

CS258 Software Engineering Lab | 2017



**Indian Institute of Technology,
Indore**

Experimental Game Design

**SOFTWARE
REQUIREMENTS
SPECIFICATION
DOCUMENT**



**Mohit Mohta (150001018)
E. Ranjith (150001011)
Dip Kumar Das (150001010)
Prakash Choudhary (150001022)**

Revision History

Comments	Author	Description	Date
<First Revision>	Group 9	<Version 1>	13-01-17
<Second Revision>	Group 9	<Version 2>	-

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Date	Title	Printed Name	Signature

Table of Contents

REVISION HISTORY

DOCUMENT APPROVAL

1. INTRODUCTION

1.1 PURPOSE	5
1.2 SCOPE	6
1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS.....	6

2. GENERAL DESCRIPTION

2.1 ROUGH IDEA FOR STORY OF THE GAME.....	7
2.2 PRODUCT PERSPECTIVE.....	7
2.3 PRODUCT FUNCTIONS.....	8
2.4 FLOW CHART OF THE GAME FLOW.....	8
2.5 USER CHARACTERISTICS.....	8
2.5 GENERAL CONSTRAINTS.....	9
2.6 ASSUMPTIONS AND DEPENDENCIES	9

3. SPECIFIC REQUIREMENTS.....

3.1 EXTERNAL INTERFACE REQUIREMENTS	9
3.1.1 USER INTERFACES	9
3.1.2 HARDWARE INTERFACES	9
3.1.3 SOFTWARE INTERFACES	9
3.2 FUNCTIONAL REQUIREMENTS	9
3.2.1 MODULE FOR VISUAL SEARCH (VS) EXERCISES.....	9
3.2.1.1 Introduction	9
3.2.1.2 Input	10
3.2.1.3 Processing	10
3.2.1.4 Output	10
3.2.1.5 Error Handling	10
3.2.2 MODULE FOR MENTAL ROTATION ((MR) EXERCISES	10
3.2.2.1 Introduction	10
3.2.2.2 Input	11
3.2.2.3 Processing	11
3.2.2.4 Output	11
3.2.2.5 Error Handling	11
3.2.3 MODULE FOR SPATIAL NAVIGATION (SN) EXERCISES	11
3.2.3.1 Introduction	11
3.2.3.2 Input	12
3.2.3.3 Processing.....	12
3.2.3.4 Output	12
3.2.3.5 Error Handling	12
3.2.4 MODULE FOR VISUAL PUZZLE (VP) EXERCISES	12
3.2.4.1 Introduction	12
3.2.4.2 Input	13
3.2.4.3 Processing	13
3.2.4.4 Output	13
3.2.4.5 Error Handling	13
3.3 NON-FUNCTIONAL REQUIREMENTS.....	13
3.3.1 PERFORMANCE	13

3.3.2 AVAILABILITY	13
3.3.3 SECURITY	13
3.3.4 MAINTAINABILITY	14
3.3.5 PORTABILITY	14
3.4 DESIGN CONSTRAINTS	14
3.5 OTHER LOGICAL REQUIREMENTS	14

1. Introduction

The introduction of the Software Requirements Specification (SRS) provides an overview of the entire SRS with purpose, scope, definitions, acronyms, abbreviations, references and overview of the SRS. The aim of this document is to gather and analyze and give an in-depth insight of the complete **Video Game** by defining the problem statement in detail. Nevertheless, it also concentrates on the capabilities required by users. The detailed requirements of the **Experimental Video Game** are provided in this document.

1.1 Purpose

The purpose of the document is to collect and analyze all assorted ideas that have come up to define the system, its requirements with respect to help our clients in their research. Also, we shall predict and sort out how we hope this document will be used in order to gain a better understanding of the project, outline concepts that may be developed later, and document ideas that are being considered, but may be discarded as the product develops.

In short, the purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This document describes the project's target audience and its user interface, hardware and software requirements. It defines how our client, team and audience see the product and its functionality. Nonetheless, it helps any designer and developer to assist in software delivery lifecycle (SDLC) processes.

1.2 Scope

The software to be produced is an experimental game that is aimed at helping Dr. Sanjram P.K. in his research. The game design is very scientifically customized and every module is crafted so as to incorporate the parameters that Ms. Akanksha Tiwari (Doctoral Student), BSBSE IIT Indore has concluded upon that affect the human visuospatial working memory, in general, in her research at Human Factors and cognition lab, IIT Indore.

The thought is, basically, to test the parameters that are used to judge the visuospatial memory of the players before and after the game and thereafter observe and analyze the changes in the visuospatial capabilities of the player due to a session of gaming.

The game control is of first person type to let the subject feel that he himself is in that scenario. The game contains different modules integrated all in a single flow. Game lasts for a duration of 25 minutes. In the game, player aims to score high, gain maximum health and score minimum distraction. In between player will face obstacles in the form of visual puzzles, way finding, target search, pattern match by mental rotation etc. and to proceed further to achieve

score in the game. The game score, health of the player and distractions and the time left is displayed on the top throughout the game.

1.3 Definitions, Acronyms, and Abbreviations

Definition/Meaning/Full form	Term
Visual search is a type of perceptual task requiring attention that typically involves an active scan of the visual environment for a particular object or feature (the target) among other objects or features (the distractors).	Visual search (VS)
Mental rotation is the ability to rotate mental representations of two-dimensional and three-dimensional objects as it is related to the visual representation of such rotation within the human mind.	Mental rotation (MR)
Spatial navigation is the process by which organisms use multiple cue sources such as path integration, magnetic cues, landmarks, and beacons to determine the route to a goal and then travel that route.	Spatial navigation (SN)
Puzzles that can be solved by visualizing the picture provided.	Visual puzzle (VP)
The mental action or process of acquiring knowledge and understanding through thought, experience, and the senses.	Cognition
Relating to or denoting the visual perception of the spatial relationships of objects	Visuo-spatial
A person who interacts with the game	Player
Operating System	OS
Graphical User Interface	GUI

2. General Description

This videogame is designed to facilitate and collect data for study on human visuospatial working memory. We hypothesize that playing the video game based on later mentioned specific parameters can effectively improve the performance of visuospatial components of working memory.

There are few psychological test batteries which have these parameters implemented as a training module as well as test batteries. These training parameters in the form of video game will be more interesting, interactive and then more effective for improving one's working memory.

Our motive is to design a game/games with different module to provide specific cognitive training in video game environments. These modules are:

1. Visual search (VS)
2. Mental rotation (MR)
3. Spatial navigation (SN)
4. Visual puzzle (VP)

2.1 Aim of the Game:

In the game, the aim of the player is to maximize his score and health and minimize the distractions.

2.2 Product Perspective

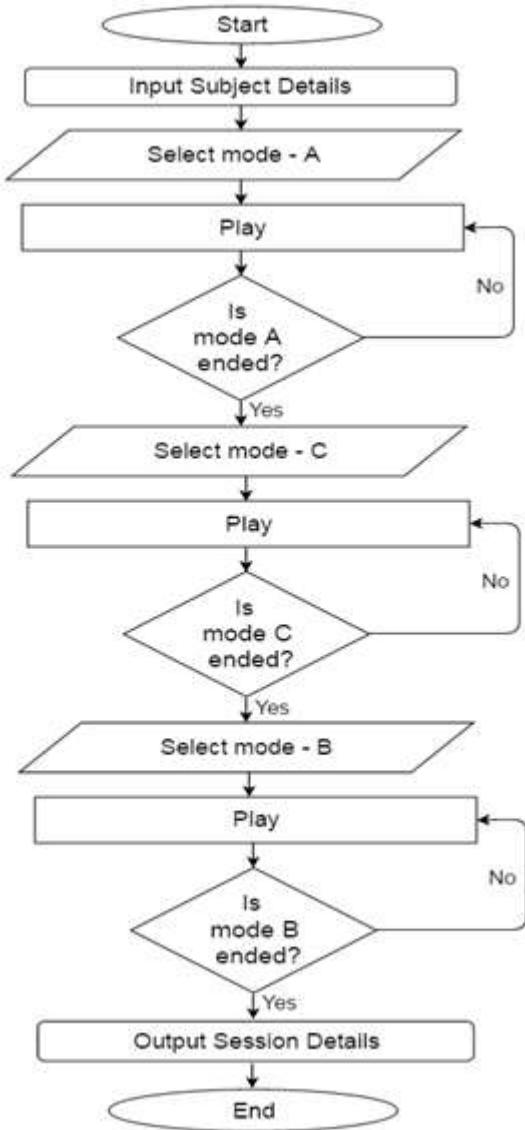
The product i.e the game in this case is more to be an interactive source in which player interacts with game software and viceversa, related to the game. It will be a standalone desktop application.

The two principal actors in this product are “Player” and “System”.

2.3 Product Function

- ❖ Player has to start as per the game storyline.
- ❖ As he passes moves ahead in the game, it is quite essential for him/her to pass maximum modules to satisfy the aim of the game (maximize score and health and minimize distractions).
- ❖ Game score, health, distraction and time must be displayed all the time at the top of the game.
- ❖ The Environment of the game must be an attractive setting in the lap of nature.

2.4 Flow chart for Game flow:



2.5 User Characteristics

In terms of a superset, in general way, it is considered that the user does have the basic knowledge of operating the game and the basic English language (As all the text appear in the user interface will be of English language and so will be the readme). However, the users at which we aim will be actually a very small set of players on whom our clients have already tested on the parameters to judge the visuospatial capabilities.

2.6 General Constraints

The game will not work on an OS other than Windows 8.1 or newer.

2.7 Assumptions and Dependencies

OS: Windows 8.1 or newer.

GPU: Graphics card with DX9 (shader model 3.0) or DX11 with feature level 9.3 capabilities.

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

We will have to provide a 3d video interface to the user so that he can get himself more and more into the game. A first player controller, hence, would be the best suited for the game.

3.1.2 Hardware and Software Interfaces

In order to play the game, we will facilitate the users to use their keyboards, touchpad, mouse or joystick to input the commands and real time effects of the input will be reflected in the game. The user should have a good graphic card installed in his/her machine. Basically, mouse shall be used to change rotate the screen and arrow keys to move in the respective directions.

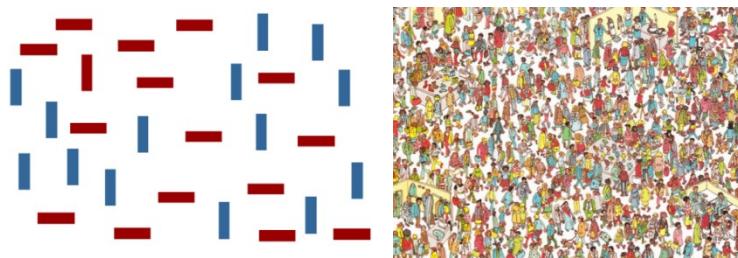
3.2 Functional Requirements

This section describes specific features of the experimental game which we are working on.

3.2.1 Module for Visual Search(VS) exercises

3.2.1.1 Introduction

Visual search is a type of perceptual task requiring attention which typically involves an active scan of the visual environment for a particular object or feature (the target) among other objects or features (the distractors). Visual search can take place with or without eye movements. In this module we can put different level of complexity for target and distractors. It could be static or dynamic as per increase the difficulty level.



Example video clip: <https://www.youtube.com/watch?v=XreVWeMdYk0>

3.2.1.2 Inputs

The medium of input will depend on the surrounding the player is in the game. For instance, if the player is in a room and is surrounded by a lot of objects and is asked to work upon finding a particular type of object in the room than the mode of input will be mostly mouse or touchpad to rotate the eye view in the room and keyboard for moving the player.

3.2.1.3 Processing

As the user drags in the cursor inside the gameview, we shall process the camera view in a manner that it appears to the player that he has actually rotated inside the room.

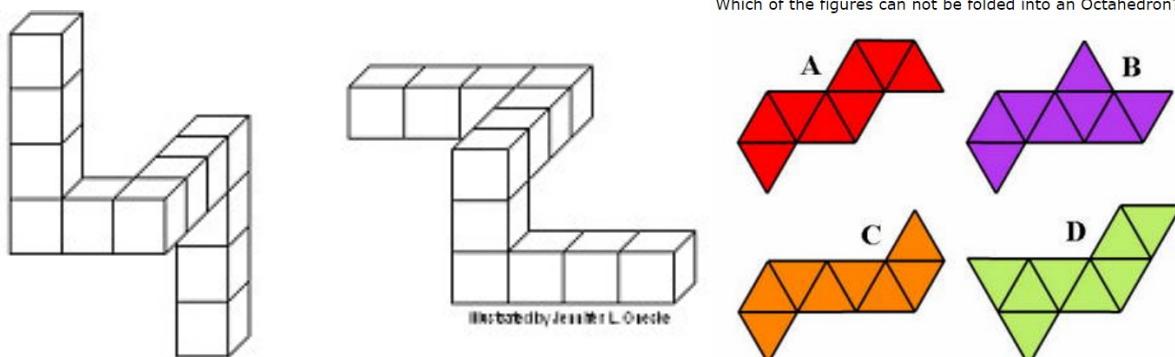
3.2.1.4 Outputs

The outputs will be displayed in the form of the changed view inside the gameview.

3.2.2 Module for Mental Rotation(MR) Exercises

3.2.2.1 Introduction

Mental rotation is an ability to mentally represent and rotate 2D and 3D objects in hypothetical space quickly and accurately, while the object's features remain unchanged.). In this module we can put some objects to recognize after rotating the objects from the earlier representation.



Example video clips: <https://www.youtube.com/watch?v=nEMm-M5Bt3o>
<https://www.youtube.com/watch?v=xW8JPJO9-VI>
<https://www.youtube.com/watch?v=S2evy8EY8IQ>
<https://www.youtube.com/watch?v=SsGXkoL0FNE>

3.2.2.2 Inputs

By mouse only, for instance, suppose 5 shapes are given and the player has to choose the two that exactly match or fit each other as per the game level and requirement of the level, then he/she may select the two pieces by clicking over them.

3.2.2.3 Processing

We will give an id to every part that is displayed and if the player clicks on the two right parts, he qualifies for the next level.

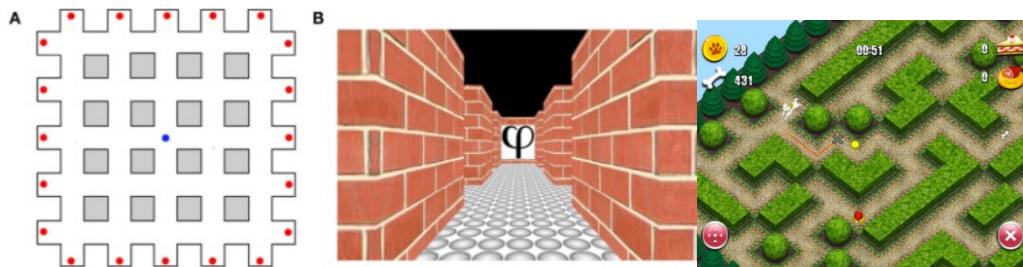
3.2.2.4 Outputs

As the user selects the two pieces and submits his choice, based on whether the selection made is correct or not or how close it is, the user will get points as per the storyline of the game.

3.2.3 Module for Spatial Navigation(SN) Exercises

3.2.3.1 Introduction

Spatial memory is the part of memory responsible for recording information about one's environment and its spatial orientation. It is a phenomenon by which we can recognize or find the appropriate path to reach the target point. In this module we can add few tasks to reach a target by selecting without training or after training.



Example video clips: <https://www.youtube.com/watch?v=4sBznC7T9uU>
https://www.youtube.com/watch?v=_UdgULkKxKI

3.2.3.2 Input

Using WASD and arrow keys on the keyboard to make the player move and touchpad or mouse to make the camera view change.

3.2.3.3 Processing

When a user presses any of the arrow key the player (which acts as a rigid body) will move in the direction the force is applied. We will write functions in c# and javascript in unity that will process the input from the keyboard to move the player. When a combination of keys is the input, our function will treat every direction as a vector and will lastly move the player in the direction of the resultant vector.

3.2.3.4 Output

As the arrow keys are pressed, force is exerted on the player and it shows movement in the 3d space on the screen.

3.2.4 Module for Visual Puzzle(VP) Exercises

3.2.4.1 Introduction

Visual Puzzles are based on traditional logic and lateral thinking, are accompanied by original artwork. The player will solve the puzzles in his mind by using his own strategies. In this module, we can design an interactive task to solve visual puzzles with increasing complexity level and score gain.

Example video clips: https://www.youtube.com/watch?v=_wtK3LBTCjE

3.2.4.2 Input

Sometimes the player will have to use keyboard and sometimes just a touchpad or a mouse might do depending on the type of game in the module.

3.2.4.3 Processing

The player's response to a puzzle is collected and is compared to the pre-saved correct answer and if both of them match, the player unlocks the next level of the game.

3.2.4.4 Output

As the moves are made, the responses are reflected on the screen.

3.3 Non-Functional Requirements

3.3.1 Performance

All the modules should be in a single game and should be followed by a single story or theme of the game. The games should be as much interactive as look like a game environment not a test. Different tasks in the game should require the above mentioned parameters. Responses should be added by mouse or keyboard and a log file should be generated having the score information.

3.3.2 Availability

The game should be of distributable over the other PCs. And, since we're publishing it on the web it will be accessible and available for all.

3.3.3 Security

In order to leave no vulnerabilities for hackers to exploit, secure protocols shall be used for transfer of data from server to the clients and for processing and storage of data on server side.

3.3.4 Maintainability

Collaboration using version control systems like Git and storage of the code on cloud of github and proper comments and description for every code will ensure that a new programmer who

gets to work or improve this very piece of code faces minimal hurdles. A proper documentation for every module shall be included.

3.3.5 Portability

All the laptop/PC machines that posses Windows 8.1 or versions newer than that are capable to run this.

3.4 Design Constraints

The outdated assets of unity in the unpaid version might be a constraint but it shall be compensated by making the modules more interactive and user friendly to drag the player more into the story of the game.

3.5 Other Logical Requirement

Raw data which is actually the output of the research of Dr. Sanjram P.K. will be of great help in scientifically designing the modules of the game and assigning weightage to a module according to its types like Mental rotation(MR), Visual Puzzle(VP) etc.