

# GroupE4 Requirements Specification

*BuzzNet*

**Group E4**

## **Group Members:**

**LI Chun Leung** 155193164  
**PENG Minqi** 1155191548  
**WONG Kwok Kam** 1155192018  
**ZENG Bai Chuan** 1155193167  
**ZHANG Ka Sing** 1155194769

**Department of Computer Science and Engineering**

**Version 1.2**

**9 Feb 2025**

## **Document Revision History**

<b>Version</b>	<b>Revised by</b>	<b>Revision Date</b>	<b>Comments</b>
0.1	WONG Kwok Kam	4 Feb 2025	Initial draft.
1.0	LI Chun Leung, PENG Minqi, WONG Kwok Kam, ZENG Bai Chuan, ZHANG Ka Sing	6 Feb 2025	Major updates to all sections after 1st general meeting. Especially on functional requirements, clarify the roles and their functions.
1.1	WONG Kwok Kam	7 Feb 2025	Commenting and modifying functional requirements and document layout after 2nd general meeting.
1.2	WONG Kwok Kam	9 Feb 2025	Major updates based on comments.

# 1 Introduction

This Software Requirements Specification (SRS) document is intended to capture the complete software requirements for the Project ABC. This document is intended to capture the scope and requirements of the desired system to be approved by the client and to be used by the software development team as a way to understand what needs to be done, and to serve as a basis for software design, development, and testing.

In addition to system functionality, the SRS also describes the non-functional requirements, design constraints, and other factors necessary to provide a comprehensive view of the requirements for the software.

## 1.1 Overview

The ***Buzznet*** web application is designed to provide a public social media platform for users to share text and image-based content. The system will support core social media functionalities including post creation and user interactions, including likes, comments, follow and repost.

The application will implement role-based access control with distinct privileges for standard users and administrators. Core features include content moderation tools. Users can delete their own posts and comments. Also they can post anonymously which means that their identity will not be revealed under anonymous posts (cannot be followed).

A successful implementation will enable seamless content discovery while maintaining community guidelines through automated content flagging and user reporting mechanisms. The system aims to foster real-time public discourse while ensuring secure and moderated user interactions.

## 1.2 Definitions, acronyms, and abbreviations

Term, Acronym, or Abbreviation	Definition
UI	User Interface
UX	User Experience
2FA	Two-Factor Authentication
CAPTCHA	Completely Automated Public Turing test to tell Computers and Humans Apart

## 1.3 References

The following is a list of all references used to produce this SRS.

Reference	Date	Published by	Source
Software Requirements Specification For (TMS)	January 2003	U.S. Agency for International Development	<a href="#">Software Requirements Specification For (TMS)</a>

## 2 Assumptions

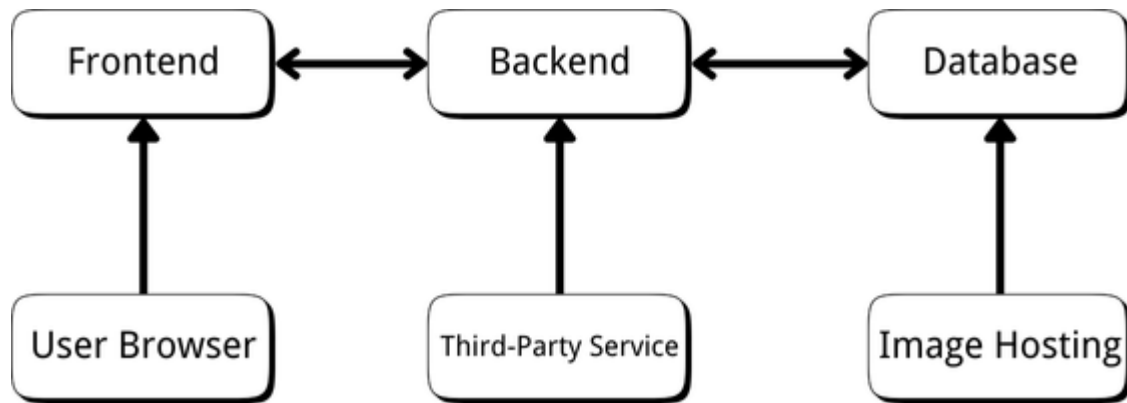
This document is based on the following assumptions.

Assumption No.	Title	Description
1	User Connectivity and Skilled Level	All users of this system are assumed to have the ability to connect to the Web application (via Internet or Intranet) and to have Web browsing capability.
2	Web Browser	Supported Web Browsers will be needed, including Chrome, Edge, Firefox.
3	Supported Operating Systems	Able to run on Linux, Windows, or Android released within 5 years
4	Supported Language	The user interface and data will be English or Chinese. The user can set the UI language in Setting. And some other languages can also be used in certain situations, like user names.
5	Runtime Environment	The runtime hardware, software, and networking environment will be supplied in fully operational condition by the client.
6	Screen Resolution	Optimal end user screen resolution will be 1280 * 720.
7	Public Events and Program Announcements	There are text (free-form) announcements to be accessible from the system's home page without login.
8	Testing Prerequisite	Employees can directly take tests.
9	Change Request	The development of the system will proceed based on the client requirements. Any additional requirements and/or changes of existing requirements may result in changes to the development schedule and/or the project price. Change Requests must be submitted formally.
10	Resource Accessibility	The development group should keep the period within 2 months. Individuals in groups can take more than 20 hours per week. The project group can use the Linux server and other Cloud resources provided by the school.
11	Teamwork	Engineers in the group should be proficient in Git or other version control methods to make the program organized.
12	Report Technique	There is no specific reporting technology identified as standard. But the group should keep reporting per week.

## **3 High-level System Architecture**

The system will follow a client-server architecture with a clear separation between the frontend (client) and backend (server). Below is a high-level overview of the architecture:

### **3.1 Architecture Overview:**



### **3.2 System Flow:**

#### **3.2.1 User Interaction**

The user interacts with the frontend by the web browser (e.g., writes a blog post, likes a post, etc).

#### **3.2.2 API Request**

The frontend sends an API request to the backend (e.g., to save a post or retrieve comments, etc).

#### **3.2.3 Backend Processing**

The backend processes the request, performs necessary operations (e.g., database queries, etc), and returns a response.

#### **3.2.4 Data Storage**

The backend interacts with the database to store or retrieve data.

#### **3.2.5 Response to User**

The frontend updates the UI based on the backend's response.

## **4 Functional Requirements**

### **4.1 Functional Requirements**

The following table summarises the functional requirements for the *Buzznet*.

<b>Req. No.</b>	<b>Title</b>	<b>Description</b>
FR1.1	Registration	Users shall register using a valid email address and password.
FR1.2	Login	Users shall log in with email/password and 2FA (e.g., OTP via email or authenticator app).
FR1.3	Password Resetting	Users shall reset their password via email.
FR2.1	Post Creation	Users shall create posts with text and/or images.
FR2.2	Post Deletion	Users shall delete their own posts/comments.
FR2.3	Anonymous Post	Users shall post anonymously, hiding their identity (username not displayed, no follow option).
FR3.1	Post Reaction	Users shall like, comment on, and repost posts.
FR3.2	Subscription	Users shall follow other users through user profile or non-anonymous posts.
FR3.3	Post Categories	Users shall be able to categorize their posts into predefined categories (e.g., News, Sports, etc.).
FR3.4	Search Functionality	Users shall be able to search for posts and comments by keywords or categories.
FR4.1	Regulatory Control	Administrators shall delete any post and ban users.
FR4.2	Report	Users shall report posts/comments for violating guidelines.
FR4.3	Assessment	The system shall enforce role-based access control (user vs. admin).
FR5.1	Anonymity	Anonymous posts shall not reveal user identity in any UI component.
FR5.2	Logging	The system shall log all actions for audit purposes.
FR6.1	User Comments Storage	The system shall allow users to submit comments on posts, which will be stored in the database.
FR6.2	Display Comments	Users can view comments on posts, and comments will be shown although the pages refresh.
FR6.3	Comment Moderation	Users and Administrators shall have the ability to delete inappropriate comments.

## 4.2 Actors

An actor is anything that interacts with the data (e.g., viewing, modifying, etc). The following are identified actors of the software.

### 4.2.1 Standard User

Registered users can create content, interact with posts, and manage their account.

### 4.2.2 Administrator

Administrators can have privileges on top of standard users, such as, manage content moderation, ban users, and modify system settings.

### 4.2.3 System

Automated processes (e.g., sending 2FA codes, notifications, etc).

## 4.3 Use Cases

A use case defines a set of use-case instances, where each instance is a sequence of actions a system performs that yields an observable result of value to a particular actor. Following are the identified use cases.

### 4.3.01 Register Account for User

**Initiator:** Standard User

**Description:** User provides unique username, unique email and password to create an account.

#### Basic Flow of Events

User provides a username, email, and password → System validates the input → If valid, the system creates the account → System sends a confirmation email to the user's email address.

#### Exceptional Flow of Events

E-1: Account name and/or email inputted by the user is used. The user should use another username and/or email.

### 4.3.02 Register Account for Administrator

**Initiator:** Administrator

**Description:** Administrator creates the account in the account database directly.

#### Basic Flow of Events

Administrator provides a username, email, and password directly.

#### Exceptional Flow of Events

E-1: Account name and/or email inputted by the user is used. The user should use another username and/or email.

### 4.3.03 Login

**Initiator:** Standard User, Administrator

**Description:** User / Administrator logs in with password, if correct then enters a 2FA code to login.

**Precondition:**

The user has already finished the registration of the account and is present in the database.

#### Basic Flow of Events

User inputs their username and password → System validates the credentials → System prompts the user for a 2FA code → User inputs the correct 2FA code → System grants access to the user.

#### Exceptional Flow of Events

E-1: The user forgot the password. Then the user should reset the password via email (See **4.3.13**).

E-2: The credentials (username, password and 2FA code) of the user cannot be validated. Then the user cannot log in unless they provide the correct credentials.

### 4.3.04 Create Post

**Initiator:** Standard User, Administrator

**Description:** User / Administrator publishes text/image content. Can toggle "anonymous" mode.

#### Basic Flow of Events

User (logged in) navigates to the post creation page → User inputs text and/or uploads an image → User optionally adds category to the post → User selects whether to post anonymously or not → User submits the post → System saves the post and displays it on the user's profile.

#### Exceptional Flow of Events

E-1: The user tries to submit an empty post. The system prompts the user to enter content.

### 4.3.05 Delete Post/Comment

**Initiator:** Standard User, Administrator

**Description:** User removes their own post/comment. Administrator removes inappropriate posts reported by the users.

#### Basic Flow of Events

For User (logged in): Navigates to their posts or comments → User selects the post/comment they want to delete → User confirms the deletion → System removes the post (and all reposted posts associated with this post) or comment from the database.

For Administrator (logged in): Navigates to a tab showing inappropriate posts/comments → Administrator selects the post/comment they want to delete → Administrator confirms the deletion → System hides the post/comment.

#### Exceptional Flow of Events

E-1: The user attempts to delete a post/comment that does not belong to them. The system denies the action.

#### **4.3.06 Follow User**

**Initiator:** Standard User, Administrator

**Description:** User / Administrator follows another non-anonymous account.

##### **Basic Flow of Events**

User / Administrator views the profile of another user → User / Administrator clicks the 'follow' button → System updates the following list of the user.

##### **Exceptional Flow of Events**

E-1: The User / Administrator cannot follow the account in an anonymous post. The follow button will not be shown and the system will deny it if the user tries to follow.

#### **4.3.07 Send 2FA Code**

**Initiator:** System

**Description:** System generates and sends a 2FA code via email after password validation.

##### **Basic Flow of Events**

User logs in with their username and password → System validates the credentials → System generates and sends a 2FA code to the user's registered email.

##### **Exceptional Flow of Events**

E-1: The system fails to send the 2FA code due to a server error. The system notifies the user to try again later.

#### **4.3.08 Anonymous Post**

**Initiator:** System

**Description:** System hides user identity for posts marked as anonymous.

##### **Basic Flow of Events:**

User toggle on the anonymous option while creating a post → System saves the post to the database → System displays the anonymous post without revealing the user's identity.

##### **Exceptional Flow of Events:**

Nil.

#### **4.3.9 Post Categories**

**Initiator:** Standard User, Administrator

**Description:** User / Administrator categorizes his / her posts into predefined categories.



**Basic Flow of Events:**

User navigates to the post creation page → User selects a category from a predefined list, if not then the post will have a default category → User submits the post → System saves the post with the selected category → System shows the post in corresponding category.

**Exceptional Flow of Events:**

Nil.

**4.3.10 Search Functionality**

**Initiator:** Standard User, Administrator

**Description:** User / Administrator searches for posts and comments by keywords or categories.

**Basic Flow of Events:**

User enters a keyword or tag in the search bar → System retrieves relevant posts/comments → System displays the search results to the user.

**Exceptional Flow of Events:**

E-1: The user enters a keyword that does not match any posts/comments. The system informs the user that no results were found.

**4.3.11 Report**

**Initiator:** Standard User, Administrator

**Description:** User /Administrator reports posts/comments for violating guidelines.

**Basic Flow of Events:**

User views a post/comment they wish to report → User clicks the report button → System prompts the user to select or provide a reason for the report → User submits the report → System logs the report for administrator's review.

**Exceptional Flow of Events:**

E-1: The user attempts to report their own post/comment. The system notifies the user that they cannot report their own content.

**4.3.12 User Comments**

**Initiator:** Standard User, Administrator

**Description:** User submits comments on posts.

**Basic Flow of Events:**

User navigates to a post → User enters a comment in the comment section → User submits the comment → System stores the comment in the database and displays it under the post.

**Exceptional Flow of Events:**

E-1: The user submits an empty comment. The system warns the user that the comment cannot be empty.

### 4.3.13 Forget Password

**Initiator:** Standard User, Administrator

**Description:** The user (standard user and administrator) initiates a password recovery process to reset password for their account.

#### Basic Flow of Events:

User clicks on the password reset link → System prompts the user to enter their registered email address → System verifies the email address and if it is present in the database, then sends a password reset link to the user's email → User checks their email, clicks on the password reset link and inputs a new password → System updates the user's password with the new password.

#### Exceptional Flow of Events:

E-1: The user enters an email address that is not registered. The system informs the user that this email is not found.

E-2: The user submits the password reset form that is not fulfilling the requirement of the strong password setting (e.g. Uppercase, lowercase letter and special characters should be involved). The system warns the user to have a strong password.

## 5 Non-functional Requirements

### 5.1 System Requirements

Req. No.	Title	Description
SYS01	Web based	The application should be web-based, it must support major decent browsers, including Firefox and Chrome.
SYS02	Languages	The application must support both Chinese and English languages to meet the needs of different users. Users can conveniently select the interface language in the settings and switch dynamically. The default language should be automatically selected based on the user's browser language settings.

### 5.2 Performance Requirements

Req. No.	Title	Description
PER01	Response Time	The average response time for user interface requests should be less than 2 seconds. For complex queries, the response time should be less than 5 seconds.

PER02	Concurrent Users	The system should support at least 50 concurrent users without impacting performance. During peak times, the system should handle up to 100 concurrent requests.
PER03	Resource Usage	CPU usage should remain below 70% to ensure stable operation under high load. Memory usage should stay below 80% to prevent application crashes or performance degradation.

### 5.3 Security Requirements

Req. No.	Title	Description
SEC01	Authentication	Multi-factor authentication (MFA) should be available for all users to enhance account security.
SEC02	Authorization	Role-based access control (RBAC) must be enforced to ensure users can only access resources relevant to their roles. The system should support fine-grained permissions to control access at the data level.
SEC03	Compliance	The system must comply with relevant security standards and regulations (GDPR, HIPAA) to protect user data and privacy.
SEC04	Logging and Monitoring	Security-related events (e.g., login attempts, data access) should be logged and monitored in real-time for suspicious activities.

### 5.4 Documentation Requirements

Req. No.	Title	Description
DOC01	Requirements Document	A document specifying the functional requirements and non-functional requirements.
DOC02	Design And Implementation Document	A document that outlines the architecture, components, and processes involved in the development of a software application.
DOC03	Testing Document	Testing documentation encompasses a set of documents that outline the testing process, strategy, and results for a software application.
DOC04	User Manual	Document to help users understand and utilize the application features.