Project Plan

Confide Instant Messaging Application

Table of Contents

[2 Acronyms and Abbreviations 4](#_Toc485021076)

[3 Summary of Project Schedule 4](#_Toc485021077)

[4 Detailed Project Deliverables 6](#_Toc485021078)

[4.1 Server and Client API Evaluation 6](#_Toc485021079)

[4.2 Description: 6](#_Toc485021080)

[4.2.1 Justification: [Risk 1] 6](#_Toc485021081)

[4.2.2 Deliverables: 7](#_Toc485021082)

[4.2.3 Outcome: 7](#_Toc485021083)

[4.3 Implement CCRD “Chat” Use Case 7](#_Toc485021084)

[4.3.1 Description: 7](#_Toc485021085)

[4.3.2 Justification: [CCRD UC 1] 7](#_Toc485021086)

[4.3.3 Deliverables: 7](#_Toc485021087)

[4.3.4 Outcome: 7](#_Toc485021088)

[4.4 Implement “Create Account” Use Case 8](#_Toc485021089)

[4.4.1 Description: 8](#_Toc485021090)

[4.4.2 Justification: [CCRD UC 2] 8](#_Toc485021091)

[4.4.3 Deliverables: 8](#_Toc485021092)

[4.4.4 Outcome: 8](#_Toc485021093)

[4.5 Implement “Add User To Contacts” Use Case 8](#_Toc485021094)

[4.5.1 Description: 8](#_Toc485021095)

[4.5.2 Justification: [CCRD UC 3] 8](#_Toc485021096)

[4.5.3 Deliverables: 8](#_Toc485021097)

[4.5.4 Outcome: 8](#_Toc485021098)

[4.6 Implement “View Contacts” Use Case 9](#_Toc485021099)

[4.6.1 Description: 9](#_Toc485021100)

[4.6.2 Justification: [CCRD UC 3] 9](#_Toc485021101)

[4.6.3 Deliverables: 9](#_Toc485021102)

[4.6.4 Outcome: 9](#_Toc485021103)

[4.7 Implement “View Chats” Use Case 9](#_Toc485021104)

[4.7.1 Description: 9](#_Toc485021105)

[4.7.2 Justification: 9](#_Toc485021106)

[4.7.3 Deliverables: 9](#_Toc485021107)

[4.7.4 Outcome: 9](#_Toc485021108)

[4.8 Implement “Notifications” Sub-Use Case 10](#_Toc485021109)

[4.8.1 Description: 10](#_Toc485021110)

[4.8.2 Justification: 10](#_Toc485021111)

[4.8.3 Deliverables: 10](#_Toc485021112)

[4.8.4 Outcome: 10](#_Toc485021113)

[4.9 Group Chat Sessions 10](#_Toc485021114)

[4.9.1 Description: 10](#_Toc485021115)

[4.9.2 Justification: 10](#_Toc485021116)

[4.9.3 Deliverables: 10](#_Toc485021117)

[4.9.4 Outcome: 11](#_Toc485021118)

[4.10 Local User Data Caching, Message Confidentiality. 11](#_Toc485021119)

[4.10.1 Description: 11](#_Toc485021120)

[4.10.2 Justification: 11](#_Toc485021121)

[4.10.3 Deliverables: 11](#_Toc485021122)

[4.10.4 Outcome: 11](#_Toc485021123)

[4.11 User workflow, Delete Chats, Edit Account 11](#_Toc485021124)

[4.11.1 Description: 11](#_Toc485021125)

[4.11.2 Justification: 11](#_Toc485021126)

[4.11.3 Outcome: 12](#_Toc485021127)

[4.12 Manage Contacts 12](#_Toc485021128)

[4.12.1 Description: 12](#_Toc485021129)

[4.12.2 Justification: 12](#_Toc485021130)

[4.12.3 Outcome: 12](#_Toc485021131)

# Acronyms and Abbreviations

|  |  |
| --- | --- |
| Abbreviation | Description |
| API | Application Programming Interface |
| C-1 to C-4 | Construction phase 1 to 4 |
| CCRD | Critical Core Risky Difficult |
| E-1 to E-4 | Elaboration Iteration 1 to 4 |
| OpenFire | OpenFire XMPP server |
| GUI | Graphical User Interface |
| LCAM | Lifecycle Architecture Milestone |
| PRM | Project Release Milestone |
| T-1 to T-3 | Transition into production 1 to 3 |
| UC | Use Case |

# Summary of Project Schedule

The project achieved the Lifecycle Objectives Milestone (LCOM). Table 2 provides a summary of the project deliverables planned up to the end of the project. Further details of the status of each item are available by following the links in the table.

Table 2 - Project Schedule

| Date | Phase | Deliverables | Justification |
| --- | --- | --- | --- |
| 10 April  to  21 April | E-1 | -[Ensure server is suitable and determine most suitable client API](#gjdgxs)  -[Implement “Create Account” Use Case](#1fob9te) | Risk 1  CCRD UC 2 |
| 24 April  to  8 May | E-2 | -[Implement “Add User To Contacts” Use Case](#3znysh7)  -[Implement “View Contacts” Use Case](#2et92p0)  -[Implement “View Chats” Use Case](#tyjcwt) [-Implement “Chat” Use Case](#30j0zll) | CCRD UC 1  CCRD UC 3 |
| 9 May  to  21 May | E-3 | -Send/Accept Contact Request  -Manage and View Contact's (Presence)  -Storing/retrieving info on server (Chats, Messages)  -Message Integrity  -View Chats GUI  -Chat GUI | CCRD UC 3 |
| 22 May  to  12 June | E-4 | Review and refine contact, message and chat functionality. Tidy user interface and refactor/clean code. Update and improve docs for [submission at LCAM](https://interact2.csu.edu.au/webapps/Bb-wiki-BBLEARN/wikiView?course_id=_24293_1&wiki_id=_13523_1&page_guid=e8a1ab976512435d8f3f7c6d9e0c4add) | Project Deliverable |
| 12 June | LCAM | LCAM Submission | Demonstrate capacity to complete project objectives |
| 10 Jul  To  23 Jul | C-1 | [Group Chat Sessions](#53d7o77iv97y) | Necessary to reach our Standard Target |
| 24 Jul  To  6 Aug | C-2 | [Manage Contacts](#xzde7359orkw) | Necessary to reach our Standard Target |
| 7 Aug  To  20 Aug | C-3 | [Local User Data Caching, Message Confidentiality.](#nr6o9zl8291z) | Necessary to reach our Stretch Target |
| 21 Aug  To  3 Sep | C-4 | [-User workflow](#vljinq66lrxm)  [-Delete Chats](#vljinq66lrxm)  [-Edit Account](#vljinq66lrxm)  Notifications1 | Necessary to reach our Stretch Target |
| 4 Sep  To  17 Sep | T-1 | Complete 1st Round External  User Acceptance Testing  Resolve Any Identified Issues | Deploy Application in Trial Environment |
| 18 Sep  To  1 Oct | T-2 | Complete 2nd Round External  User Acceptance Testing  Resolve Any Identified Issues | Deploy Application in Production  Environment |
| 2 Oct  to  15 Oct | T-3 | Contingency for last minute changes | Final bug fixes/ enhancements |
| 20 Oct | PRM | PRM Submission | Complete Final Project Assessment |

Note 1: Basic data which could be used for notifications is already available. Main task is to implement this in the GUI.

# Detailed Project Deliverables

This section is a brief summary of the main deliverables to be provided in the executable architecture. Green bars next to the main headings indicate that the outcome has already be partially or fully achieved. Precise deliverables for some items still need to be determined in consultation with the client.

## Server and Client API Evaluation

## Description:

Ensure server is suitable and determine most suitable client API“Hello world” application with OpenFire server and client API’s (Smack, Babbler, Emite)

### Justification: [Risk 1]

This is our highest priority risk, and everything else is built on top of this. We cannot afford to spend long not knowing which client API we are using, as this will result in lots of effort being put in the wrong direction

### Deliverables:

”Hello world” application with OpenFire server the smack API, and hopefully the other two API’s.

To Be Completed: E-1

### Outcome:

Successfully completed in E-1.

## Implement CCRD “Chat” Use Case

### Description:

Two users on different client devices can join and participate in a chat. Users can start a chat with another user who is in their contacts, by selecting that contact.

### Justification: [CCRD UC 1]

This is our highest priority CCRD UC, as it is the essence of the whole project. We need to show that we can have this basic functionality before moving on to improving it and making the system more usable.

**Involves:**

* Log In
* Add User To Contacts
* View Chats
* Send Message
* Receive Message

### Deliverables:

* A user can start a chat with a contact
* A message can be sent
* A sent message is received
* A user can open an existing chat

To Be Completed: E-2

### Outcome:

Successfully completed in E-2

## Implement “Create Account” Use Case

### Description:

Users can create accounts (authorization), and then login to their account (authentication).

### Justification: [CCRD UC 2]

This is our second highest priority use case, as it lays the groundwork for fully implementing our highest priority CCRD UC; that is abstracting the creation and entering of chats, via contacts.

### Deliverables:

* A person can create an account
* A created account can be logged into from the device that created it and other devices.

To Be Completed: E-1

### Outcome:

Successfully completed in E-1

## Implement “Add User To Contacts” Use Case

### Description:

Users can add other users via their account, either by searching for them, or directly by their user handle/id.

### Justification: [CCRD UC 3]

These are our third highest priority CCRD UC, as it is the next step towards fully implementing our highest priority UC.

### Deliverables:

* A user can search for another user, and request to add them as a contact
* A user can accept a contact add request
* A user can decline a contact add request

To Be Completed: E-2

### Outcome:

Successfully completed in E-2, with GUI completed in E-3, bug with retrieving a user's contact list has been resolved.

## Implement “View Contacts” Use Case

### Description:

Users can view all their contacts, and search for a specific contact.

When viewing each contact, they can see their presence.

### Justification: [CCRD UC 3]

These are our third highest priority CCRD UC, as it is the next step towards fully implementing our highest priority UC.

### Deliverables:

* A user can view all of their contacts.
* A user can view a specific contact.

To Be Completed: E-2

### Outcome:

Successfully completed in E-2, with GUI developed in E-3. Bug with adding contacts successfully resolved.

## Implement “View Chats” Use Case

### Description:

Users can view all their chats and switch between them.

### Justification:

This needs to be implemented to some extent during “Chat” use case.

### Deliverables:

* A user can view all of their chats
* A user can select and view a specific chat from the chats view
* A user can select and view a specific chat from the contacts view

**To Be Completed**: E-2

### Outcome:

Successfully completed in E-2, with GUI developed in E-3.

Implemented as a list of contacts, each of which is associated with one chat (can only have one 2-person chat session with a given contact).

## Implement “Notifications” Sub-Use Case

### Description:

Users receive notifications for events such as: contact add request, contact add request accepted, new message.

### Justification:

Notifications are not part of our CCRD UC’s, however they are an essential part of creating the real time and fast response environment of an instant messenger, and ensuring that important events are not unseen for some time or missed altogether.It may also exercise some aspect of the architecture that is not by the other work items.

### Deliverables:

* Notifications can be displayed on the client

To Be Completed: C4

### Outcome:

This is essentially spread out through the other use cases and iterations. All functionality we have completed so far, the system is alerted when any relevant event occurs, and we can use this information to notify the user in the GUI.

## Group Chat Sessions

### Description:

Group chat sessions will allow our chat sessions between 2 users to expand into more flexible group chats with multiple people. Also the 2-person chat sessions we have implemented so far, are meant for rapid exchange of messages between two people, and therefore cannot be saved. Therefore, implementing group chats between 2 people should allow us to save the chat between 2 people.

### Justification:

Now that all the critical core use cases have been implemented, group chats is naturally the next best use case to pursue, as it expands functionality greatly.

### Deliverables:

* Groups of 2 or more people can all chat together in the one group chat.
* These chats can be saved on the server, and loaded upon login.

To Be Completed: C-1

### Outcome:

To be advised.

## Local User Data Caching, Message Confidentiality.

### Description:

Local user data caching will help to improve performance, speed, and efficiency of our application, by reducing the amount of data that has to be requested and received from the server. It will also allow users to view some information on the app when they do not have an internet connection.Message confidentiality will help to ensure that people's messages are safe and secure.

### Justification:

This cover lots of NFRs in areas such as performance, efficiency, and security.

### Deliverables:

* Some chosen data is cached locally, resulting in less data to be requested from the server, and offline access to this data.
* Messages have a given level of confidentiality.

To Be Completed: C-3

### Outcome:

To be advised.

## User workflow, Delete Chats, Edit Account

### Description:

Users workflow is enforced so that the user can only follow the specified allowed workflow, and it is made obvious what this is and how a user would complete a given task.Users can delete chats via the apps GUI.Users can edit their account details via the apps GUI.

### Justification:

Managing accounts and chats will get us to our stretch target. User workflow is important in making the application nice and usable.

To Be Completed: C-4

### Outcome:

To be advised.

## Manage Contacts

### Description:

Users can manage their contacts from the app GUI. Performing tasks such as adding a contact, deleting a contact, muting a contact, or blocking a contact.

### Justification:

Allowing the user to manage their contacts is necessary to reach our standard target.  
To Be Completed: C-2

### Outcome:

To be advised