ICT313 Natural User Interfaces

Tempest

Requirements and Analysis



# Title Page



**Project name:**

Neuromend

**Client/organisation:**

Shri Rai

**Supervisor:**

Fairuz Shiratuddin

**Tempest team members:**

Ary Bizar

Anopan Kandiah

Hannah Klinac

Alex Mlodawski

Bryan Yu

**Date of document:**

v1.0 - 20/08/2014

v2.0 - 12/11/2014

**Version of document:**

2.0

# Contents Page



[Title Page 2](#_Toc403595673)

[Contents Page 3](#_Toc403595674)

[Executive Summary 4](#_Toc403595675)

[Introduction 5](#_Toc403595676)

[Outline of the Solution Proposed to Solve the Problem 6](#_Toc403595677)

[Functional Requirements 8](#_Toc403595678)

[Non-functional Requirements 10](#_Toc403595679)

[Diagrammatic Representation of the Requirements 15](#_Toc403595680)

[Conclusion 16](#_Toc403595681)

[Appendices 17](#_Toc403595682)

# Executive Summary



The purpose of a Requirements and Analysis document is to produce a guide or plan that the developers will follow to build the project to meet the client’s needs. During the Requirements and Analysis process, the domain and scope of the system are discovered, the tasks are documented, and all elements are conceptualised.

In this document the there will be the following sections:

* Outline of the solution proposed to solve the problem
* Functional Requirements
* Non-functional requirements
* Diagrammatic representation of the requirements

# Introduction

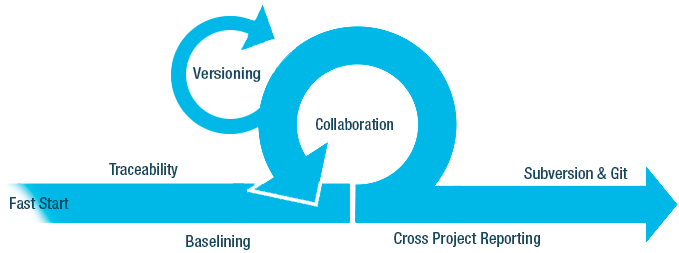


The projects purpose is to explore the possibilities of using virtual reality hardware such as the Oculus Rift in coherence with other devices such as the Leap Motion, Kinect (Windows), Razer Hydra and keyboard & mouse to find the best combination that will eventually be used to try and rehabilitate stroke patients. The system created is a virtual simulation, which uses a combination of devices to achieve a virtual reality goal.

When someone has a stroke, the brain is starved of oxygen for an amount of time and some parts of the brain may die. Research has found that if those areas of the brain are forcefully used again, the brain may eventually begin to rewire itself resulting in possible repairing of the damaged areas. The aim of this project is to develop a simulation that fully immerses the user and has 3 simple tasks that the user should be able to complete. The system has the potential if adapted for future expansion, to try and rehabilitate stroke patients by getting them to use their affected areas of their brains to hopefully regain functionality of limbs and other motor control functions. For the scope of this project, our goal is to find the best tools in the hardware and software aspects to solve that problem and have a product that is ready to be tested on patients.

By solving the problem we can then conduct tests to see how users respond to the different environments using the different combination of devices to further expand the project and maybe in the future put it into some form of practice to actually try and rehabilitate patients.

Methodology is how the risk management is performed. For this project the SCRUM methodology was used. Sprints are the basis of scrum, which are typically one to three weeks in duration. At the end of each sprint, team members and stakeholders meet to assess the progress of a project and to plan the next steps. This means that the projects progress is always being kept in check, and also the direction of the project can be adjusted based on completed work rather than speculation and predictions. This methodology is by nature adaptive and thus flexible to changes in requirements, while ensuring that there is always a stable product at the end of every week so that development does not get out of control or fall behind schedule. For this project there were weekly meetings at 1pm on Wednesday, which involve all team members and the project supervisor. The following diagram depicts the workings of the SCRUM methodology where Collaboration is usually a 1 week process and versioning is a day.



# Outline of the Solution Proposed to Solve the Problem



The scope of the solution was to build a system that meets all of the client’s requirements in the allocated time.

* The system functionality

The system is to consist of three virtual environments in which a user will be placed in for immersive interaction. Each of the three virtual environments will be a level that makes up the overall game. Users will use combinations of the Oculus Rift and remote sensing devices or mouse and keyboard to carry out the interaction in the environments. Each level consists of a different form of interaction. The order of the type of interaction wasn’t strict but there ended up being a level focused on object manipulation, another on object avoidance and lastly way finding. There is a training video at the beginning of the game to teach users how to carry out interactions required using a desired Oculus Rift combination.

* A description of the interface through which the user will access that functionality

Users will require a windows computer to run the game. After installation, the game will be able to run and display on the computer screen and Oculus Rift. The Oculus Rift will be the key device for displaying output to users as the device’s proximity to eyes and heard tracking functionality will make the users feel as if they are inside the game environment. The interaction will be carried out through sensory devices, which will be described in detail later in this section.

* The output expected from the system.

User will have their performance tracked during the game. This data is collected and processed into information that is meaningful and better readable. The idea here is for users and/or supervisors to be able to monitor progress made by playing the game. The data is stored in a database so it can be accessed remotely and without needing to run the game.

* Any constraints imposed on the project

The game was not aimed to be large with a lot of levels and a storyline. It is limited to a small three level game that involves simple but fun interaction in virtual environments. The actual look was not of primary concern but the game was will still made to looking as appealing as possible.

* The hardware/software and other computing environments to be used in the solution

The game is targeted towards windows computers. The hardware that the game makes use of are the Oculus Rift, Kinect for Windows, Leap Motion, Razer Hydra, mouse and keyboard. The use of these devices, excluding mouse and keyboard, requires their accompanied drivers to be installed. These drivers are included in the package and should be installed along with the game.

# Functional Requirements



Functional requirements are the requirements for the projects system to be completed. These requirements are the key stone in the development of our system.  
  
**Device compatibility:**   
This is one of the most important functional requirements as it defines the entire project. If the devices have compatibility issues with each other than the project can be halted in its objectives. Technical issues that may arise are if the computers and devices do not work together through their software and hardware issues. The risk we take in development is by not specifying some form of minimum system requirements and the device driver install have random error occurrences when all operations seem to be working according to specification. Device compatibility is also dependent on device cohesion as we have worked with multiple devices together. All of the device compatibility issues should was not necessarily solved by the time of the first prototype and demo because it was discovered that some of the devices had conflicting drivers. This resulted in having to have an Oculus + Device specific version of Neuromend for each device.  
  
**Device Cohesion**  
Device cohesion was one of the hardest requirements to fulfil as some devices encountered errors when working with one another. Our best way to solve this has been only equipping computer with the devices upon which the certain group individuals are tasked with working on. The risks involved are very minimal as the purpose of the project is to find out how to get certain devices working with one another and how effective they are at working cohesively. Device cohesion had a work around devised by milestone 3.

**Easy use (pickup and play)**

Easy use and user adaptability to the hardware and software that we will be presenting is important because it will be how our end users perceive the technology and how well it can be applied towards our main goal of rehabilitating stroke patients. The main risk with this is that if the end product is not as easy to use as hoped or as we have tested. The project we will be tested on actual test subjects but so far testing has been done through our own means. This means there is a small risk of users not understanding the concepts of the actual devices we are using.

**Environment (Immersion)**

The environment immersion is an important functional requirement as it is the purpose of the project to make a realistic enough environment that the user can actually think that they are not sitting down in a room wearing a headset but are actually in a different world exploring and completing tasks. Environment was covered in one of the later milestones but is still very important so that the users can feel comfortable using the technology that the project offers.

**Safety**

Being safe while using the project’s hardware and software is a very important requirement. If the user feels uncomfortable they should be able to withdraw from participation without feeling any burden or reason to continue their discomfort. The main risk we run is that we do not cater for this properly and in turn end up making people afraid of using the projects technology to try and rehabilitate. We do not want patients to feel so uncomfortable that they would not like to try the rehabilitation process again. The requirement is dependent on a couple of other requirements like environment and Easy use. If the system is too much of a burden or the environment scares the user then the safety goal has not been accomplished.

# Non-functional Requirements



**Security**

Some of the users’ personal details will be required for personalization of the game. These details are stored in a server that will have restricted access to those who have permission rights. It’s a key non-functional requirement to impose such user/supervisor authentication because the privacy and identity of users need to be protected to prevent misuse. To reduce the risk of any breach being detrimental users are only identified by a uniquely assigned id number. They are required to input details such as their full name, age, address, etc.

**Maintainability**

The changing of device drivers is inevitable so the system was designed in such a way that will allow for such changes to be introduced with the absolute minimum impact on users. The cost of such changes is also unavoidable but the system wasn’t designed to force users to buy new hardware should new ones hit the market. In the physical sense, the devices will need to be properly stored away to prevent damage and theft. They should be cleaned regularly between use for good hygiene and optimal operation such as making sure there are no smudges on the lenses of the Oculus Rift.

**Aesthetics**

The virtual environments the game takes place in have some aesthetic appeal to better draw in users and heighten the sense of realism. This requirement went beyond just looking nice and involved taking into account the contrast of any text on screen and non-invasive objects that may irritate users. The goal for this requirement was to make the simulation look clean and orderly enough to encourage users to keep playing and coming back to improve their performance.

**Useability**

This was a difficult requirement to fulfil due to the immense variety of paralysis in stroke patients. For the scope of this project, it was required for the devices to be functional and easy to use. For future adaptations, the range of devices that can be used should cater for several motor control situations but it’s impossible to suit every circumstance. Certain devices will better suit specific physical limitation such as focused movement of a slightly paralysed hand using Leap Motion. It will be up to users/supervisors to decide which sensory device should be used depending on the situation.

**Price**

The actual software package will be provided freely but the devices will need to be bought or borrowed. Support should also be free because it’s in our best interest to help and learn in this relatively new field of virtual reality rehabilitation.

**Fault tolerance**

The game should be as error/fault free as possible because of the possibility of further use in the medical field. Faults are unfortunately more likely to occur due to the fact that some of the devices are still in early stages of development. In order to deliver a highly fault tolerant package a greater amount of resources was allocated to testing during development.

**Extensibility**

Technology develops rapidly and it’s likely that within a year the devices that are currently being used will be outdated. In such cases this would require a major update to the software to accommodate the latest devices. Extra functionality may also be desired in the future such as voice recognition so the software was developed in such way that allows easy slotting in and out of functionality. This would save time and money in the long run and reduce any user inaccessibility.

**Compliance**

Rules, ethics and laws were all considered during the development of the simulation. Anything that may be deemed offensive or inappropriate was not included in the game. Interaction that could be seen as exploiting or belittling a user’s physical capability were avoided.

The below sections takes into consideration the system as a whole and makes general discussion of the non-functional requirements.

* Documentation

The documents that are provided along with the final game package will include some sort of a user manual. This manual provides key details such as health warnings, system requirements, getting started, rules and credits. The medium in which this manual will be provided is in digital format. The manual addresses a wide audience of users but the primary audience are those who have suffered some sort of paralysis due to stroke. The health warning in the manual greatly needs to take into account such primary audience because the game will require physical activity from the user so it’s in our best interest to provide proper warnings and instructions to maximize the safety of the users.

* Hardware Consideration

The game requires a relatively modern windows computer in order to run smoothly. The graphics card will need to be able to handle multiple monitor setups to be able to make use of the Oculus Rift while simultaneously being able to display the game on the computer’s regular monitor. An internet connection will also be required in order to store user performance onto a server.

* Performance Characteristics

Performance is widely dependant on the hardware the game is run on. Computers with high-end graphics cards and CPUS are more likely to run the game without lag and compatibility issues. The ultimate aim was to make the game run smoothly with the lowest system requirements possible in order to be able to target more users who will have a range of different computing hardware. Smoothness is vital because it will be disorientating for users wearing the Oculus Rift if the environment in which they are carrying out interaction is lagging and not responding correctly. A fast internet connection is not required but it will need to be relatively stable to send user performance details to and from a server.

* Error Handling and Extreme Conditions

The game will respond to errors by displaying messages to the user. Small errors such as devices not being recognized will be explicitly displayed. Small errors such us devices not being plugged in are out of our control but the system will be tested to minimize potential large errors as much as possible.

* System Interfacing and Compatibility

The game communicates with a server to send and receive user details. Input comes from the chosen sensory device, mouse and keyboard. The output is displayed on the computer’s monitor as well as the Oculus Rift. The data outputted will contain some of the user’s personal details and performance record.

* Quality Issues

To be reliable the game will need to be run on a modern windows computer. Compatibility on other types of computers cannot be assured due to the variety of devices the game makes use of. Some of these devices are still in development stages and may even have conflict issues with each other so in order to deliver better quality product only windows computers were targeted. Sever access is needed for the game to gather user data so ideally the server is required to run indefinitely so there are no issue with game downtime.

* System Modifications

Once the game has been delivered to users no vital modification will be required to keep it operational. Small modifications that are related to the user him/herself are expected to be made most frequently by themselves or their supervisor. These would include the user’s personal details and possibly their performance records. Major modifications may occur when there are firmware updates for the devices drivers and/or updates/bug fixes for the game itself.

* Physical Environment

The game will be able to be played anywhere with a windows computer and good room space. Good room space is important because the user will be required to use some physical activity to carry out interaction. The type of physical activity will depend on the game level, the devices in use and the physical limitations of the user. As a brief guide, the Kinect (Windows) will require users to be standing a fair distance away from it so it will require the most room space. The Leap Motion, Razer Hydra, mouse and keyboard all have a much more limited range of motion so users can operate them while seated or standing at their most comfortable distance from the computer. It’ll be ideal to have some soft flooring around the user if they are operating a device while standing to better prevent injury should the user get disorientated and lose balance.

* Security Issues

User’s personal details will be stored on a server and will only be accessible by users and supervisors with the associated permission rights. Collection of user’s personal details will be restricted to the bare minimum required for effective operation of the game. They will have their own profile, which they can log into before starting the game. The devices used in the game come at quiet a high cost so they should be properly locked away after use to prevent theft and damage.

* Resource Issues

A server will contain all users’ data and it’ll be running indefinitely so it can remain accessible at any time. However, server failure can happen and users/supervisors should make local backups of their own data. Users will be required to do an initial installation of the game. The device drivers will also need to be installed during this installation process so all that will be required of the user is to connect the devices themselves and run the game. General maintenance of the devices will be the user’s responsibility, cleaning, storage, etc.

# Diagrammatic Representation of the Requirements



Refer to appendix’s B and C.

# Conclusion



**Purpose of document**

The purpose of the requirement and analysis document is to conduct exactly what is encompassed in the analysis phase of the software development cycle. This includes defining project goals and specifying the main functional and non-functional requirements of the system.

During analysis, it is common to model domain entities to identify what are the actual things, root out what are simply inputs or outputs of the system and what are attributes that describe the characteristics of the thing. Other aspects modelled are the flow of data describing how data is stored and how it is passed on from one entity to the next in which at the outset or “context level”, the entire system is the process showing the interaction amongst agents that belong within the system’s boundary. Use cases are also constructed to emphasize upon the functional requirements by showing each external user’s interaction with the system as well as what the system does in response to the user.

With all the models constructed in the analysis phase, its primary purpose is later extended to the design phase in which implementation details are appended, describing the system in such detail that is sufficient for the developer to begin creating the actual system with minimal involvement in any design plans.

**What the document discussed**

The document highlighted exactly what is stated in the conclusion above which also happens to the purpose of the requirements and analysis document. It starts with an outline describing the problem along with any problems, opportunities and constraints foreseeable and a textual description of the system that is purported to solve the problem.

The document proceeds to identify functional defining the elementary functionality of the system and non-functional requirements describing just the quality aspects of the system and lastly accompanied by any analysis stage models that diagrammatically conveys the system’s needs.

# Appendices



Appendix A – Deliverable Task Breakdown Statement

Appendix B – Context Flow Diagram of the system.

Appendix C – Use Case Diagram of the system.

Appendix A



Appendix B

Appendix C

