City University of Hong Kong

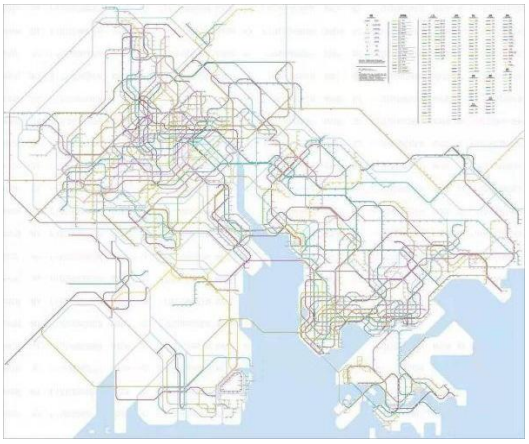
Department of Computer Science

CS3343 Software Engineering Practice

2021/22 Semester A

Hong Kong-Shenzhen Metro Route Planning System in Greater Bay Area

Project Plan



Project Group 6

HUANG Yuqin (56195499)

LIU Wei (56198489)

WANG Zhixuan (56198557, APM)

XU Jiakai (56197616)

XU Rui (56200188, PM)

ZHANG Xun (56198705)

TABLE OF CONTENTS

1 SUMMARY OF THE PROJECT **1**
1.1 Project Description 2
1.2 Stakeholders 2
1.3 Objectives of the Project 2
2 SUMMARY OF METHODOLOGY **2**
2.1 Software Development Methodology 2
2.2 Project Team Organization 3
2.3 Development Tools 3
2.3.1 Development IDE 3
2.3.2 Development Platform 4
2.3.3 Test Cases 4
2.3.4 Coverage Checking 4
2.3.5 Testing Platform 4
2.3.6 Documentation 4
2.3.7 Project Management 4
2.4 Configuration Management 4
2.4.1 Version Control 4
2.4.2 Reporting 5
3 WORK BREAKDOWN STRUCTURE **5**
4 PROJECT SCHEDULE **6**

1 SUMMARY OF THE PROJECT

1.1 Project Description

In the past, the metro systems in Hong Kong and Shenzhen were operated in a relatively independent way. With the promotion of the *Outline Development Plan for the Guangdong-Hong Kong-Macao Greater Bay Area*, the regional integration construction has become the trend of the future. Inspired by the development of the Greater Bay Area, the Hong Kong-Shenzhen Metro Route Planning System is proposed. It is expected to strengthen the connection of the metro network and the economic exchanges between Shenzhen and Hong Kong.

1.2 Stakeholders

Stakeholders	**Description**
Project Team Members	● Develop the system ● Improve the system
Metro Corporation	● Provide the construction information (e.g. length of the line, time for travel, etc.)
Maintenance Personnel	● Maintain the system ● Update the information (e.g. when there are new lines constructed, add the lines and stations)
Users	● Provide the starting point and the destination point ● Obtain the planned route and relevant information

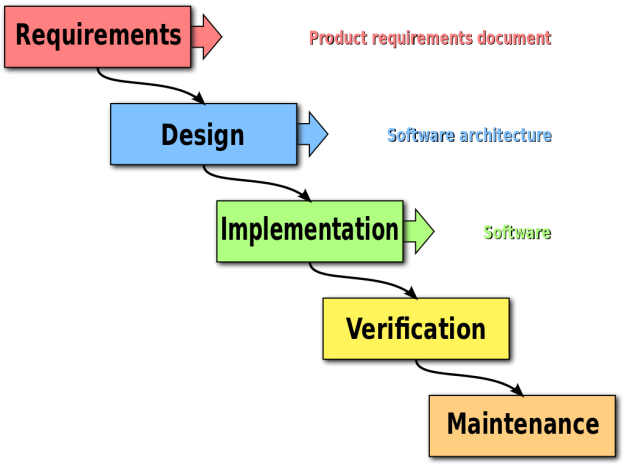
1.3 Objectives of the Project

The aim of the project is to make it more convenient for the civilians of Hong Kong and Shenzhen to plan the route across the two cities to better arrange their travel plan. It can better improve the commuting efficiency, integrate the economic and trade exchange and cultural communications and consequently promote the development of the two cities and the Greater Bay Area.

2 SUMMARY OF METHODOLOGY

2.1 Software Development Methodology

Waterfall development model is adopted in our project. It is a traditional and commonly used sequential development approach.



In our project, we will follow the development steps in the above picture. First, we collect the information to analyze the requirements. Then, design the outline of the system according to the information. After that, we will build the main body of the code to realize the required functions. We will execute testing on the program to ensure the code runs without bugs. Finally, maintenance is needed to keep the system working smoothly, in the meantime keeping the timeliness of information.

2.2 Project Team Organization

Name	Position	Work Description
HUANG Yuqin	Program Developer, Program Tester	Algorithm Core & Testing
LIU Wei	Program Developer, Program Tester	UI Data Processing & Drawing
WANG Zhixuan	Assistant Program Manager	UI Data Processing & Drawing
XU Jiakai	Program Designer, Program Developer	Algorithm Design Principle & Pattern
XU Rui	Program Manager	Data Integration & I/O Module
ZHANG Xun	Program Developer, Interface Designer	UI Design & Programming

One thing that is worthy to note is that we arrange our group into two teams to do parallel development: the algorithm team (Member: XU Rui, XU Jiakai, HUANG Yuqin) is responsible for the structure and functions of the algorithm, and the UI team (Member: ZHANG Xun, LIU Wei, WANG Zhixuan) is in charge of UI-aspect work.

2.3 Development Tools

2.3.1 Development IDE

- Software:
  - IntelliJ IDEA 2021.2
    https://www.jetbrains.com/idea/

    - Eclipse IDE for Java Developers - 2021-09
      https://www.eclipse.org/downloads/
- External libraries: org.apache.poi (Apache POI - the Java API for Microsoft Documents)
  https://poi.apache.org/

2.3.2 Development Platform

- Windows 10 with Java Development Kit (JDK) v16.0.2
- macOS Monterey 12.0.1 (x86_64)

2.3.3 Test Cases

- JUnit v5.7.0

2.3.4 Coverage Checking

- IntelliJ IDEA built-in coverage runner

2.3.5 Testing Platform

- Windows 10
- macOS Monterey 12.0.1 (x86_64)

2.3.6 Documentation

- Visual Paradigm 16.2 - Use Case Diagram, Class Diagram and Sequence Diagram
- Bugzilla - Bug Report
- Microsoft Word - Report
- Microsoft Powerpoint - Presentation

2.3.7 Project Management

- TeamGantt - Project Scheduling
- GitHub - Project Collaboration
- Fork - Git Client for Project File Management
- Google Drive - Project Documentation

2.4 Configuration Management

To strengthen the collaboration, we use some measures to keep the software and documentation in a consistent state.

2.4.1 Version Control

Github is a good platform for project file sharing and progress synchronization. Project members can fetch the progress of the team in time and make modifications. When pushing the file back, all the modifications will be recorded. If there are conflicts, members can handle it by judging the difference between the conflicted files to ensure the correctness of the overall project. Our members will work under respective branches, and make pull requests to merge with the main branch so as to enable independent development without interrupting the main branch, thus improving the efficiency.

We also exploit Fork to better arrange the local files and Github files. With it we can intuitively see the difference between the newest version (files to be committed) and the older version. It also allows rollback of the version in case that some pull requests are handled by mistake.

2.4.2 Reporting

We created a shared folder on Google Drive to edit the documents in a parallel way. Members can also check the content in real-time. It will improve the speed and accuracy of writing.

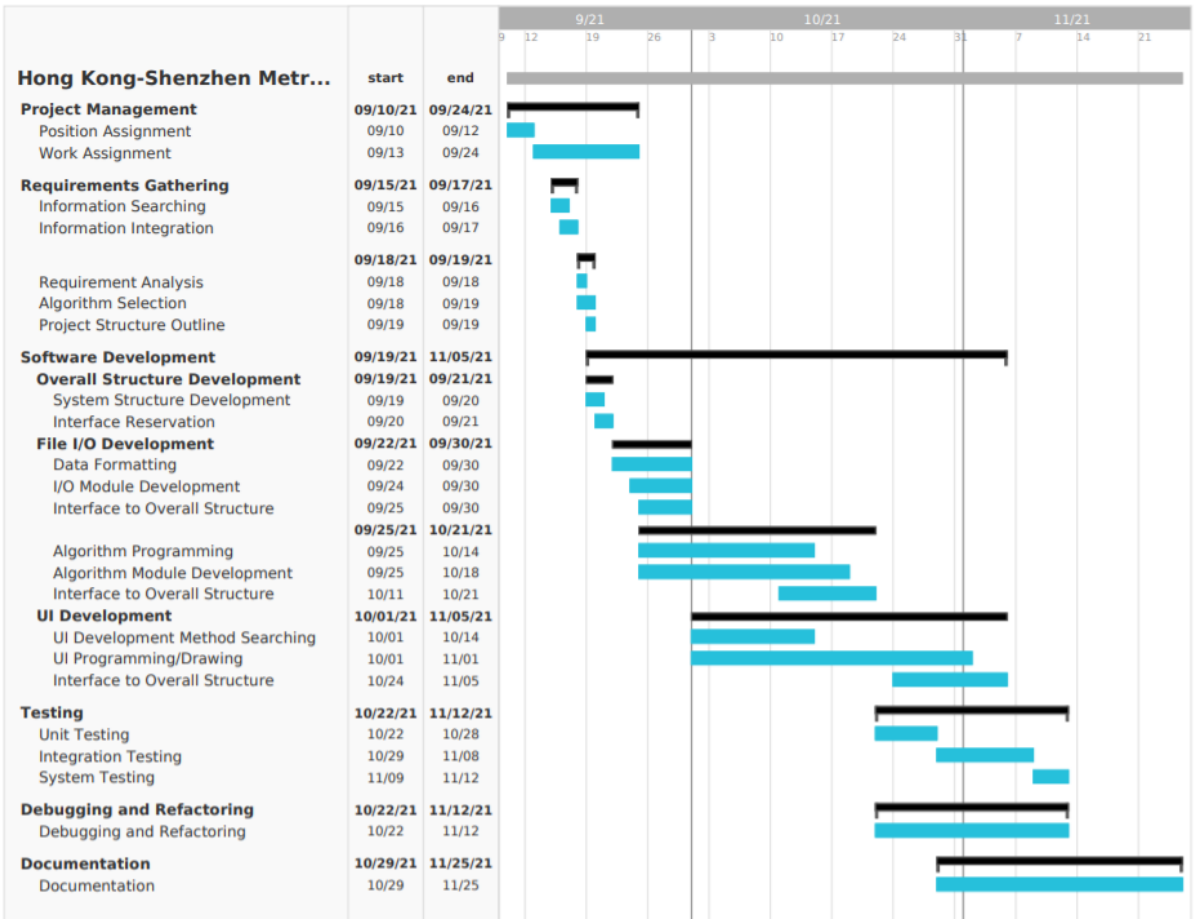
3 WORK BREAKDOWN STRUCTURE

We adopt process-type WBS to generalize and define the overall scope of work of the project. Because the graphical tree (organizational chart) is not easy to expand in the document, we use the outline (indented format) to represent our WBS. We use the top-down technique to generate the WBS. That is, outline the overall phase first, then break the phases down into smaller details.

0.0 Hong Kong-Shenzhen Metro Route Planning System
1.0 Project Management
    1.1 Position Assignment
    1.2 Work Assignment
2.0 Requirements Gathering
    2.1 Information Searching
    2.2 Information Integration
3.0 Analysis and Design
    3.1 Requirement Analysis
    3.2 Algorithm Selection
    3.3 Project Structure Outline
4.0 Software Development
    4.1 Overall Structure Development
        4.1.1 System Structure Development
        4.1.2 Interface Reservation
    4.2 File I/O Development
        4.2.1 Data Formatting
        4.2.2 I/O Module Development
        4.2.3 Interface to Overall Structure
    4.3 Algorithm Development
        4.3.1 Algorithm Programming
        4.3.2 Algorithm Module Development
        4.3.3 Interface to Overall Structure
    4.4 UI Development
        4.4.1 UI Development Method Searching
        4.4.2 UI Programming/Drawing
        4.4.3 Interface to Overall Structure
5.0 Testing
    5.1 Unit Testing
    5.2 Integration Testing
    5.3 System Testing

6.0 Debugging and Refactoring
7.0 Documentation

4 PROJECT SCHEDULE



Hong Kong-Shenzhen Metr...	start	end
Project Management	**09/10/21**	**09/24/21**
Position Assignment	09/10	09/12
Work Assignment	09/13	09/24
Requirements Gathering	**09/15/21**	**09/17/21**
Information Searching	09/15	09/16
Information Integration	09/16	09/17
	09/18/21	**09/19/21**
Requirement Analysis	09/18	09/18
Algorithm Selection	09/18	09/19
Project Structure Outline	09/19	09/19
Software Development	**09/19/21**	**11/05/21**
Overall Structure Development	**09/19/21**	**09/21/21**
System Structure Development	09/19	09/20
Interface Reservation	09/20	09/21
File I/O Development	**09/22/21**	**09/30/21**
Data Formatting	09/22	09/30
I/O Module Development	09/24	09/30
Interface to Overall Structure	09/25	09/30
	09/25/21	**10/21/21**
Algorithm Programming	09/25	10/14
Algorithm Module Development	09/25	10/18
Interface to Overall Structure	10/11	10/21
UI Development	**10/01/21**	**11/05/21**
UI Development Method Searching	10/01	10/14
UI Programming/Drawing	10/01	11/01
Interface to Overall Structure	10/24	11/05
Testing	**10/22/21**	**11/12/21**
Unit Testing	10/22	10/28
Integration Testing	10/29	11/08
System Testing	11/09	11/12
Debugging and Refactoring	**10/22/21**	**11/12/21**
Debugging and Refactoring	10/22	11/12
Documentation	**10/29/21**	**11/25/21**
Documentation	10/29	11/25