

# Test Plan and Cases (TPC)

**We Are Trojans (WAT) Network**

**Team No.01**

<b>Team members</b>	<b>Roles</b>
<b>Eirik Skogstad</b>	<b>Project Manager, Life Cycle Planner</b>
<b>Min Li</b>	<b>Feasibility Analyst, Operational Concept Engineer</b>
<b>Pittawat Pamornchaisirikij</b>	<b>NDI/NCS Acquirer &amp; Evaluator, Tester</b>
<b>Saloni Priya</b>	<b>Requirements Engineer, UML Modeler</b>
<b>Suleyman Erten</b>	<b>Operational Concept Engineer, Requirement Engineer</b>
<b>Kamonphop Srisopha</b>	<b>Prototyper, UML modeler</b>
<b>Ameer Elkordy</b>	<b>IIV&amp;V, Quality Focal Point</b>
<b>Punyawee Pakdiying</b>	<b>System Architect, Feasibility Analyst</b>

**11/30/2014**

# Version History

Date	Author	Version	Changes made	Rationale
11/30/14	PS	0.5	<ul style="list-style-type: none"><li>Completed section 1,2 and 3</li></ul>	<ul style="list-style-type: none"><li>Initial draft for WAT Network</li></ul>

# Table of Contents

<b>TEST PLAN AND CASES (TPC).....</b>	<b>I</b>
<b>VERSION HISTORY .....</b>	<b>II</b>
<b>TABLE OF CONTENTS.....</b>	<b>III</b>
<b>TABLE OF TABLES .....</b>	<b>IV</b>
<b>TABLE OF FIGURES .....</b>	<b>V</b>
1. Introduction .....	6
2. Test Strategy and Preparation.....	6
2.1 Hardware preparation .....	7
2.2 Software preparation .....	7
2.3 Other pre-test preparations .....	7
2.4 Requirements Traceability .....	7
3. Test Identification.....	9
3.1 Test Identifier .....	9
3.2 Test Identifier .....	13
3.3 Test Identifier .....	15
4. Resources and schedule .....	17
4.1 Resources .....	17
4.2 Staffing and Training Needs .....	17
4.3 Schedule .....	17

# Table of Tables

<i>Table 1: Requirements Traceability Matrix .....</i>	<i>7</i>
<i>Table 2: TC-01-01 User Role to use the thread/post functionality .....</i>	<i>9</i>
<i>Table 3: TC-02-01 Update user WAT point information .....</i>	<i>13</i>
<i>Table 4: TC-03-01 User Role to use online redemption system .....</i>	<i>15</i>

# Table of Figures

*Figure 1: <Figure Title> ..... **Error! Bookmark not defined.***

# 1. Introduction

This Test Plan and Cases (TPC) document describes the test cases and acceptance criteria of the WAT Network system. The purpose of doing this is to identify draft acceptance test for specific win-win requirements documented in winbook. The most critical capability requirements are tested in this document. Test cases are drafted from various use-cases developed for the SSAD document. Testing is focused at software level to test the key features of our system.

## 2. Test Strategy and Preparation

Based on the key requirements for our system in winbook and taking help from the use-case diagram for specific features we built a series of acceptance test cases for functionalities. We defined the project's test strategy as "Value-based test prioritization" and "Requirements-test traceability".

## 2.1 Hardware preparation

Users will be able to access the web interface using any handheld device. So for our hardware preparation we just need to test the system that we will use for development that includes our personal laptops.

## 2.2 Software preparation

We plan to obtain information about SeleniumHQ, PHPUnit, PHPSpec in order to implement testing the framework.

## 2.3 Other pre-test preparations

NA

## 2.4 Requirements Traceability

*Table 1: Requirements Traceability Matrix*

<b>Requirement ID</b>	<b>Use-Case ID Number</b>	<b>Verification Type</b>	<b>Test Case ID (if applicable)</b>
CR-1: Start/Edit/Delete a thread: Users are able to start/edit/delete threads	WC_3240 WC_3241	Testing	TC01- User Role to use the thread/post functionality
CR-2: Calculate the WAT points: The system should correctly calculate three types of WAT points.	WC_3260 WC_3252	Testing	TC02- Update user WAT point information
CR-3: Like/Dislike: Users are able to like or dislike threads and posts		Inspection	
CR-4: Make a post: Users could post on the thread.	WC_3240 WC_3241	Testing	TC01- User Role to use the thread/post functionality
CR-5: Redeem the gift card:	WC_3435	Testing	TC03- User Role to use

Users could use the usable points to redeem the gift card	WC_3245		online redemption system
CR-6 Automated useless threads/posts deletion: The system is capable of deleting the useless threads/posts		Inspection	
CR-7 Automated notification: The system is able to notify the users the number of likes/dislikes/posts on their threads and posts and recent events.		Inspection	
CR-8 Create/Edit/Delete the event: The maintainer is able to create/edit the event on the event activity board.		Inspection	
CR-9 Categorize the threads		Inspection	



## 3. Test Identification

### 3.1 Test Identifier

TC01- User Role to use the thread/post functionality

Test Case 01 covers the feature of the WAT Network, where the user interacts with the web-interface using the handheld device. This test includes test cases relating to user role; to start/edit/delete a thread on the forum or add a post for a specific thread.

#### 3.1.1 Test Level

Software item level

#### 3.1.2 Test Class

Capability Test

#### 3.1.3 Test Completion Criteria

User is able to add a thread on the forum that does not contain any illegal information, and the thread details are stored in back-end system.

If necessary user is able to edit his thread, and the changes he made is visible on the forum. The necessary updates are made to the system back-end.

If user want he can delete the thread at his wish, and the thread is no longer visible on the forum. The necessary updates are made to the system back-end.

If user create a post on a specific thread, the post is visible on the forum under the specific thread. The data specific to the post is stored in the database.

#### 3.1.4 Test Cases

Table 2: TC-01-01 User Role to use the thread/post functionality

Test Case Number	TC-01-01
Test Item	This test case test the basic features of our system where user can start/edit/delete a thread on forum and add a post

	for a specific thread.
Test Priority	M (Must Have)
Pre-conditions	<p>Start a thread: We need to have a forum page where there is an option to start a thread.</p> <p>Edit/Delete Thread: Once the user creates a thread, he can view the option to edit his thread or delete the thread created.</p> <p>Start a Post: User is on the forum page and is able to view the threads on forum. To make a post, select the option of adding post.</p>
Post-conditions	<p>The thread should be posted on the forum page with some basic description as when the thread was created, who created the thread.</p> <p>Other users of the system should be able to view the thread on the forum.</p> <p>The thread should have the option for like/dislike</p> <p>The thread data should be stored in Database in correct format. the necessary Database links should be established properly</p> <p>The necessary update to the forum page and the Database should be made when the either delete or edits his thread.</p> <p>Start a Post: The post created by user is visible on the forum below the specific thread.</p>
Input Specifications	<p>Start a thread: The user is on the forum page. Selects the option to start a thread on forum. Enters valid content on the text-area and clicks start the thread.</p> <p>Edit: User view his thread and selects the option to edit</p>

	<p>Delete: User view his thread and selects the option to delete the thread.</p> <p>Make a post: The user is on forum page The user selects the option to make a post for a specific thread Enter the post specific data and clicks start the post.</p>
Expected Output Specifications	<p>Start a thread: Thread is posted on the forum, and thread specific data is stored in Database.</p> <p>Edit: The changes made to a specific thread are visible on the forum page, and thread specific data is updated in database.</p> <p>Delete: Thread is no longer visible on forum, and the data related to that thread is deleted from the Database.</p> <p>Start a post: Post is posted on the forum, and is visible below the specific thread.</p>
Pass/Fail Criteria	<p>Pass: The system correctly loads the forum page If the data entered to be posted does not violate any rules, the system post the thread.</p> <p>If user chooses to start a post, the post is posted on the forum.</p> <p>Fail: User tries to click start a thread without entering any data, the thread will not be visible on forum.</p>
Assumptions and Constraints	<p>Assumption: The thread data does not violate any rules</p> <p>Constraint: Connectivity between user's handheld device and server</p>

	must available and properly configured
Dependencies	N/A
Traceability	<p>WC_3240: As a student, I can start a thread by posting a question so that other users might answer that question with credible information.</p> <p>WC_3241: As a user, I can answer a question posted by other users.</p>

## 3.2 Test Identifier

TC02- Update user WAT point information

Test Case 02 covers the feature of the WAT Network, where the system award point to the user based on their interaction with the WAT point system. This test include test cases covering when the user earn points, when the points are reduced and updates to the three point system.

### 3.2.1 Test Level

Software item level

### 3.2.2 Test Class

Capability Test

### 3.2.3 Test Completion Criteria

If a user like/dislike a post the points for a specific users are updated by the system. The system sends a notification to the specific user when users points are updated.

### 3.2.4 Test Cases

**Table 3: TC-02-01 Update user WAT point information**

Test Case Number	TC-02-01
Test Item	This test case test the basic features of our system where the system award points to the user based on like/dislike and update all the three points for the user.
Test Priority	M (Must Have)
Pre-conditions	We need to have like/dislike option for each thread/post. For each user their is three points system like Lifetime/Usable/Semester.
Post-conditions	If a user like/dislike post/thread the system increase/decrease the point for the specific user respectively.

	<p>Based on the updates of points the changes are made to the Semester/Usable/Lifetime points.</p> <p>The new calculated points are visible to the user.</p> <p>A notification relating to the update is sent to the user.</p>
Input Specifications	A user like/dislike a post/thread.
Expected Output Specifications	Based on the user input the points for specific users are updated.
Pass/Fail Criteria	<p>Pass:</p> <p>When a user like someone post, the specific user usable and semester points are updated immediately.</p> <p>A notification relating to the update is sent to the specific user.</p> <p>Fail:</p> <p>The points are not updated based on user like/dislike of post/thread by the system.</p>
Assumptions and Constraints	Connectivity between user's handheld device and server must available and properly configured
Dependencies	TC-01-01: The system should work properly for the CR-1, this feature is highly dependent on CR-1.
Traceability	<p>WC_3260: As a user, my thread specific point balance cannot drop below zero so I won't be over-penalized for an inappropriate answer.</p> <p>WC_3252: As a user, I can track and compare my points with my peers' points so that I can gauge how I measure up to others.</p>

## 3.3 Test Identifier

TC03- User Role to use online redemption system

Test Case 03 covers the feature of the WAT Network, where the user interact with the web-interface using the handheld device. This test include test cases that cover user making use of their WAT points to redeem items or gift card from the online redemption system.

### 3.3.1 Test Level

Software item level

### 3.3.2 Test Class

Capability Test

### 3.3.3 Test Completion Criteria

User is able to redeem an item if user has enough WAT points to redeem a particular item.

On redemption the system decrease the user usable WAT points.

Notification about the redemption and WAT points update is sent to the specific user.

### 3.3.4 Test Cases

Table 4: TC-03-01 User Role to use online redemption system

Test Case Number	TC-03-01
Test Item	This test case test the basic features of our system where user make use of our online redemption system to redeem his WAT points.
Test Priority	M (Must Have)
Pre-conditions	The user is on the Redemption page. The user selects an item he wants to redeem using WAT points.

Post-conditions	<p>If the user has enough points to redeem the point, user is able to get the item.</p> <p>The system decrease the users usable point and notify the user about the redemption.</p> <p>The system decrease the quantity of the item redeemed by the user.</p> <p>Specific updates are made to the database.</p>
Input Specifications	<p>The user is on the Redemption page.</p> <p>The user selects an item he wants to redeem using WAT points.</p> <p>User clicks Redeem item to redeem the specific item.</p>
Expected Output Specifications	<p>If the user has enough points to redeem the point, user is able to get the item.</p> <p>The system decrease the users usable point and notify the user about the redemption.</p> <p>The system decrease the quantity of the item redeemed by the user.</p>
Pass/Fail Criteria	<p>Pass:</p> <p>If the user has enough points to redeem the point, user is able to get the item.</p> <p>The system decrease the users usable point and notify the user about the redemption.</p> <p>The system decrease the quantity of the item redeemed by the user.</p> <p>Specific notification is sent to user by the system.</p> <p>Fail:</p> <p>User tries to redeem an item without having enough WAT points. The system display an error message.</p>
Assumptions and Constraints	Connectivity between user's handheld device and server must available and properly configured
Dependencies	N/A
Traceability	<p>WC_3435: As a user I can use my WAT points to redeem items from a virtual store.</p> <p>WC_3245: As a user, I can redeem usable WAT points so that I can receive a gift card.</p>



## 4. Resources and schedule

### 4.1 Resources

Currently as per the clients requirements we are developing the system offline. Initial testing would involve generating random data set for testing specific features.

Once the clients approve the offline system, we will put the web-interface on the server for actual users. The data would be collected and testing will be performed based on the data obtained.

Data Set:

Initially, team will enter their own data set to test the system

Gather actual user information

Software:

Currently, we have identified software to test our UI mockup SeleniumHQ, PHPUnit, PHPSpec. We will further identify software to test our system for other functionalities that include server speed, database connectivity etc.

Budget: We will be using all free testing software for our system, as the budget for our project is \$0.

### 4.2 Staffing and Training Needs

All team members will perform testing, with Pittawat Pamornchanisirikij being the lead.

Training need: Tem needs to spend time to familiarize themselves with SeleniumHQ, PHPUnit, PHPSpec resources that will be used for testing.

### 4.3 Schedule

Table 5: Testing Schedule

Date	Test Identifier	Responsible person	Resources	Training needs
11/30/2014	TC-01-01, TC-02-01 and TC-03-01	Pittawat Pamornchaisirikij	Use the following for testing UI SeleniumHQ, PHPUnit, PHPSpec	N/A