Software Requirements Specification

A Serious Game for COE-DAT As a Virtual Policymaking Laboratory

Merve YORULMAZ - 201311062

Sümeyye TEZALAN - 201311056

Büşra YILMAZ – 201212075

Advisor: Murat YILMAZ, Assist. Prof. Dr.

Co-advisor: Ulaş GÜLEÇ, Specialist

06.12.2017

Table of Contents

List of Tables

List of Figures

1. INTRODUCTION

1.1. Purpose

The aim of software requirements specification (SRS) document is describing military serious game. A serious game will be created for COE-DAT as a virtual policymaking laboratory. This serious game' aim is to train military personnel about policymaking and strategic decision making. This document contains detailed information about requirements of the project, proposed software functionalities and identified constraints.

1.2. Scope of the Project

COE-DAT is a NATO accredited multinational sponsored entity. The mission of the COE-DAT is supporting NATO and nations by providing key decision-makers with realistic solutions (Coedat.nato.int, 2017). Mobile education and advanced training courses have been given to NATO and partner countries' personnel by COE-DAT. The courses contribute to strategic training and skills of the personnel (Coedat.nato.int, 2017). However, only theoretical courses are not sufficient to handle with real life conflicts. Generally, theoretical courses are supported with practical courses to improve success rate. Therefore, military personnel who takes these types of courses needs a practicing platform. However, COE-DAT could not provide practicing platform for strategic decision-making counter terrorism courses. Additionally, there are not enough platform that provides virtual policymaking to deal with this problem all around the world. The project for COE-DAT aims to solve this lack of virtual policymaking laboratory problem.

A serious game is created to meet the need of policymaking laboratory in NATO COE-DAT. Military personnel will have an opportunity to practice that they learn at the theoretical course. The project is supported with realistic event scenarios in accordance with experience of Turkish Armed Forces personnel. The scenarios are based on realistic events or situations which include terrorism actions. The game includes agents that can make decisions depending upon different situations. The strategic success of military personnel will be evaluated with the help of scenarios by giving responses, taking actions and responses of military personnel in the serious game.

The serious game includes actors of participant and admin. A participant can interact with other non-player characters (NPCs) to share information that she/he holds, and NPCs can give response towards actions of participant. The admin is able to add an event to the

game event list, and can change action and response points in the payoff matrices (see Glossary). In addition, admin can make alteration in the virtual simulation game with the help of granted password to admins.

1.3. Glossary

Terms	Definitions
Trainee	The military personnel that plays the game.
Stakeholders	Any person who is a contributor of this project.
Military Serious Game	The game that is utilized by Military institutions with the aims of education, training, etc. (Michael & Chen, 2011)
Virtual Environment	"It is synthetic sensory information that leads to perceptions of environments and their contents as if they were not synthetic." (Blascovich, et al., 2002)
Simulation	It is a close representation of reality in the virtual environment (Thiagarajan & Stolovitch, 1978).
Payoff Matrix	A matrix that contains points for some situations such as decision making in order to measure performance of participants.
NPC (Non-player Character)	The characters (that have counter-terrorism roles and allies' roles of the participant) are played by computer (Merrick and Maher, 2006).

Table 1: Glossary

1.4. Overview of the Document

The functionalities of the project: a serious game for COE-DAT as a virtual policymaking laboratory will be described in the second part of the document. Informal requirements will be expressed before technical requirements, since informal requirements are a context between them in Requirement Specification part. Functionality and technicality of the virtual military serious game are detailed in the Requirements Specification chapter for software developers.

2. OVERALL DESCRIPTION

2.1. Product Perspective

A serious game for COE-DAT as a virtual policymaking laboratory is a military serious game project that aims to give strategic practice opportunity for trainees that take theoretical strategic counter-terrorism course. At the end of the project, the serious game will be used by military institutions to support their courses.

2.1.1. Development Methodology

The most common types of agile software development methodology that is Scrum has been determined to use in development methodology. Scrum development methodology provides easy planning, fast feedback, frequent checkpoints and improved return on investment (Yilmaz, 2017). Scrum is used to manage complicated software projects. It gives descriptions about how everything is to be done in details instead of providing complete on project. Scrum development methodology divides project time into the periods, and divides works into sub-works. In the project development process, the sub-works are completed within a certain period of time. All of these periods are called "sprints". At the end of each sprint, software product should be available and workable (Cohn, 2017).

The scrum team should decide the number of sprints to divide the project process. The scrum teams are small. The ideal size of the team should be 7 +/- 2 people (Scrum-institute.org, 2017).

Sprint 1	Sprint 2	Sprint 3	Sprint 4

1. Design Scenario Characters
2. Design Environment Model
3. Design Military Characters
4. Create Textures
   4.1. Environmental Textures
   4.2. Object Textures
5. Animation
   5.1. Event Animation
   5.2. Character Animation
6. Testing
	1. Create Response
	   1.1. Responses of NPCs
	   1.2. Responses of Trainees
	2. Integrate Decision Making Tree (DMT)
	3. Create Scenarios
	4. Create Scripting
	   4.1. Environment Coding
	   4.2. Character Coding
	   4.3. Animation Coding
	   4.4. Implementation Coding
	5. Audio Design
	6. Testing
		1. Design GUI
		   1.1. Login Panel
		   1.2. Progress Bar
		   1.3. Messagebox & Dialogbox
		   1.4. Scenario Settings Panel
		   1.5. Response Settings Panel
		   1.6. Score Displaying Panel
		2. Edit Audio
		3. Integrate Video
		4. Integrate Points to Payoff Matrices
		5. Create Scripting
		6. Testing
			1. Design Database
			2. Edit Decision Making Tree
			3. Testing & Integration
			   3.1. Alfa Testing
			   3.2. Usability Testing
			   3.3. Functional Testing
			4. Maintenance
			5. Release

Figure 1: Sprints of the Project

Sprints are significant part of scrum methodology in the scope of the quality objectives. Each sprint provides a definition about what will be produced, a design guide and a flexible plan, the work itself and the output to release at the end (Schwaber & Sutherland, 2017). Four sprints of the project are demonstrated in the Figure 1. At the end of each sprint, the output is tested, and required arrangements and maintenances are done. Due to mentioned features above, scrum methodology is an appropriate method for this project.

Figure 2: Scrum Board with Tasks

A scrum board (in Figure 2) have been also used in order to complete tasks in the current sprints. Tasks are written on different cards. Scrum boards is divided into three stages. These stages are "To Do", "In Progress", "Done" respectively. "To Do" stage includes the task cards which will be planned to complete according to priority of the tasks. "In Progress" stage represent the task which are currently continued. In the "Done" stage, successfully completed tasks are indicated. "To Do" stage contains tasks which are determined in the first scrum meeting. These specified tasks are listed below.

- Design Scenario Characters
- Design Environment Model
- Design Military Characters
- Create Textures
- Animation
- Test

2.2. User Characteristics

2.2.1. Trainee

- Trainee must be NATO COE-DAT military personnel.
- Trainee must know English that is common language for COE-DAT personnel in order to understand terror scenarios within this application.
- Trainee must participate theoretical course against terrorism that is given by competent institutions.

2.2.2. Admin

- Admin must be NATO COE-DAT military personnel.
- Admin must know English that is common language for COE-DAT personnel in order to understand terror scenarios within this application.
- Admin must be specialist of theoretical course against terrorism.

3. REQUIREMENTS SPECIFICATION

3.1. External Interface Requirements

3.1.1. User Interfaces

The user interface will be worked on cross platform.

3.1.2. Hardware Interfaces

There are no external requirements for hardware interface.

3.1.3. Software Interfaces

There are no external requirements for software interface.

3.1.4. Communications interfaces

There are no external requirements for communication interfaces.

3.2. Functional Requirements

3.2.1. Profile Management Use Case

Use Case

- Login

- Register

- Display Old Scores

- Start Game

- Get Event

- Check Stamps

- Take Response

- Give Response

- Display Event Score

- Sharing Decision of Event Information

- Share

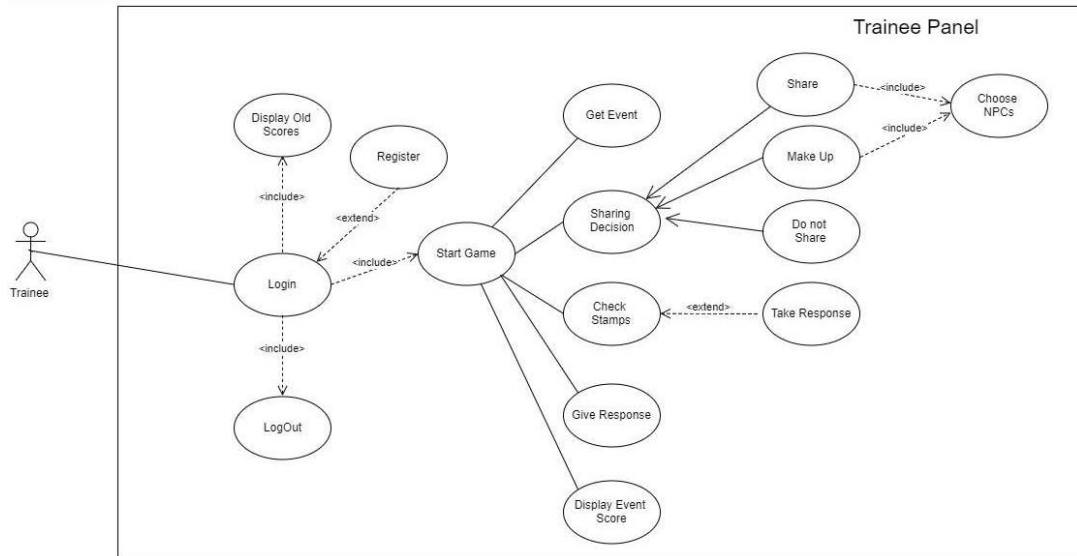- Make Up

- Do not Share

- Choose NPCs

- Logout

Diagram:

Figure 3: Trainee Panel Use Case

Brief Description:

The use case of trainee panel is shown in Figure 3. Trainee can use the "Login", "Register", "Display Old Scores", "Start Game", "Get Event", "Check Stamps", "Take Response", "Give Response", "Display event Score", "Sharing Decision of Event Information", "Do not Share", "Choose NPCs", and "Logout" functions in the trainee panel.

Trainee must make decisions about the terror event that comes to him/her. The trainee can decide whether to share the terror event or not with NPCs. In addition, trainee can make up information about the terror event to share. Trainee can respond to terrorism or NPCs, display event score and see all old scores.

Initial Step by Step Description:

1. Trainee can access to system by logging in with id information and password.

   1.1. If the entered password does not match with their information, trainee must try to login again.

1.2. If trainee does not have a membership in the system, trainee must register to be able to login to the system.

2. Trainee can start the game.

2.1. Game simulation starts when game panel is opened, and game simulation is repeated for each scenario.

2.2. Trainee can decide to not share information about the scenarios. Trainee can share information correctly or can make up information about the scenarios to share. Trainee must choose at least one of the NPC's to share. S/he has the right to not share information with anyone.

2.3. Trainee can give responses to NPCs. S/he should check whether there is enough stamp, before giving response.

2.4. Trainee can take responses from NPCs.

2.5. Trainee can display event score at the end of each event cycle.

3. Trainee can display all the game scores played until that day.

4. Trainee can logout from the system.

3.2.2. Admin Panel Use Case

Use Case:

- Display All Old Scores

- Scenario Settings

- Add Scenario

- Update Scenario

- Delete Scenario

- Response Setting

- Add Response

- Calculate Pay-Off Matrices
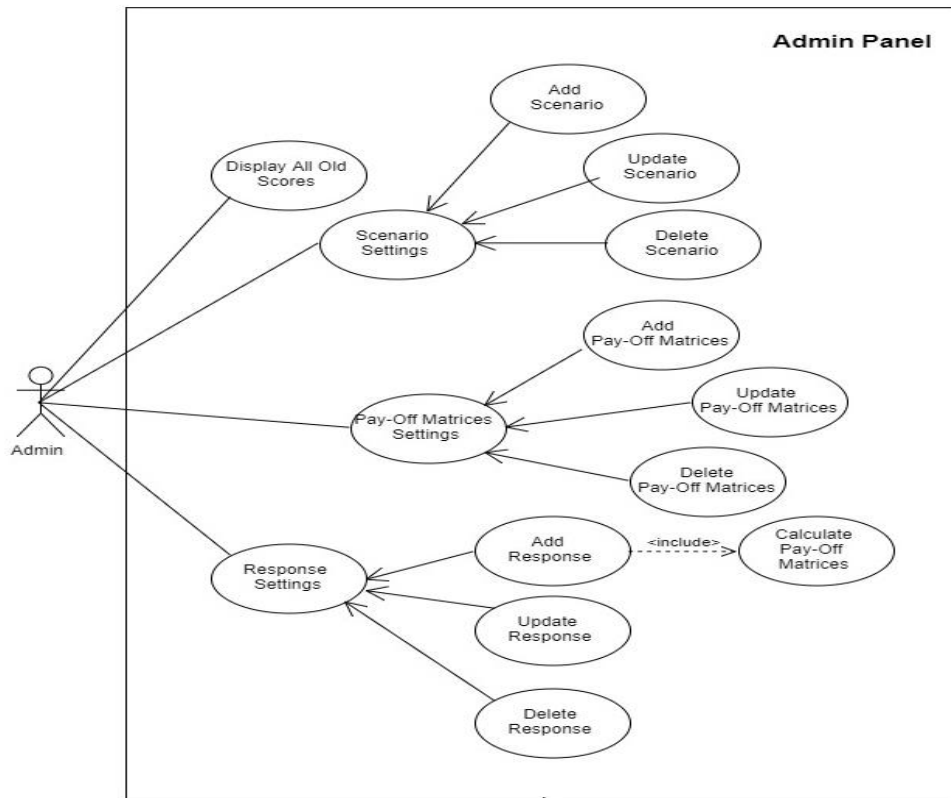
- Update Response

- Delete Response

Diagram:

Figure 4: Admin Panel Use Case

Brief Description:

Figure 4 represents the use case of admin panel. In the application, admin have some rights about game regulation. These rights include adding, deleting terror scenarios and updating some features of an existing scenario. Admin have rights to update payoff matrices points, and terror event points. Furthermore, admin can make some arrangements and additions for responses that are given by NPCs or trainee. That is, admin can add-delete responses corresponding to each terror event. Additionally, updating responses can be done by admin.

Initial Step by Step Description:

1. Admin can manage scenarios, which are in terror event list.

1.1. When admin selects the 'Add Scenario' option, the related panel appears to enter information of a new scenario.

1.2. When admin selects the 'Delete Scenario' option, the related panel appears to select an existing scenario to delete.

1.3. When admin selects the 'Update Scenario' option, the related panel appears to update information of the selected scenario.

2. Admin can manage response options of the related events.

2.1. When admin selects the 'Add Response' option, the related panel appears to display event lists. Admin can add a new response after an event is selected from event list.

2.2. When admin selects the 'Delete Response' option, response deletion panel appears. Admin can delete the response of a selected event after an event is selected from the event list.

2.3. When admin selects the 'Update Response' option, response update panel appears. Admin can update information of a selected event response after an event is selected from the event list.
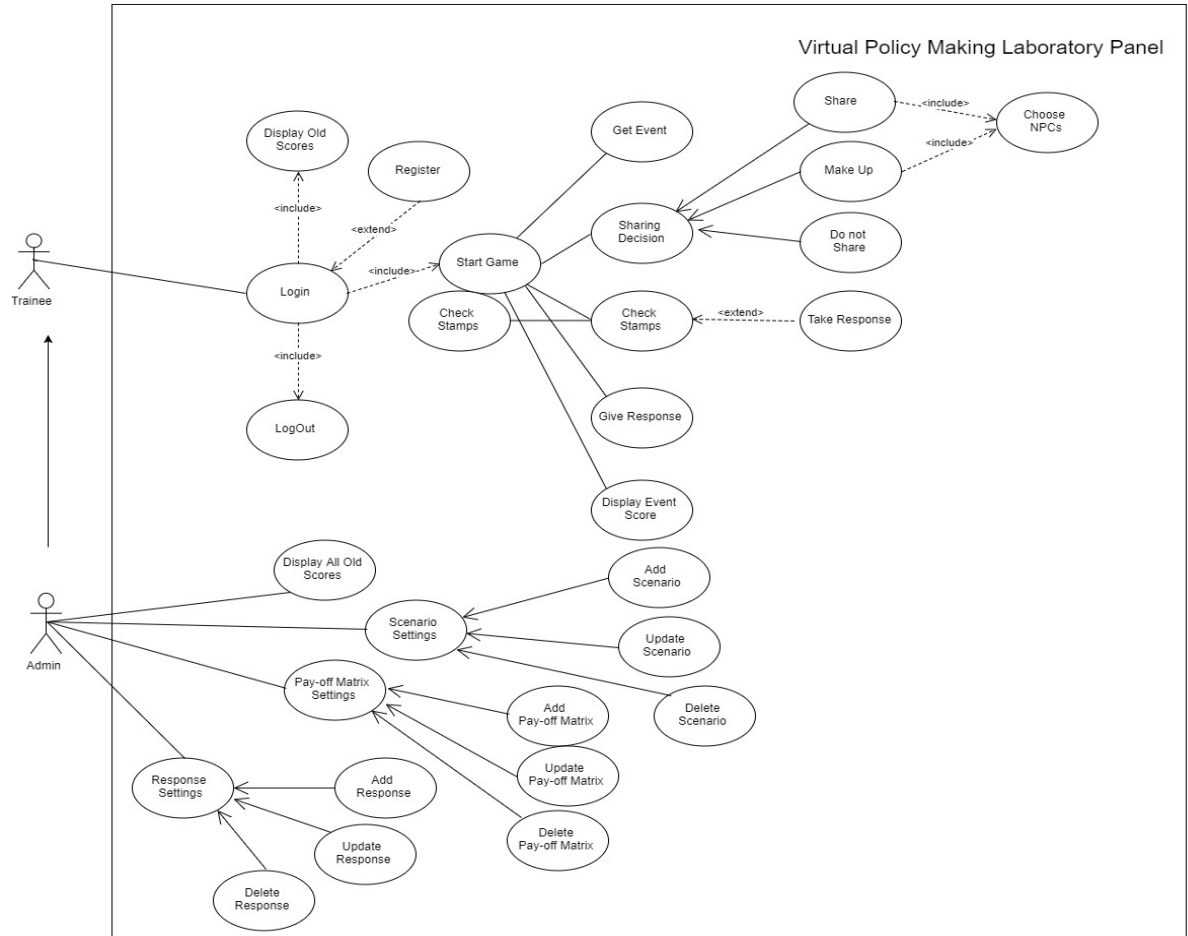
Figure 5: Virtual Policy Making Laboratory Panel Use Case

The use case in the Figure 5 is the combination of Figure 2 and Figure 3. Figure 4 shows that admin can do everything that trainee can do.

3.3. Performance Requirements

The application will be created by using Unity 3D. Therefore, version of Unity 3D 2017.1 or upper versions of Unity 3D 2017. 1 must be supported. Operating system must be Windows 8.1 or its upper versions.

3.4. Software System Attributes

3.4.1. Portability

- The fact that the plant does not need equipment other than the basic hardware elements makes the portability of the plant easier.
- The program which will be developed with Unity 3D supports cross-platform. Thus, it provides advantages for portability to different platforms.

14

3.4.2. Performance

- Trainee cannot see scoring values that are effective in the decision mechanism of each terrorist event.

- Animations about responses should not be played unless the trainee is unresponsive.

- The trainee cannot see the consequences of that decision before they decide.

3.4.3. Usability

- The output of the project should be passed from usability tests. The application will be appropriate for users who have intermediate computer knowledge.

- The application doesn't require complex hardware components. Thus, installation and configuration of the application is simple for any system.

3.4.4. Maintainability

Developing technology leads to new adaptations, additions and innovations for the system. Therefore, object oriented programming will be used in order to make alterations and maintaining easier.

3.4.5. Scalability

Designed to measure the performance of multiple personnel, the system is now handled as a single player at the same time. There is a demand of multi-player requirement. Therefore, multi-player feature is determined as future plan to develop the system in terms of scalability.

3.4.6. Security Requirement

There is a requirement to prevent the possible external threats, access and usage of the application in the scope of security issue. Therefore, hashing of user's password will be used to protect the application from external threats.

4. REFERENCES

1. Blascovich, J., Loomis, J., Beall, A. C., Swinth, K. R., Hoyt, C. L., & Bailenson, J. N. (2002). TARGET ARTICLE: Immersive Virtual Environment Technology as a Methodological Tool for Social Psychology. *Psychological Inquiry, 13*(2), 103-124. doi:10.1207/s15327965pli1302_01

2. Coedat.nato.int. (2017). Centre of Excellence Defence Against Terrorism. [online] Available at: http://www.coedat.nato.int/functions.html [Accessed 8 Dec. 2017].

3. Coedat.nato.int. (2017). Centre of Excellence Defence Against Terrorism. [online] Available at: http://www.coedat.nato.int/courses.html [Accessed 9 Dec. 2017].

4. Cohn, M. (2017). Scrum Methodology and Project Management. [online] Mountain Goat Software. Available at: http://www.mountaingoatsoftware.com/agile/scrum [Accessed 9 Dec. 2017].

5. Merrick, K. and Maher, M. (2006). Motivated reinforcement learning for non-player characters in persistent computer game worlds. Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology - ACE '06.

6. Michael, D. and Chen, S. (2011). Serious games. Mason, Ohio: Course Technology, pp.47-82.

7. Schwaber, K. and Sutherland, J. (2017). Scrum Guide | Scrum Guides. [online] Scrumguides.org. Available at: http://www.scrumguides.org/scrum-guide.html [Accessed 10 Dec. 2017].

8. Scrum-institute.org. (2017). Scrum Roles – The Scrum Team - International Scrum Institute. [online] Available at: http://www.scrum-institute.org/Scrum_Roles_The_Scrum_Team.php [Accessed 10 Dec. 2017].

9. Thiagarajan, S., & Stolovitch, H. D. (1978). *Instructional simulation games*. Englewood Cliffs, NJ: Educational Technology Publications.

10. Yilmaz, M. (2017). CENG 396 – Software Engineering --- Kanban.