

School of Computer Science and Engineering

CZ3002 Advanced Software Engineering

**Design Report on Software Maintainability**

Project Name: HangOut

****

Group Name: Mac & Cheese

Lab Group: TDDP1

| **Group Members** |
| --- |
| Jethro Phuah An Ping (Team Leader) |
| Lam Zhi Fah (Deputy Team Leader) |
| Alicia Chua Jieying |
| Ernest Ang Cheng Han |
| Ong Jing Heng Shaun |
| Ong Sim Hao |
| Shannon Tan Xinyi |

**Document Change Record**

| **Revision** | **Description of Change** | **Approved by** | **Date** |
| --- | --- | --- | --- |
| 0.10 | Initial Template | Jethro Phuah | 02/10/2021 |
| 0.20 | Add Design Strategies, Architectural Design Patterns | Lam Zhi Fah | 05/10/2021 |
| 0.30 | Software Configuration Management Tools | Lam Zhi Fah | 07/10/2021 |
| 1.00 | Revised Edition | Jethro Phuah | 10/10/2021 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Contents**

1. [Design Strategies](#_heading=h.gjdgxs) 3
   1. [Planning Phase Before Development](#_heading=h.30j0zll) 3
   2. Development Process3
   3. [Correction by Nature](#_heading=h.3znysh7) 3
   4. [Enhancement by Nature](#_heading=h.2et92p0) 4
2. [Architectural Design Patterns](#_heading=h.tyjcwt) 4
3. [Software Configuration Management Tools](#_heading=h.3dy6vkm) 5

| 3.1. | MediaWiki | 5 |
| --- | --- | --- |
| 3.2. | GitHub | 6 |
| 3.3  3.4. | Google Drive  TortoiseSVN | 6  6 |

# Design Strategies

## Planning Phase Before Development

During team meetings, the group will discuss the important features that should be included in the mobile application to enhance user experience. We would also perform analysis and predict the different kinds of improvements we would be implementing in the future after the release of the application. As such, we will be focusing on the Usability aspect of Hangout for the members of the public as we foresee a broad user base. The software must flow according to the user’s expectations, offer required information and provide navigation controls that are clear to understand.

When the widespread usage of the application becomes the norm, it will have to be scaled up to handle much higher traffic rates as the user base continuously expands. Hence, we also targeted Scalability as one of the factors that we have to look into in the future.

Since HangOut is heavily dependent on the user experience, we decided to adapt the 3-tier architectural model as our architectural mode. This system infrastructure framework will be beneficial for Graphic User Interface design. It promotes modularity since each tier can run on its own infrastructure. This means that each tier can be easily modified, updated or scaled without impacting the other functions.

## Development Process

We are testing out in a small, test driven development for initial runs. Due to safe distancing measures and contact minimization in place in response to COVID-19, we were unable to organise focus groups and gatherings with our target user base to test our product. However, we can conduct online zoom sessions to provide instructions to participants on the testing procedures and receive feedback from them after using the application. We can then collate their responses and carry out analysis to improve the application. Furthermore, team members, non-developers and developers alike, can also perform the role of testers and provide continuous feedback on the design and usability of the application.

## Correction by Nature

For official testing of Hangout, we will be implementing the following procedures and this is what we will look out for:

* + 1. **Corrective Maintainability**

Fault detection done through testing, followed by reactive modification of the software product to correct any discovered problems.

* + 1. **Preventive Maintainability**

Features implemented in atomic manner, and each feature is tested independently. This allows for errors to be detected easily and respective modifications can be carried out before they become effective faults.

## Enhancement by Nature

We will enhance our application while testing the application and this is what we will look out for:

* + 1. **Adaptive Maintainability**

Software can be easily adapted to a new operational environment without much issues.

* + 1. **Perfective Maintainability**

After the application is up and running, we are able to quickly detect any errors and correct them when required, which helps reduce maintenance costs and time needed.

* 1. **Maintainability Practices**

To uphold quality in both process and product, we have implemented the following maintainability practices over the course of our project:

* Readable Code
* Version Control
* Standardised Documentation
* Modularity

# Architectural Design Patterns

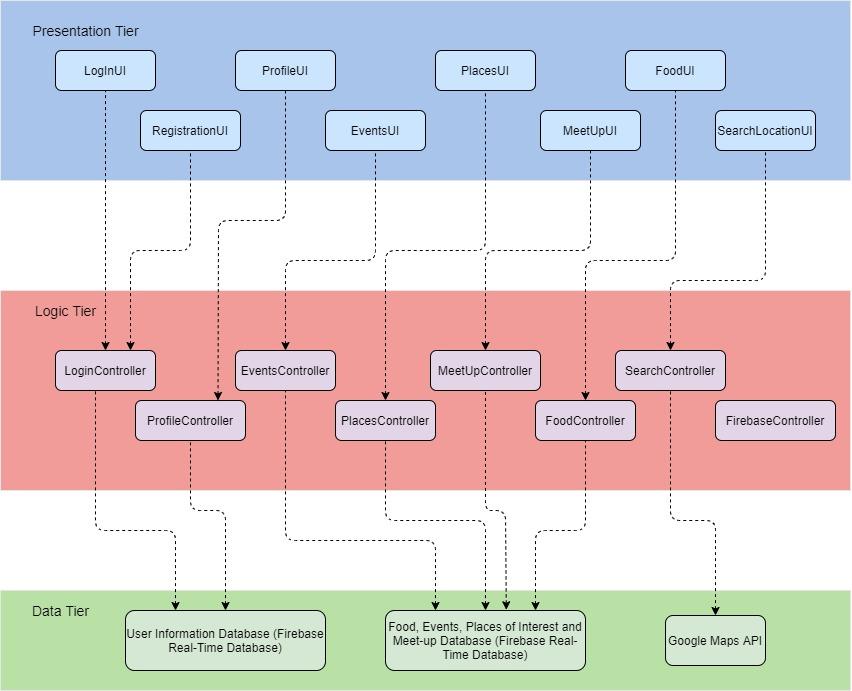
Hangout is using the 3-tier Architecture design pattern.

The presentation tier is where it holds all the user interface programming in the app. Its main function is to translate tasks and results to something that the user can understand.

The Logic tier coordinates the application, processes commands, makes logical decisions and evaluations, and performs calculations. It also moves data between the two surrounding layers.

Lastly, the data tier stores all the user’s and foods/events/places of interest/meet-up database in Firebase. These information is then passed back to the logic tier for processing and eventually back to the user.

Following is the design:



# Software Configuration Management Tools

Having Software Configuration Management Tools (SCMT) allows us to do the following:

* Access control.
* Traceability between software and test ware and different versions of the variants.
* Build and release management

## MediaWiki

MediaWiki is a free and open-source application. This service is used as it is easy for beginners to pick up. There are many FAQs provided which can teach users the functions required by the users. There is a wide range of functions which allows users to create their information in different styles. It also allows users to concurrently edit the page at the same time. Hence, editing of the page will not result in a loss of information.

## GitHub

GitHub is a source code hosting platform using the distributed version control and source code management Git. GitHub is chosen for its familiarity and support provided by various IDE applications. GitHub also supports issue tracking similar to a ticketing system. Whether it’s a software bug, code enhancement or documentation, users can open an issue, label them appropriately and assign them for other team members to resolve. All users involved will receive timely updates on the progress of the issue. This allows us to push and pull the relevant code from the shared repository, and in the case there are any errors, we can easily track the changes and revert the necessary commits.

## Google Drive

Google Drive is used as a file storage system to back up any documents initially created. This service allows us to share and store files within the group easily, and also allows members to edit the documents concurrently after delegation of work, which can help speed up the completion of the documents. Version control is also available in this service.

## TortoiseSVN

SVN is used for revision and version control, which enables us to track different versions of the source code files and documents being uploaded and edited. It allowed users to move, remove, rename and copy files while preserving the full revision history. TortoiseSVN can also be used to integrate windows explorer and hence, allowing us to access the files easily for editing.