

**Friendstagram**  
**Release Plan**

Version: 1.0  
Last Modified: March 27, 2022

## APPROVALS

**Submitting Organization's Approving Authority:**

<b>Signature</b>	<b>Printed Name</b>	<b>Date</b>	<b>Phone Number</b>
Clarence Hong	Clarence Hong	26 March 2022	91521337

**CMS' Approving Authority:**

<b>Signature</b>	<b>Printed Name</b>	<b>Date</b>	<b>Phone Number</b>
Dr Shen Zhiqi	Dr Shen Zhiqi		

## REVISION HISTORY

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
0.10	Clarence Hong	11/03/2022	Clarence Hong	11/03/2022	Initial Document Baseline
0.20	Royce Ang Jia Jie, Jovan Huang Tian Chun, Zhu Weiji	19/03/2022	Clarence Hong	19/03/2022	Added Sections 1, 2, 3 and 4
0.30	Jovan Huang Tian Chun, Tan Hui Zhan	26/10/2021	Clarence Hong	26/10/2021	Added Sections 5, 6 and 7
1.0	Clarence Hong	27/03/2022	Clarence Hong	27/03/2022	Final Document check

## **TABLE OF CONTENTS**

<b>1. INTRODUCTION</b>	<b>6</b>
<b>2. REFERENCED DOCUMENTS</b>	<b>6</b>
<b>3. OVERVIEW</b>	<b>6</b>
<b>4. ASSUMPTIONS, CONSTRAINTS, RISKS</b>	<b>8</b>
4.1. Assumptions	8
4.2. Constraints	8
4.3. Risks	9
<b>5. RELEASE APPROACH</b>	<b>11</b>
5.1. Rationale	11
5.2. Release Strategy	11
5.2.1. Release Content	12
5.2.2. Release Schedule	13
5.2.3. Release Impacts	<b>15</b>
5.2.4. Release Notification	16
<b>6. GLOSSARY</b>	<b>16</b>
<b>7. ACRONYMS</b>	<b>17</b>

## **LIST OF FIGURES**

Figure 1: Use Case Diagram for Friendstagram.....	7
---	---

## **LIST OF TABLES**

Table 1: Referenced Documents.....	6
Table 2: Risks and Impacts and Strategy for the release of Friendstagram.....	9
Table 3: Release Content of each release version.....	12
Table 4: Release Schedule of each release version.....	13

## 1. INTRODUCTION

This document contains the release life cycle of the Friendstagram web application. This includes the beginning of the development stage to the official release as well as the future updates to the web application. All releases follow the format of release date and release version, such as YYYYMMDD\_v2.0. The release plan intended audience are the development team, the release team and relevant stakeholders of the Friendstagram web application. The release plan is updated every time a verified working new version of the web application is released. All necessary information regarding the web application's release life cycle will be recorded in the release plan.

## 2. REFERENCED DOCUMENTS

*Table 1: Referenced Documents*

Document Name	Issuance Date
Project Proposal	20 February 2022
Software Requirement Specification	20 February 2022
Project Plan	20 February 2022
Software Quality Assurance	6 March 2022
Risk Management	6 March 2022

### 3. OVERVIEW

The objective of ‘Friendstagram’ is to provide opportunities for users to find new potential friends of similar interest and locations. The web application will be optimized for mobile devices as students are regularly on their mobile phones. Various requirements have been identified for the web application, along with the proposed solution to fulfill these requirements.

Since the goal of the project is to create a web application that recommends new potential friends to students without having any unessential features, we decided to use a Minimum Viable Product (MVP) approach. This MVP approach is taken after considering various factors such as duration of the project, team size, and the technical skills of each individual member. The following functionalities will be included in the web application:

- a. Register with personal account
- b. Log in with personal account
- c. Log out personal account
- d. View user’s information
- e. Edit user’s information
- f. Recommend new potential friends with similar interests to users through our questionnaire and algorithm
- g. Display contact information and interests of a new potential friend from the list of recommended new potential friends

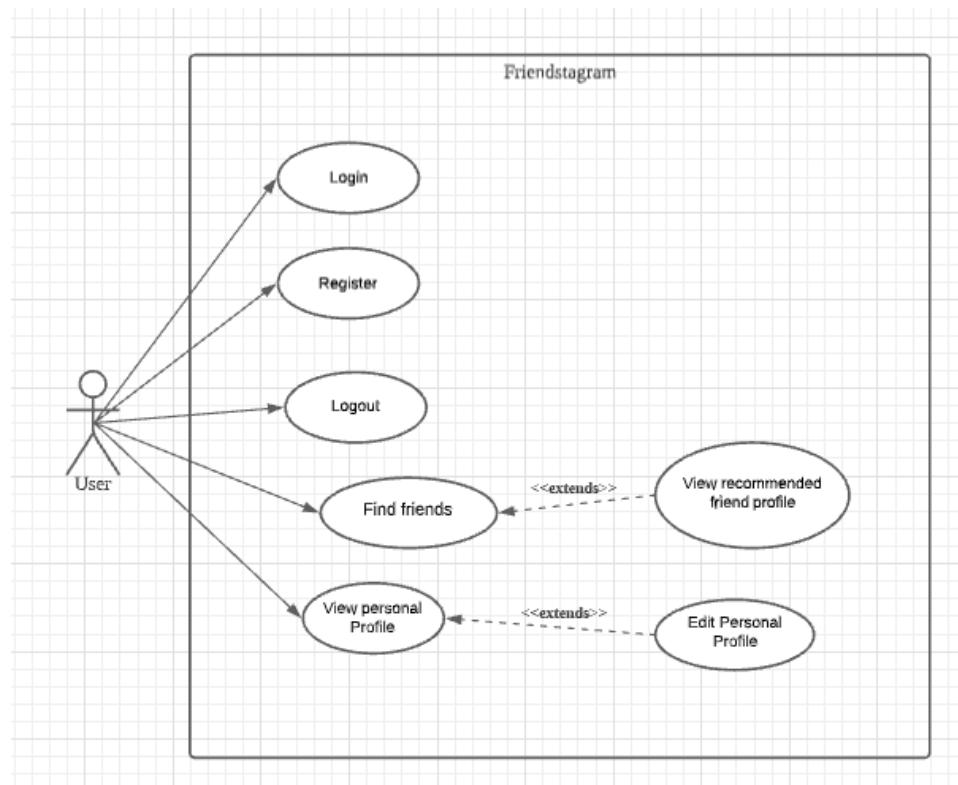


Figure 1: Use Case Diagram for Friendstagram

## 4. ASSUMPTIONS, CONSTRAINTS, RISKS

### 4.1. Assumptions

1. Users are assumed to be students of Nanyang Technological University
2. Users have internet access
3. Users have access to a device with a browser installed that is capable of accessing the Friendstagram web application
4. The web application should still function regardless of the number of concurrent users

### 4.2. Constraints

The duration of the Friendstagram project is limited to only three months, leading to our major constraint of duration. As the members of the team have other modules to study, there will be tests and assignments that will conflict with the release day of the application. The members must let the team know about their test and assignment dates in advance to ensure that the team can organize and plan the project schedule and release date in advance. This will reduce the likelihood of having clashes with the release day of the application.

The manpower for the Friendstagram project is limited to only 6 people. The members of the teams are just students with not much experience in developing major applications. Thus, the requirements that the team sets needs to be appropriate and feasible for a team of 6 students that can be completed in time.

There is a budget constraint for the Friendstagram project as only free softwares such as Netlify and PythonAnywhere can be used. Proper discussion and planning must be done before starting the project to decide on the technological stack that is going to be used without requiring any funds.

### 4.3. Risks

*Table 2: Risks and Impacts and Strategy for the release of Friendstagram*

Risk	Impact	Strategy
Team member unavailable before project completion	There will be a lack of manpower required to complete the project on time. Remaining team members will have increased workload to meet the project deadline, leading to increased stress levels and potentially conflicts.	Team members are separated into pairs who can take over each other's role in the event that one needs to leave due to emergencies such as medical injury, catching COVID-19, etc. Otherwise, reassignment of tasks can be done if an individual is unable to cope with their deadlines.

Requirement miscommunication	This can lead to increased workload due to the development of features that are not required or missing out required features of the software. This will result in missing the deadline for the project due to the increased workload.	Sufficient time must be allocated for requirement solicitation and to draft detailed documentation. The team is required to follow the detailed documentation to prevent requirement miscommunication. In the event that a requirement changes, all team members must be notified before making changes to documentation.
Internal conflict between teammates	Work productivity will drop as teammates are unwilling or avoid communicating with one another. Internal conflicts will lead to friction that reduces the overall productivity of the team.	To reduce the likelihood of conflict, ensure that clear and concise communication is exercised at all times. When a conflict arises, the project manager must quickly deescalate the situation and ensure that the issue is resolved immediately. The project manager must be objective and avoid taking sides. If the project manager is involved in the conflict, other team members must step in to defuse the situation. The group must seek assistance from outside sources such as mentors/counselors/school/government when the conflict cannot be resolved internally.
Deadline delayed due to other assignments	Assignments and tests will reduce the amount of time each team member can spend on the development of Friendstagram. If the reduced time is not taken into account during the project scheduling, deadlines will be delayed as there will not be enough time to finish the project on time.	Team members must have good personal time management. Team members need to inform the team of future assignments and tests at the start of the project. The project schedule is then constructed based on the different assignments and tests of each team member, planning ahead of potential time constraints. This will reduce the probability of delaying the deadline due to other assignments clashing with the project schedule.
Lack of physical team meetings	Unable to discuss project progress physically, leading to lack of awareness on each team member's progress on their assigned workload.	Due to COVID-19, physical meetings have been minimized. Weekly online meetings will be held on platforms such as Zoom to mitigate this risk. Regular updates on the project will be done on online messaging platforms

		such as WhatsApp or Telegram. This allows the team to communicate with one another easily even without physically meeting.
--	--	--

## 5. RELEASE APPROACH

### 5.1. Rationale

Each new release of the application will offer new functionalities of the web application. With reference to the System Requirement Specification(SRS) document, functional requirements can be separated into multiple versions of release for the web application. This allows the team to easily keep track and stay on schedule by planning each release to have specific functionalities early.

The team must ensure that the new functionalities are tested thoroughly and working accordingly before each release. Testing must be done concurrently with the development of the functionalities to detect issues as early as possible during the development phase. This will reduce the likelihood of delaying a release version due to functionalities having issues. Therefore, the team must work closely together in parallel to be able to test and develop concurrently.

### 5.2. Release Strategy

Phased rollout will be the release strategy used. Each functionality will be extracted from the SRS and placed in a list. The list will then be sorted according to the importance of each function in terms of the interoperability of the application. The sorted list of functions will then be separated into multiple releases, each release containing a manageable amount of work for the team. The recommendation feature must be the first function to be released as it is the foundation of the web application.

Every release must follow the release plan, which documents the steps and actions required for each release. The release will go through User Acceptance Testing(UAT) where the client will be given the finalized web application. The client will evaluate the web application and will inform the team of any errors that were missed by the team during testing. Otherwise, if there are no errors, the client approves the web application and it will be prepared for deployment.

#### 5.2.1. Release Content

Table 3: Release Content of each release version

Release version	Functionality
Version 1.0	<ul style="list-style-type: none"><li>• User can create an account</li><li>• User can choose their interests and halls</li><li>• User can find new friends through recommendations</li></ul>

Version 1.1	<ul style="list-style-type: none"> <li>User can rate their interests</li> <li>User can login to an existing account</li> <li>User can log out of the current account</li> </ul>
Version 1.2	<ul style="list-style-type: none"> <li>User can view/edit their personal profile</li> <li>User can view recommended friends' profile</li> <li>User can view recommendation history</li> </ul>
Version 2.0	<ul style="list-style-type: none"> <li>Bug fixes and improvements</li> </ul>

### 5.2.2. Release Schedule

*Table 4: Release Schedule of each release version*

Release Version	Milestone	Start Date	End Date
Version 1.0	Requirement Specification	25/02/22	28/02/22
	Design (Architectural and UI)	29/01/22	02/02/22
	Implementation	03/02/22	05/02/22
	Testing	06/02/22	08/02/22
	Release plan	09/02/22	09/02/22
	Build release	10/02/22	10/02/22
	User Acceptance Testing	11/02/22	11/02/22
	Prepare release	12/02/22	12/02/22
	Deploy release	13/02/22	13/02/22
<b>Version 1.0 Release Date: 13/02/2022</b>			
Version 1.1	Analyze and Update Requirements	14/02/22	16/02/22
	Implementation	17/02/22	19/02/22

	Testing	20/02/22	22/02/22
	Release plan	23/02/22	23/02/22
	Build release	24/02/22	24/02/22
	User Acceptance Testing	25/02/22	25/02/22
	Prepare release	26/02/22	26/02/22
	Deploy release	27/02/22	27/02/22

**Version 1.1 Release Date: 27/02/2022**

Version 1.2	Analyze and Update Requirements	28/02/22	02/03/22
	Implementation	03/03/22	05/03/22
	Testing	06/03/22	08/03/22
	Release plan	09/03/22	09/03/22
	Build release	10/03/22	10/03/22
	User Acceptance Testing	11/03/22	11/03/22
	Prepare release	12/03/22	12/03/22
	Deploy release	13/03/22	13/03/22

**Version 1.2 Release Date: 13/03/22**

Version 2.0	Analyze and Update Requirements	14/03/22	16/03/22
	Implementation	17/03/22	19/03/22
	Testing	20/03/22	22/03/22
	Release plan	23/03/22	23/03/22
	Build release	24/03/22	24/03/22
	User Acceptance Testing	25/03/22	25/03/22
	Prepare release	26/03/22	26/03/22

	Deploy release	27/03/22	27/03/22
<b>Version 2.0 Release Date: 27/03/22</b>			

### 5.2.3. Release Impacts

Version 1 in full is the foundation of the web application, it must be tested thoroughly to ensure that there is a solid foundation for the whole system. All main functionalities will be included in this version.

Version 1.0 includes the core functionality of the web application, where users are required to input their interests and halls, which will be used to recommend other users through the system. Since inputting interests and halls is done during the account creation stage, account creation is part of this version.

Version 1.1 includes other main functionality of the web application, such as allowing users to log in and out of existing accounts, as well as rating their interests. Rating interests is an important functionality but not a core functionality which is why it is under this version. Rating interests helps the system to match more closely related profiles, giving the user better recommendations.

Version 1.2 includes additional functionalities of the web application, such as viewing/editing personal profiles and viewing of recommendation history. Viewing/editing personal profiles allows users to change their interests and halls to their current circumstances. Viewing of recommendation history helps users find past recommendations if they were to forget or misclick out of the web application.

### 5.2.4. Release Notification

Every release version must be completed 3 days in advance before the official release date. The 3 day window is to cater for unforeseen issues such as undetected bugs in the web application. The release manager will supervise each release and ensure that the team is on track with the release schedule without any delays. All team members and stakeholders will be notified with the information of the release. Information includes version number, new functionalities added or changed, official release date and testing results.

## **6. GLOSSARY**

### **GitHub**

GitHub is a provider of Internet hosting for software development and version control using Git. It offers the distributed version control and source code management functionality of Git, plus its own features.

### **Git**

Git is software for tracking changes in any set of files, usually used for coordinating work among programmers collaboratively developing source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows.

### **Zoom**

Zoom is a cloud-based video communications app that allows you to set up virtual video and audio conferencing, webinars, live chats, screen-sharing, and other collaborative capabilities.

## **7. ACRONYMS**

**COVID-19:** CoronaVirus Disease of 2019

**MVP:** Minimum Viable Product

**SRS:** System Requirement Specification

**UAT:** User Acceptance Testing