# Inception Phase Status Assessment

## 1. Assessment against Objectives of the Inception Phase

### 1.1 Do we know what we are trying to achieve?

The aim of the project is to create an ADHD Task Manager mobile application that is designed to help individuals with ADHD manage their daily tasks and improve their productivity. This is embodied in the completed Vision Document.

<https://github.com/commet003/ITC303-9-Team1-Project/blob/main/LCOM%20Documents/Iteration1/LCOMProjectVision.docx>

We understand the main functional requirements of the project which are:

* User registration through username and password authentication
* Sqlite database system
* Points awarding with the external app integration Email, Map, Calendar app
* Other user comparison communication i,e leader board.

This is shown in the completed Use Case model within our designated folder:

<https://github.com/commet003/ITC303-9-Team1-Project/tree/main/Use_Cases>

We understand the main Non-Functional requirements of the project which are:

* allows users to create and manage their tasks easily. This functionality will enable users to add due dates, prioritize tasks, and categorize tasks by project or topic.
* system will include a Pomodoro timer, which allows users to work in intervals of focused work and rest.
* the system will include a habit tracker and reward system.
* the system will include insights and analytics. This functionality will include graphs and charts.

This is shown in the completed Non-Functional Requirement model embodied in [*https://github.com/commet003/ITC303-9-Team1-Project/blob/main/LCOM%20Documents/Iteration1/LCOMInitialRequirementModel.docx*](https://github.com/commet003/ITC303-9-Team1-Project/blob/main/LCOM%20Documents/Iteration1/LCOMInitialRequirementModel.docx)

### 1.2 Do we know how we are going to achieve it?

We have a good idea of how we are going to achieve our aims. We are going to use Separation of Concerns (SoCs) to effect scalability, testability, performance, and maintainability. Consequently, 3 major components, namely model, view, and controller, with specific responsibilities will be created. This is shown in the completed Architectural Notebook

<https://github.com/commet003/ITC303-9-Team1-Project/blob/main/LCOM%20Documents/Iteration1/LCOMProposedArchitecture.docx>

We have a good understanding of the project specific risks facing our project and how we are going to deal with them. The risks are:

* Secure user data if external database is used.
* Making the app attractive aesthetically to keep user motivated.
* Functional inaccuracy
* Compatibility issues with different IDEs

Our evolving understanding of risks is shown in the ongoing risk list and discussed further below in Section 4 and within our initial risk management list.

<https://github.com/commet003/ITC303-9-Team1-Project/blob/main/LCOM%20Documents/Iteration2/LCOMRiskList.xlsx>

We have a good understanding of how we are going to check that our application delivers the intended functionality and system properties. Our key areas of concern and the test strategies we will use to address these concerns are as follows:

* Make certain that the primary components of the Task List are functional. The ability to include, eliminate, and finish tasks represents the central operations of the app.
* It is crucial to preserve user information for future access. The app needs to recall saved tasks and determine the appropriate moments for issuing notifications.
* The app must provide accomplishment incentives upon task completion. One of the primary draws of this application is the rewards users receive after finishing their tasks.

This is shown in the completed Master Test Plan

<https://github.com/commet003/ITC303-9-Team1-Project/blob/main/LCOM%20Documents/Iteration2/LCOMMasterTestPlan.docx>

We have a good understanding of the dependencies and likely completion times for different parts of the project. Target completion dates for key aspects of the project are as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Subject** | **Phase** | **Iteration** | **Dates** | **Primary objectives** (risks and use case scenarios) |
| ITC303 – Software Development Project 1 | Inception Phase | I-1 | 13/03 – 26/03 | Establish Vision; Establish Initial Use Case Model; Complete Preliminary Non-functional Requirement Analysis; Identify/Document Candidate Architectures; Establish Version Control |
| I-2 | 27/03 – 9/04 | Establish Risk List; Complete Full Description for Critical Core Risky Difficult (CCRD) Use Case; Implement Technical Competency Demonstrator; Create Test Plan; Establish Initial Project Plan; Deliver Life Cycle Objectives Milestone (LCOM); Complete Inception Phase Project Assessment |
| Elaboration Phase | E-1 | 10/04 – 23/04  (Session Break) | Mitigate Highest Priority Risk(s); Implement Highest Priority Architectural Element(s) to Support CCRD Use Case; Complete Development Testing for Highest Priority Architectural Element(s) |
| E-2 | 24/4 – 7/05 | Mitigate 2nd Highest Priority Risk(s); Implement 2nd Highest Priority Architectural Element(s) to Support CCRD Use Case; Complete Development and Integration Testing for 2nd Highest Priority Architectural Element(s) |
| E-3 | 8/05 – 21/05 | Mitigate 3rd Highest Priority Risk(s); Implement 3rd Highest Priority Architectural Element(s) to Support CCRD Use Case; Complete Development and Integration Testing for 3rd Highest Priority Architectural Element(s); Deploy Executable Architecture in Trial Environment; Complete Internal User Acceptance Testing for CCRD Use Case in Trial Environment |
| E-4 | 22/05 – 2/06 | Contingency; Deliver Life Cycle Architecture Milestone (LCAM); Complete Elaboration Phase Project Assessment |
| Mid-year Semester Break | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Mid-year Semester Break | | | | |
| ITC309 – Software Development Project 2 | Construction Phase | C-1 | 10/07 – 23/07 | Implement 2nd Highest Priority Use Case(s); Complete Development and Integration Testing for 2nd Highest Priority Use Case(s); Complete Internal User Acceptance Testing for 2nd Highest Priority Use Case(s) |
| C-2 | 24/07 – 6/08 | Implement 3rd Highest Priority Use Case(s); Complete Development and Integration Testing for 3rd Highest Priority Use Case(s); Complete Internal User Acceptance Testing for 3rd Highest Priority Use Case(s) |
| C-3 | 7/0 – 20/08 | Implement 4th Highest Priority Use Case(s); Complete Development and Integration Testing for 4th Highest Priority Use Case(s); Complete Internal User Acceptance Testing for 4th Highest Priority Use Case(s) |
| C-4 | 21/08 – 3/09  (Session Break) | Contingency; Deliver Initial Operation Capability Milestone (IOCM); Complete Construction Phase Project Assessment |
| Transition Phase | T-1 | 4/09 – 17/09 | Deploy Application in Trial Environment; Complete 1st Round External User Acceptance Testing; Resolve Any Identified Issues |
| T-2 | 18/09 – 1/10 | Complete 2nd Round External User Acceptance Testing and resolve any identified issues |
| T-3 | 2/10 – 13/10 | Contingency period if the project runs over schedule. Deliver Product Release Milestone (PRM) and Complete Final Project Assessment |

This is shown in the Initial Project Plan.

### 1.3 Skills required

Our project requires skills using the following key tools and technologies:

* Calendar integration possibly incorporating OAuth 2.0
* HTTPS protected data transfer between mobile device and backend webserver. Possible use of WebSocket.
* Software connectivity using common formats like CSV and JSON
* Secure communication protocols (HTTPS, SSL/TSL) for protocols.
* Android Studio to write and implement project using Kotlin language
* SQlite database integration for storing user data

We have demonstrated that we have the skills to use these technologies through the implementation of a technology competency demonstrator.

## 2. Deliverables

- Master Test Plan No Issues

- Project Plan No Issues

- Risk List No Issues

- Initial Requirements Model No Issues

- Project Vision No Issues

- Proposed Architecture No Issues

## 3. General Issues

1. Communication

An area we are lacking as a team is communicating to one another. We have analyzed that this needs to be addressed moving forward to ensure everybody is on the same page when it comes to decisions and input from all of the members. All the team members have acknowledged that this is a problem and steps have been implemented to improve this in the future. Some of these are check in messages, as well as more regular meetings.

1. Design

The Team has not settled on a final design of the product, which can make some of the documentation seem off. The documentation will be rectified when the final design has been confirmed.

## 4. Risks

*For each risk (max 7)*

### 4.x *<insert risk name here>*

*<identify any key points you wish to make about this particular risk>*

*<state the mitigation strategy you are using to address the risk>*

*<say whether the risk is ongoing or resolved>*

4.1 .gitignore does not have undesired files created by windows/macOS system

* files not to be add to repo must be added to .gitignore at the beginning to avoid complications when members do a pull.
* From ITC205, when undesired files are not added to .gitignore can cause complications.
* This risk is ongoing and the mitigation process will be to add system generated files to .gitognore at the earliest.

4.2 Iteration 1.1 not completed

* Incomplete iteration 1.1 will negatively impact iteration 1.2.
* The mitigation strategy is to finish assigned tasks by the stated deadline and to follow the team charter.
* This risk is resolved.

4.3 Core use cases not explicitly stated

* Scope creep can delay the project.
* The mitigation strategy is to clearly state core use cases and non-core use cases.
* this risk is ongoing.

4.4 App functional accuracy

* Points must be correctly added so that rewards can be correctly given
* The mitigation strategy is to thoroughly test the app
* this risk is ongoing

4.5 Different app properties such as compileSdk and/or minSdk

* Group members to use same properties to maintain consistency.
* the mitigation strategy is to decide on such properties.
* this risk is ongoing.

## 5 Summary – Overall Project Progress

The project has made good progress during the Inception Phase, with all required tasks being completed. However, there are ongoing issues that need to be addressed, including poor communication between team members. This can lead to misunderstandings, delays, and other issues, and it is important for the team to work together to improve communication and ensure that everyone is on the same page.

In addition to poor communication, there is an ongoing issue with the design of the application, which has not been settled on. The team is exploring different design options to ensure that the app meets the needs of its target audience.

There are also several ongoing risks associated with app development that need to be monitored throughout the project, including technical issues such as bugs and compatibility issues, scope creep which could result in delays and security risks associated with storing and handling sensitive data.

To mitigate these risks and address ongoing issues, the team needs to communicate effectively, identify potential issues and risks early on, and take appropriate measures to address them. Overall, the project is moving forward according to plan, but the team needs to remain vigilant in identifying and addressing potential issues and risks to ensure a successful outcome for the project.