

Software Methodology - Project

Team Size: 3-5 students

Midterm Checkpoint: 30 October, 2024

Final Due Date: 12 December, 2024

Email: miqing@bjut.edu.cn

Description

In this assignment, you are required to design and implement an **Android Diary APP** that allows you to keep a record of your daily thoughts, feelings, and experiences in a structured and private manner.

- ✧ The APP lets users customize the layout of diary entries with different fonts, colors, and sizes.
- ✧ The APP lets users add photos, videos, files, and even locations to diary entries.
- ✧ The APP lets users categorize diary entries.
- ✧ The APP lets users search past entries by keyword, date, or tag using the search bar.
- ✧ The APP lets users use passwords to protect diary entries from prying eyes and intruders.

You are encouraged to add **AI-based features** to the APP, such as Image-to-Text, automatic tagging and summarization.

Deliverables

1. Working Code (5% Grade)

Each team must submit the completed code of your project as well as **an APK file**.

2. Video for Rank (5% Grade)

Each team must submit a 5-minute video (in MP4 format) in which you present and explain your project.

3. Report (10% Grade)

Each team must submit a formal report, which contains all the information that is relevant to your project. The exact format of the report is up to you, but it should be well written and include the following parts.

Cover Page

Preface

- ✧ Define the expected readership of the document and describe its version history, including a rationale for the creation of a new version and a summary of the changes made in each version.

Table of Contents

System Specification

1. Customer Problem Statement
 - ✧ Put yourself into a customer's role. Describe the problem that your customer is facing and his or her suggestions about how a software system could help.
2. Glossary of Terms
 - ✧ List important terms and their definitions to ensure consistency and avoid ambiguity in the system specification.
3. User Requirements Definition
 - ✧ List, prioritize, and describe functional requirements, nonfunctional requirements, and user interface requirements in a table, one row per requirement.
 - ✧ Instead of user requirements, you may wish to write user stories. Write one or the other, but not both.
4. System Requirements Specification
 - ✧ Identify stakeholders who have interest in the system.
 - ✧ Identify actors that will directly interact with the system and their goals.
 - ✧ Draw the use case diagram with all the use cases. For all use cases that you can think of, write a brief or casual text description. Select a few most important use cases and provide detailed description.
 - ✧ Add more details to nonfunctional requirements. Note that your requirements should be testable.
 - ✧ Write the acceptance test cases to show how your requirements will be tested for acceptance by the customer.

System Design

1. Interaction Diagrams
 - ✧ Draw interaction diagrams for a few most important use cases. You should do at least sequence diagrams, but you may do some other UML interaction diagrams as well.

- ✧ Explain and justify the design patterns that you use in the particular interaction diagrams.
- 2. Class Diagram and Interface Specification
 - ✧ Draw the class diagram to show all classes and their associations.
 - ✧ Define the meaning of each class, operation, and attribute in plain language.
- 3. System Architecture and System Design
 - ✧ Describe the architectural styles used in your design.
 - ✧ Draw and describe UML package diagram of subsystems in your system.
 - ✧ Describe the database schema of your system (format of database tables).
 - ✧ Describe the global control flow of your system (execution orderness, time dependency, and concurrency).
 - ✧ Describe the hardware requirements of your system (the minimal and optimal configurations).
- 4. Algorithms and Data Structures (if applicable)
 - ✧ Use activity diagrams to describe the algorithm design.
 - ✧ Describe your data structures, such as arrays, linked lists, hash tables, or trees, and why you use them.
- 5. User Interface Design
 - ✧ Do the GUI design for your system using prototyping tools.
- 6. Design of Tests
 - ✧ List and describe the test cases that will be used for unit testing of your system.
 - ✧ Discuss the test coverage of your tests.
 - ✧ Describe your integration testing strategy and plans on how you will conduct it.

Project Management

- ✧ Describe organizational matters that you used for effective teamwork (e.g., organizing group meetings, managing shared resources and resolving ambiguities).
- ✧ List the projected milestones and dates using Gantt charts.
- ✧ Provide an itemized list of each team member's contribution. **Quantify each team member's contribution as a percentage.**

References

| |
|--|
| ✧ Provide a list of URL's, books, or other sources that you used for your project. |
|--|

IMPORTANT NOTES

1. A zip file containing all deliverables is submitted by the whole team, and points will be split among the team members according to the declared contributions.
2. It is mandatory to use a software tool for producing UML diagrams. Hand-drawn UML diagrams are not acceptable.
3. Do not just show the UML diagrams. For all figures, tables, and diagrams, provide some narrative discussion.
4. The report should have professional appearance. Make sure that it is neat, easy to read and understand, with clearly labeled section headings, figure captions, pagination, and without grammatical and typographical errors.
5. Late submission receives NO mark except with strong justifiable reasons or under extenuating circumstances.