Inception Phase assessment

## Goals of Inception Phase and How They’re Being Achieved

#### Gather accurate system requirements

We’ve successfully achieved this thorough analysis of the system requirements by gaining an understanding of the target audience’s desired functionality. This was refined through various detailed discussions regarding references to other similar systems on 20th of March 2018 and determining an accurate use case model during 20-27th of March 2018. This allowed us to produce a Project Vision document and Requirements Model document.

#### Analyze functional and non-functional requirements

Following on from accurate system requirements earlier, we were able to determine an accurate list of functional and non-functional requirements and conduct detailed analysis on them, justifying them against the business case. This was conducted between the 19th and 24th of March. This allowed us to add our analysis into the Project Vision and Requirement Model documents.

#### Produce diagrams expressing system functionality

These high level requirements expressed as both functional and non functional were then cultivated into the Domain Model. Essentially this process has formed the crux of the elaboration phase of the project so far allowing clear insight into the architectural requirements.

#### Propose system’s architecture and highlighting architecturally significant requirements

Through a combination of reviews of the functional and nonfunctional requirements and the newly produced Domain models we have been determine the architecturally significant requirements and propose a suitable system architecture. This was completed through 31st of March and 7th of April, and allowed us to complete our Architectural Proposal document.

#### Produce a minimal proof of concept for demonstrating technical competency

Once the main components were identified from the system architecture stage, they needed to be shown to be technologically possible. This was done through the creation of a Technical Competency Demonstrator. This proof of concept was written to be above and beyond spec on 29 of March, and then was refined and documented on 7th of April. The reason we invested the extra time to go above and beyond spec was so that the code was reusable later in the project, and not a wasted effort.

#### Determine and analyze project risks

As project management is an established process there are multiple references to general project risks. We took various examples of these lists and analysed if, and to what extent, each of those risks applied to our project. We then looked at the risks included within our architecture, and analyzed those as well. We did this on the 7th of April to produce the Risk List document.

#### Provide clear plan for ensuring quality of project through testing

For this document, we took all the functional requirements, non-functional requirements, and use cases, then described how each of them would be tested to verify they’re achieved during the project. By targeting these features, we were able to most accurately ensure the project will match design requirements. This was done on the 13th of April.

#### Produce a timeline for various milestones throughout the project

From our project requirements we’re given a set time to complete the project in, and a section of stop work where no progress will be made. Working within these limitations, each of the use cases, risks, and testing steps were distributed upon the available iterations, focusing on the highest priorities first. The time for various documentation milestones (LCAM, IOCM, PRM) were isolated from development work due to lessons learnt about the amount of time it takes to achieve respectable quality standards, also there are contingencies added into the plan to ensure these are reliability completed without affecting the plan too much even if they run over time. This assessment was conducted on 7th of April, and allowed us to create our Project Plan document.

## Justification of Technical Capability to Achieve Those Goals

The technical capability of the members to achieve this project is seen as a substantially low risk as team members have experience using each of the skills required for this project, such as Test Driven Development, .NET, Git version control, and code review. Additional to these technical skills is Object Oriented System Analysis and Design (OOSAD), this incorporates all the documentation and design stages of the assessment, and both members also have formal qualification with this subjects.

As a practical proof of team member’s understanding of the technologies used in this project a technical competency demonstrator (TCD) software program was produced. This TCD program showed the successful serialisation, transport and deserialization of JSON messages over a UDP network connection. This was later later analysed in the “Technical Competency Demonstrator” document on 30th March 2018.

## Issues encountered

Generally speaking we have not experienced technical issues thus far, however there is a real indication that we have overly optimistic view on how much time goes into tasks.This is highlighted by our initial estimates in the Iteration Plan 1 where our time estimations were out by around 1000%. Applying our learnings from this our second iteration plan’s time estimations were out by 450%, an improvement nonetheless. Looking forward, this is one of our main focuses on project management which we’d like to become better accurate with, however we believe this is considered a skill and will mature over time.

## Status of Any Important Risks

|  |  |  |
| --- | --- | --- |
| **Risk** | **Open/Closed** | **Status** |
| Code quality issues | Open | Neither symptom or trigger has occurred, migration currently not required |
| Poor software quality | Open | Neither symptom or trigger has occurred, migration currently not required |
| Networking library issues | Closed | Risk has successfully been resolved and demonstrated in Tech Competency Demonstrator |
| Security too complex | Open | Neither symptom or trigger has occurred, migration currently not required |
| Prototype failure | Open | Neither symptom or trigger has occurred, migration currently not required |
| Multithreading introduces high level of difficult in troubleshooting | Open | Neither symptom or trigger has occurred, migration currently not required |
| Network level security issues related to usage of UDP | Closed | Risk has successfully been resolved and demonstrated in Tech Competency Demonstrator |

## Current progress of project

The current state of progress is *satisfactory*; the reason we aren’t considering the state as “*good*” is we are currently working within our contingency allowance iteration of LCOM. Moving forward we would prefer to finish our targeted outcomes on time and not to rely on these contingency time allowances when meeting our deliverables.

At this stage, the project has merit, the team has the the technical skills required as demonstrated in the TCD. Provided we can improve our time estimation to adhere to the project plan, a minimal functional feature set of the library as a prototype can be delivered by end of the Construction Phase on the 3rd June 2018.