# **Messenger application**

# **1) Project Identification and Selection**

\* A messenger application for a local group in an area and goal is to asynchronous communication channel or application between local community.

\* Number of users > 30 users

\*Age group <+14+years

=> Features

* Cross-platform support for Android, iOS, Mac, Windows, and web
* End-to-end encryption (E2EE)
* Groups (channels) for up to 30 users with customizable notifications
* Text, photo, video, and audio messages
* Voice and video call over the internet.
* Photo sharing from device storage or direct using built-in camera app.
* File sharing for PDFs, documents, spreadsheets, slideshows, etc., up to 100MB.
* Support for payment gateways.
* The storage capacity of the server will be 10 TB.

**Functional Requirements**

1. Support one-to-one chats
2. Support offline sending message
3. Support to send messages to other users even the user is offline
4. support group chats
5. WhatsApp group with up to **30 participants**
6. Support video chat
7. Support group video chat
8. Allow voice messages
9. Support image, video, and file-sharing capabilities
10. Support encrypted message
11. Video cannot exceed 16 MB or 90 seconds to 3 minutes
12. Indicate read/receipt of messages
13. Last seen time of users (depends on a few scenarios)
14. Sent + Delivered + Read receipts ticks
15. File size can be shared is limited to 100MB
16. Support to play YouTube, audio, and video on the display screen
17. Supports video formats — MP4, 3GP, MKV, AVI, and MOV
18. Notification will be shown once your file, video, audio, and image exceed the size limitation
19. Support to share the file, video, audio, and image to other applications which includes
20. Support reply to a particular message in a group chat
21. Allow us to forward any messages
22. Allow us to copy the message
23. Allow us to save the message
24. Allow us to archive the message
25. Allow us to reply privately to the particular message
26. Allow us to send and view videos and images one time and delete them once opened

**2)Project Initiation**

\* Feasibility study

\*Assigning project characters

\*Review

\*Project office location identification

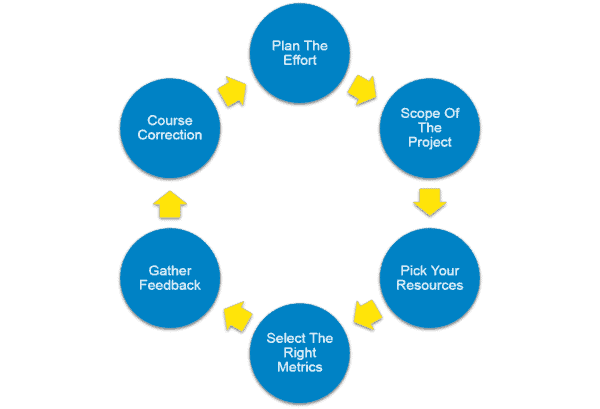
\*Scheduling of project

Here are some more points

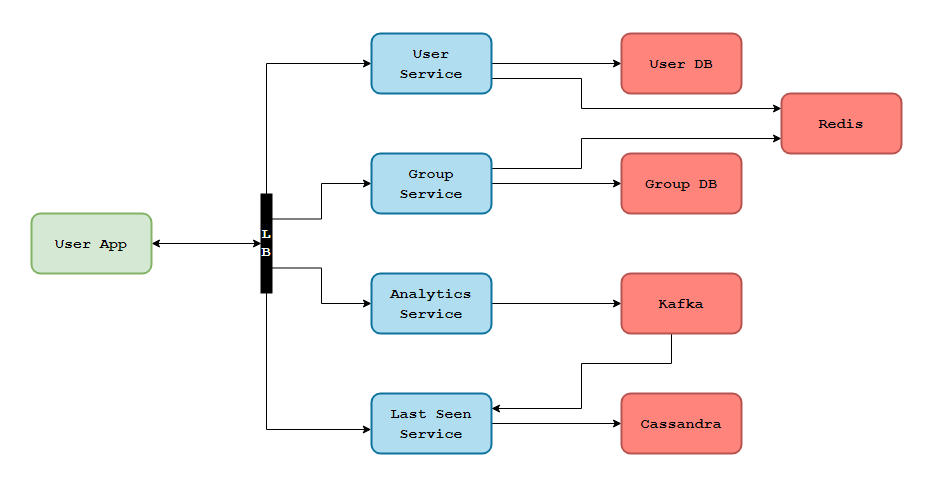
* **Define the Project:** Start by defining the project's objectives and goals. Clearly state the purpose of the project and the things you hope to achieve.
* **Understand the Project's Scope and Boundaries**: Identify the scope and boundaries by determining what is in and out of the scope.
* **Stakeholder Engagement:** Before the planning phase begins, stakeholders should approve the project's objectives, scope, budget, and timeline.
* **Conduct Feasibility Analysis:** This step involves assessing whether the project is feasible within the given constraints, such as time, budget, resources, and technology.
* **Generate a Project Charte**r: A project charter captures key aspects of the project, including the project's objectives, scope, constraints, stakeholders, risks, budget, and timeline.
* **Allocate Resources, Plan, Assess, and Manage Risks**: This step involves assessing potential risks and allocating the necessary resources to manage them.
* **Establish Project Governance**: This means you have to define the roles and responsibilities of the project team and the project's governance structure.
* **Form the Project Team:** Form a team of experts who will work together to deliver the project successfully.
* **Communicate and Obtain Buy-In**: Communication is critical during the project initiation stage. Ensure that stakeholders and team members know what is expected of them.
* **Kick-off the Project and Initial Planning**: Now that the project initiation phase is complete, you can kick off the project and begin initial planning, which involves developing detailed project plans, assigning tasks, and setting timelines

**3)Analysis of project**

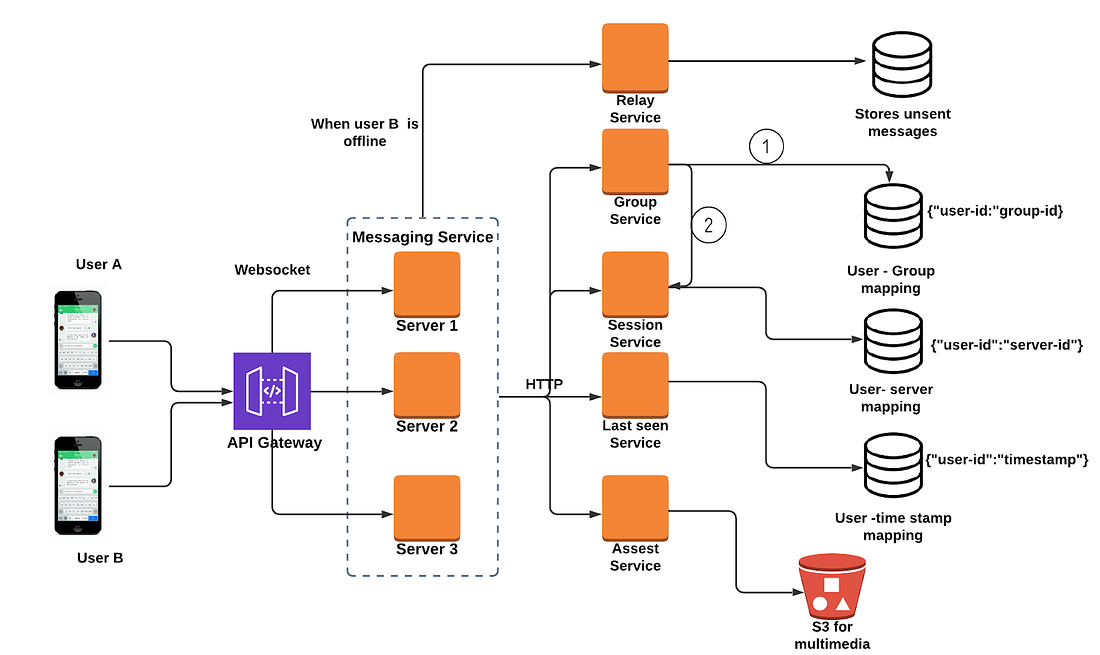
**\* Conceptual Model**



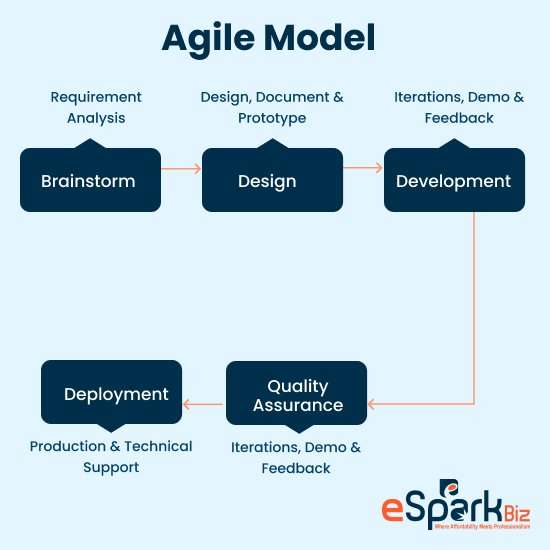
**\*Logical model**



**\*Workflow Chart**



**\*Agile Model**



**4)Logical Design**

**Hardware Interface**

Android phone 256 MB minimum RAM required Internet or LAN connections Processor with speed of 500MHz

**Tools and Technology**

Quality Planning -Software QFD

Product Innovation- Brainstorming, Mind-Map, TRIZ/ARIZ, Innovative algorithms

Software Analysis –Brainstorming, Mind-Map Design, patterns UML tools and technique

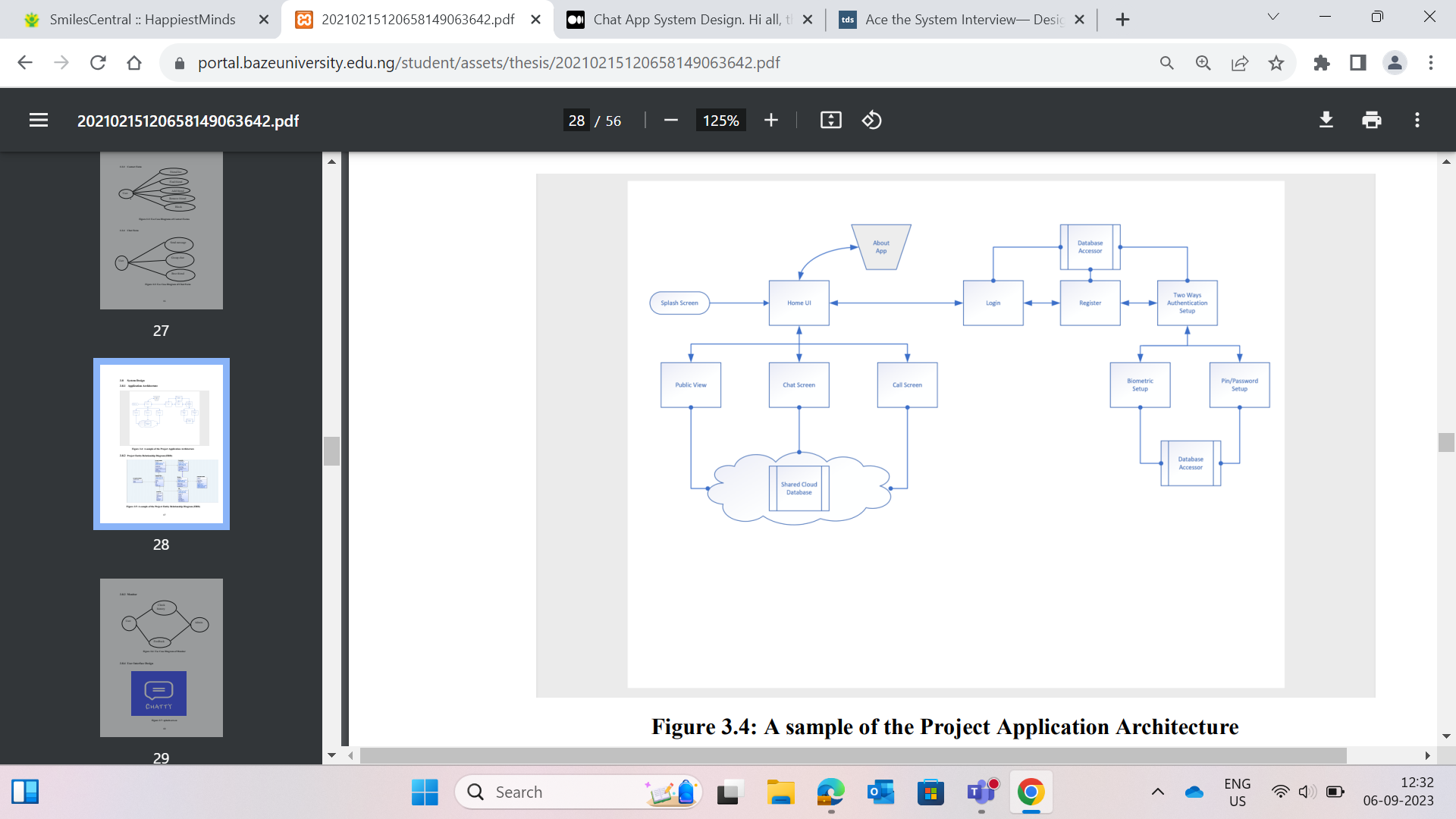
Database modelling tools- Mongo dB Compass, Mongoose, Mongo dB Driver

Software Development Methodology- Agile, scrum

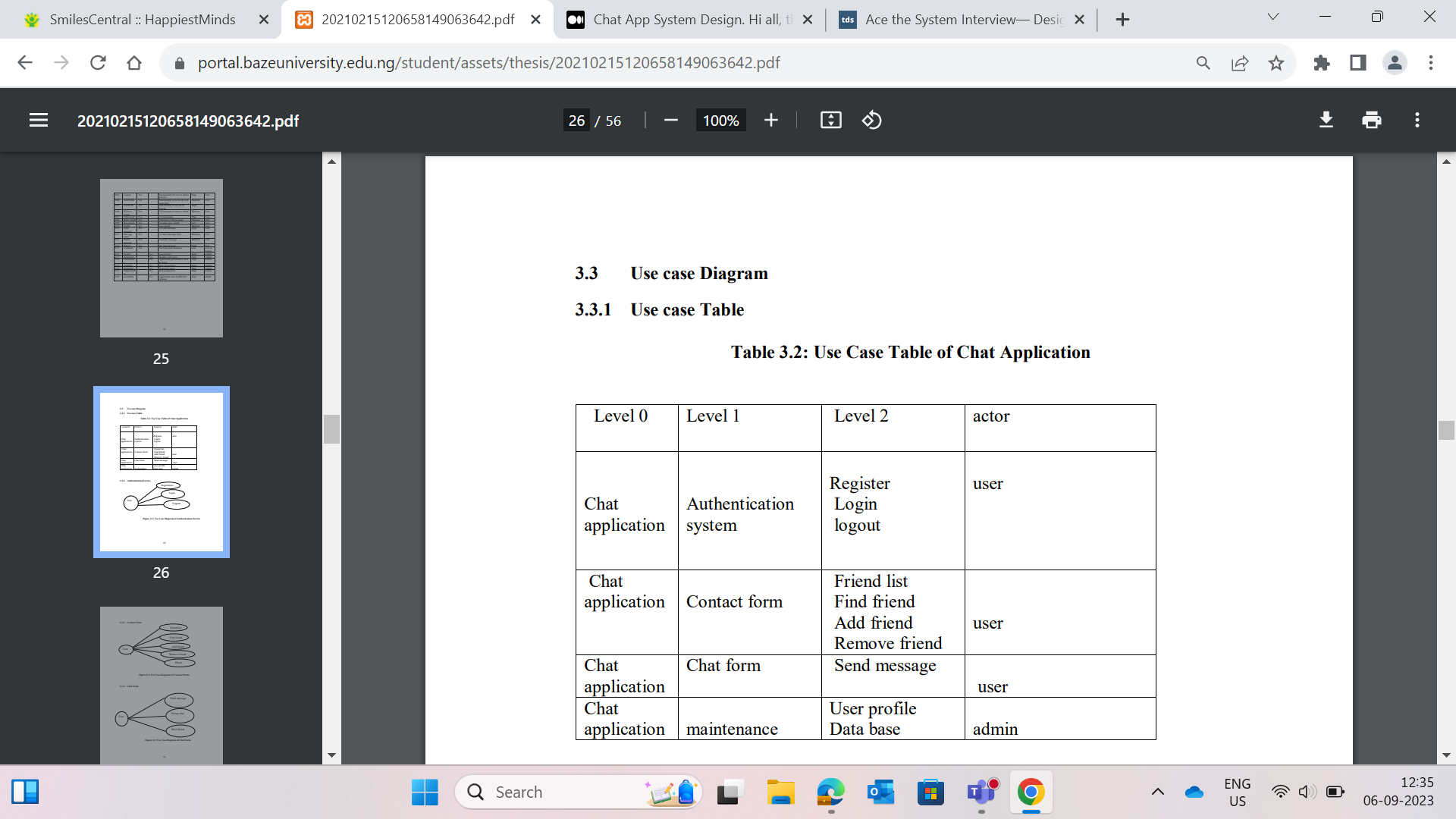
Programming Language -Typescript, JavaScript, Dart, Json, yaml, swift, java

**4)Physical Design**

\*Project architecture

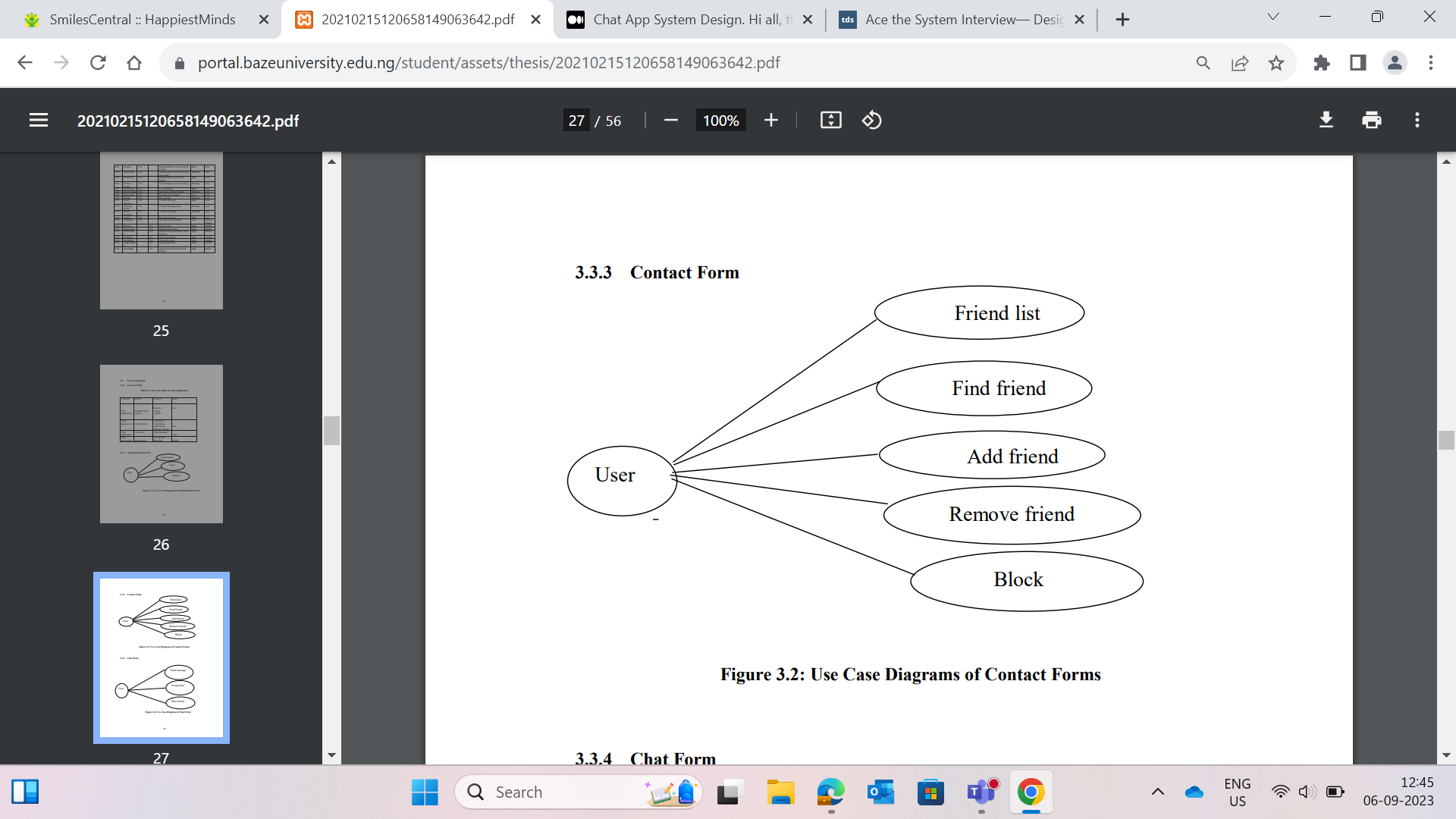


Physical Workflow

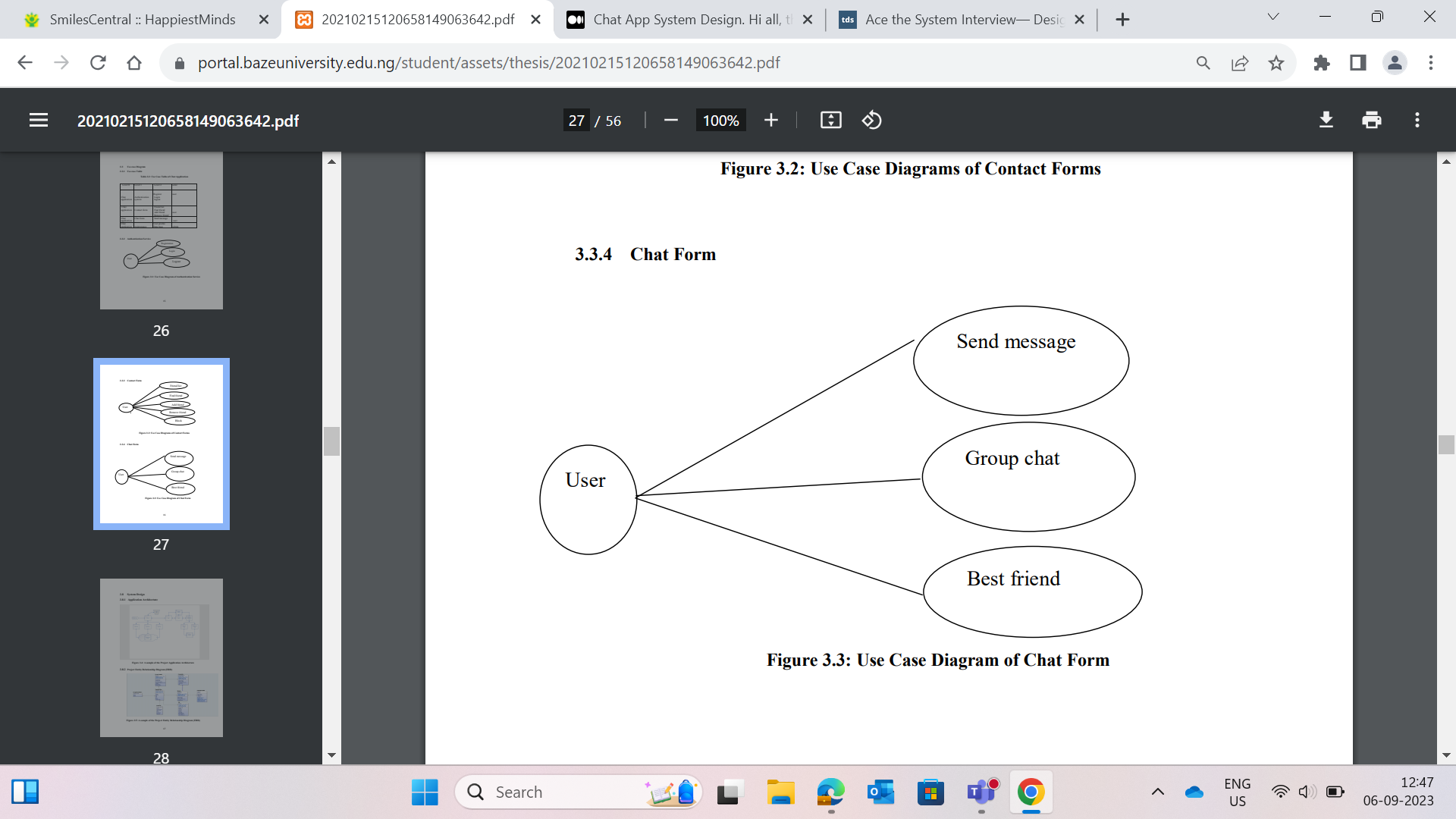


2) **Authentication Service**

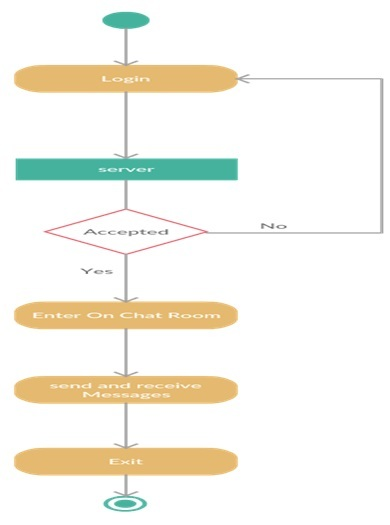
**3)Contact form**



**4)Chat form**



**Activity diagram**



**5) Testing and Maintenance**

Testing

Installation test, Functional test, Load test, Performance profiling, Data integrity test, & automated test.

HIGHER LEVEL ITEMS TO BE TESTED

1. Chat application and supporting infrastructure

2. Application running on different client devices

HIGHER LEVEL ITEMS NOT TO BE TESTED

1 SRS of chat application

2 User Manual of chat application

3 Already existing chat application

4 Manual processes related to the application

5 Any legacy system

LOWER-LEVEL ITEMS TO BE TESTED

1 User Profile

2 Chatting

3 Add Friend

4 Remove Friend

5 Find Friend

6 Registrar

7 Login

8 verifications

9 biometric Login

10 Logout

LOWER-LEVEL ITEMS NOT TO BE TESTED

Use Guide

1 Processes

2 Registrations for new Members

3 Login Features

4 Adding Friend

5 Chat Forms

6 Settings

There is always room for improvement in any application. This project deals extensively with text communication with special feature addition of Pidgin English which makes it user friendly to those at the bottom of the societal pyramid as well, thereby capturing a sizeable numerical group within the West Africa region. With the app’s friendly user interface, it brings technology closer to the general populace and the grass-rooters

### **Maintenance**

The maintenance of the application occurs throughout the Development Life Cycle Stages, but sometimes, you would notice a shorter period of it.

### **Closure, Hand-off, and Support**

In the last phase of the SDLC Example, you still need to put in more effort. You have to gather everything about the application you created and its specifications and pass it to another support team.

