Test Plan and Cases (TPC)

Team 04

Name	First Role	Second Role	Email Address
Chengyu Shen	Project Manager	Operational concept developer	**shenchen@usc.edu**
Shiji Zhou	Prototyper	Software Architect	**shijizho@usc.edu**
Yufei Hong	Feasibility Analyst	Project manager	**yufeihon@usc.edu**
Guanghe Cao	Software Architect	Life Cycle Planner	**caog@usc.edu**
Yang Wei	Operational concept developer	Prototyper	**Wei495@usc.edu**
Lin Xia	Life Cycle Planner	Feasibility Analyst	**xialin@usc.edu**
William Goishi	IIV & V	Quality Focal Point	**wgoishi@usc.edu**

12/04/2017

Version History

Date	Author	Version	Changes made	Rationale
12/04/17	Yufei HONG	1.0	• Original template for use with Populic	• Initial draft for use with Populic

Table of Contents

TEST PLAN AND CASES (TPC) ..... **I**

VERSION HISTORY ..... **II**

TABLE OF CONTENTS ..... **III**

TABLE OF TABLES ..... **IV**

TABLE OF FIGURES ..... **VII**

1. Introduction 8

2. Test Strategy and Preparation 9

2.1 Hardware preparation 9

2.2 Software preparation 9

2.3 Other pre-test preparations 9

2.4 Requirements Traceability 10

3. Test Identification 12

3.1 TC-01 Navigation 12

3.2 TC-02 Notification 21

3.3 TC-03 Pick a competitor 27

3.4 TC-04 Challenge Content 36

3.5 TC-05 Verification and Score 42

3.6 TC-06 Challenge Others 52

3.7 TC-07 Pot 58

4. Resources and schedule 63

4.1 Resources 63

4.2 Staffing and Training Needs 63

4.3 Schedule 64

Table of Tables

Table 1: Requirements Traceability Matrix *10*

Table 2: TC-01-01 Opening and closing the challenge homepage *13*

Table 3: TC-01-02 Viewing today's challenge in detail *14*

Table 4: TC-01-03 Opening the competitor selection page *15*

Table 5: TC-01-04 Submitting topic suggestion to the system *16*

Table 6: TC-01-05 Rank system page *17*

Table 7: TC-01-06 Fetching and demonstrating today's challenge name and upcoming 4 challenge names from the database *18*

Table 8: TC-01-07 Navigating to the embedded "message" application in the iPhone when a user challenge others via contact list *19*

Table 9: TC-01-08 "Back button" in "Competitor Selection page", "Challenging Others page", "Shot page", and "Verification page" should navigate users to the previous page *20*

Table 10: TC-02-01 Check the system should send/receive challenge notification when user has challenged others/been challenged by others *22*

Table 11: TC-02-02 Check system should send notification to a user if he / she has successfully been picked as a competitor *23*

Table 12: TC-02-03 Check system should be able to send/receive post notification when user has sent/received a post. *24*

Table 13: TC-02-04 Check system should send the approved/disapproved notification to a user as soon as his/her post has been verified *25*

Table 14: TC-02-05 TC-02-05 Whenever a user receives a notification, the badge number should be shown on the top-right corner *26*

Table 15: TC-02-06 The application should take users to Message page and fill in "To:", textfield automatically when the user want to change friends in contact list. *27*

Table 16: TC-03-01 Check that user can see the list of today's challengers *28*

Table 17: TC-03-02 Check that the cell will be highlighted when clicked *29*

Table 18: TC-03-03 Check there will be a hint message below the list when one cell is clicked *30*

Table 19: TC-03-04 Check there will be a "Go" button below the hint message when one cell is clicked *31*

Table 20: TC-03-05 Check there will be an alarm dialog when misselecting *32*

Table 21: TC-03-06 Check the list will be updated after closing the alarm dialog *33*

Table 22: TC-03-07 Check the page is initialized when the list has been updated *34*

Table 23: TC-03-08 Check the page is updated to an information page when picking successfully *35*

Table 24: TC-04-01 This item involves testing that a daily challenge content and upcoming challenge content will display correctly. This is a boundary condition that will need to be tested for fetching abnormal data from the server and database. *36*

Table 25: TC-04-02 Display challenge detail *38*

Table 26: TC-04-03 Submit suggestion about challenge *39*

Table 27: TC-04-04 Display challengers and inviting friends *40*

Table 28: TC-04-05 Display challenge remaining time *41*

Table 29: TC-05-01 Approve the competitor's post which was posted later than user's *42*

Table 30: TC-05-02 Approve the competitor's post which was posted earlier than user's *44*

Table 31: TC-05-03 Update the competitor's score when the user was decline *45*

Table 32: TC-05-04 Declines the competitor's post *46*

Table 33: TC-05-05 Not update the competitor's score when the user was decline *47*

Table 34: TC-05-06 Use green word show user or compoetitor is approved *48*

Table 35: TC-05-07 Use red word show user or compoetitor is declined *49*

Table 36: TC-05-08 Never can do today's challenge after being verified *50*

Table 37: TC-05-09 Update rank page after being approved or declined *51*

Table 38: TC-06-01 Check user can see list of all the in-app friends *53*

Table 39: TC-06-02 Check user can see a list of all the phone contacts *54*

Table 40: TC-06-03 Check user can see counter counted selected name correctly *55*

Table 41: TC-06-04 Check the clicked icon shown correctly *56*

Table 42: TC-06-05 Check alphabet view list and list scroll up and scroll down *57*

Table 43: TC-07-01 Test if community button can navigate user to send photo/video interface *58*

Table 44: TC-07-02 Check if the finishing time of the challenge task could be displayed.....*60*

Table 45: TC-07-03 Check if the photo could be previewed *61*

Table 46: TC-07-04 Check if the text could be added on photo.....*62*

Table 47: Testing Schedule.....*64*

Table of Figures

No table of figures entries found.
This is an automatic table of contents. To use it, apply heading styles (on the Home tab) to the text that goes in your table of contents, and then update this table.

If you want to type your own entries, use a manual table of contents (in the same menu as the automatic one).

1. Introduction

The fundamental purpose of testing in this project is checking whether we have met all the requirements that the client have promoted before. Different from other groups, our client also has his own developing teams separated from us. We may only finish this project by the end of this semester, however, the other developers or even upcoming developers will continue to carry out this project based on our achievement. So that we have to make sure the interfaces that we could provide now are perfect.

In general, we used both white box testing and black testing for this project. Particularly, when we wrote our code, we use unit testing, which is under the class of white box testing, to test whether each line of codes can be reached and verified the correctness of them. In terms of the black box testing, the testers/client spent several hours in scrutinizing whether the output matches the expected result. It was intended to qualify every feature of the system and improve the code performance.

We focus on the functional testing in this testing, which means we don't spend plenty of time on unit testing, instead, we have several testing under the class of functional testing and system testing and usability testing to validate that we developed the right product.

This report provides the information about both the entire test cases that we ran during the process and their results. In total, there are 7 different test cases have been tested with either white box testing or black box testing. However, due to the scale of this project, it was separated to several tasks and were assigned to different developers. Even though the Test Driven Development requires to create the test unit prior to coding, the remaining time for integration test is not enough.

2. Test Strategy and Preparation

The testing process is totally followed the agile test-first strategy. Like TDD, before we started to develop a new functionality for the system, we write the test code in advance. By doing so, the unit testing code be done immediately as we pass those pre-written test codes. Because this social networking application is a user-orientation application, the test cases that we have already built in this documentation include both normal cases and off-nominal cases. We followed the win-conditions listed in WinBook 3 months ago and check the feasibility of each test case.

2.1 Hardware preparation

We must have at least 4 iPhones which are used to challenge each other and accept invitations from different users.

2.2 Software preparation

1. Xcode for run the project.
2. At least one Macbook to launch the project in each individual iPhone.
3. The Ubuntu Server which is used to communicate with applications, receiving and responding useful data to the front end pages.
4. React-Native environment and Node.js environment
5. We have in total 843 node modules, in order to test them successfully, we should download them in advance.

2.3 Other pre-test preparations

Because our application use Firebase to send notifications and each iPhone should have a certainly unique access token. So, before testing, we have to make sure each iPhone that we are going to test should have a valid access token and the database saves them in the corresponding place correctly.

2.4 Requirements Traceability

Table 1: Requirements Traceability Matrix

Requirement ID	Use Case	Requirement Description	Verification Type	Test Case ID
WC_4596	UC-7 Selecting Competitor From Challenger List	User can select competitor from challenger list.	Testing	TC-01-03, TC-03-02, TC-03-03
WC_4601	UC-10 Giving Suggestion	User can post own challenge post.	Testing	TC-01-04 TC-05-07 TC-07-01
WC_4595	UC-2 Seeing the Upcoming challenge	User can see upcoming challenges.	Testing	TC-01-02
WC_4600	UC-8 Verify competitor challenger result	User can verify, accept, or decline challenge post.	Testing	TC-01-06 TC-01-12
WC_4599		User receives notification when picked as a competitor.	Testing	TC-02-01 TC-02-02 TC-02-03 TC-02-04 TC-02-05 TC-02-06
WC_4598	UC-8 Verify competitor challenger result	User can verify opponent's post.	Testing	TC-01-05
WC_4625		User can see finish time of challenge task on the game page.	Testing	TC-07-02
WC_4624		User can preview the photo before post it.	Testing	TC-07-03
WC_4623	UC-7 Selecting Competitor From Challenger List	User can pick competitor from challengers who	Testing	TC-03-01

		challenged user.		TC-03-02 TC-03-03 TC-03-04 TC-03-05 TC-03-06 TC-03-07 TC-03-08
WC_4623		User can add text on photo	Testing	TC-07-04
WC_4621		User can navigate backwards to previous page.	Testing	TC-01-14
WC_4620	UC-4 Seeing own post challenge result	User can see ranking among other users.	Testing	TC-01-05
WC_4550	UC-9 Inviting friend from contact list	User can invite friends to compete via the contact list.	Testing	TC-01-06
WC_4536		User can receive notification from completed challenge that was sent.	Testing	TC-02-03
WC_4535	UC-6 Sending Challenge Request	User can give suggestion to challenge topic.	Testing	TC-01-04
WC_4534		User can receive challenge notification from app.	Testing	TC-02-01 TC-02-05 TC-02-06
WC_4533		User can see the remaining time of	Testing	TC-04-05

		challenge.		
WC_4532	UC-6 Sending Challenge Request	User can send challenge request to to friends.	Testing	TC-06-01 TC-06-03 TC-06-04 TC-06-05
WC_4531	UC-1 Seeing the daily challenge	User can see daily challenge on pop-up screen.	Testing	TC-01-01 TC-01-02 TC-01-06

3. Test Identification

3.1 TC-01 Navigation

TC-01 Navigation

3.1.1 Test Level

Software item level

3.1.2 Test Class

Functionality test

3.1.3 Test Completion Criteria

This test will finish when:

- Pages are linked by the workflow which was discussed in the prototype and buttons have the corresponding behaviors described in the prototype as well.

- The front-end pages could fetch and demonstrate the corresponding data from the database blamelessly.

3.1.4 Test Cases

- Opening and closing the challenge homepage
- Viewing today's challenge in detail
- Opening the competitor selection page
- Submitting topic suggestions to the system
- Opening the ranking page
- Fetching today's challenges as well as upcoming four days challenges in the homepage
- Sending text message to people who the user has challenged
- "Back button" in "Competitor Selection page", "Challenging Others page", "Shot page", and "Verification page" should navigate users to the previous page

Table 2: TC-01-01 Opening and closing the challenge homepage

Test Case Number	TC-01-01: Opening/closing the challenge homepage
Test Item	The user may open and close the challenge homepage for several times in a short time.
Test Priority	H
Pre-conditions	None
Post-conditions	None
Input Specifications	The user should open the app and stay in the map page.
Expected Output Specifications	The bottom right button in the home map page should link to the pop up challenge page.
Pass/Fail Criteria	If the pop-up page could show up and close by turns, then this test would pass. Otherwise, it would fail.

Assumptions and Constraints	1. The device should be able to connect to the database
Dependencies	iOS / Simulator
Traceability	WC-4531

Table 3: TC-01-02 Viewing today's challenge in detail

Test Case Number	TC-01-02: Viewing today's challenge in detail
Test Item	The user may not catch the point for a certain challenge topic, we should allow the user to have a look of today's challenge description in detail.
Test Priority	M
Pre-conditions	The challenge information should be stored in the database, including pictures, description and name.
Post-conditions	None
Input Specifications	When user click the challenge topic name in the homepage, a request of viewing detail will be sent to the backend
Expected Output Specifications	A pop-up screen with all the detail information of the selected topic will be presented.
Pass/Fail Criteria	If the pop-up screen cannot show or if any information doesn't match the choiced topic, fail. Otherwise, pass.
Assumptions and Constraints	1. The request has a parameter called date which determines which topic information is required. Thus, this date should be legal and have the corresponding row in the database 2. The device should connect to the internet
Dependencies	MySQL, server remotely
Traceability	WC-4531

Table 4: TC-01-03 Opening the competitor selection page

Test Case Number	TC-01-03: Opening the competitor selection page
Test Item	The user should be able to select a competitor from a list of users who have challenged him / her already.
Test Priority	M
Pre-conditions	The user should have been challenged by at least one of his friend
Post-conditions	None
Input Specifications	1. The user id
Expected Output Specifications	The user will be taken to the competitor list which contains all the user's nickname who have challenged him / her before.
Pass/Fail Criteria	Pass when the user can be navigate to the competitor list and the information listed there should match this user. Fail otherwise.
Assumptions and Constraints	1. The device should be able to connect to the database 2. The user should be challenged by others
Dependencies	MySQL, Ubuntu server
Traceability	WC-4596

Table 5: TC-01-04 Submitting topic suggestion to the system

Test Case Number	TC-01-04: Submitting topic suggestions to the system
Test Item	This system check if the user has capability to promote their own suggestions to the system successfully.
Test Priority	M
Pre-conditions	The user should hold some ideas and type them in the pop-up submitting page
Post-conditions	The suggestion data will be stored in the backend server
Input Specifications	1. The user id 2. The user's suggestion (one for per time) 3. Click the send button
Expected Output Specifications	The pop-up suggestion dialog will be dismissed and user will be taken back to the challenge page
Pass/Fail Criteria	Pass if the navigation could perform as expected ad data could be stored into the system flawlessly.
Assumptions and Constraints	1. The device is able to connect to the remote database
Dependencies	MySQL, Ubuntu Server, iOS / simulator
Traceability	WC-4535

Table 6: TC-01-05 Rank system page

Test Case Number	TC-01-05: Rank system page
Test Item	This test case checks the rank system functionality.
Test Priority	M
Pre-conditions	There is a rank table in the database which contains all the user's score information for today's challenge.
Post-conditions	None
Input Specifications	1. Today's date
Expected Output Specifications	The rank button in challenge homepage should demonstrate the rank page of all users' scores in non-ascending order.
Pass/Fail Criteria	Pass if the rank page could should all the users in this app rank in non-ascending order. Fail otherwise.
Assumptions and Constraints	1. The device is able to connect to the remote server.
Dependencies	MySQL, Ubuntu Server, iOS / simulator
Traceability	WC-4620

Table 7: TC-01-06 Fetching and demonstrating today's challenge name and upcoming 4 challenge names from the database

Test Case Number	TC-01-06: Fetching and demonstrating today's challenge name and upcoming challenge names from the database
Test Item	This test case checks whether today's challenge as well as upcoming four days' challenge topics from the database successfully.
Test Priority	M
Pre-conditions	The database should store the today's challenge and upcoming challenges before.
Post-conditions	None
Input Specifications	1. Today's date 2. Upcoming four day's dates
Expected Output Specifications	The homepage is supposed to fetch and demonstrate those information in the correct area.
Pass/Fail Criteria	Pass if all of those names can be fetched successfully.
Assumptions and Constraints	1. The device should be able to connect to the remote server
Dependencies	MySQL, Ubuntu Server, iOS / simulator
Traceability	WC-4531, WC-4595

Table 8: TC-01-07 Navigating to the embedded "message" application in the iPhone when a user challenge others via contact list

Test Case Number	TC-01-07: Navigating to the embedded "message" application in the iPhone when a user challenge others via contact list.
Test Item	This test checks whether the project can navigate the user to message app and fill in the text field as well as receiver phone number automatically.
Test Priority	H
Pre-conditions	1. The user should allowed us to access their contact list 2. The user should have challenged some people via the contact list.
Post-conditions	This challenge information will be stored in the database.
Input Specifications	1. The challenged phone number. 2. The user id
Expected Output Specifications	The user will be navigated to the Messages app when he / she hits the challenge button in contact list challenge page. And this system will take them back after they hit the "↑" button in the Messages.
Pass/Fail Criteria	Pass if the application could navigate them in the correct direction in this flow and send text msg successfully.
Assumptions and Constraints	1. The device should be able to connect to the remote server 2. The device should have SIM card 3. The SIM card should allow the user (iPhone) typically to send their friends text message.
Dependencies	MySQL, Ubuntu Server, iPhone.
Traceability	WC-4534

Table 9: TC-01-08 "Back button" in "Competitor Selection page", "Challenging Others page", "Shot page", and "Verification page" should navigate users to the previous page

Test Case Number	TC-01-06: "Back button" in "Competitor Selection page", "Challenging Others page", "Shot page", and "Verification page" should navigate users to the previous page they visited before.
Test Item	Cause many users perhaps go to a page undesirably, we have to allow users to move back to the previous page following the user-friendly design
Test Priority	M
Pre-conditions	The user enters an page by mistake / mistyping.
Post-conditions	None
Input Specifications	1. The user click the "GO BACK" BUTTON
Expected Output Specifications	The user is linked to the previous page.
Pass/Fail Criteria	Pass if the user can be taken back to the previous page blamelessly.
Assumptions and Constraints	None.
Dependencies	iOS / Simulator
Traceability	WC_4621

3.2 TC-02 Notification

TC-02: Notification

3.2.1 Test Level

System Performance

3.2.2 Test Class

Correctness and Erroneous Tests.

3.2.3 Test Completion Criteria

- Check system should send/receive challenge notification when user has challenged others/been challenged by others.
- Check system should send notification to a user if he / she has successfully been picked as a competitor
- Check system should be able to send/receive post notification when user has sent/received a post.
- Check system should be able to send/receive post notification when user has sent/received a post.
- 5
- Whenever a user receives a notification, the badge number should be shown on the top-right corner.
- The application should take users to Message page and fill in “To:”, textfield automatically when the user want to change friends in contact list.

3.2.4 Test Cases

Table 10: TC-02-01 Check the system should send/receive challenge notification when user has challenged others/been challenged by others

Test Case Number	TC-02-01
Test Item	When user send challenge invitation to others, system will send notification to user's challengee and the system should receive notification when user was challenged by others
Test Priority	M
Pre-conditions	User open the notification setting of the app
Post-conditions	Firebase receive notification request
Input Specifications	User send challenge invitation or receive challenged
Expected Output Specifications	System will send notification to user's challenge and the system should receive notification when user was challenged by others
Pass/Fail Criteria	Pass, if system send notification to user's challenge if user challenge him and the system should receive notification when user was challenged by others
Assumptions and Constraints	User should have friends to challenge and user's friends are willing to challenge user.
Dependencies	iOS, Firebase, MySQL, Ubuntu
Traceability	WC-4599

Table 11: TC-02-02 Check system should send notification to a user if he / she has successfully been picked as a competitor

Test Case Number	TC-02-02
Test Item	System should send notification to a user if he / she has successfully been picked as a competitor
Test Priority	M
Pre-conditions	User open the notification setting of the app
Post-conditions	Firebase receive notification request
Input Specifications	The user is picked by other user
Expected Output Specifications	The one who is picked should receive notification with correct information.
Pass/Fail Criteria	Pass, if the one who is picked can receive notification with correct information
Assumptions and Constraints	The user is picked by other user
Dependencies	iOS, Firebase, MySQL, Ubuntu
Traceability	WC-4599

Table 12: TC-02-03 Check system should be able to send/receive post notification when user has sent/received a post.

Test Case Number	TC-02-03
Test Item	When user took a photo/video, he/she should be able to sent a notification to his/her competitor. When user received a post from his/her competitor, user should be able to receive a notification from his/her competitor.
Test Priority	H
Pre-conditions	User open the notification setting of the app
Post-conditions	Firebase receive notification request
Input Specifications	User sent post to competitor or received post from competitor
Expected Output Specifications	System will send notification to user's competitor and the system should receive notification when user was challenged by others
Pass/Fail Criteria	Pass, when user post a photo/video, system will send a notification to competitor. When competitor post a photo/video, user will receive a notification from system.
Assumptions and Constraints	User should have friends to send challenge task and user's friends are willing to send challenge task to user.
Dependencies	iOS, Firebase, MySQL, Ubuntu
Traceability	WC-4536, WC-4534

Table 13: TC-02-04 Check system should send the approved/disapproved notification to a user as soon as his/her post has been verified.

Test Case Number	TC-02-04
Test Item	The system should send the approved/disapproved notification to a user as soon as his/her post has been verified.
Test Priority	M
Pre-conditions	User open the notification setting of the app
Post-conditions	Firebase receive notification request
Input Specifications	The competitor send approved or declined the user's challenge result.
Expected Output Specifications	System will send notification to user's app and the system should receive notification when user's challenge post is approved or declined.
Pass/Fail Criteria	Pass, if system send notification to user's app if his competitor approve or decline and the system should receive notification when user was approved or decliend by others
Assumptions and Constraints	User should have done the challenge and post his/her challenge result.
Dependencies	iOS, Firebase, MySQL, Ubuntu
Traceability	WC-4534,WC-4600

Table 14: TC-02-05 TC-02-05 Whenever a user receives a notification, the badge number should be shown on the top-right corner

Test Case Number	TC-02-05
Test Item	Test the badge number correctly shown on the top-right corner of app icon
Test Priority	M
Pre-conditions	1. User open the notification setting of the app 2. User should can and already get the notification, user did not clean the notification yet.
Post-conditions	Firebase receive notification request
Input Specifications	The system get notification
Expected Output Specifications	If notification not be clear, the badge number should be shown on the top-right corner
Pass/Fail Criteria	Pass, whenever the user get notification, the badge number increase.
Assumptions and Constraints	The badge number will be set to 0, when any notification be checked by user.
Dependencies	iOS, Firebase, MySQL, Ubuntu
Traceability	WC-4598

Table 15: TC-02-06 The application should take users to Message page and fill in "To:", textfield automatically when the user want to change friends in contact list.

Test Case Number	TC-02-06
Test Item	User can see "To: ", text field automatically filled with names selected by user in contact list. "Content: " automatically filled by app.
Test Priority	M
Pre-conditions	Current page is on the iPhone message page
Post-conditions	Future page navigate to challenge others page
Input Specifications	Names were selected by user in contact list
Expected Output Specifications	"To: ", text field automatically filled with names selected by user in contact list. "Content: " automatically filled by app.
Pass/Fail Criteria	Pass if "To:" and "Content: " filled automatically
Assumptions and Constraints	User give permission for application to use iPhone message system
Dependencies	iOS
Traceability	WC_4596

3.3 TC-03 Pick a competitor

Description: In this page, users can pick a person to pair with among those who have sent challenge invitation to him/her.

3.3.1 Test Level

Functional Testing: This adheres to a type of black-box testing that is based on the functionalities or the requirements to be tested. User is expected to test the application by providing certain input for the requirement to be tested after which the outcome is thoroughly examined.

3.3.2 Test Class

Correctness and Erroneous Tests.

3.3.3 Test Completion Criteria

The tests will be completed when the below conditions are satisfied:

If failed, an alarm dialog appears, the list will be updated and users can repick; If successful, users will go to the challenge information page and the game is on.

3.3.4 Test Cases

Table 16: TC-03-01 Check that user can see the list of today's challengers

Test Case Number	TC-03-01
Test Item	This test will check that user can see the list of today's challengers
Test Priority	M
Pre-conditions	User has be navigated to this page after clicking the "challengers" button in homepage.
Post-conditions	User can choose a cell in the list.
Input Specifications	Clicks the "challengers" button in homepage.
Expected Output Specifications	List of today's challengers
Pass/Fail Criteria	Pass if user can see a list of today's challengers

Assumptions and Constraints	The user has not been paired with others and at least one person has challenged him / her today.
Dependencies	None
Traceability	WC_4623

Table 17: TC-03-02 Check that the cell will be highlighted when clicked

Test Case Number	TC-03-02
Test Item	When the user clicks a cell in the “Pick a competitor” page, it will be highlighted and only one is highlighted.
Test Priority	S
Pre-conditions	User has be navigated to this page and at least one cell is in the list.
Post-conditions	User can click the “Go” button below the list.
Input Specifications	Clicks one of the cells in the list
Expected Output Specifications	The cell clicked is highlighted
Pass/Fail Criteria	Pass if the cell clicked is highlighted and other cells are not.
Assumptions and Constraints	At least one cell is in the list.
Dependencies	None
Traceability	WC_4623

Table 18: TC-03-03 Check there will be a hint message below the list when one cell is clicked

Test Case Number	TC-03-03
Test Item	When the user clicks a cell in the “Pick a competitor” page, there will be a hint message below the list.
Test Priority	S
Pre-conditions	User has be navigated to this page and at least one cell is in the list.
Post-conditions	User can click the “Go” button below the list.
Input Specifications	Clicks one of the cells in the list.
Expected Output Specifications	There will be a hint message below the list.
Pass/Fail Criteria	Pass if there is a hint message below the list.
Assumptions and Constraints	At least one cell is in the list.
Dependencies	None
Traceability	WC_4623

Table 19: TC-03-04 Check there will be a "Go" button below the hint message when one cell is clicked

Test Case Number	TC-03-04
Test Item	When the user clicks a cell in the "Pick a competitor" page, there will be a "Go" button below the hint message.
Test Priority	M
Pre-conditions	User has be navigated to this page and at least one cell is in the list.
Post-conditions	User can click the "Go" button below the list.
Input Specifications	Clicks one of the cells in the list.
Expected Output Specifications	There will be a "Go" button below the hint message.
Pass/Fail Criteria	Pass if there is a "Go" button below the hint message.
Assumptions and Constraints	At least one cell is in the list.
Dependencies	None
Traceability	WC_4623

Table 20: TC-03-05 Check there will be an alarm dialog when misselecting

Test Case Number	TC-03-05
Test Item	When the user clicks the “Go” button with selecting a competitor who has been paired with someone else, there will be an alarm dialog asking the user to repick.
Test Priority	M
Pre-conditions	The user picks a person who has been paired.
Post-conditions	None
Input Specifications	User clicks the “Go” button with selecting a competitor who has been paired with someone else.
Expected Output Specifications	There will be an alarm dialog asking the user to repick.
Pass/Fail Criteria	Pass if there is an alarm dialog asking the user to repick when the user clicks the “Go” button with selecting a competitor who has been paired
Assumptions and Constraints	The user picks a person who has been paired.
Dependencies	None
Traceability	WC_4623

Table 21: TC-03-06 Check the list will be updated after closing the alarm dialog

Test Case Number	TC-03-06
Test Item	When the user clicks the “OK” button in the alarm dialog, it will disappear and the list will be updated, removing all people who have been paired
Test Priority	M
Pre-conditions	None
Post-conditions	None
Input Specifications	Clicks the “OK” button in the alarm dialog
Expected Output Specifications	The list will be updated, removing all people who have been paired.
Pass/Fail Criteria	Pass if the list is updated and removes all people who have been paired after clicking the “OK” button in the alarm dialog.
Assumptions and Constraints	None
Dependencies	None
Traceability	WC_4623

Table 22: TC-03-07 Check the page is initialized when the list has been updated

Test Case Number	TC-03-07
Test Item	When the list has be updated, no one is in selected status. In other words, no one is highlighted and there is no hint message and “Go” button in this page.
Test Priority	M
Pre-conditions	The list has been updated.
Post-conditions	None
Input Specifications	Clicks the “OK” button in the alarm dialog
Expected Output Specifications	No one is highlighted and there is no hint message and “Go” button in this page.
Pass/Fail Criteria	Pass if no one is highlighted and there is no hint message and “Go” button in this page.
Assumptions and Constraints	None
Dependencies	None
Traceability	WC_4623

Table 23: TC-03-08 Check the page is updated to an information page when picking successfully

Test Case Number	TC-03-08
Test Item	When the user clicks the “Go” button with selecting a competitor who has not been paired with someone else, there will be an information page showing the information about today’s challenge
Test Priority	M
Pre-conditions	User click someone not paired
Post-conditions	None
Input Specifications	Clicks the “Go” button
Expected Output Specifications	There will be an information page showing the information about today’s challenge
Pass/Fail Criteria	Pass if there is an information page showing the information about today’s challenge
Assumptions and Constraints	None
Dependencies	None
Traceability	WC_4623

3.4 TC-04 Challenge Content

TC-04: challenge content

3.4.1 Test Level

Software level testing

3.4.2 Test Class

Timing testing, Alpha testing, UI testing

3.4.3 Test Completion Criteria

- The front-end connects to our backend and fetch data from our database. and according to different content, font end will display different style and content.
- Front-end can deal with the situation of no internet, no data fetching from database and send data to server
- Front-end dose not crash during testing

3.4.4 Test Cases

Table 24: TC-04-01 This item involves testing that a daily challenge content and upcoming challenge content will display correctly. This is a boundary condition that will need to be tested for fetching abnormal data from the server and database.

Description: This item involves testing that a daily challenge content and upcoming challenge content will display correctly. This is a boundary condition that will need to be tested for fetching abnormal data from the server and database.

Test Case Number	TC-04-01 display right challenge content
Test Item	Testing that the challenge page should display the daily challenge and upcoming challenge
Test Priority	S
Pre-conditions	You must be click the challenge in the map page.

Post-conditions	You can close the challenge
Input Specifications	NONE
Expected Output Specifications	The page will display the challenge content and upcoming challenge content which is the same as the database
Pass/Fail Criteria	This page should display right content according to the specific date and input.
Assumptions and Constraints	The front-end has a working connection to the server and database and can fetch all the detail data of the challenge.
Dependencies	This application should not crash when the server doesn't return any data to front-end and the front-end doesn’t connect to the server and database
Traceability	WC_4626: As a user, I can check daily challenge and upcoming challenge.

Table 25: TC-04-02 Display challenge detail

Description: This item involves testing the popup box of challenge detail. The user have specific pages to get the detail of challenge content. So this popup box will have GIF about how to finish the challenge and the way of proof.

Test Case Number	TC-04-02 Display challenge detail
Test Item	Testing the ability of challenge page can pop out a new box to display the detail information of daily challenge
Test Priority	S
Pre-conditions	User must be wait until the challenge main page is opened and data is fetched from server.
Post-conditions	User can click the close button to close this popup page.
Input Specifications	User open the challenge page and click the text of daily challenge name.
Expected Output Specifications	The challenge detail page will display and the content of the this page must include the GIF, challenge name, challenge description and proof way
Pass/Fail Criteria	The test will pass if all the detail are included and the contents are same with the content which maintainer input in the database.
Assumptions and Constraints	Popup box has a working connection to the database backend and can retrieve and store information. And when there is no internet, the page will not crash.
Dependencies	This application does not crash when opened and the database backend server can accept requests. User can enter text into the text box to add additional curriculum items.
Traceability	WC_4627:As a user, I can get the detail of daily challenge content

Table 26: TC-04-03 Submit suggestion about challenge

Description: This item involves testing the ability of submit button. The user can submit the suggestion about the challenge content to the back-end server. The user can input the sentence and click submit in the bottom to submit the content to our server and if it success, there would display a toast the to user.

Test Case Number	TC-04-03 Submit suggestion about challenge
Test Item	Testing the ability of the user can submit the suggestion about the challenge content to the back-end server.
Test Priority	S
Pre-conditions	User must be wait until the challenge main page is opened and data is fetched from server. And click the submit in the bottom.
Post-conditions	User can close the popup box by click close bottom or click ok bottom.
Input Specifications	The user have to input some words in the TextView otherwise the suggestion page will not send any data to server.
Expected Output Specifications	If the user input null, there is no toast which could remind the user. If the input some content when the server receive the content, user will see a toast to remind him that he already send the suggestion successfully.
Pass/Fail Criteria	The server will receive and store the suggestion which is the same as the user input.
Assumptions and Constraints	System has a working connection to the database backend and can retrieve and store information.
Dependencies	This page does not crash when opened and the database backend server can accept requests. The page has right display when the user can not connect the server.
Traceability	WC_4628: As a user, I can submit some idea which can help application improve.

Table 27: TC-04-04 Display challengers and inviting friends

Description: This item involves testing the ability of challenge page fetch the right number of inviting friends and competitor. When other users were challenged by other users, in users challenge page, the right number of users will be display. The challenge page will also display right number of friends which user has invited daily.

Test Case Number	TC-04-04 display right number of challengers and inviting friends
Test Item	Testing the ability that challenge page can use right API to fetch the right number of challengers and inviting friends
Test Priority	S
Pre-conditions	The user must click the bottom right challenge button in the map page.
Post-conditions	The user can close the challenge page after clicking the close button
Input Specifications	NONE
Expected Output Specifications	The page will display the the right number of inviting friends and competitor which is fetch the right number.
Pass/Fail Criteria	The page will display those two type of data and the content would be the same as the data which we store in the database.
Assumptions and Constraints	System has a working connection to the database backend and can retrieve and store information.
Dependencies	This page does not crash when opened and the database backend server can accept requests. The page will display right content when there is no internet.
Traceability	WC_4629: As a user, I can know, how many friend I invite and how many friend challenge me.

Table 28: TC-04-05 Display challenge remaining time

Description: This item involves testing the ability of how much time the user can have before the challenge expired. The user can have timer to remind them the deadline of the daily challenge

Test Case Number	TC-04-05 display right challenge remaining time.
Test Item	Testing the ability that challenge page can get the time of the phone and count the deadline of challenge
Test Priority	S
Pre-conditions	The user must click the bottom right challenge button in the map page.
Post-conditions	The user can close the challenge page after clicking the close button
Input Specifications	NONE
Expected Output Specifications	The page will display the the right time
Pass/Fail Criteria	The page will display remaining time would be the same as the data which we store in the database.
Assumptions and Constraints	System has a working connection to the database backend and can retrieve and store information.
Dependencies	This page does not crash when opened and the database backend server can accept requests. The page will display right content when there is no internet.
Traceability	WC_4533: As a user, I will see the time remaining of the challenge

3.5 TC-05 Verification and Score

Description: In this page, users can verify his/her competitor's post(approve or decline), and the score they get should show on the main challenge page.

3.5.1 Test Level

Functional Testing: This adheres to a type of black-box testing that is based on the functionalities or the requirements to be tested. User is expected to test the application by providing certain input for the requirement to be tested after which the outcome is thoroughly examined.

3.5.2 Test Class

Correctness and Erroneous Tests.

3.5.3 Test Completion Criteria

The tests will be completed when the below conditions are satisfied:

3.5.4 Test Cases

Table 29: TC-05-01 Approve the competitor's post which was posted later than user's

Test Case Number	TC-05-01 Approve the competitor's post which was posted later than user's.
Test Item	When the user approved the competitor's post which was posted later than his, the competitor could get 300 points.
Test Priority	M
Pre-conditions	The user should post his challenge result to the server and the go to confirm button is shown in the main challenge page.
Post-conditions	None
Input Specifications	Clicks the "Approve" button in post page.

Expected Output Specifications	The go to confirm button disappear and the main challenge show of the user and the competitor show the competitor get 300 points.
Pass/Fail Criteria	Pass if competitor and user both can see correct approved result in his/her own main challenge page.
Assumptions and Constraints	The competitor's post was posted later than the user's post.
Dependencies	None
Traceability	WC_4600

Table 30: TC-05-02 Approve the competitor's post which was posted earlier than user's

Test Case Number	TC-05-02 Approve the competitor's post which was posted earlier than user's.
Test Item	When the user approved the competitor's post which was posted earlier than his, the competitor could get 400 points.
Test Priority	M
Pre-conditions	The user should post his challenge result to the server and the go to confirm button is shown in the main challenge page.
Post-conditions	None
Input Specifications	Clicks the "Approve" button in post page.
Expected Output Specifications	The go to confirm button disappear and the main challenge show of the user and the competitor show the competitor get 400 points.
Pass/Fail Criteria	Pass if competitor and user both can see correct approved result in his/her own main challenge page.
Assumptions and Constraints	The competitor's post was posted earlier than the user's post.
Dependencies	None
Traceability	WC_4600

Table 31: TC-05-03 Update the competitor's score when the user was decline

Test Case Number	TC-05-02 Update the competitor's score when the user was decline.
Test Item	When the user approved the competitor's post which was posted later than his, and then the competitor decline this user's post, the competitor score update from 300 points to 400 points.
Test Priority	S
Pre-conditions	The user have approved competitor's post which was posted later than his, and then the competitor declined the user's post.
Post-conditions	None
Input Specifications	The competitor clicks the "decline" button in post page.
Expected Output Specifications	The main challenge show of the user and the competitor show the competitor's score change from 300 to 400.
Pass/Fail Criteria	Pass if competitor and user both can see correct scire change in his/her own main challenge page.
Assumptions and Constraints	The competitor's post was posted later than the user's post, and approved earlier than the user.
Dependencies	None
Traceability	WC_4600

Table 32: TC-05-04 Declines the competitor's post

Test Case Number	TC-05-02 Declines the competitor's post
Test Item	When the user declined the competitor's post, the competitor could get 0 points.
Test Priority	M
Pre-conditions	The user have decline competitor's post which was posted later or earlier than his/her.
Post-conditions	None
Input Specifications	The user clicks the "decline" button in post page.
Expected Output Specifications	The go to confirm button disappear and the main challenge show of the user and the competitor show the competitor get 0 points.
Pass/Fail Criteria	Pass if competitor and user both can see correct score in his/her own main challenge page.
Assumptions and Constraints	The competitor's post his challenge result.
Dependencies	None
Traceability	WC_4600

Table 33: TC-05-05 Not update the competitor's score when the user was decline

Test Case Number	TC-05-02 Not update the competitor's score when the user was decline.
Test Item	When the user approved the competitor's post which was posted earlier than his, and then the competitor decline this user's post, the competitor score do not change, which is still 400 points.
Test Priority	S
Pre-conditions	The user have approved competitor's post which was posted earlier than his, and then the competitor declined the user's post.
Post-conditions	None
Input Specifications	The competitor clicks the "decline" button in post page.
Expected Output Specifications	The main challenge show of the user and the competitor show the competitor's score is not change, and show the user's score is 0.
Pass/Fail Criteria	Pass if competitor and user both can see correct score in his/her own main challenge page.
Assumptions and Constraints	The competitor's post was posted earlier than the user's post, and be approved earlier than the user.
Dependencies	None
Traceability	WC_4600

Table 34: TC-05-06 Use green word show user or compoetitor is approved

Test Case Number	TC-05-02 Use green word show the user or the competitor is approved.
Test Item	When the competitor is approved by the user, the main challenge page use green word to show the user is approved, and his/her score.
Test Priority	S
Pre-conditions	The user or the competitor have been approved .
Post-conditions	None
Input Specifications	The competitor or the user clicks the “Approve” button in post page.
Expected Output Specifications	The main challenge show of the user and the competitor show the competitor’s or the user’s challenge result is comfirmed, and word color is green
Pass/Fail Criteria	Pass if competitor and user both can see correct challenge result and the world color is correct.
Assumptions and Constraints	The competitor or the user have post his/her challenge result and been approved.
Dependencies	None
Traceability	WC_4600

Table 35: TC-05-07 Use red word show user or compoetitor is declined

Test Case Number	TC-05-02 Use red word show the user or the competitor is declined.
Test Item	When the competitor is declined by the user, the main challenge page use red word to show the user is approved, and his/her score is 0.
Test Priority	S
Pre-conditions	The user or the competitor have been declined .
Post-conditions	None
Input Specifications	The competitor or the user clicks the “Declined” button in post page.
Expected Output Specifications	The main challenge show of the user and the competitor show the competitor’s or the user’s challenge result is declined, and word color is red
Pass/Fail Criteria	Pass if competitor and user both can see correct challenge result and the world color is correct.
Assumptions and Constraints	The competitor or the user have post his/her challenge result and been declined.
Dependencies	None
Traceability	WC_4600

Table 36: TC-05-08 Never can do today's challenge after being verified

Test Case Number	TC-05-07 Never can do today's challenge after being verified
Test Item	When the user is declined by the competitor,he/she can never do the challenge again.
Test Priority	S
Pre-conditions	The user or the competitor have been approved or decliend.
Post-conditions	None
Input Specifications	The competitor or the user clicks the "Approve" or "Decline"button in post page.
Expected Output Specifications	The main challenge show of the user and the competitor show the competitor's or the user's challenge result is declined or approved, and he/she can never have a do it button to do today's challenge.
Pass/Fail Criteria	Pass if competitor and user both can never have do it button in the main challenge page.
Assumptions and Constraints	The competitor or the user have post his/her challenge result and been approved or declined.
Dependencies	None
Traceability	WC_4601

Table 37: TC-05-09 Update rank page after being approved or declined

Test Case Number	TC-05-08 Update rank page after being approved or declined
Test Item	When the user or the competitor is declined or approved, the score they get will be updated to the rank page immediately.
Test Priority	M
Pre-conditions	The user or the competitor have been approved or decliend.
Post-conditions	None
Input Specifications	The competitor or the user clicks the “Approve” or “Decline”button in post page.
Expected Output Specifications	The score of the user or the competitor in the rank page will be added with the score(300, 400, or 0) that the user or the competitor should get.
Pass/Fail Criteria	Pass if the score in the rank page is updated correctly.
Assumptions and Constraints	The competitor or the user have post his/her challenge result and been approved or declined.
Dependencies	None
Traceability	WC_4620

3.6 TC-06 Challenge Others

TC-06: Challenge Others

3.6.1 Test Level

System Performance

3.6.2 Test Class

Correctness and Erroneous Tests.

3.6.3 Test Completion Criteria

The test for Challenge Others will be completed when

- User can see in-app friend list shown correctly
- User can see contact list shown correctly
- Counter counted selected name correctly
- User can see the clicked icon shown correctly when names were selected
- User can quickly search name via alphabet view list and scroll up or scroll down the page

3.6.4 Test Cases

Table 38: TC-06-01 Check user can see list of all the in-app friends

Test Case Number	TC-06-01
Test Item	This test will check that user can see a list of all the in-app friends
Test Priority	M
Pre-conditions	The user not paired with others
Post-conditions	None
Input Specifications	The interaction of user contacts phones and user register phones
Expected Output Specifications	List of all in-app friends
Pass/Fail Criteria	Pass if user can see a list of all the in-app friends
Assumptions and Constraints	At least one friend in user's contact list should already register our application
Dependencies	iOS / Simulator
Traceability	WC_4532

Table 39: TC-06-02 Check user can see a list of all the phone contacts

Test Case Number	TC-XX-02
Test Item	The test will check user can see a list of all the phone contacts
Test Priority	M
Pre-conditions	The user not paired with others
Post-conditions	None
Input Specifications	The user's iPhone contact information
Expected Output Specifications	List of all in-app friends
Pass/Fail Criteria	Pass if all right format contact
Assumptions and Constraints	1. User has at least one contact in his iPhone Contacts 2. Contacts should at least have a First name or Last name
Dependencies	iOS / Simulator
Traceability	WC_4550

Table 40: TC-06-03 Check user can see counter counted selected name correctly

Test Case Number	TC-06-03
Test Item	When user click any name shown on contacts page, the counter shown on bottom left will increase one and when user re-click the name, the count will decrease one
Test Priority	S
Pre-conditions	None
Post-conditions	None
Input Specifications	At least two names shown on contact list or friend list
Expected Output Specifications	The count changed correct after user operation
Pass/Fail Criteria	Pass if user click any name shown on contacts page, the count shown on bottom left will increase one and when user re-click the name, the count will decrease one
Assumptions and Constraints	None
Dependencies	iOS / Simulator
Traceability	WC_4550, WC_4532

Table 41: TC-06-04 Check the clicked icon shown correctly

Test Case Number	TC-06-04
Test Item	When user click any name shown on contacts page, an icon shown on the right side of the name and when user re-click the name the icon will disappear.
Test Priority	M
Pre-conditions	None
Post-conditions	None
Input Specifications	At least two names shown on contact list or friend list
Expected Output Specifications	The icon shown and disappear correctly
Pass/Fail Criteria	Pass if an icon shown on the right side of the name if user click any name shown on contacts page, and disappear when user re-click the name
Assumptions and Constraints	At least one name shown on contacts list or friend list
Dependencies	iOS / Simulator
Traceability	WC_4550, WC_4532

Table 42: TC-06-05 Check alphabet view list and list scroll up and scroll down

Test Case Number	TC-06-05
Test Item	Check the challenge page can be quickly viewed by alphabet list and challenge page can be scroll up and down to see all the list information.
Test Priority	S
Pre-conditions	None
Post-conditions	None
Input Specifications	Flatlist and alphabet list API in React
Expected Output Specifications	Alphabet view list and page scroll up and down are working.
Pass/Fail Criteria	Pass when user can quick search the name on the list through the alphabet view list and can scroll up or down the page to see all list information
Assumptions and Constraints	None
Dependencies	iOS / Simulator
Traceability	WC_4550, WC_4532

3.7 TC-07 Pot

Description: In this page, users can preview the photo, send a challenge related photo to competitor, add text on the photo and view the time when user send the photo.

3.7.1 Test Level

System Performance

3.7.2 Test Class

Correctness and Erroneous Tests.

3.7.3 Test Completion Criteria

The test for Post will be completed when

- User can preview photo after took a photo
- User can add text on the photo after took a photo
- User can see the challenge task finish time after finished challenge task
- User can be navigated to send picture/video interface when user picked a community

3.7.4 Test Cases

Table 43: TC-07-01 Test if community button can navigate user to send photo/video interface

Test Case Number	TC-07-01
Test Item	After took a picture/video, when user clicked community button to choose which community to post, the button should be able to navigate user to send picture/video interface.
Test Priority	M
Pre-conditions	1. User has taken a picture/video successfully 2. User has joined one or more communities.

Post-conditions	Server received the upload data
Input Specifications	User picks a community by clicking community
Expected Output Specifications	The send picture/video page will be displayed once a certain community has been picked
Pass/Fail Criteria	Pass, when user picked a certain community, user will be navigated to send picture/video interface
Assumptions and Constraints	User has community and would like to post the photo/video to the community.
Dependencies	iOS / Simulator
Traceability	WC-4601

Table 44: TC-07-02 Check if the finishing time of the challenge task could be displayed

Test Case Number	TC-07-02
Test Item	When user clicked the send button to send the challenge task which has been completed to the competitor, user should be able to see his/her finish time of the challenge task on the game on interface
Test Priority	M
Pre-conditions	1. User has Internet 2. User has already finished challenge tasks and clicked send button
Post-conditions	Server received request
Input Specifications	User has send the challenge post
Expected Output Specifications	The challenge finish time of challenge task will be displayed on game on interface.
Pass/Fail Criteria	Pass, when user or competitor or both of them finished challenge task, the finish time will be displayed on game on page
Assumptions and Constraints	User would like to do challenge task and would like to send challenge photo/video to competitor
Dependencies	iOS / Simulator, MySQL, Ubuntu
Traceability	WC-4625

Table 45: TC-07-03 Check if the photo could be previewed

Test Case Number	TC-07-03
Test Item	When user hit the shooting button and take a photo, a preview photo should on the screen to allow user to preview the photo that he/she just took
Test Priority	M
Pre-conditions	User has already taken a photo
Post-conditions	None
Input Specifications	User opened camera to take a photo
Expected Output Specifications	The photo will be displayed on the screen after took a photo
Pass/Fail Criteria	Pass, when user took a photo the photo will be displayed on screen to allow user to preview
Assumptions and Constraints	User would like to take a photo for challenge task and preview the photo
Dependencies	iOS / Simulator
Traceability	WC-4624

Table 46: TC-07-04 Check if the text could be added on photo

Test Case Number	TC-07-04
Test Item	When user took a photo, he/she should able to add text on the photo
Test Priority	M
Pre-conditions	User has already taken a photo
Post-conditions	Photo could be previewed
Input Specifications	User input text on the photo
Expected Output Specifications	Text will be showed on the photo
Pass/Fail Criteria	Pass, when user took a photo, user could add text on the photo
Assumptions and Constraints	User would like to add text on the photo
Dependencies	iOS / Simulator
Traceability	WC-4622

4. Resources and schedule

We have finished the Unit and Integration test. And all the bugs we have found also were fixed in time.

4.1 Resources

JIRA is our bug tracking tools. The team assign the different test job to different team member. So that team member would be the tester of that part and he will be responsible to the performance of that part. The test will create the test case and perform the test process. All the test will provide those information including test case, result and test process. After all the test finish the unit test. We have the cross-test to make sure every test process is correct and the result is reliable.

For our back-end, we use some tool and write some script to test all the API to make sure all of them work well. Our back-end team member run our back-end code and database on our client server.

4.2 Staffing and Training Needs

All the team member was included in the test process. And each of them are responsible to one part. And we require all the team member note down test case and test process. If we find any bugs, all the team member will offer logs and summary all those bugs to help other team member avoid the similar problem. For the back-end test, our team member use tools and write some script to help test API. There is basic rule when we start to do some test job. Each team member will follow standard instruction to write the test case and generate the test process

4.3 Schedule

Table 47: Testing Schedule

Date	Test Identifier	Responsible person	Resources	Training needs
11/01/16	TC-01-01, TC-01-02, TC-01-03, TC-01-04, TC-01-05, TC-01-06, TC-01-07	Chengyu Shen	Computer, Phone and a tablet with any HTML5 supported browser	N/A
11/05/16	TC-02-01, TC-02-02, TC-02-03, TC-02-04, TC-02-05, TC-02-06	Shiji Zhou, Lin Xia, Yang Wei, Yufei Hong, Guanghe Cao	Computer, Phone and MySQL	N/A
11/26/17	TC-03-01, TC-03-02, TC-03-03, TC-03-04, TC-03-05, TC-03-06, TC-03-07, TC-03-08,	Shiji Zhou	IOS mobile phones.	N/A

11/20/17	TC-04-01, TC-04-02, TC-04-03, TC-04-04	Lin Xia	Computer, Phone and MySQL.	N/A
11/18/16	TC-05-01, TC-05-02, TC-05-03, TC-05-04, TC-05-05 TC-05-06, TC-05-07, TC-05-08	Guanghe Cao	IOS mobile phones, MySQL.	N/A
11/20/16	TC-06-01, TC-06-02, TC-06-03, TC-06-04, TC-06-05,	Yang Wei	iOS phones, Computer	N/A
11/25/16	TC-07-01, TC-07-02, TC-07-03, TC-07-04,	Yufei Hong	Mac, iOS mobile Phones.	N/A