



PROJECT REPORT

JAVA   
  
Client-Server Based UTM Onboarding System

|  |  |  |  |
| --- | --- | --- | --- |
| **Created By:** | Vedant Patil | **Approved By:** | <Domain Lead Name> |
| **Created On:** | 28/06/2025 | **Approved On:** | DD-MMM-YYYY |

Page left blank intentionally

**INDEX**

[**1** **PROJECT DETAILS** 2](#_Toc143445375)

[**2** **SUMMARY** 2](#_Toc143445376)

[**3** **INTRODUCTION** 2](#_Toc143445377)

[3.1 Background 2](#_Toc143445378)

[3.2 Stakeholders 2](#_Toc143445379)

[3.3 Objectives 2](#_Toc143445380)

[**4** **METHODOLOGY** 2](#_Toc143445381)

[4.1 Considerations & Assumption 3](#_Toc143445382)

[4.2 Approach 3](#_Toc143445383)

[4.3 Activities 3](#_Toc143445384)

[**5** **TARGETTED V/S ACHIEVED OUTPUT** 3](#_Toc143445385)

[**6** **CONCLUSION** 3](#_Toc143445386)

[**7** **APPENDICES** 4](#_Toc143445387)

[7.1 Appendix A – Title 4](#_Toc143445388)

**General Instructions for using the Live Project Report Template**

* This template and the subsequent document created using this template is a confidential document and is the intellectual property of Cloud Counselage Pvt. Ltd. Circulating it outside of the organisation without the consent of Cloud Counselage Pvt. Ltd. is the breach of company policies and will lead to legal actions
* This template is a guideline document to communicate the implementation of design ideas and the results of the work to the stakeholders.
* The **text between inequality (< >) is to be replaced** by relevant text
* Please **remove the yellow highlight on the Text** between the inequality (< >). This is done to help you notice the text to be changed/replaced
* The text in *italics* highlighted in grey is just for reference and should be removed after adding the relevant text

# **PROJECT DETAILS**

|  |  |  |  |
| --- | --- | --- | --- |
| **Project Name** | Client-Server Based UTM Onboarding System | | |
| **Project Sponsor** |  | | |
| **Project Manager** |  | | |
| **Start Date** | 24/05/2025 | **Completion Date** | 27/06/2025 |

# **SUMMARY**

This is a console-based Java application that automates the onboarding process of Community Ambassadors (CA) for the Industry Academia Community (IAC). The system allows users to enter their name and email through the console. It then generates a unique UTM link for each new ambassador and sends a welcome email containing this link using Java Mail API. The application uses Java Socket Programming to simulate communication between the client (user) and the server (which processes data and sends the email). This project helps reduce manual effort in the onboarding process and ensures faster and accurate UTM link generation and delivery.

# **INTRODUCTION**

## Background

The Industry Academia Community (IAC) aims to help 10 million students become job-ready by the end of 2028. To achieve this, they onboard student volunteers known as Community Ambassadors (CAs) who spread awareness and encourage other students to join the initiative.

Each CA receives a unique UTM link that helps the organization track their performance and contribution. This UTM link is currently sent manually after the CA submits a form, which takes time and effort.

To improve efficiency and accuracy, there is a need to automate this onboarding process—specifically the generation of the UTM link and sending it to the CA’s email. This project addresses that need by providing a Java-based console application that handles the process automatically.

## Stakeholders

Community Ambassadors (CAs) – Use the UTM links to share IAC and need quick onboarding.

IAC Team – Want to track CA performance and save time by automating the process.

## Objectives

To develop a console-based Java application that automates the Community Ambassador (CA) onboarding process by:

Generating a unique UTM link from the name submitted in the New Joinee Form.

Automatically sending a Welcome Email with the UTM link to the CA’s email address.

Helping the IAC team save time and track performance of CAs through UTM-based referrals.

# **METHODOLOGY**

These conventions are all about the positions of line breaks, how many characters should go on a line, and everything in between.

## Considerations & Assumption

The application will run in a console-based environment like VS Code.

Internet connection is required to send emails using JavaMail.

User will enter correct name and email while onboarding.

Gmail account used for sending emails must have App Password enabled.

Server and client are assumed to be running on the same machine (localhost).

No database is used; only real-time processing is done.

The application is for internal use to automate onboarding, not for public deployment.

## Approach

Used a simple client-server model in Java.

The client takes name and email from the user.

The server receives the data and creates a UTM link.

Then it sends a welcome email with the UTM link using JavaMail API.

This approach automates the onboarding process in a lightweight, console-based system.

## Activities

Collected requirements from the problem statement

Planned the structure of client-server communication

Wrote and tested Java code for client, server, and email sending

Integrated email functionality using JavaMail API

Performed testing to ensure correct UTM link generation and delivery

Documented the project details and outputs

# **TARGETTED V/S ACHIEVED OUTPUT**

Targeted Output:

The aim was to create a Java-based system that automatically sends a welcome email with a personalized UTM link when a new CA (Community Ambassador) joins.

Achieved Output:

Successfully built a console-based Java application that accepts user input, generates a UTM link, and sends the email using the JavaMail API.

# **CONCLUSION**

This project successfully automated the CA onboarding process by generating UTM links and sending them via email. It helped reduce manual effort and ensured quick and accurate communication. The console-based Java application meets the problem statement requirements and is simple, effective, and easy to use

# **APPENDICES**

## Appendix A – Title Project Overview and Components

This appendix includes a brief summary of the project, key components like Client, Server, EmailSender, and how they work together to generate and send UTM links for CA onboarding