**MARKING SCHEDULE**

1. Overall 10 marks
2. Executive summary 6 marks
3. Introduction 10 marks
4. Outline of the solution proposed to solve the problem(s) 10 marks
5. Functional requirements 25 marks
6. Non-functional requirements 10 marks
7. Diagrammatic representation of requirements for the system 20 marks
8. Conclusion 5 marks
9. Appendices 4 marks

Appendix A – Deliverable Task Breakdown Statement

Appendix B – Copies of client documents (as applicable)

Appendix C – Glossary

## Total 100 marks

ICT313 Natural User Interfaces

Tempest

Requirements and Analysis



# Title Page



**Project name:**

**Client/organisation:**

**Supervisor:**

**Team members:**

**Date of document:**

**Version of document:**

# Contents Page



[Total 100 marks 1](#_Toc396426899)

[Title Page 3](#_Toc396426900)

[Contents Page 4](#_Toc396426901)

[Executive Summary 5](#_Toc396426902)

[Introduction 6](#_Toc396426903)

[Outline of the Solution Proposed to Solve the Problem 7](#_Toc396426904)

[Functional Requirements 8](#_Toc396426905)

[Non-functional Requirements 9](#_Toc396426906)

[Diagrammatic Representation of the Requirements 12](#_Toc396426907)

[Conclusion 13](#_Toc396426908)

[Appendices 14](#_Toc396426909)

# Executive Summary



Purpose of the document:

A summary of the whole document; that is, what is presented in each section below.

# Introduction



Project purpose

The background to the client’s business/organisation

Narrative outline of the current system (if applicable) and the system to be created

Problems and Opportunities

* The basic problem(s) that needs solving
* The opportunities afforded by solving the problem(s)

The objectives and goals of the system that will be produced

Brief discussion of methodology

The projects purpose is to explore the possibilities of using virtual reality hardware such as the Oculus VR in coherence with other devices such as the Leap Motion, Kinect, Razer Hydra and keyboard & mouse to find the best combination to try and rehabilitate stroke patients. The system to be 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 to try and rehabilitate stroke patients by getting them to use their affected areas of their brains to hopefully to regain functionality of limbs and other motor control functions. Our goal is to find the best tools in the hardware and software aspects to solve that problem.

By solving the problem we can then conduct tests to see how the patients respond to the different environments to further expand the project and maybe 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 is being 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 net 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 not on speculation and predictions. This methodology is flexible if requirements need to be changed, and there is always a stable product at the end of every week so that development doesn’t get out of control. For this project there will be weekly meetings at 1pm on Tuesdays which involve all team members and the project supervisor.

# Outline of the Solution Proposed to Solve the Problem



The scope of the solution

A narrative description of the proposed solution in terms of:

* The system functionality
* A description of the interface through which the user will access that functionality
* The output expected from the system.
* Any constraints imposed on the project
* The hardware/software and other computing environments to be used in the solution

# Functional Requirements



A list of the functional requirements broken up by requirement and including the following information – see lecture slides for more detail:

* Name of highest ranked requirement
* A full description task in terms of the work required; also include:
* Criticality
* Technical issues
* Cost and schedule
* Risks
* Dependencies with other requirements
* Name of second highest ranked requirement

….etc

Functional requirements are the requirements for the projects system to be completed. These requirements are the key stone in the development of our system.  
  
Functional requirements: ----------------------------------  
-Device compatibility-  
This is one of the most important fucntion requirements as it define the entire project. If the devices have compatibility issues with each other then the project can be halted in its objectives.  
The technical issues: Computers and device do not work togehter through device and driver issues  
Risks: We take a risk in developemnt by not specifing some form of minimum system requirements and the device driver install have random error occurences when all operations seem to be working according to specification  
Dependencies: Device compatibility is also dependent on device cohesion as we will be working with multiple devices together.  
  
-Device Cohesion-  
  
-Easy use (pickup and play)-

# Non-functional Requirements



A list of non-functional requirements - see lecture slides for more detail:

For each non-functional requirement consider the following:

* Documentation
* Hardware Consideration
* Performance Characteristics
* Error Handling and Extreme Conditions
* System Interfacing and Compatibility
* Quality Issues
* System Modifications
* Physical Environment
* Security Issues
* Resource Issues

A list of non-functional requirements - see lecture slides for more detail:

For each non-functional requirement consider the following:

* Documentation

The documents that will be provided along with the final game package will include some sort of a user manual. This manual will provide key details such as health warnings, system requirements, getting started, rules and credits. The medium in which this manual will be provided may be in digital, hard copy or both. The manual will address 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 will require 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 run the game without lag and compatibility issues. The ultimate aim is 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 but large errors will output a code that will refer users to the manual for more details. Small errors such us devices not being plugged in are out of our control but the system will be vigorously tested to minimize potential large errors as much as possible.

* System Interfacing and Compatibility

The game will be communicate with a sever to send and receive user details. Input will come from the chosen sensory device, mouse and keyboard. The output will be 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. It will be viewable in the form of an Excel spreadsheet and accessed by anyone who has the right permission from any computer connected to the server.

* 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 will make 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 are being targeted. Sever access will be needed for the game to gather user data so ideally the sever will be 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. Development of the game will be taking into consideration these potential modifications so they can be made with the least amount of impact on users.

* 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 will require users to be standing/sitting a fair distance away from it so it will require the most room space. The Leap Motion, Razar 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. These details may include the user’s height, physical limitation, etc. 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 anytime. 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 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



Model the processes in the system. For example: Context diagram, level 0 / use-case models

Model the data in the system eg ERD

Model the dynamic events in the system – if using object orientation

Model the physical components and other software involved in the running of the system

eg. network topology diagram

# Conclusion



Restate the purpose of the document

What the document discussed

# Appendices



Appendix A – Deliverable Task Breakdown Statement (signed, scanned and inserted into document)

Appendix B - Copies of client documents on which you based the analysis.

Appendix C - Glossary