

Hazard Analysis Software Engineering

Team #2, Campus Connections

Waseef Nayeem

Zihao Du

Matthew Miller

Firas Elayan

Abhiram Neelamraju

Michael Kim

Table 1: Revision History

Date	Developer(s)	Change
Oct 20th	All	Revision 0
...

Contents

1 Introduction

Based on the STPA Handbook, a system hazard is a system state or set of conditions that, together with a particular set of worst-case environmental conditions will lead to a loss. Regarding CampusConnections, our AR-based social networking application, a hazard can be a condition in the game when it fails to perform the intended functions or performs unexpected behaviors when coupled with environmental conditions. This document aims to detect, analyze, assess, and eliminate or migrate potential safety and security hazards that are applicable to this application.

2 Scope and Purpose of Hazard Analysis

The scope of hazard analysis is to specify all potential system hazards that may arise when using the application and discover safety and security requirements to migrate and eliminate the effects of those hazards. However, it will not include hazards related to the hardware the application is running on. It will be the choice of the user and we cannot account for all mobile devices on the market. Hazard to the user and the society will be out of the scope as well. We will assume users intend to run the application on a normally functioning mobile device properly and efficiently. The purpose of the document is to highlight various hazards associated with the system, effects and causes of corresponding failures along with new requirements for further mitigation steps.

3 System Boundaries and Components

The system will be divided into the following components:

1. The application's in-game feature components:
 - Social Media
 - AR & Location Services
 - Event/Lecture Management
 - General application features
2. The database being used which will store all of users' data

General app features include user login system, it will be responsible for user login and account creation, as well as notification and user accessibility management. The other three features are just responsible for corresponding in-game functionalities, more details can be found in the figure below. The database and user interaction are considered external to the system, the interaction between the system and external systems is described in the previous [document](#).

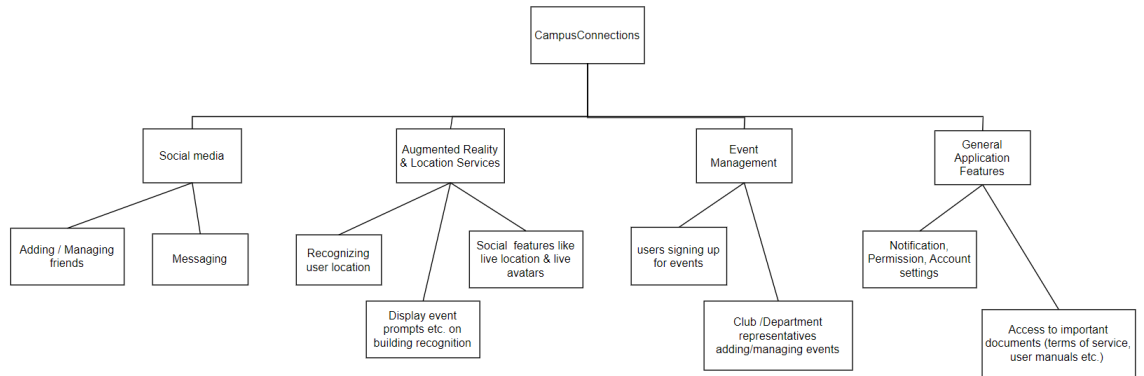


Figure 1: System Components

4 Critical Assumptions

- Assume the users of the application do not intend to misuse it
- Assume the user's device will have all necessary hardware components with sufficient computing/output power such as sensors, processors, etc.
- Assume the routes to the backend of the system will always be ready to serve requests and not blocked due to unnecessarily locked resources

5 Failure Mode and Effect Analysis

[Include your FMEA table here —SS]

Design Function	Failure Modes	Causes of Failure	Effects of Failure	Detection	Recommended Action	SR	Ref. No.
F21: AR Object Recognition	App is unable to detect object	Poor lighting conditions Camera angle	User is unable to view information from the AR element	Keep track of prior incomplete scan attempts	Implement a failsafe mechanism that shows the scannable information if enough previous attempts have failed		
14.1 Backend Server	Server is inaccessible	No internet connection Hosting service is down Invalid access key	Users are unable to make changes to their profile Users cannot receive updated information from the server	Keep track of the connection to the server using periodic heartbeats or similar methods	Display an error message stating that the connection the the server has been lost		
User Profile View	Sensitive user data is publicly visible	Invalid visibility permissions Access control software malfunction	Confidential user information could be exposed and used for malicious purposes	Software failsafes and checks that verify the integrity of the permission structure	Prevent other users from accessing unauthorized information by creating a robust permissions system		
12 App Performance	App performance is poor	Large amounts of assets to be rendered Slow internet connection App is running on an old phone	User experiences lag or slowdown App feels unresponsive	Track of frame times or other performance metrics	Only show up to a maximum number of avatars or lower the level of detail		

12.2 User Safety	App is used where and when it is not intended	User is distracted by the app	User could get into an accident	N/A	Display a warning message when opening the app that tells the user to be aware of their surroundings at all times while using the app		
AR Module	Device is not compatible with AR features	Device is missing required hardware or software	User cannot use the AR features	Check if device supports the required feature set using API functions.	Display a warning message if the user's device is not compatible		
F1: Account Creation	Duplicate ID Creation	1. Account login failure 2. Function returns incorrect results 3. Incorrect user added as friend	1. Duplicate ID was used during account creation 2. Duplicate ID was used during account update	Compare ID with existing ID in database	Checks whether the ID is duplicate or not when entered and when saving		

F7: Share location with friends	Location shown unintentionally	1. Location known without application being focused 2. Incorrect location display on other users	1. User did not turn off their GPS and application	Application only sends location information for too long	If user has not interacted with the application for a prolonged period, check for activity, and turn off all processes if no response is received		
15.1 Unauthorized user activity	1. Incorrect login attempts 2. Account activity in unknown devices 3. Unknown administrator activity	1. Account locked for user until further checks 2. Compromise of user and friend data	1. Predictable passwords allowed, improper password change rules 2. Security measures such as second factor authentication not used 3. Account lock features were not used	1. Account used from previously unknown account 2. Suspicious activity in the logs	1. Too many incorrect passwords will result in account lock. The lock can be undone from email. The account ID and password can be changed with rules preventing predictable passwords such as cannot use same password. 2. Recommend all users to use multi-factor authentication and have logs of their previous activity available to them. 3. Notify users of the security measures of the application and procedures of using them		

14.1 Main- tenance and up- dates of features	Update can- not be fin- ished in time or major er- ror is found	1. Users could ex- perience unexpected down times 2. Features could be delayed	1. Update was not properly tested 2.					
---	---	---	---	--	--	--	--	--

6 Safety and Security Requirements

[Newly discovered requirements. These should also be added to the SRS. (A rationale design process how and why to fake it.) —SS]

6.1 Safety Requirements

- The product shall not transmit information while not in use.
 - **Rationale:** This requirement limits the battery usage of the product.
 - **Fit Criterion:** The product will not execute any code that involves the transmission of information outside of the product.

6.2 Access Requirements

6.3 Integrity Requirements

6.4 Privacy Requirements

6.5 Audit Requirements

6.6 Immunity Requirements

7 Roadmap

[Which safety requirements will be implemented as part of the capstone timeline? Which requirements will be implemented in the future? —SS]

Safety Requirements to be implemented for capstone:

- The product shall not transmit information while not in use.

Safety Requirements to be implemented after capstone:

-