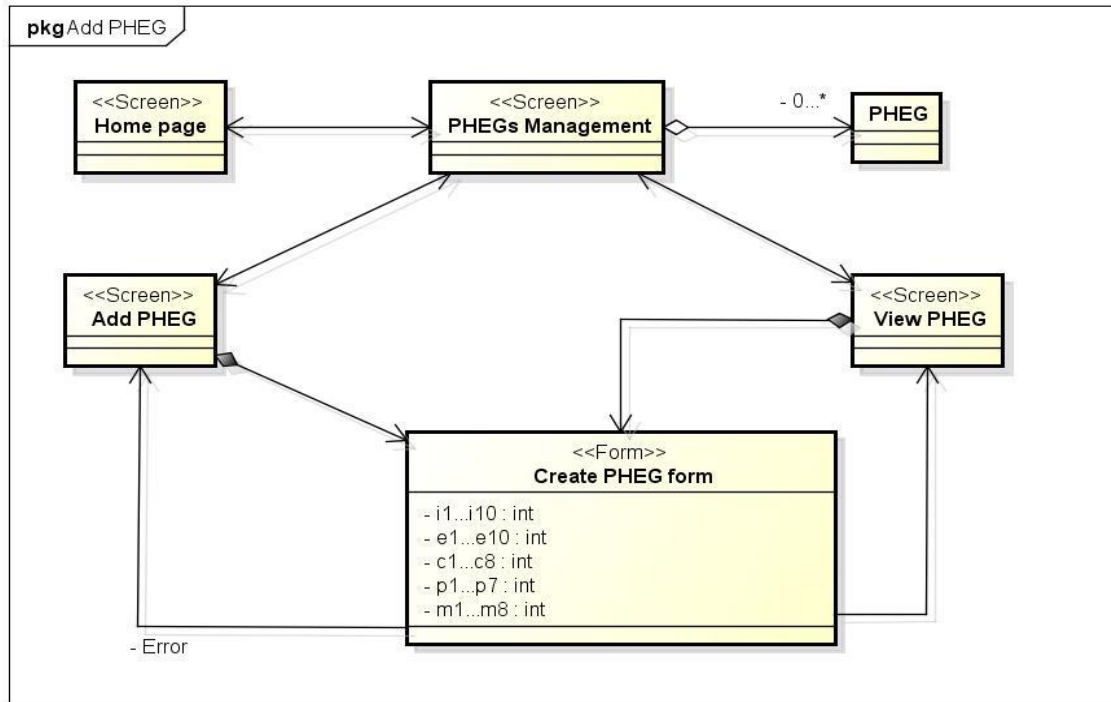


Figure D.44. Create PHEG dataflow diagram and sequence diagram

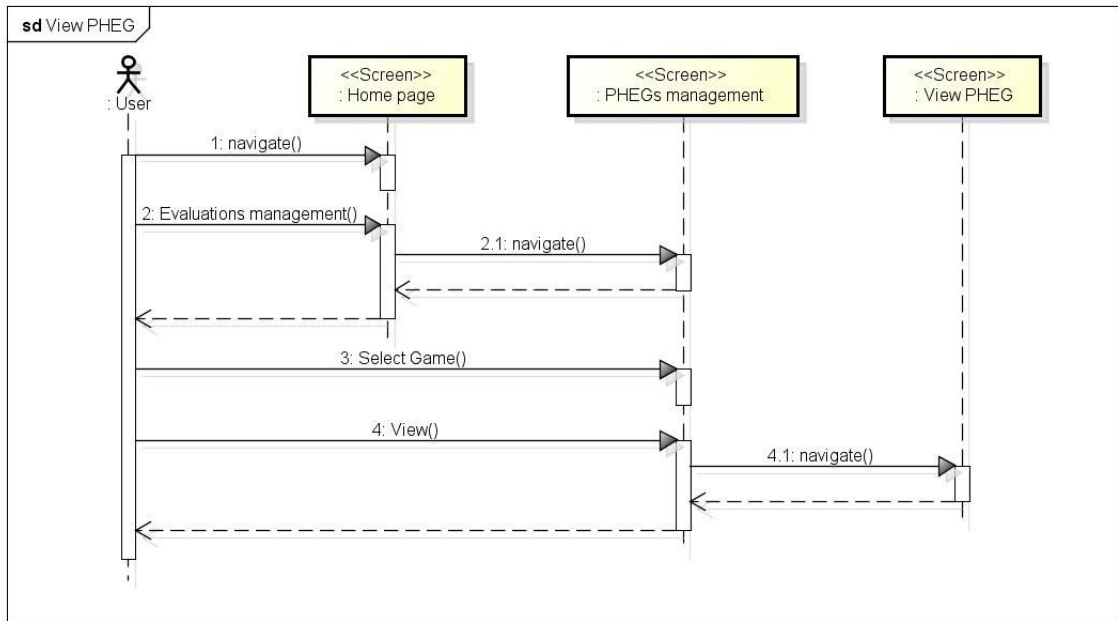
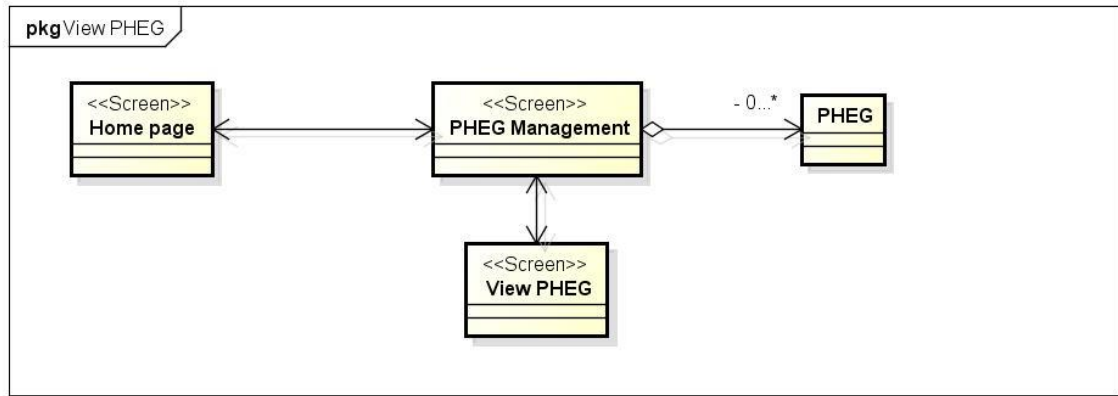


Figure D.45. View PHEG dataflow diagram and sequence diagram

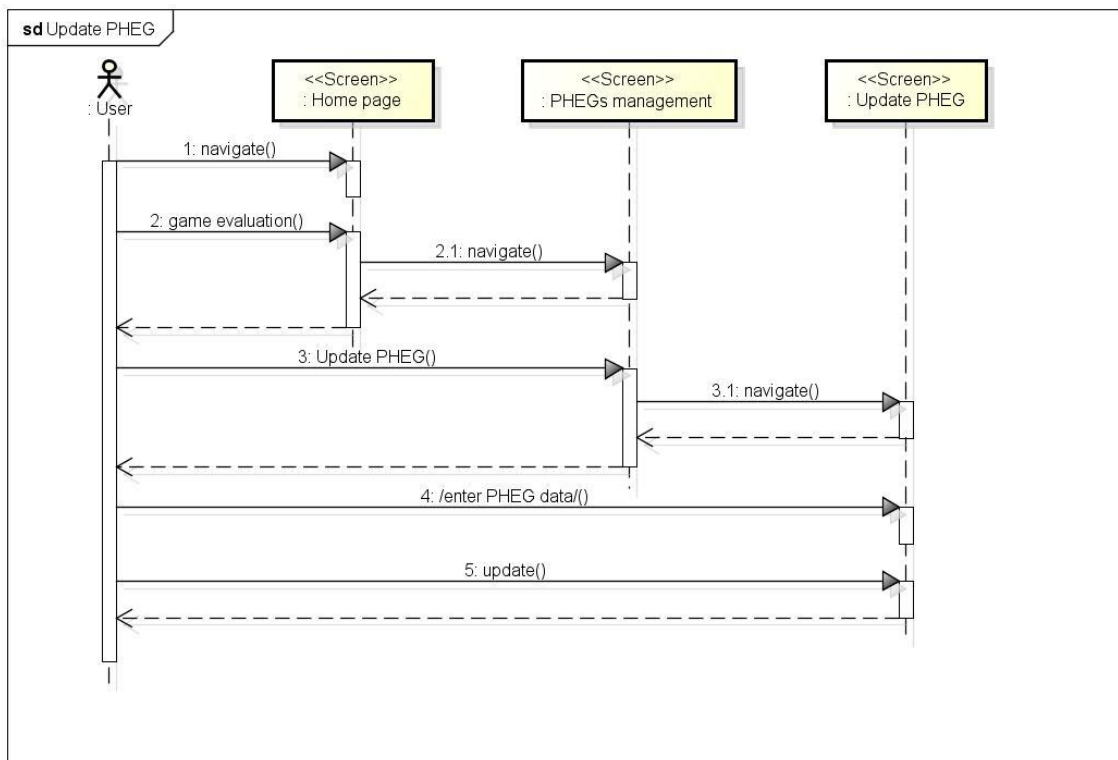
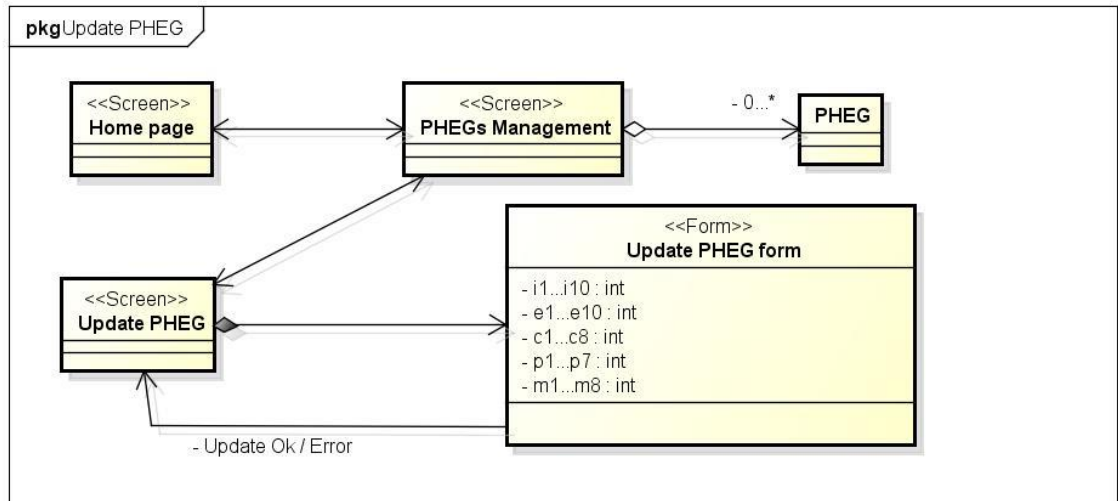


Figure D.46. Update PHEG dataflow diagram and sequence diagram

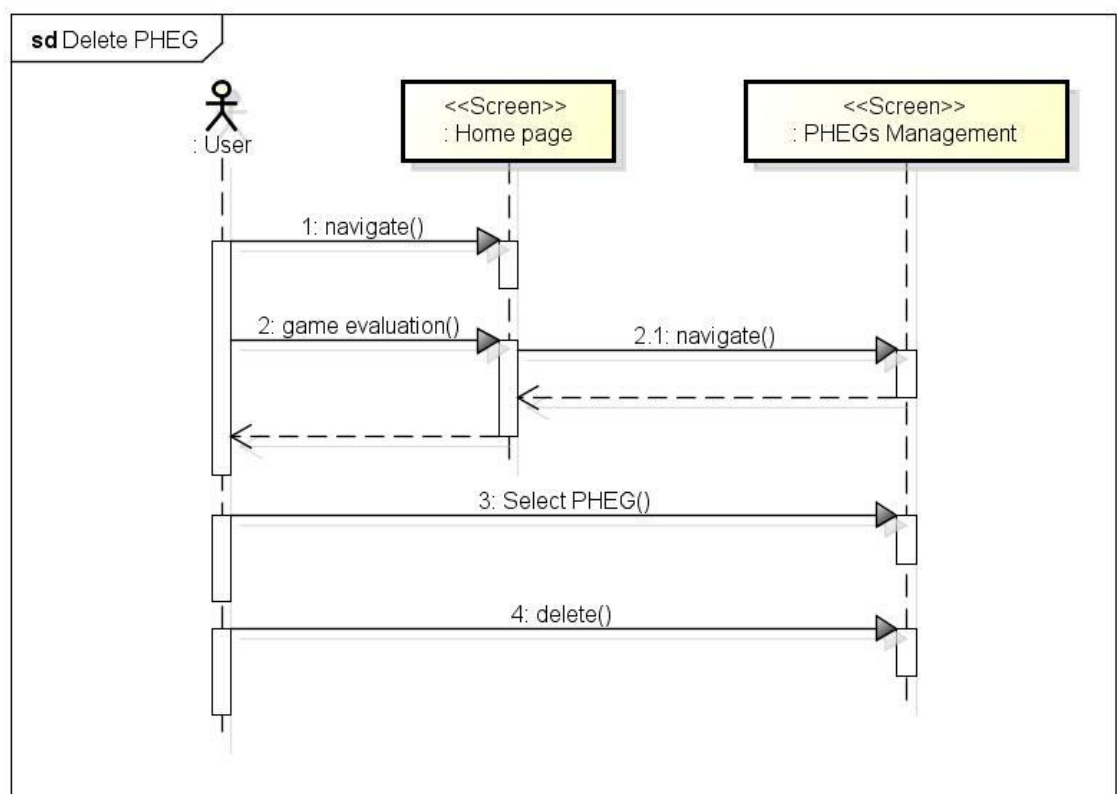
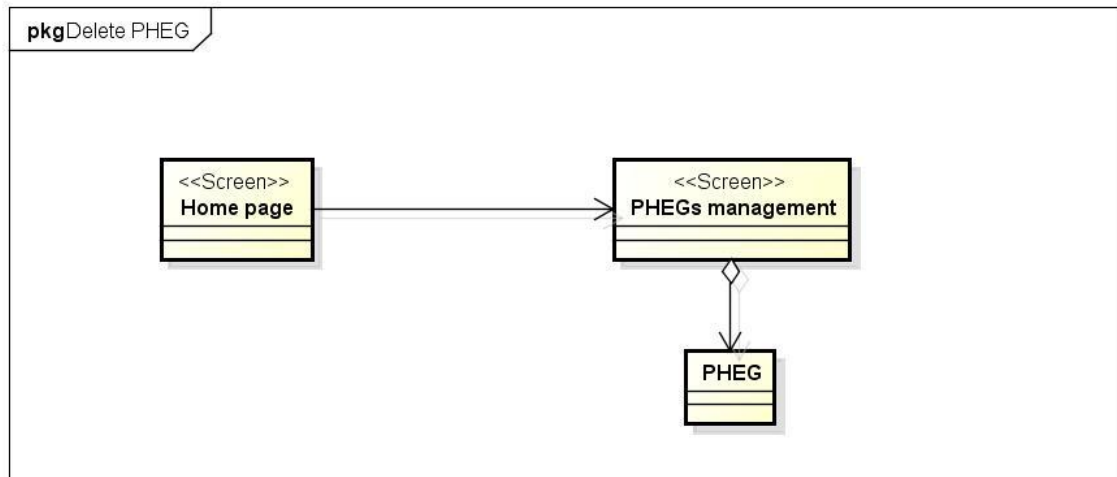


Figure D.47. Delete PHEG dataflow diagram and sequence diagram

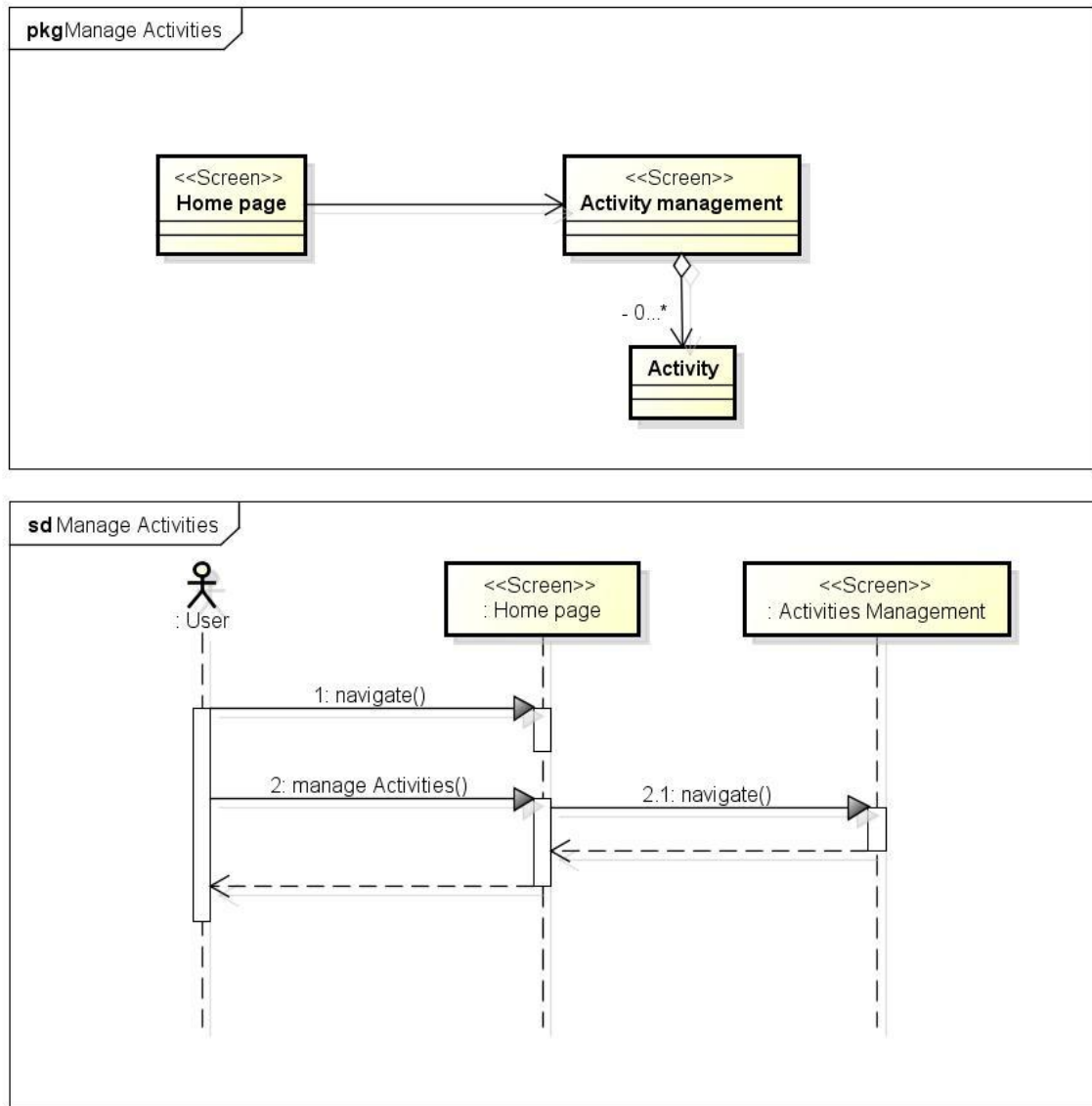


Figure D.48. Manage activities dataflow diagram and sequence diagram

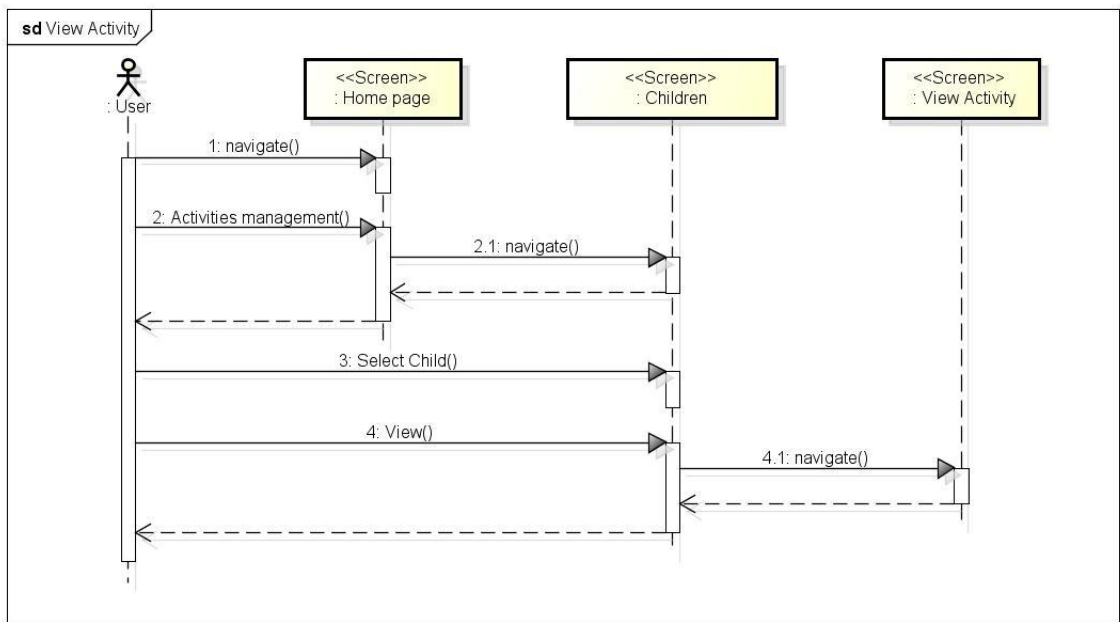
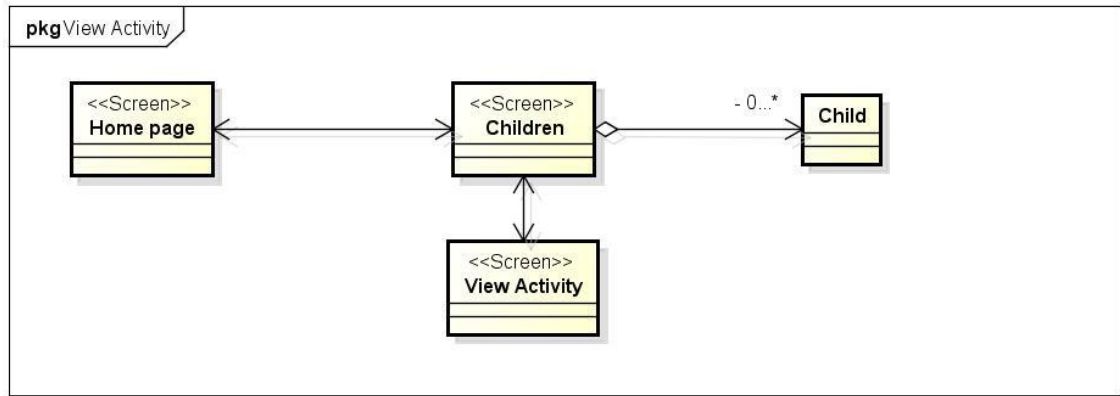


Figure D.49. View activity dataflow diagram and sequence diagram

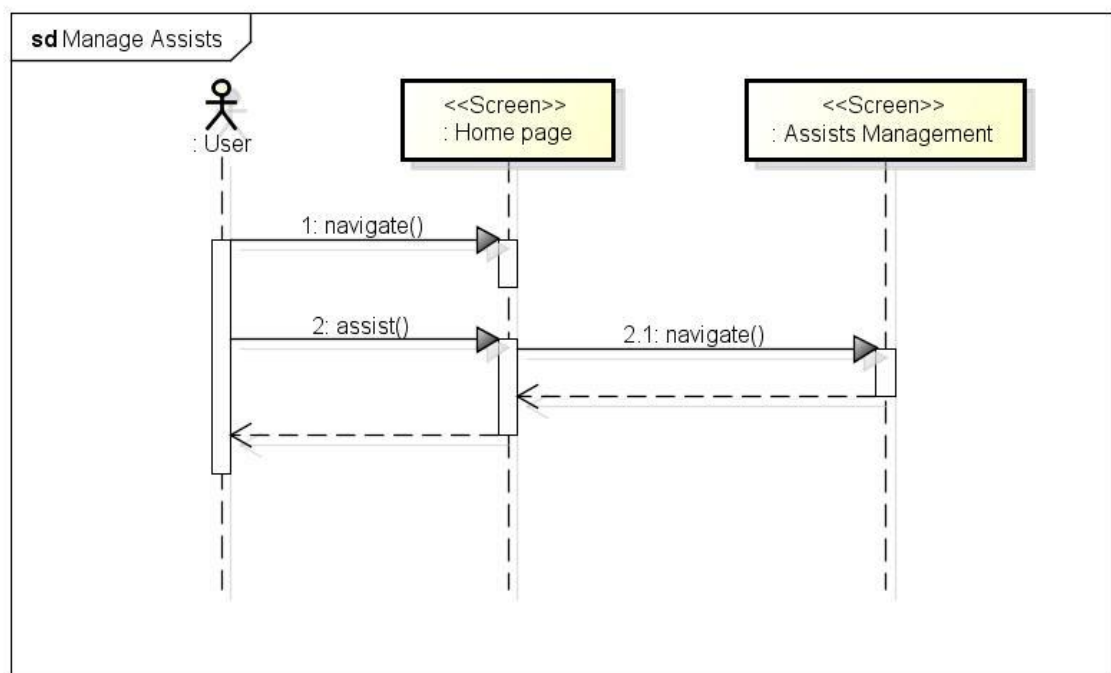
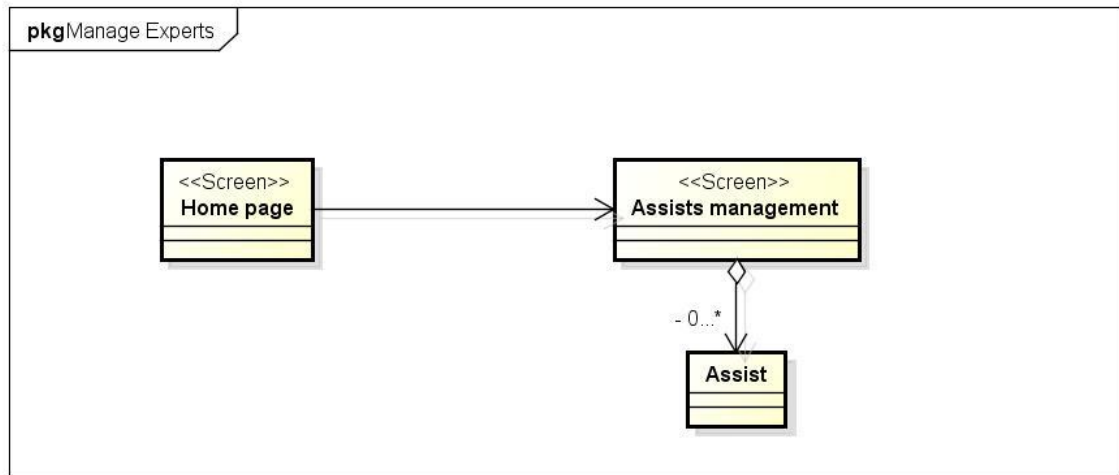


Figure D.50. Manage assists dataflow diagram and sequence diagram

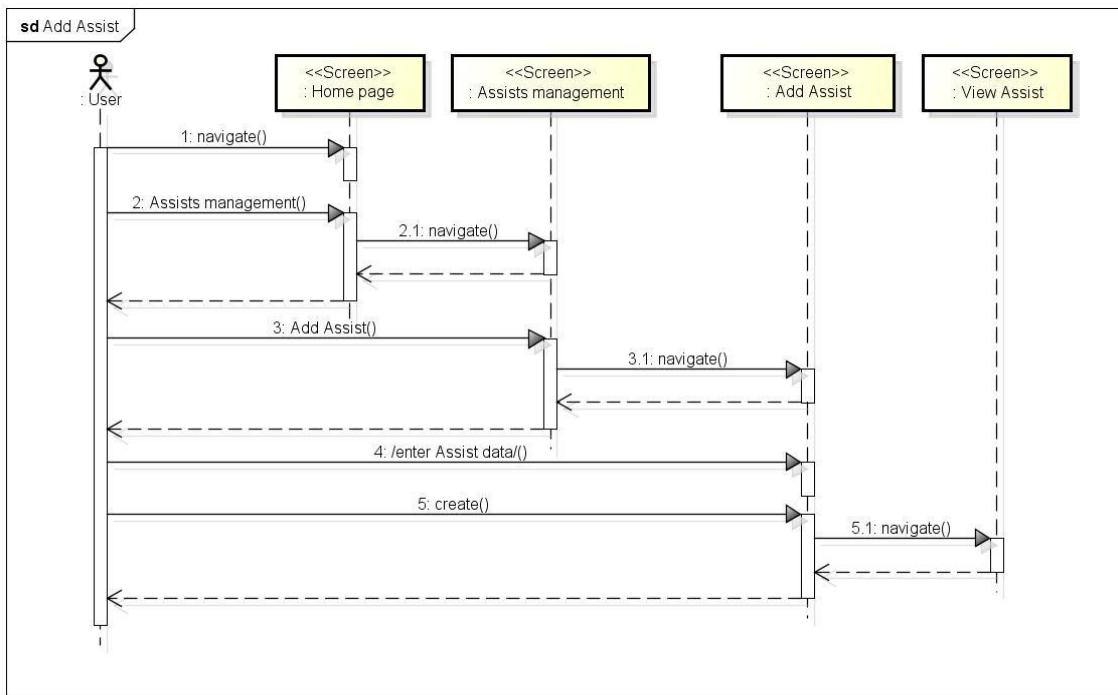
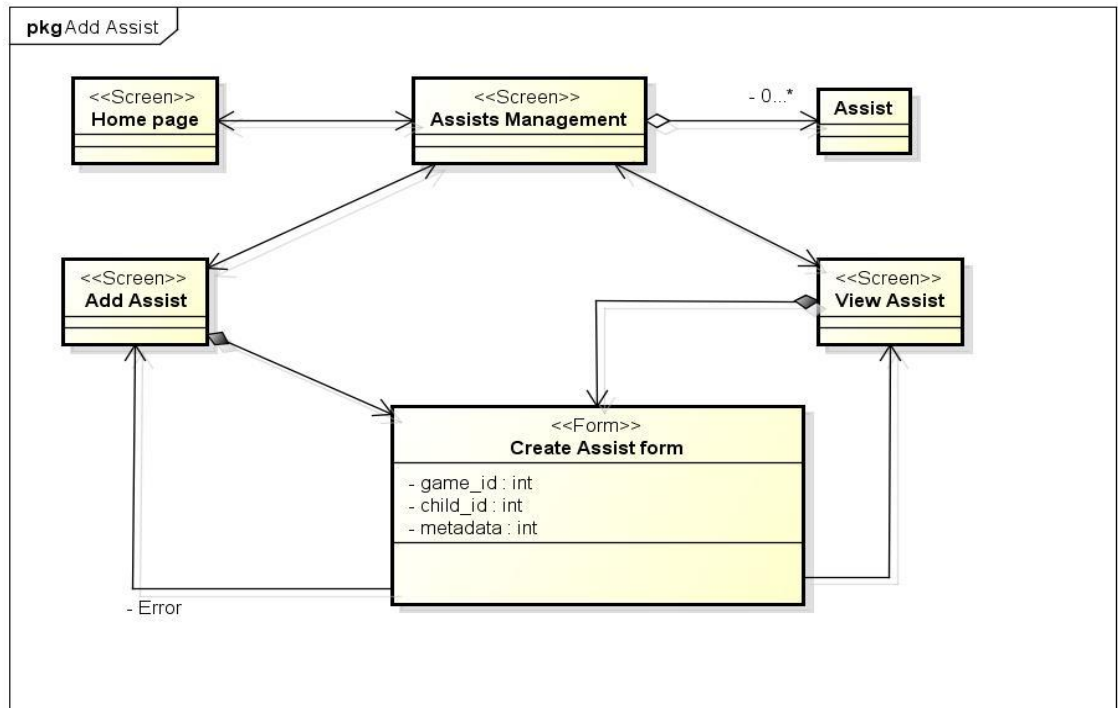


Figure D.51. Create assist dataflow diagram and sequence diagram

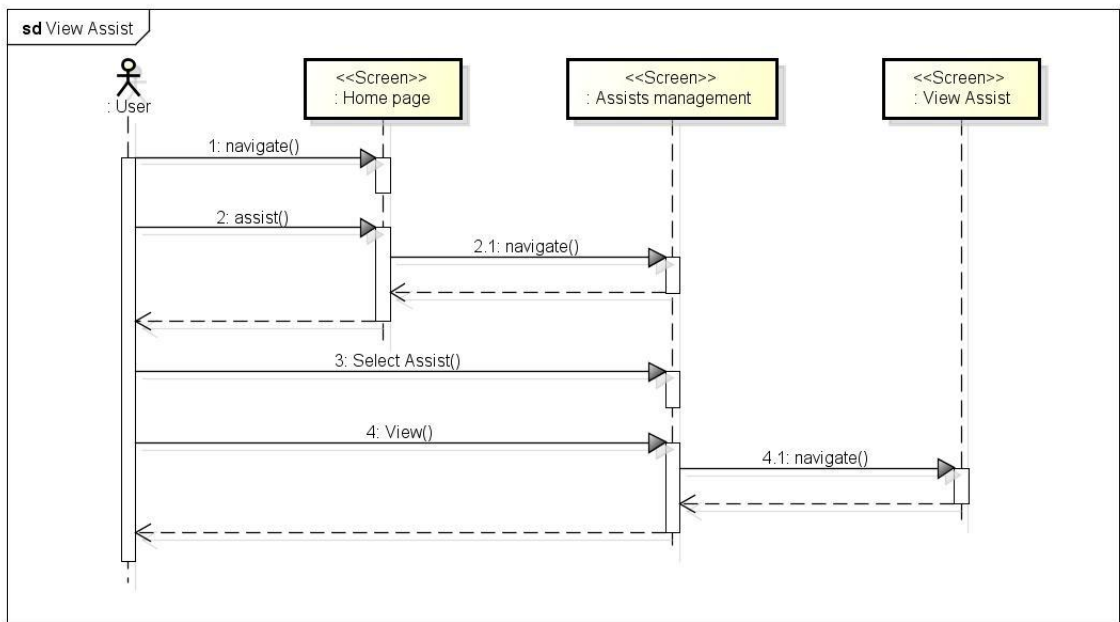
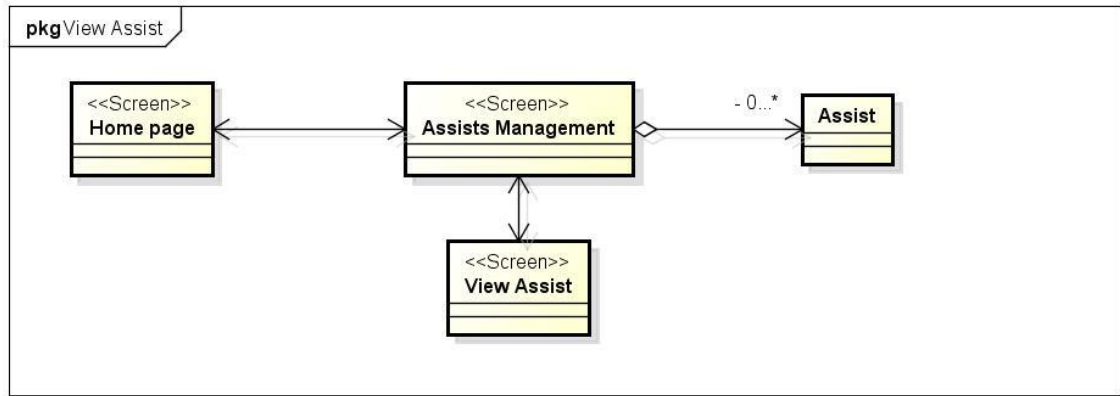


Figure D.52. View assist dataflow diagram and sequence diagram

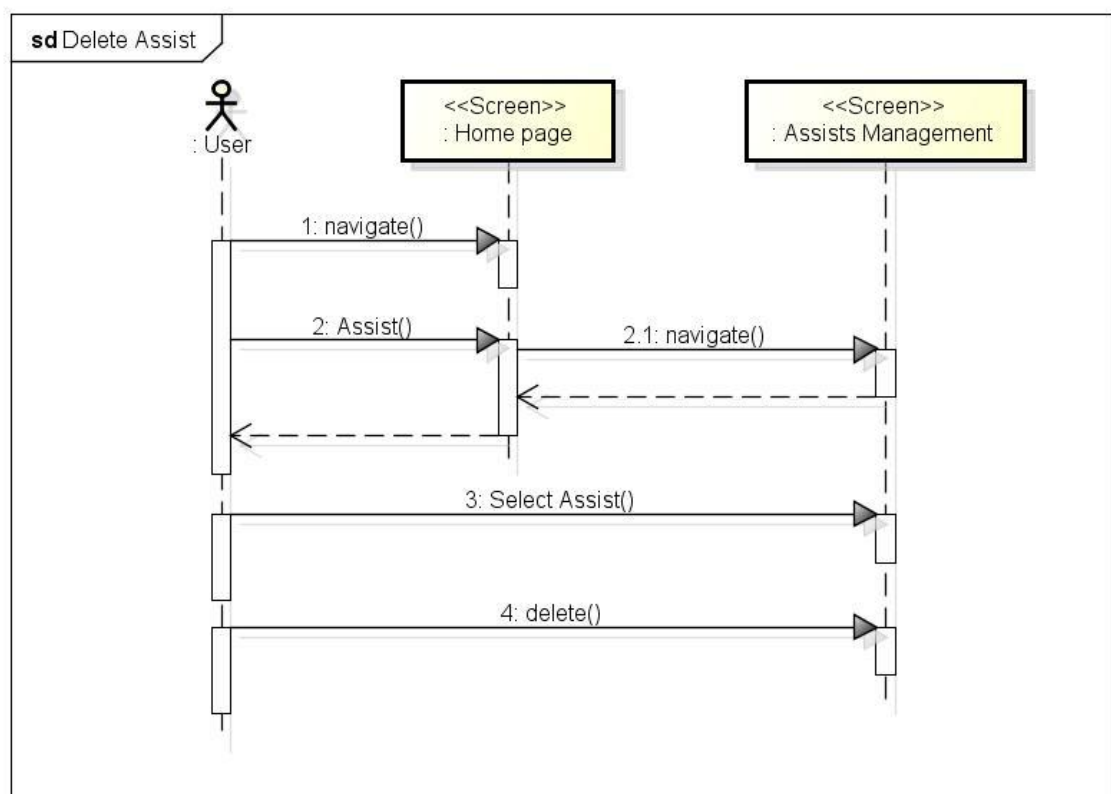
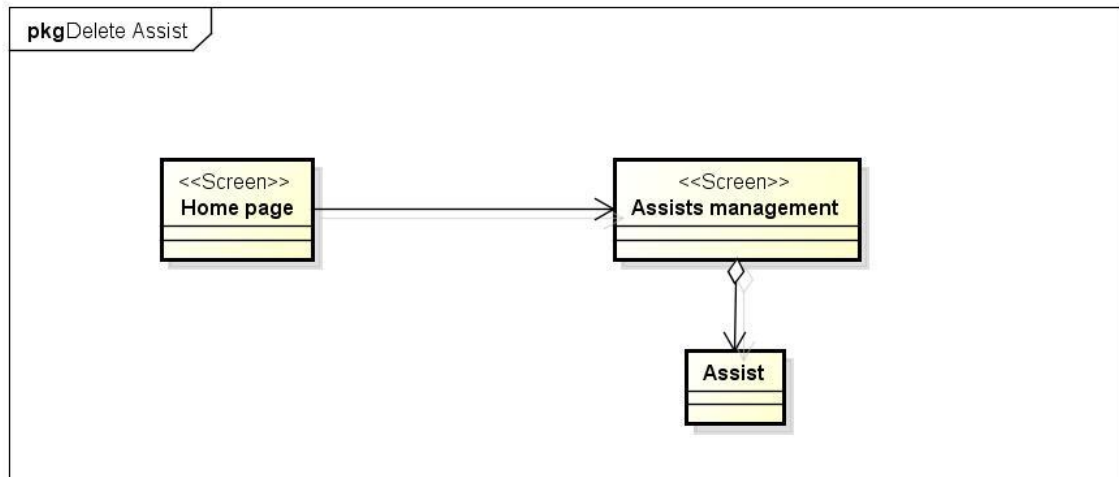


Figure D.53. View assist dataflow diagram and sequence diagram

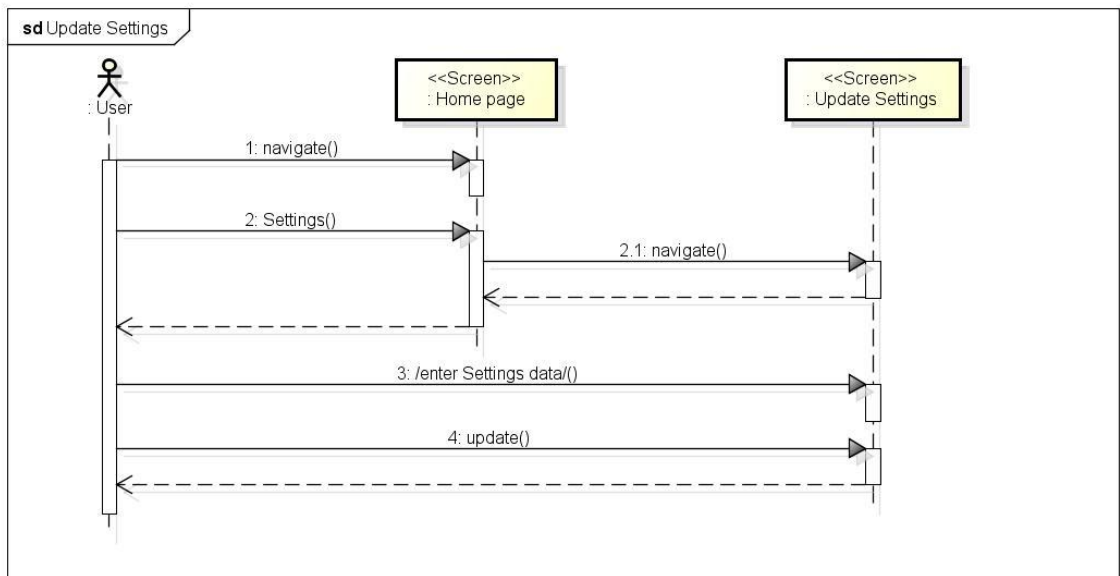
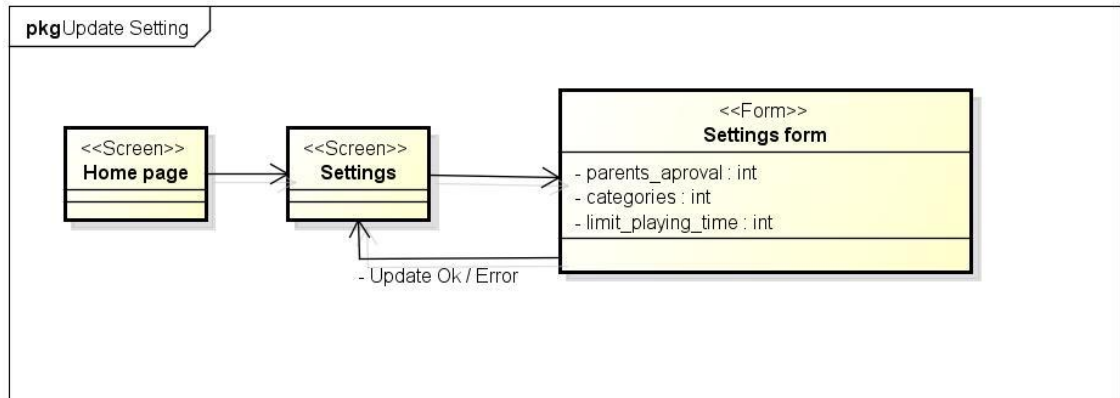


Figure D.54. Update setting dataflow diagram and sequence diagram

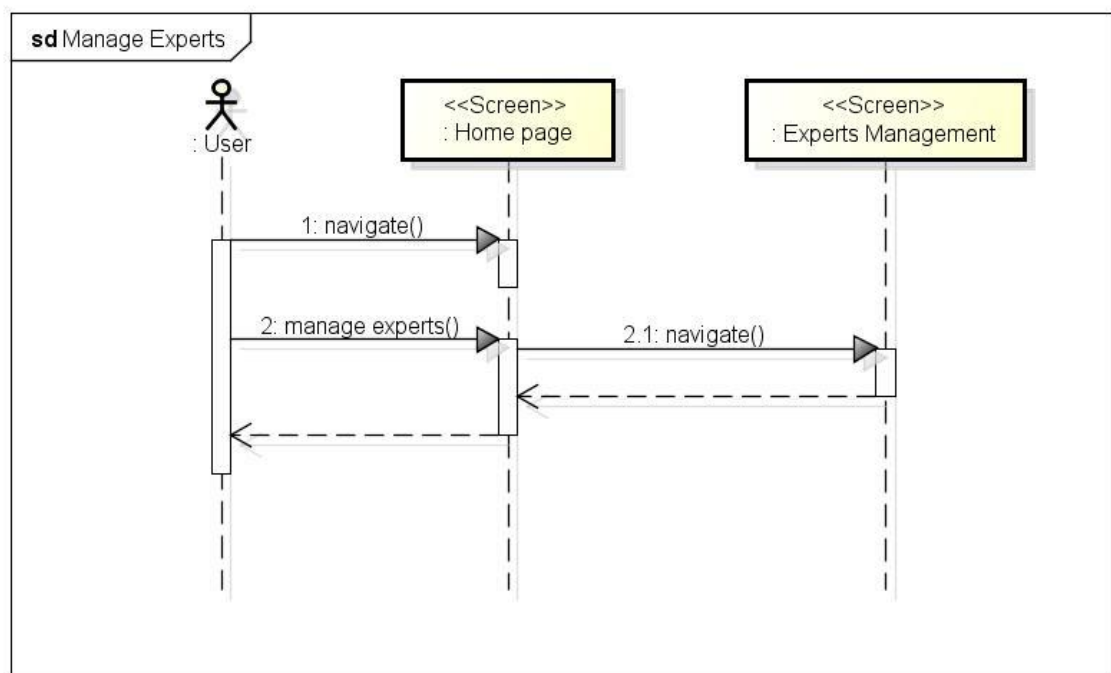
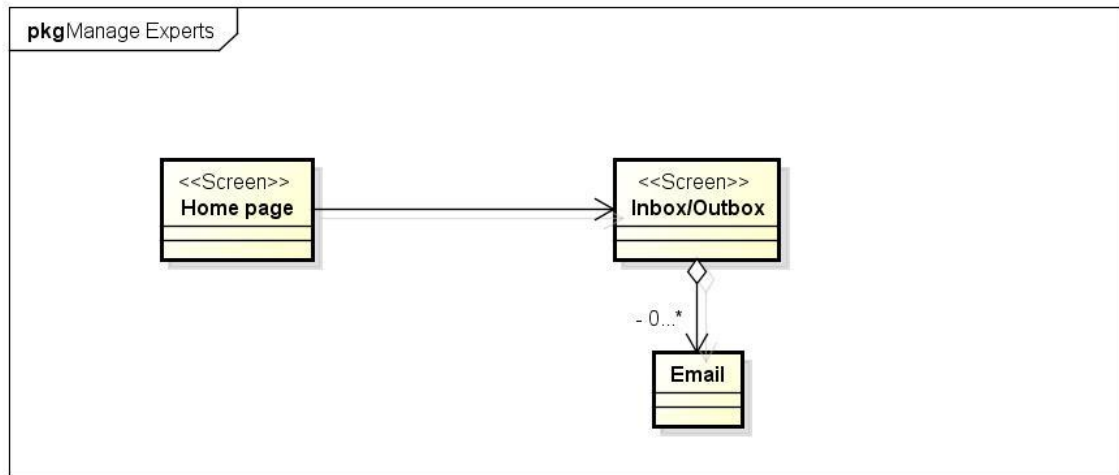


Figure D.55. Manage Inbox/Outbox dataflow diagram and sequence diagram

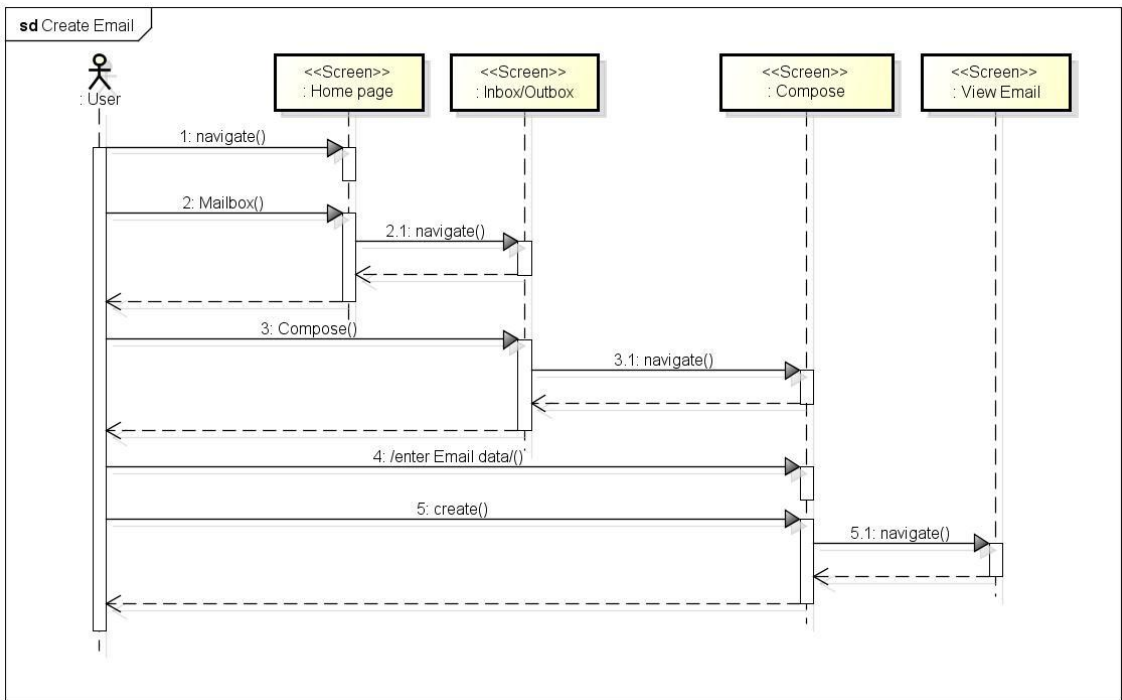
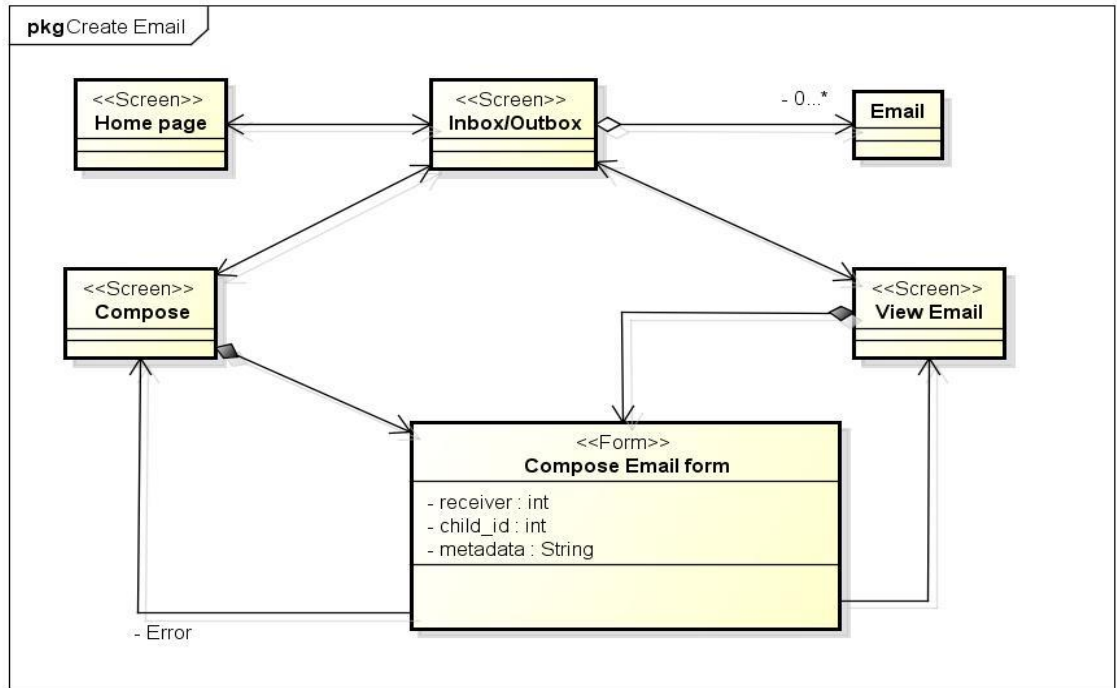


Figure D.56. Compose Mail dataflow diagram and sequence diagram

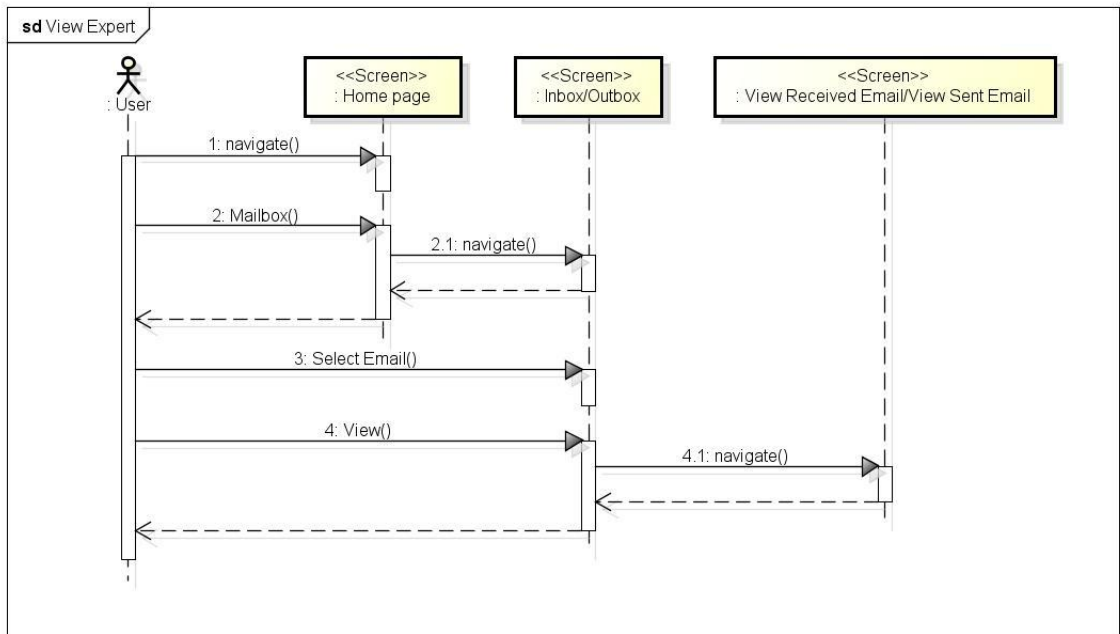
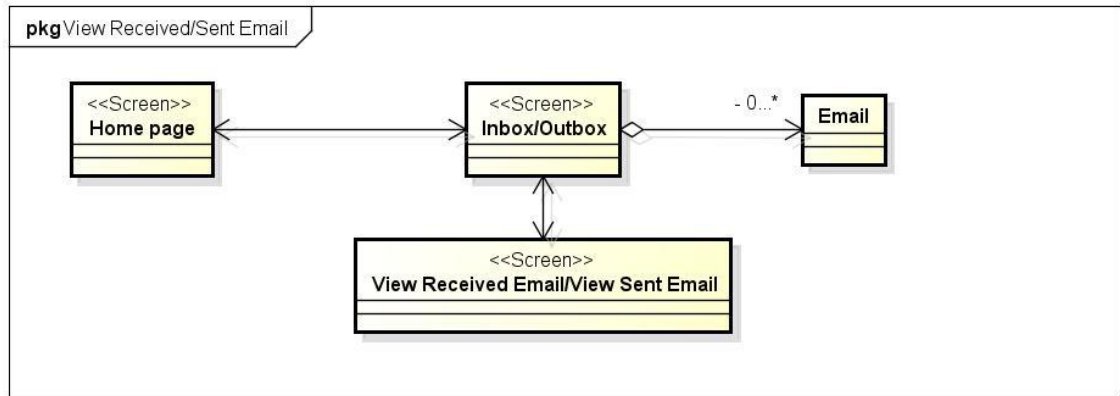


Figure D.57. View E-mail dataflow diagram and sequence diagram

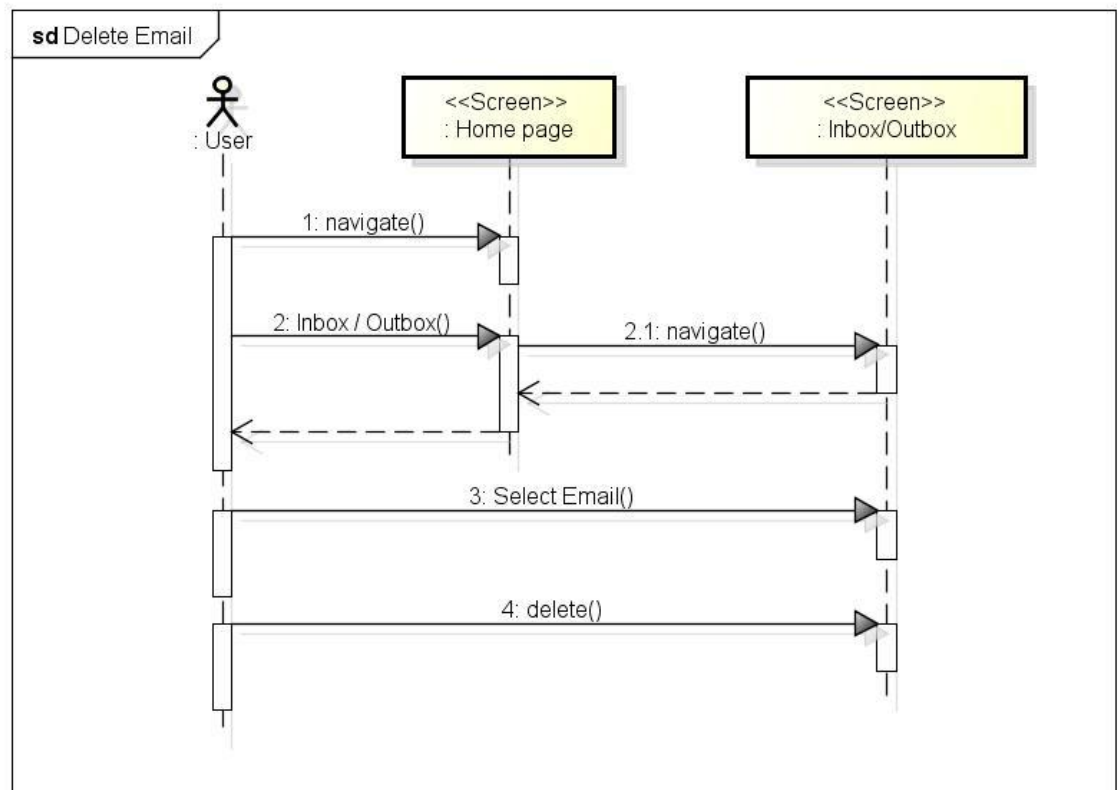
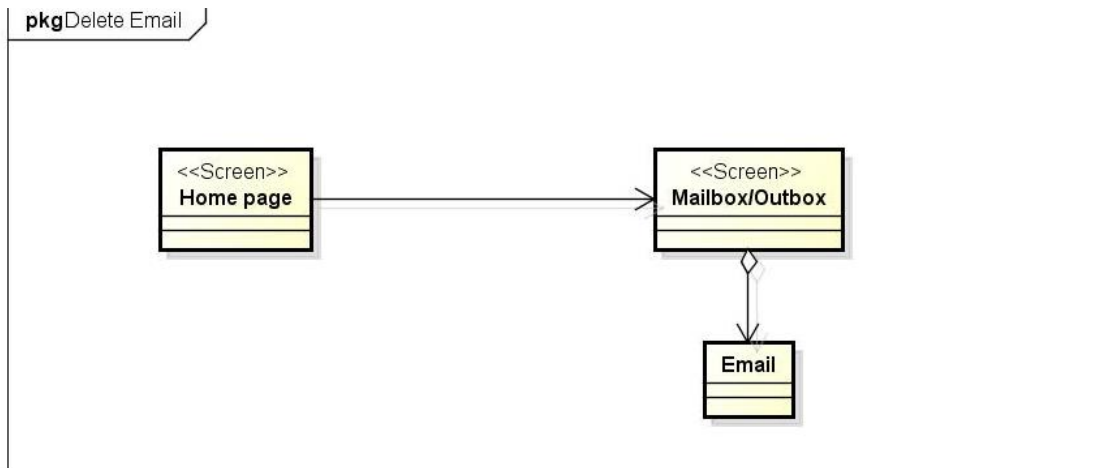


Figure D.58. Delete e-mail dataflow diagram and sequence diagram

Appendix E. Models

Table E.1. tableName() method

Input	-
Output	Returns the name of the associated database table
Pre-conditions	-
Post-conditions	-

Table E.2. model() method

Input	-
Output	Returns the static model of the specified Active Record class
Pre-conditions	-
Post-conditions	-

Table E.3. rules () method

Input	-
Output	Returns array of the validation rules for attributes
Pre-conditions	-
Post-conditions	-

Table E.4. attributeLabels () method

Input	-
Output	Returns the array of the attribute labels. Attribute labels are mainly used in error messages of validation. This method allows you to explicitly specify attribute labels
Pre-conditions	-
Post-conditions	-

Table E.5. relations () method

Input	-
Output	Returns array of database objects relations
Pre-conditions	-
Post-conditions	-

Table E.6. search () method

Input	-
Output	Returns an instance of CActiveDataProvider (Yii Framework's class) which gives database records that match the conditions
Pre-conditions	-
Post-conditions	-

Table E.7. beforeValidate () method

Input	-
Output	Boolean
Pre-conditions	-
Post-conditions	This method is invoked before validation starts

Table E.8. afterValidate () method

Input	-
Output	Boolean
Pre-conditions	This method is invoked after validation ends
Post-conditions	-

Table E.9. defaultScope () method

Input	-
Output	Returns array of the default named scope that should be implicitly applied to all queries for this model
Pre-conditions	-
Post-conditions	-

Table E.10. encrypt () method

Input	password. String
Output	Returns the MD5 hash of the input
Pre-conditions	-
Post-conditions	-

Table E.11. getChild () method

Input	-
Output	Returns array of children which the user have access
Pre-conditions	-
Post-conditions	-

Table E.12. getGame () method

Input	-
Output	Returns array of games which the user have access
Pre-conditions	-
Post-conditions	-

Table E.13. getReceiver () method

Input	-
Output	Returns array of mailboxes which the user have access to send e-mail to them
Pre-conditions	-
Post-conditions	-

Table E.14. inboxSearch () method

Input	-
Output	Returns an instance of CActiveDataProvider (Yii Framework's class) which gives database records that match the conditions and they are received e-mails
Pre-conditions	-
Post-conditions	-

Table E.15. outboxSearch () method

Input	-
Output	Returns an instance of CActiveDataProvider (Yii Framework's class) which gives database records that match the conditions and they are sent e-mails
Pre-conditions	-
Post-conditions	-

Table E.16. searchNotEvaluated () method

Input	-
Output	Returns an instance of CActiveDataProvider (Yii Framework's class) which gives database records that match the conditions and they are not evaluated games
Pre-conditions	-
Post-conditions	-

Appendix F. Users' Permissions

Table F.1. List of users' permissions

User Type	Administrator	Expert	Parent
Controller / Action			
Site Controller			
Index. Represents base HTML contents	X	X	X
Error. Represents error messages	X	X	X
Login. Represents login Web page	X	X	X
Logout. Handles logout procedure	X	X	X
Admin Controller			
Dashboard. Represents home page	X		
View. Displays a particular model	X		
Create. Creates a new model	X		
Update. Updates a particular model	X		
Admin. Manages all models	X		
Delete. Deletes a particular model	X		
Chart Weekly. Generates statistical charts (weekly)	X		
Chart Monthly. Generates statistical charts (monthly)	X		
Chart Yearly. Generates statistical charts (yearly)	X		
Experts Controller			
Dashboard. Represents home page		X	
View. Displays a particular model	X	X	
Create. Creates a new model	X	X	
Update. Updates a particular model	X	X	

Admin. Manages all models	X		
Delete. Deletes a particular model	X		
Parents Controller			
Dashboard. Represents home page			X
View. Displays a particular model	X		
Create. Creates a new model	X		X
Update. Updates a particular model	X		X
Admin. Manages all models	X		
Delete. Deletes a particular model	X		
Chart Weekly. Generates statistical charts (weekly)			X
Chart Monthly. Generates statistical charts (monthly)			X
Chart Yearly. Generates statistical charts (yearly)			X
Children Controller			
View. Displays a particular model	X		X
Create. Creates a new model	X		X
Update. Updates a particular model	X		X
Admin. Manages all models	X		
Delete. Deletes a particular model	X		X
Game Controller			
View. Displays a particular model	X		
Create. Creates a new model	X		
Update. Updates a particular model	X		
Admin. Manages all models	X		
Delete. Deletes a particular model	X		
Category. Lists games categories	X		
Lesson Controller			

View. Displays a particular model	X		
Create. Creates a new model	X		
Update. Updates a particular model	X		
Admin. Manages all models	X		
Delete. Deletes a particular model	X		
Category Controller			
View. Displays a particular model	X		
Create. Creates a new model	X		
Update. Updates a particular model	X		
Admin. Manages all models	X		
Delete. Deletes a particular model	X		
Tips Controller			
View. Displays a particular model	X		
Create. Creates a new model	X		
Update. Updates a particular model	X		
Admin. Manages all models	X		
Delete. Deletes a particular model	X		
PHEG Controller			
View. Displays a particular model	X	X	
Create. Creates a new model		X	
Update. Updates a particular model		X	
Admin. Manages all models	X		
Delete. Deletes a particular model		X	
Activity Controller			
View. Displays a particular model		X	X
Create. Creates a new model			X

Update. Updates a particular model			X
Admin. Manages all models	X	X	
Delete. Deletes a particular model		X	X
Assist Controller			
View. Displays a particular model		X	X
Create. Creates a new model			X
Update. Updates a particular model		X	X
Admin. Manages all models	X	X	
Delete. Deletes a particular model		X	X
Score Controller			
View. Displays a particular model		X	X
Create. Creates a new model			X
Update. Updates a particular model		X	
Admin. Manages all models	X	X	
Delete. Deletes a particular model		X	X
Session Controller			
Create. Creates a new model			X
Admin. Manages all models	X		
Login. Starts a session			X
Logout. Stops a session			X
Settings Controller			
View. Displays a particular model			X
Create. Creates a new model			X
Update. Updates a particular model			X
Admin. Manages all models	X		
Delete. Deletes a particular model			X

Mailbox Controller			
Inbox. Lists incoming messages	X	X	X
Outbox. Lists outgoing messages	X	X	X
View Inbox. View a particular incoming message	X	X	X
View Outbox. View a particular outgoing message	X	X	X
Create. Sends a new message	X	X	X
Delete. Deletes a particular message	X	X	X

Appendix G. RESTful Web service

Table G.1. Get list of the children

URI	/API/children	
Method	GET	
Request	-	
Response format	JSON	
Response	A JSON object with the following attributes	
Name	Type	Description
Success	Boolean	Indicates successful execution of action
Message	String	Response message

Table G.2. Get a child's details

URI	/API/children/<child_id>	
Method	GET	
Request	-	
Response format	JSON	
Response	A JSON object with the following attributes	
Name	Type	Description
Success	Boolean	Indicates successful execution of action
Message	String	Response message

Table G.3. Get settings

URI	/API/settings/<parent_id>	
Method	GET	
Request	-	
Response format	JSON	
Response	A JSON object with the following attributes	
Name	Type	Description
Success	Boolean	Indicates successful execution of action
Message	String	Response message

Table G.4. Get list of the games

URI	/API/game	
Method	GET	
Request	-	
Response format	JSON	
Response	A JSON object with the following attributes	
Name	Type	Description
Success	Boolean	Indicates successful execution of action
Message	String	Response message

Table G.5. Get a game's details

URI	/API/game/<game_id>	
Method	GET	
Request	-	
Response format	JSON	
Response	A JSON object with the following attributes	
Name	Type	Description
Success	Boolean	Indicates successful execution of action
Message	String	Response message

Table G.6. Get the list of a game's lessons

URI	/API/lesson/?filter = [{"property": "game_id", "value": <game_id>, "operator": "="}]	
Method	GET	
Request	-	
Response format	JSON	
Response	A JSON object with the following attributes	
Name	Type	Description
Success	Boolean	Indicates successful execution of action
Message	String	Response message

Table G.7. Get a lesson's details

URI	/API/game/<level_id>	
Method	GET	
Request	-	
Response format	JSON	
Response	A JSON object with the following attributes	
Name	Type	Description
Success	Boolean	Indicates successful execution of action
Message	String	Response message

Table G.8. Start a session for a child

URI	/API/session/	
Method	POST	
Request	A JSON object with the following attributes	
Name	Type	Description
Child_id	INT	Children unique id
Start_session	DATETIME	Time of starting session
Session_type	TINYINT	Equal to 1
Response format	JSON	
Response	A JSON object with the following attributes	
Name	Type	Description
Success	Boolean	Indicates successful execution of action
Message	String	Response message

Table G.9. Stop a session for child

URI	/API/session/	
Method	POST	
Request	A JSON object with the following attributes	
Name	Type	Description
Child_id	INT	Children unique id
Start_session	DATETIME	Time of starting session
Session_type	TINYINT	Equal to 0
Response format	JSON	
Response	A JSON object with the following attributes	
Name	Type	Description
Success	Boolean	Indicates successful execution of action
Message	String	Response message

Table G.10. Sending score

URI	/API/score/	
Method	POST	
Request	A JSON object with the following attributes	
Name	Type	Description
Child_id	INT	Children unique id
Game_id	INT	Game unique id
Lesson_id	INT	Lesson unique id
Time	FLOAT	Time to finish this level in
Try	INT	Number of wrong answer in order to pass the level
Response format	JSON	
Response	A JSON object with the following attributes	
Name	Type	Description
Success	Boolean	Indicates successful execution of action
Message	String	Response message

Appendix H. Game module

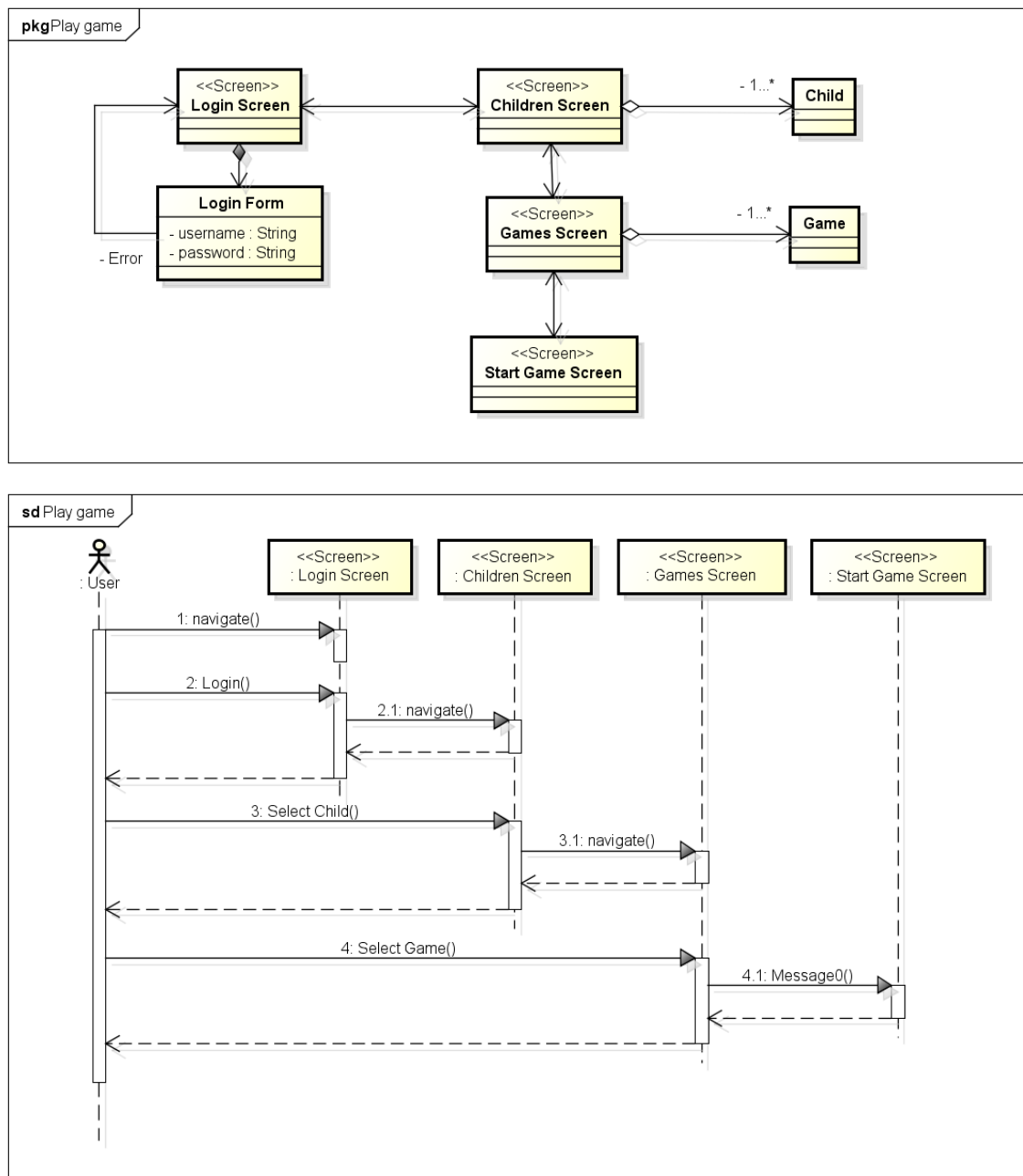


Figure H.1. Sample games' dataflow diagram and sequence diagram