# QUALITY MANAGEMENT

## Quality Management Overview

### Organization, Responsibilities, and Interfaces

|  |  |  |
| --- | --- | --- |
| **Name** | **Role** | **Responsibilities** |
| Tran Binh Duong | Supervisor | * Helps define product quality expectations. * Determines final acceptance of product’s quality. |
| Le Van Quy Hoang | PM | * Creates quality plan * Facilitates resolution of quality issues, escalating as needed |
| Nguyen Thi Hong Nhung | Test leader | - Provides test and test management |
| Team members:   * Le Van Quy Hoang * Tran Dinh Hoang Huy * Nguyen Van Quyen * Nguyen Khac Hoang * Kieu Cao Khanh | Developer | * Provides feedback on quality plan, help determine metrics and criteria for this project * Be a part of quality reviews and provide feedback on deliverables |

### Tools, Environment, and Interfaces

|  |  |
| --- | --- |
| **Tool** | **Description** |
| **Cause-and-effect diagram** | Used to analyze the causes and effect of a problem. Used to find the root cause problem when their are a complaint about quality problems. |
| **Control chart** | Used of control charts is to prevent defects. Apply Seven Run Rule. |
| **Flowchart** | Used to analyze how problems occur and how processes can be improved |

## Quality Planning

### Define Project Quality

* ***System output:***+ A Web application to support admin manages entity ship system.

***+*** A Web applicationto support store manages own orders fast and convenient.

+ An Android application to support shipper’s business (find the best way of transfer, grad order...)

* ***Functionality:***

+ Web application:

* + - For Admin: allow admin to login, add/ block/ update shipper or orders and analyze and export reports.
    - For Store: allow login to system, create order, tracking state of shipper, feedback for each shipper and view reports.

+ Android application has main functions: The system can find best way to go to target location, shipper can pick order and view report about delivered history.

* ***Performance:***

+ Time delay for find shipper who nearest with place of delivery is less than 10s.

+ Time delay on web application for tracking state of shipper on map is less than 5m.

+ Server can handle least 2000 clients concurrently.

+ Other functions of server perform well while have many order running on application.

* ***Reliability:***

+ The application is available 24/7.

+ Find the way to go to places of receive and place of delivery are at least 90%.

* ***Maintainability:***

+ Web application is easily to maintain without any crashes. Source code is readability, organized into groups of skeleton (modules) and complies with coding convention.

+Android application is easily to be updated and synchronize with Web application without any crashes. Source code is readability, complies with coding convention.

+ System’s architecture has to be design to be easy to extend.

* ***Security:***

+ Information of admin/ root admin / data on server is secured.

+ Information of store and shipper on server is secured.

### 1.2.2 Measure Project Quality

|  |  |
| --- | --- |
| **Metric** | **Goal** |
| Accurate | * Find the way to go to places of receive and place of delivery are at least 90%. * Display the current position of shipper on map exactly. |
| Response of web application | * Time delay for find shipper <= 10s |
| Bugs/Lines of Code | * UT: 8 – 9 bugs / KLOC * ST: 2 – 4 bugs / KLOC   *(based on Fsoft norms)* |
| Maximum deep of loops | <= 4 |
| Android Program Size | <= 100 MB |
| Algorithm complexity | <= O(n^2) |
| Android version support | Support Android version 4.4.2 Kitkat to 5.0.1 Lollipop |
| Android screen support | Multiscreen  Must be tested on 320x480, 768x1024, 768x1336, 1080x1920 screens |
| Website support browser | Support Chrome version 41.0.xxx, Firefox version 36.0 or later. |

## Quality Assurance

### 1.3.1 Analyze Project Quality

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestone** | **Deliverables** | **Goal** | **Review and Approved** |
| TODO | Interface design ver1.0 | * Good looking & easy-to-use. * Cover all functions specified in SRS. | TODO |
| TODO | Software architecture design ver1.0 | Design to be easy to extend. | Supervisor |
| TODO | Web application | * Information of admins / root admins and shopkeeper on server are secured. * Provide information of order and push notification for mobile application. * Time delay for searching a order <=3s. | TODO |
| TODO | Mobile application | - Tracking road and received order.  - Notify to system when have issue on delivered process. | TODO |
| TODO | Integration test report | 30 – 34 test cases / KLOC  2 – 4 bugs / KLOC | NhungNTH |
| TODO | System test report | 30 – 34 test cases / KLOC  2 – 4 bugs / KLOC | NhungNTH |

### 1.3.2. Improve Project Quality

|  |  |
| --- | --- |
| **Issue** | **Action** |
| Difficult to track project’s progress | * Submit weekly report to supervisor * Meeting up team member everyday (work 6 days / week). Sometime extending the last days to keep deadline and fix bugs. * Using kanbanflow tool to track team members’ work ([www.kanbanflow](http://www.kanbanflow)) * Using GIThub to manage source of project (documents, source code) |
| Coding application does not match with User Requirement | * Team has to define clearly requirement specification and Software Architecture Design. * Develop Team must study about the document and comply with the content of document. * Any changes in process have to approve of PM |
| Maintainability | * Specify coding conventions document * Spend a lot of time to research architecture design. Then, decide the most appropriate architecture for maintaining easily. |
| Low quality code | * Create and execute types of test (unit test, integration test, system test). * Peer review, peer coding among developers. * Using Open Source and Framework to improve the quality code: NodeJS, AngularJS, Ionic framework. * Developers have to comply with the coding convention document. |
| Technology | * Predict and list the problems of technologies that are the bottleneck of project. Then, organize research to find solution from the beginning. * Have knowledge-training schedule for members. |
| Reward and discipline | * Teambuilding to increase communication ability between project’s members * Have punishment rules when:   + Member comes late or Miss meeting  + Make mistake in member document writing. Reviewed by Supervisor.  + Submit terrible code (which causes to re-coding more than 10%)  + Miss deadline |
| Acceptance of users | Do survey to discovery what features user want from this projects. Do it before design progress. |

## Quality Control

|  |  |  |  |
| --- | --- | --- | --- |
| **Deliverables** | **Goal** | **Quality control activity** | **Frequency /**  **Interval** |
| Interface design | * Good looking & easy-to-use: don’t using many button, don’t many minor detail. * Cover all functions specified in SRS | Designer has to:   * Ask for advice of some other designer * Take comment from friends on completed work * Have approve from SRS leader | Each time design a new screen |
| Software architecture design | Design to be easy to extend | Have review and judgment from Supervisor | On completion |
| Web application | * Information of admins / root admins and shopkeeper on server are secured. * Provide information of order and push notification for mobile application. * Time delay for searching a order <=3s. | Testers execute security test and system test | On completion |
| Mobile application | - Tracking road and received order.  - Notify to system when have issue on delivered process. | Testers execute system test | On completion |
| Integration test report | 30 – 34 test cases / KLOC  2 – 4 bugs / KLOC | PM requires testers to report on work | Weekly |
| System test report | 30 – 34 test cases / KLOC  2 – 4 bugs / KLOC | PM requires testers to report on work | Weekly |

## . Action Plan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **What** | **Who** | **When** | **How** | **Output** |
| Define coding convention | HoangLVQ,HuyTDH | 25/09/2015 | Read standard coding convention of NodeJS  Extract and modify to reuse it | Coding Convention  *(10.1 in this document)* |
| Training kanbanflow | All team | 16/09/2015  to 17/09/2015 | Hoang LVQ guides team members how to use kanbanflow |  |
| Market research | All team | 29/08/2015  to 13/09/2015 | Do survey to find out what functions users need | Survey summary |
| Training NodeJS,  AngularJS, | All team | 14/09/2015  to 20/09/2015 | * Read AngularJS documents * Find out how to set up and complete ‘Hello World’ tutorial. * Try some core class and functions for image processing. |  |
| Research for Ionic frame work | HuyTDH  QuyenNV | 7/09/2015  to 20/09/2015 | - Read Ionic frame work document.  - Routing in ionic frame work  - Find how to setup Ionic using plugin of cordova: geolocation, device. |  |
| Training Unit Test | [TODO] | [TODO] | Training how to create and perform Unit Test |  |