

UniMingle: boosting student diversity

Testing Plan

Team 15

1. Introduction

Software testing is a necessary procedure of software development since it provides help verify the correctness, completeness and quality. Furthermore, both verification and validation are two benefits that software testing brings to developers so that can evaluate if their work match the draft plan as well as the expected output of stakeholders. In the project of UniMingle, software testing will be divided into two sections to comprehensive assess the website which is targeted output in this semester, software inspection and software testing. On one hand, software inspection will be conducted with development of UniMingle to assess functional characteristics such as login and register functions. On the other hand, software testing is another crucial element to examine the result of UniMingle.

In the following parts of this plan, the test scope which involves test methodologies and the test schedule which includes both execution time and specific processes will be mentioned to help the stakeholders understand testing processes of UniMingle clearly. Certainly, the current test status is the last but not the least section in this document.

2. Test Scope

In order to present a clear organization of testing processes, there are five major parts of test scopes. Each part involves targeted tests to examines whether each function achieves the desired aims. During testing, developers will report bugs through “issue” function of GitHub. After that, developers will conduct bugfix test to notarize that bugs have been solved. In the end, UniMingle developers will use the smoke test as a regression test in each test stage.

The first part is software inspection which is a stage for developers to inspect if the necessary functions have been achieved in UniMingle. In this respect, we had spent a whole week to conduct software inspection to ensure that the current processes meet the requirement of stakeholders and the goals of the initial plan. User module and event module which are two core parts of UniMingle had been inspected in the early stage of project test in week 8. Except for the functional modules, the design of UniMingle is another vital element during software inspection because it is the most intuitive part for users. After these software inspections, developers could reconfirm that there were nothing deviating from the plan.

After conducting software inspection, dynamic tests should stand on the stage to help developers discover any bugs in UniMingle. Various tests will be involved and proceeded in this period. Since the codes of UniMingle are composed of JavaScript and PHP, unit tests will cover both programming languages. Normally, unit tests will be taken place when developers are working on the codes to examine that the corresponding modules operate well. Also, a smoke test which acts as a regression test will be implemented before developers continue working on adjusted codes to make sure they work in this part.

Once developers confirm that all units are working well in UniMingle and every function had been developed, a functional test will be implemented to evaluate the output of UniMingle project. Differ from unit test, functional test is a “black box” test which means that functions

of UniMingle are examined with importing inputs and comparing received output to expected ones. Functional tests are divided into front-end and back-end part in UniMingle. In the aspect of front-end, user module and event module are expected to pass tests of all functions such as create new accounts, events map display and so on. For the back-end side, functional test will focus on the content management system. UniMingle's databased are managed and maintained by this system so that it is the most important part of the whole project.

There are two tests that UniMingle need to take after completing unit test and functional test. To guarantee and improve user experience, loading speed and jump speed of a website is conclusive. Hence, a performance test which pays attention to the speed of UniMingle is included in the test stage. Alternatively, although UniMingle had passed the speed test which indicates that users with satisfied with their experiences, there are other issues affecting user experiences and the evaluation of UniMingle. For instance, users will visit UniMingle in diverse environments with various instruments so that a compatibility test is essential. During a compatibility test, developers will use three mainstream web browsers (Google Chrome, Safari and Internet Explorer) to simulate user behaviors on UniMingle. Furthermore, since operation systems will influence UniMingle's performance, mobile systems like Android and iOS and desktop system like Mac and Windows will be used to evaluate UniMingle. What's more, as users will have diverse screen size of their devices, three major resolution ratios will act as a variable as well.

3. Test Schedule

As for the development of UniMingle, there are about four weeks left to complete the development, including various test before it is deployed to servers and released to the public. In order to ensure UniMingle can be delivered to stakeholders on time, a test schedule was created by developers after internal discussions.

Testing Schedule

PROJECT TITLE		UniMingle: boosting student diversity				TEAM		Team 15																			
TEAM MEMBER		Xiuquan Wang, Junhong Zhu, Xuankai Feng				DATE		13/05/19																			
WBS NUMBER	TASK TITLE	START DATE	DUE DATE	DURATION	PCT OF TASK COMPLETE	PHASE ONE					PHASE TWO					PHASE THREE				PHASE FOUR							
						WEEK 8					WEEK 9					WEEK 10				WEEK 11				WEEK 12			
						M	T	W	R	F	M	T	W	R	F	M	T	W	R	F	M	T	W	R	F		
1	Test Planing Stage																										
1.1	Software Inspection	06/05/19	10/05/19	4	100%																						
1.2	Software Test Methods Analysis	06/05/19	10/05/19	4	100%																						
2	Unit Test (JavaScript & PHP)																										
2.1	Test Case Design	13/05/19	16/05/19	3	100%																						
2.2	JavaScript Unit Test	15/05/19	20/05/19	5	80%																						
2.3	PHP Unit Test	15/05/19	20/05/19	5	48%																						
2.4	Bug Report & Test Result Analysis	15/05/19	21/05/19	6	33%																						
2.5	Unit Bug fix	15/05/19	21/05/19	6	49%																						
2.6	Regression Test (Smoke Test)	20/05/19	21/05/19	1	0%																						
3	Functional Test (Front-end & Back-end)																										
3.1	User Module Test Case Deisgn	20/05/19	22/05/19	2	0%																						
3.2	Event Module Test Case Deisgn	20/05/19	22/05/19	2	0%																						
3.3	Front-end Test	23/05/19	26/05/19	3	0%																						
3.4	Front-end Bug Fix	20/05/19	26/05/19	6	0%																						
3.5	Back-end Test (API/CMS) Case Design	22/05/19	23/05/19	1	0%																						
3.6	Back-end Test (API/CMS)	22/05/19	24/05/19	2	0%																						
3.7	Back-end Bug Fix	22/05/19	26/05/19	4	0%																						
3.8	Regression Test (Smoke Test)	20/05/19	26/05/19	6	0%																						
3.9	Bug Report & Test Result Analysis	21/05/19	25/05/19	4	0%																						
4	Performance Test & Compatibility Test																										
4.1	Performance Test Case Design	27/05/19	28/05/19	1	0%																						
4.2	Performance Test	28/05/19	30/05/19	2	0%																						
4.3	Compatibility Test Case Design	27/05/19	28/05/19	1	0%																						
4.4	Compatibility Test	28/05/19	30/05/19	2	0%																						
4.5	Regression Test (Smoke Test)	28/05/19	30/05/19	2	0%																						
4.6	Bug Report & Test Result Analysis	30/05/19	31/05/19	0	0%																						
5	Test Conclusion Stage																										
4.1	Test Report Organize & Release	03/06/19	07/06/19	4	0%																						
4.2	Tests Completed	05/06/19	05/06/19	0	0%																						

Figure 1. UniMingle Testing Schedule

There are four phases of UniMingle's test. In the week 8, software inspection and previous research of test methods which are the content of the first phase have been finished. As for the second phase, it is the most significant part of UniMingle's test because it contains two main tests, the unit test and the functional test. Both tests are expected to finish executions by the end of week 10. Test results will be recorded and analyzed then contribute to the final test report. The last phase is about performance test and compatibility test, similarly, all outputs in these tests will be taken down to finish the test report. During the test period, Xiuquan will mainly responsible for back-end's tests, Junhong will take the responsibility of testing front-end, and Xuankai will primarily start from testing user module. All tests will be finished by the end of week 11, and UniMingle is expected to release at week 12. Also, a test report will be finished and released.

4. Current Testing Status

So far, test has been started for more than a week. All team members of UniMingle had reached a consensus on the testing methods as well as testing schedule. Software inspection had been completed so that the emphasis of UniMingle is software testing. As shown in the testing schedule above, test cases for unit test had completed designing and ready for using. In the meanwhile, a small part of unit test in the aspect of JavaScript and PHP are operating. Developers has been submitted some bugs via "issue" in UniMingle's official GitHub and some of them had been fixed.

The whole testing plan of UniMingle is running well with the help of collaboration. In the following weeks, every team member will keep working on various testing according to the testing schedule, to ensure deliver UniMingle in schedule.