

< Journey Assistant > Software Testing Summary Report Version 1.0

Group Member

Yiwen Song

Zhihui Xie

Weizhe Wang

Huangfei Jiang

Haoping Chen

Modification History

Date	Version	Description	Author
2019-06-18	1.0	Finish the first version.	Yiwen Song

Contents

1	Intruduction	3
1.1	Purpose	3
1.2	Scope	3
1.3	Definition	3
1.4	Bibliography	3
1.5	Sketch	3
2	Testing Summary	3
2.1	Testing Requirement & Testing Use Case	3
2.2	Testing Environment	3
2.3	Testing Tool	3
3	Testing Result	4
3.1	Testing Progress	4
3.2	Testing Personnel	4
4	Conclusion Analysis	4
4.1	Result of Testing Use Case	4
4.2	Fix of Testing Failure	4
4.3	Testing Result Analysis	4
4.3.1	Covering Analysis	4
4.3.2	Hole Analysis	5
5	General Comment	5
5.1	Software Ability	5
5.2	Limitation	6
5.3	Proposal	6

1 Intruduction

1.1 Purpose

The software testing summary aims to provide a record and summary of the testing process of our Journey Assistant software. The document will show the conclusion of the tests made on Journey Assistant, as well as the analysis. The document helps the developers to conclude the testing process and make the development better.

1.2 Scope

Software the document is applied on: Journey Assistant.

Characteristics, subsystems, models and codes related to the software all fit the contents of this the document.

1.3 Definition

The terms referred to in this document are defined in the project glossary document (Glossary.pdf).

1.4 Bibliography

1. <Object Oriented Software Engineering (Version 3)> (Tsinghua University Press)
2. <Object Oriented Software Engineering Practice Guidelines>

1.5 Sketch

The document consists of 4 parts, i.e., the testing summary, the testing result, the conclusion and the general comment. Testing summary describes the testing requirements and testing methods. Testing result records the testing progress and the actual work of the testing personnel. The conclusion analyzes the condition of testing and shows the solution of the problem. The general comment gives a general comment according to the testing process. Each part of the document links with one another, and complements each other to show an overview of our testing procedure.

2 Testing Summary

2.1 Testing Requirement & Testing Use Case

Please check *Software Testing Plan* document for detailed information.

2.2 Testing Environment

Please check *Software Testing Plan* document for detailed information.

2.3 Testing Tool

We mainly use manual ways to generate the testing data. Some automatic tests can be done by Python.

3 Testing Result

3.1 Testing Progress

Testing progress is shown in Table 1.

Milestone	Start	End	Delay
Planning	2019.6.14	2019.6.15	/
Designing	2019.6.15	2019.6.16	/
Preparation	2019.6.16	2019.6.16	/
Execution	2019.6.16	2019.6.18	/
Evaluation	2019.6.18	2019.6.20	/

Table 1: Testing Progress

3.2 Testing Personnel

Testing personnel is shown in Table 2.

Role	Personnel	Job
Test Manager	Song Yiwen	Manage the whole process of testing. Adjust the testing process according to real situations.
Test Planner	Xie Zhihui	Plan the details of testing process according to technical details and user requirements.
Test Performer	All others	Perform testing as planned. Report the test result and evaluate the possible faults that leads to failure.

Table 2: Testing Personnel

4 Conclusion Analysis

4.1 Result of Testing Use Case

Result of testing use case is shown in Table 4.1

4.2 Fix of Testing Failure

Fix of testing failure is shown in Table 4.

4.3 Testing Result Analysis

4.3.1 Covering Analysis

Testing Covering Analysis Testing covering analysis is shown in Table 5.

Requirement Covering Analysis Requirement covering analysis is shown in Table 6.

Requirement Index	Use Case Index	Result	Remark
TR-01	TR-01-01-01	Pass	None'
	TR-01-01-02	Pass	None'
	TR-01-01-03	Pass	None'
	TR-01-02-01	Pass	None'
TR-02	TR-02-01-01	Pass	None'
	TR-02-01-02	Pass	None'
TR-03	TR-03-01-01	Pass	None'
TR-04	TR-04-01-01	Pass	None'
	TR-04-01-02	Fail	None'
TR-05	TR-05-01-01	Pass	None'
	TR-05-01-02	Pass	None'
	TR-05-02-01	Pass	None'
	TR-05-02-02	Pass	None'
	TR-05-02-03	Pass	None'

Table 3: Result of Testing Use Case

Test Requirement Index	Test Use case Index	Error	Error Status
TR-04	TR-04-01-02	Some sights can't be selected when a hotel is selected before.	Fixed

Table 4: Fix of Testing Failure

4.3.2 Hole Analysis

Hole analysis is shown in Table 7.

5 General Comment

5.1 Software Ability

The software can basically satisfy the functional needs designed, and each use cases have basically been realized. The software can design journey itineraries for users given their requirements, as well as let the users customize the journey itineraries. For the current application, our software has achieved the condition for delivery.

Requirement Index	Use case Number	Execution Number	Not Executed	Reason for none-execution
TR-01	4	5	0	None
TR-02	2	2	0	None
TR-03	1	49,152	0	None
TR-04	2	2	0	None
TR-05	5	5	0	None

Table 5: Testing Covering Analysis

Requirement Index	Pass? [Y][P][N][N/A]
TR-01	Y
TR-02	Y
TR-03	Y
TR-04	N
TR-05	Y

Table 6: Requirement Covering Analysis

Requirement\Level	A: Seriously influencethe system operation	B: Normal functional errors.	C: Don't affect operation but must be fixed	D: reasonable proposals	Total
TR-01	0	0	0	2	2
TR-02	0	0	0	1	1
TR-03	0	0	0	0	0
TR-04	1	0	0	2	3
TR-05	0	0	0	1	1

Table 7: Hole Analysis

5.2 Limitation

- The software has not been run on a large number of devices. Therefore, the capability of our software still needs to be confirmed.
- Some of our system's user interface is still not beautiful enough.
- As we have not raised a pressure test, the stability of our system can still not be confirmed. When facing a large number of concurrent queries, the system can still be unstable.

5.3 Proposal

- Release Beta test on our system and check the stability of our system running on different environments.
- Further reconstruct the application and improve our user interface.
- Buy more professional servers to improve the capacity of concurrent queries. Also, we may hold pressure test to check how many queries can be processed at the same time.