# FPT UNIVERSITY

## CAPSTONE PROJECT DOCUMENT

## AGC101

### Report #5 – Software Test Documents

| Tra Da | |
|---|---|
| **Group Member** | SE03451 – Soulivong Phimmathong |
| | SE03271 – Hoang Van Canh |
| | SE03030 – Vu Hoang Hieu |
| | SE03193 – Nguyen Hoang |
| **Supervisor** | Nguyen Van Sang |
| **Project Code** | AGC101 |

Hanoi, May 2016 – August 2016

# Table of Contents

# 1 INTRODUCTION

## 1.1 Purposes

This document aims to describe testing strategies, methodologies, and activities used to perform software testing for AGC system. Also helps in estimating the testing effort required, test coverage, requirement tracking/tracing, etc.

## 1.2 System Overview

AGC is a web application aiming to connect customers and restaurants together. AGC will be available on modern web browsers which is also compatible for mobile and tablets devices. The site is reliable and convenient which help users to easily access, search for information about the most suitable restaurants, promotions, food and services provided. Besides, AGC provides information of restaurants addresses, food, and table booking service.

# 2 TEST PLAN

## 2.1 Scope of Testing

- ➢ Perform three levels of testing, unit testing, integration testing and system testing.
- ➢ Perform functional testing, user interface testing and data integrity testing.
- ➢ Test all functions defined in software requirement specification of AGC101, as well as others scenarios those not existed in SRS.

## 2.2  Requirements for Testing

### 2.2.1  Test Items

The following functions must be tested and also check their user interface

| Use case No | Group of function | Function | Description |
|---|---|---|---|
| **UC001** | Admin management | Activate/deactivate user | allow users who have account to use system or not |
| | | Activate/deactivate restaurant | allow restaurant information to display on AGC website or not |
| | | Rate restaurant | increase or decrease restaurant rating |
| | | Delete restaurant | remove all restaurant data |
| | | Delete user | delete account registered in AGC system |
| | | View statistical data | view total number of restaurant, table booking, etc. |
| | | Add/delete category | add new or delete category to be used by restaurant |
| | | Write/delete/update news | fully modify and publish news |
| | | View all users | view users' accounts |

| | | View all categories | view restaurant' categories those available on AGC system |
|---|---|---|---|
| | | View all restaurant | View restaurants' information to decide whether to activate or deactivate |
| | | View all news | Views all news those have been written in AGC system |
| | | View all booking | View all tables booking |
| | | View detail booking | Views each table booking in detail |
| **UC002** | Customer management | login | general function used for login to the system |
| | | logout | general function used for logout of the system |
| | | View restaurant information | view index page of the restaurant |
| | | Book restaurant | book tables in a restaurant |
| | | View booking history | view all his/her own tables booking details |
| | | View account | view information related to personnel information |
| | | Update password | change password |

| | | | |
|---|---|---|---|
| **UC003** | Restaurant owner management | Add/update/delete restaurant information | fully modify his/her own restaurant information |
| | | Add/update/delete restaurant banner | fully modify his/her own restaurant banner images |
| | | Set restaurant profile picture | choose image from banner images to use us restaurant profile picture |
| | | Add/update/delete food information | fully modify dishes/menu in his/her own restaurant |
| | | Update booking status | change table booking status for customers |
| **UC004** | Guest management | register | become a member or customer of AGG website |
| | | View general information | view all general information which is not reserved for only specific users |
| | | View restaurant information | view restaurant information in detail in restaurant index page |
| | | Search | general function used for search restaurant based on users keyword |
| | | Like | Like a restaurant using Facebook account |

| | | Comment | comment a restaurant using Facebook account |
| --- | --- | --- | --- |
| | | Share | share a restaurant using Facebook account |
| **Note: Admin and Restaurant owners are able to perform all functionalities those customers are able to, Customer can perform all functionalities those Guest is able to except register** | | | |

Figure 1: Testing items

## 2.2.2 Acceptance Test Criteria

AGC system is developed for thesis defence in FPT University. Therefore, there is no actual end users, acceptance testing will be omitted.

## 2.2.3 Test Completion Criteria

Testing processes will be completed if it meets all of the following criteria.

- ✓ Test coverage >= 90%
- ✓ 0 severity bug found, less than 10 bugs those are less harmful found.
- ✓ Passed unit test
- ✓ Passed integration test
- ✓ Accepted by the AGC advisor

However, testing processes will be stopped if the testing deadline has met.

## 2.2.4 Testing Risks

- ▪ There might be undiscovered bugs.
- ▪ Disclosed bugs those have not enough time to fix.
- ▪ Testing devices/environment might not enough to implement testing operation.
- ▪ Personnel work slower than expected.

## 2.3  Test Strategies

### 2.3.1  Test Policies

- ➢ Try to generate as many bugs as possible.
- ➢ Identify and document all bugs found.
- ➢ Test all possible scenarios.
- ➢ Test with real devices, no simulation.
- ➢ All AGC team members much involve in testing phased.
- ➢ Test much be involved in every stage of software development life cycle (SDLC).

### 2.3.2  Test Model (change v-model image to default)

AGC system used Verification and Validation model (V − model) to implement all testing process.
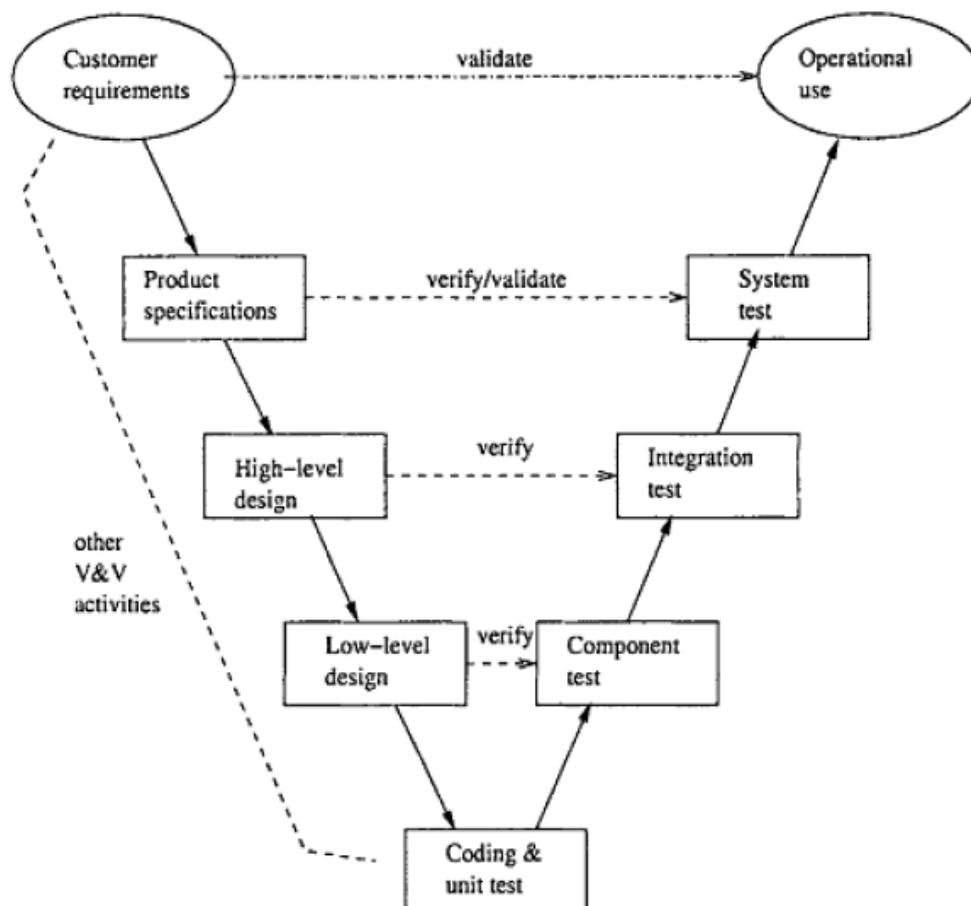


*Figure 2: V-model*

- **Unit Testing:** Unit tests designed in the module design phase are executed on the code during this validation phase. Tra Da members who will carry out this task consist of Soulivong Phimmathong, Hoang Van Canh, Vu Hoang Hieu, and Nguyen Hoang, which will use White Box testing technique.

- **Integration Testing:** Integration testing is associated with the architectural design phase. All Tra Da members will carry out this task and the techniques to be used are both black box and white box testing.

- **System Testing:** System testing is directly associated with the System design phase. System tests check the entire system functionality and the communication of the system under development with external systems. All Tra Da members will carry out this task.

- **Acceptance Testing:** Acceptance testing is associated with the business requirement analysis phase and involves testing the product in user environment. However, AGC system is develop for thesis defense in FPT University Therefore, there is no actual end users, as the result, acceptance testing will be omitted.

### 2.3.3 Types of Testing

There are 3 types of testing in AGC system: user interface, functional testing and data and data integrity testing.

#### 2.3.3.1 User Interface Testing

This type of testing is to check physical appearance of the whole system. Element to be tested will be everything that are visible to users specifically fonts, layouts, buttons, images, colours, textboxes, checkboxes, radio buttons, etc. This testing is applied to 3 type of devices: smart phone, tablet and computer.

| Test objective | To ensure every element in the system will be displayed as expected |
| --- | --- |
| Testing process | <ul><li>Testers will go through all screens then check all elements and overall looking and feel of pages</li><li>Enter both right and wrong data into the input fields to check notification and error messages</li><li>Every defect found will be recorded in Defect Log section in this documentation and send to development team to fix, then this testing will be performed again until it meets testing completion or stopping criteria</li></ul> |
| Technique | use white box testing, based on number of pages in the system |
| completion criteria | passed all test cases |

Table 1: User interface testing strategy

### 2.3.3.2 Functional testing

Functional testing is to check whether users can perform functions as needed or not, this test mainly focuses on the changes after perform specific functions as well as interaction between pages.

| Test objective | To ensure every function work correctly |
| --- | --- |
| Testing process | <ul><li>Testers will test all test cases specified in AGC software requirement specification</li></ul> |

| | |
|---|---|
| | • Enter both right and wrong data into the input fields to check whether function handle errors correctly or not<br>• Every defect found will be recorded in Defect Log section in this document and send to development team to fix, then this testing will be performed again until it meets completion or stopping criteria |
| **Technique** | use both black box and white box testing, based on use case specification |
| **Completion criteria** | passed all test cases |

Table 2: Functional testing strategy

### 2.3.3.3  Data and data integrity testing

This is Relational Database Management System (RDBMS) test which will check all data in the database, this test should be performed together with functional test since most of the processes has interaction with database. However, pure database test is also carried out to check data integrity.

| | |
|---|---|
| **Test objective** | To check data integrity and to ensure the system reads, inserts, updates, and deletes data in the database correctly |
| **Testing process** | • while performing system test, developer team will also check data in the database<br>• check data being displayed on AGC system if it matches with data in the database |

| | |
|---|---|
| | • check primary key, foreign key, null able values<br><br>• Every defect found will be recorded in Defect Log section in this document and send to development team to fix, then this testing will be performed again until it meets completion or stopping criteria |
| **Technique** | use RDBMS development environment tool |
| **Completion criteria** | passed all test cases |

Table 3: data and data integrity testing strategy

### 2.3.4 Test Stages

Common tests are carried out as in the table below:

| Type of Tests | Stage of Test | | |
|---|---|---|---|
| | **Unit** | **Integration** | **System** |
| Function Test | x | x | x |
| User Interface test | x | | x |
| Data and Data integrity test | x | | |

Table 4: common test stages

### 2.3.5  Tools

#### 2.3.5.1  Hardware

| Name | Purpose |
|---|---|
| Laptop computers | test all activities in laptop computers |
| Android phones | test if the system work correctly in android system, with responsive screen |
| iPad mini 2, iPad 4 | test if the system work correctly in iPad/tablet system, with responsive screens |
| iPhone 5 | test if the system work correctly in IOS system, with responsive screens |

Table 5: testing devices

#### 2.3.5.2  Software

| Name | version | Purpose |
|---|---|---|
| Microsoft word | 2013 | write test plan and keep tracking defects |
| Microsoft excel | 2013 | write test case, checklist, and test report |
| MySQL workbench | 6.3 CE | perform database testing |
| Google chrome | 52.0.2743.116 m | web browser testing |
| Firefox | 48.0 | web browser testing |
| Safari | 5.0.6 | Apple web browser testing |
| Coc Coc | 41.0.2272.119 | Vietnamese web browser testing |

Table 6: testing software

## 2.3.6  Resources

| Member | Position | Responsibilities |
|---|---|---|
| **Nguyen Hoang** | Test leader | o  creates and views overall testing plan<br>o  assigns and keep track of testing tasks<br>o  approve testing completion |
| **Nguyen Huu Thao** | Tester | o  creates and executes test cases<br>o  records defects in defect logs section in this documentation |
| **Vu Hoang Hieu** | Tester | o  creates and executes test cases<br>o  records defects in defect logs section in this documentation |
| **Hoang Van Canh** | Tester | o  creates and executes test cases<br>o  records defects in defect logs section in this documentation |
| **Soulivong Phimmathong** | Tester | o  creates and executes test cases |

| | | o  records defects in defect logs section in this documentation |
|---|---|---|

<div align="center">Table 7: human resources testing planning</div>

# 3  TEST CASES

See AGC_Test_Cases_V1.0.xlsx for detail.

# 4  CHECKLISTS

## 4.1  Functionalities checklists

| No. | Question | Yes | No | N/A |
|---|---|---|---|---|
| 1 | Do all links take users to the right pages or position? | | | |
| 2 | Have all form validation and submission tested? | | | |
| 3 | Have all searching and filtering functions tested? | | | |
| 4 | Have all Viewing function tested? | | | |
| 5 | Have others functions tested? | | | |

<div align="center">Table 8: Functionalities checklists</div>

## 4.2  User interface checklists

### 4.2.1  Colour

| No. | Question | Yes | No | N/A |
|---|---|---|---|---|
| 1 | Are hyperlink colours standard? | | | |
| 2 | Are the field backgrounds the correct colour? | | | |
| 3 | Are the field prompts the correct colour? | | | |
| 4 | Are the screen and field colours adjusted correctly for non-editable mode? | | | |
| 5 | Are all the buttons are in standard format and size | | | |
| 6 | Is the general screen background the correct colour? | | | |

Table 9: UI color checklists

### 4.2.2  Content

| No. | Question | Yes | No | N/A |
|---|---|---|---|---|
| 1 | Are contents displayed to the correct types of users? | | | |
| 2 | All fonts to be the same? | | | |
| 3 | Are all the screen prompts specified in the correct screen font? | | | |
| 4 | Does content remain if you need to go back to a previous page, or if you move forward to another new page? | | | |
| 5 | Is all text properly aligned? | | | |
| 6 | Is the text in all fields specified in the correct screen font? | | | |

| 7 | Is all the heading are left aligned? | | | |
|---|---|---|---|---|

Table 10: UI content checklists

### 4.2.3  Images

| No. | Question | Yes | No | N/A |
|---|---|---|---|---|
| 1 | Are all graphics properly aligned? | | | |
| 2 | Are graphics being used the most efficient use of file size? | | | |
| 3 | Banner style & size & display exact same as existing windows | | | |
| 4 | Does text wrap properly around pictures/graphics? | | | |
| 5 | Are font-awesome and glyphicon correctly displayed? | | | |

Table 11: UI images checklists

### 4.2.4  Instructions

| No. | Question | Yes | No | N/A |
|---|---|---|---|---|
| 1 | Is all the error message text spelt correctly on screens? | | | |
| 2 | Is all notification message text spelt correctly on screens? | | | |

Table 12: UI instruction checklists

### 4.2.5  Navigation

| No. | Question | Yes | No | N/A |
|---|---|---|---|---|
| 1 | Are all disabled fields avoided in the TAB sequence? | | | |

| 2 | Can all screens accessible via buttons from previous screen be accessed correctly? | | | |
|---|---|---|---|---|
| 3 | Does a scrollbar appear if required? | | | |
| 4 | Is there a link to home on every single page? | | | |

<div align="center">Table 13: UI navigation checklists</div>

### 4.2.6 Usability

| No. | Question | Yes | No | N/A |
|-----|----------|-----|----|----|
| 1 | Are some English text displayed understandable by Vietnamese people? | | | |
| 2 | Are fonts too large or too small to read? | | | |
| 3 | Are names in command button & option box names are not abbreviations. | | | |
| 4 | Assure that option boxes, option buttons, and command buttons are logically grouped together in clearly demarcated areas "Group Box" | | | |
| 5 | Can the typical user run the system without frustration? | | | |
| 6 | Do pages print legibly without cutting off text? | | | |
| 7 | Does AGC system convey a clear sense of its intended audience? | | | |
| 8 | Does the site have a consistent, clearly recognizable "look-&-feel"? | | | |

| 9 | Does AGC system correctly displayed on smartphone, iPad/tablet, and computer | | | |
|---|---|---|---|---|
| 10 | Is all terminology understandable for all of the AGC system's intended users? | | | |

<div align="center">Table 14: UI usability checklists</div>

# 5 TEST LOGS

## 5.1 Defect Logs

See AGC_Defect_Logs.xlsx for detail.

## 5.2 Test Reports

| Project Name | AGC | Creator | Nguyen Hoang |
|---|---|---|---|
| Project Code | AGC101 | Reviewer/Approver | Nguyen Van Sang |
| Document Code | AGC_Report5_STD | Issue Date | 22 Aug 2016 |

| No | Module code | Pass | Fail | Untested | N/A | Number of test cases |
|---|---|---|---|---|---|---|
| **Admin Management Module** | | | | | | |
| 1 | Login | 5 | 0 | 0 | 0 | 5 |

| 2 | write news | 9 | 0 | 0 | 0 | 9 |
|---|---|---|---|---|---|---|
| 3 | create category | 5 | 0 | 0 | 0 | 5 |
| 4 | update news | 2 | 0 | 0 | 0 | 2 |
| 5 | update category | 2 | 0 | 0 | 0 | 2 |
| 6 | delete news | 2 | 0 | 0 | 0 | 2 |
| 7 | delete category | 2 | 0 | 0 | 0 | 2 |
| 8 | activate/deactivate restaurant | 4 | 0 | 0 | 0 | 4 |
| 9 | delete restaurant | 2 | 0 | 0 | 0 | 2 |
| 10 | rate restaurant | 1 | 0 | 0 | 0 | 1 |
| 11 | delete user | 2 | 0 | 0 | 0 | 2 |
| 12 | activate/deactivate  user | 4 | 0 | 0 | 0 | 4 |
| 13 | view all news | 2 | 0 | 0 | 0 | 2 |
| 14 | view all categories | 2 | 0 | 0 | 0 | 2 |
| 15 | view all restaurants | 2 | 0 | 0 | 0 | 2 |
| 16 | view all users | 2 | 0 | 0 | 0 | 2 |
| 17 | view all booking | 1 | 0 | 0 | 0 | 1 |
| 18 | view detail booking | 1 | 0 | 0 | 0 | 1 |
| 19 | report statistics | 1 | 0 | 0 | 0 | 1 |
| 20 | Logout | 2 | 0 | 0 | 0 | 2 |

| | Customer Management Module | | | | | |
|---|---|---|---|---|---|---|
| 21 | login | 5 | 0 | 0 | 0 | 5 |
| 22 | Logout | 2 | 0 | 0 | 0 | 2 |
| 23 | View account | 2 | 0 | 0 | 0 | 8 |
| 24 | update account | 8 | 0 | 0 | 0 | 8 |
| 25 | update password | 6 | 0 | 0 | 0 | 6 |
| 26 | book restaurant | 5 | 0 | 0 | 0 | 5 |
| 27 | view booking history | 2 | 0 | 0 | 0 | 2 |
| 28 | update avatar | 3 | 0 | 0 | 0 | 3 |
| | Restaurant Owner Management Module | | | | | |
| 29 | Login | 5 | 0 | 0 | 0 | 5 |
| 30 | Logout | 2 | 0 | 0 | 0 | 2 |
| 31 | add restaurant | 8 | 0 | 0 | 0 | 8 |
| 32 | update restaurant | 1 | 0 | 0 | 0 | 1 |
| 33 | add restaurant banner | 2 | 0 | 0 | 0 | 2 |
| 34 | Set restaurant profile picture | 1 | 0 | 0 | 0 | 1 |
| 35 | delete restaurant banner | 1 | 0 | 0 | 0 | 1 |
| 36 | add food | 6 | 0 | 0 | 0 | 6 |
| 37 | update food | 1 | 0 | 0 | 0 | 1 |

| 38 | update booking status | 1 | 0 | 0 | 0 | 1 |
|----|-----------------------|---|---|---|---|---|
| **Guest Management Module** | | | | | | |
| 39 | Home screen | 10 | 0 | 0 | 0 | 10 |
| 40 | register | 15 | 0 | 0 | 0 | 15 |
| 41 | view news by category | 2 | 0 | 0 | 0 | 2 |
| 42 | view news detail | 1 | 0 | 0 | 0 | 1 |
| 43 | view user guide | 2 | 0 | 0 | 0 | 2 |
| 44 | view about as | 1 | 0 | 0 | 0 | 1 |
| 45 | restaurant by category | 1 | 0 | 0 | 0 | 1 |
| 46 | restaurant detail | 2 | 0 | 0 | 0 | 2 |
| 47 | social | 6 | 0 | 0 | 0 | 6 |
| 48 | search | 3 | 0 | 0 | 0 | 3 |
| Total | | 152 | 0 | 0 | 0 | 152 |
| Test coverage | | 100% | | | | |
| Test successful coverage | | 100% | | | | |

# 6 References

1. *http://www.tutorialspoint.com/sdlc/sdlc_v_model.htm*
2. *www.sqaforums.com/attachments/437212-WebApplicationTesting.doc*
3. *AGC_Test_Cases_V1.0.xlsx*
4. *AGC_Defect_Logs_V1.0.xlsx*