

RAILWAY INFORMATION SYSTEM

CENG 492

Computer Engineering Design

Test Specification Report

BUGBUSTERS

26.04.2010



Burkay SUCU

Caner ÇAKMAK

Halim Çağrı ATEŞ

Işıl Özge PEKEL

Contents

- 1. INTRODUCTION 3**
 - 1.1. Goals & Objectives..... 3
 - 1.2. Scope of the Document..... 3
 - 1.3. Major Constraints..... 3
 - 1.3.1. Time 3
 - 1.3.2. Staff..... 4
 - 1.3.3. Hardware 4
 - 1.3.4. Software 4
- 2. TESTING PLAN 4**
 - 2.1. Overview..... 4
 - 2.2. Testing Approach..... 4
 - 2.2.1. Unit Testing..... 4
 - 2.2.1.1. User Module 4
 - 2.2.1.2. Best Path Search Module 5
 - 2.2.1.3. Train Tracking Module..... 5
 - 2.2.2. Integration Testing 5
 - 2.2.3. Device Testing..... 5
 - 2.2.3.1. GPS Device Testing 5
 - 2.2.3.2. Accelerometer Device Testing..... 5
 - 2.2.4. Performance Testing 6
 - 2.2.5. Stress Testing..... 6
- 3. TESTING TOOLS 6**
 - 3.1. Netbeans IDE 6
 - 3.2. JUnit..... 6
 - 3.3. Nagios 6
- 4. RISK & CONTINGENCIES 7**
- 5. TESTING SCHEDULE..... 7**
- REFERENCES..... 8**

1. INTRODUCTION

Turkish State Railways (TCDD) recently introduced some high speed trains, namely "Yüksek Hızlı Tren" (YHT). Perhaps TCDD does not use the most modern railway systems but tries to improve the on-use train systems and while doing this some other kind of improvements on trains become desired. Introducing YHT is a big step and our team wants to contribute the development process. Of course this contribution is not designing or making new trains; but as mentioned earlier, improving the service quality on currently used trains for passengers and TCDD employees (especially locomotive drivers).

In Railway Information System ^[1] project, each member of our team is working on what is assigned to him/her. As soon as a member develops a new feature, he/she immediately test it, ensure that the new feature operates correctly, and then commit the part to SVN. On the other hand, if a member discovers a problem, he/she opens a ticket on Track and assigns it to the related member. So SVN and Track are the core of project testing.

1.1. Goals & Objectives

The purpose of this document is to identify and describe the testing process for *Railway Information System*. The intended audience for this document is BugBusters Team members and METU assistants and professors.

What we will consider are:

- Unit testing : modules operating correctly
- Integration testing : modules interacting correctly
- Device testing : GPS and accelerometers operates correctly
- Performance testing : System uses reasonable amount of resources and responses in reasonable amount of time (free of system freezes)
- Stress testing : System not crashes under high load

1.2. Scope of the Document

The scope of this document is to clarify steps of the testing process of the Railway Information Systems Project.

As a part of this report, what will be tested, the test constraints, methods to handle bugs, testing tools to be used, relation between testing procedure and our project schedule, and team members-testing tasks assignments will be discussed in detail.

1.3. Major Constraints

1.3.1. Time

Since we have very limited time, we have to both develop and test the project at the same time, so we have to be very fast in order to meet the deadlines.

1.3.2. Staff

Similar to time constraint, we are only four people; the human resource is also very limited. So the responsibility per member is very high. Everyone is responsible for his/her tasks for the project and it seems very difficult that these tasks are checked by another member.

1.3.3. Hardware

GPS and accelerometer devices have to be tested whether or not they operate correctly. Since we do not have an actual GPS device, we have to test the simulator accordingly.

1.3.4. Software

Server side of the project is easier to test because everything is on our computers. But user side is unknown, which browser and operating system does the client use, whether or not JavaScript enabled, etc. So the client side of the project has to be tested on different platforms and we will spend much time on it.

2. TESTING PLAN

2.1. Overview

Considering the limited time and human resources, the testing process has to be very well planned. The testing approach and strategies will be discussed in the following section.

2.2. Testing Approach

2.2.1. Unit Testing

2.2.1.1. User Module

New users may register to, existing users may login to or logout from system through user module's registration and login sub-modules respectively. There are three types of users: passengers, administrators and machinists. Every user type has its own access levels, access to different kind of sources are granted or denied considering the user type (authorization). Also users may change their profile information through update sub-module.

Testing process includes new user registrations (information field tests, such as e-mail address or phone number validation), existing user login, non-existing user login attempt, profile update, error message presentations, etc.

Future tests include validation of project web page using Nagios ^[2]. Nagios is capable of validating a web-sites SMTP, HTTP and DNS network layers are up or down.

2.2.1.2. Best Path Search Module

Best path search module presents an interface to users to search for possible optimum paths according to user choices on some parameters such as maximum amount of money or maximum time to wait.

Testing of this module is done manually by comparing the search results with database records. If there exist a search result which does not fit into user choices or if there exists a database record fitting into user choices but not exist on search results considered as bugs and fixed.

2.2.1.3. Train Tracking Module

Train tracking module enables users to follow train positions on maps. Tracking is accomplished by using GPS device. Since we do not have an actual GPS device, we implemented a simulator. Related tests are mentioned in Device Testing section.

2.2.2. Integration Testing

We develop our project module by module. So at the end, these modules have to be integrated and the integration has to be tested to verify the compatibility. These integrations includes tracking map – browser integration, accelerometer – GPS – machinist module integrations, best path search module – user interface module integration, etc. Unfortunately, modules may become not working after integration although they are robust while they operate separately. But we need the project as a collection of integrated modules which operates successfully. So finding the integration problems and fixing them are the part of integration testing.

2.2.3. Device Testing

Basically, we use two separate hardware devices. They also need careful testing.

2.2.3.1. GPS Device Testing

Our project requires GPS devices but we do not have actual GPS hardware. So we implemented a simulator of it.

Simulator may contain some bugs which are very hard to find because tracking may take hours. It may seem working for a while but the position calculation may crash after a while. So GPS simulator testing requires extra care.

GPS positions also have to be tested to be on the railways but not outside. On the other hand, two trains' positions should not seem to be on each other, meaning a crash.

2.2.3.2. Accelerometer Device Testing

We intend to use accelerometer device to measure the horizontal angle of trains and prevent the train from falling over (banking to one side).

Accelerometer device gives us the acceleration on three coordinates. We have to calculate the horizontal angle somehow by using the acceleration values. The conversion process has to be carefully handled and is open to include many bugs. Testing the correctness of operation will be done manually by changing the device position and read the angle values from computer.

2.2.4. Performance Testing

Our project's all modules have to response in a reasonable amount of time.

User module performance has to be very well in order not to keep our clients waiting in front of the monitor and not to make them bored.

Machinist module performance is more important because the GPS and accelerometer device readings may be crucial. If the position calculation delays a bit, wrong speed limits may be showed to machinist and some undesirable consequences may occur.

2.2.5. Stress Testing

Since we intend to implement a public service, we have to consider the case many users are connected to our system. We have to ensure that the system will not crash under high stress, namely when many clients use our system simultaneously.

3. TESTING TOOLS

3.1. Netbeans IDE

NetBeans^[3] is decided to be the used as the IDE to develop the project. Also the NetBeans SVN plug-in will be used to connect central SVN repository and integrate the individual work done by the members to the main project. Its debugging and time measuring features are used for testing purposes.

3.2. JUnit

JUnit is used for its unit testing capabilities of Java programs.

3.3. Nagios

Although Nagios cannot be described as a testing tool, it enables to see whether or not the SMTP, DNS and HTTP layers are successfully operating or not. We will use it to test our project web-site services, and keep using it afterwards to monitor these services. Whenever a layer is down, a notification will be send to the administrator.

4. RISK & CONTINGENCIES

The biggest risk is to miss a bug during testing process and release the project with the bug. As having the title BugBusters, we intend to release our project bug free. Encountering a bug before the release forces us to fix it until release, but after a release we may announce a newer version of the project.

5. TESTING SCHEDULE

Similar to project schedule, we prepared a testing schedule. We have to strictly obey it to meet the deadlines. Testing procedure of each module immediately starts when the implementation of that module is finished. Our testing schedule is as follows:

TASK	RESPONSIBLE MEMBER	START DATE	END DATE	COMPLETED
Unit testing	Caner ÇAKMAK İşil Özge PEKEL	17 March	23 May	70%
User module		17 March	7 April	100%
Best path module		25 March	15 May	90%
Train tracking module		1 April	30 May	50%
Integration	H. Çağrı ATEŞ Burkay SUCU	21 April	30 May	60%
Device	H. Çağrı ATEŞ Burkay SUCU	15 April	20 May	60%
GPS device		15 April	30 April	80%
Accelerometer device		20 April	15 May	10%
Performance	Caner ÇAKMAK İşil Özge PEKEL	20 April	30 May	10%
Stress	All members	23 April	23 May	15%

REFERENCES

[1] *Railway information system*. (2010). Retrieved from

<http://senior.ceng.metu.edu.tr/2010/bugbusters/>

[2] *Nagios - the industry standard in it infrastructure monitoring*. (2010). Retrieved from

<http://www.nagios.org/>

[3] *Welcome to netbeans*. (2010). Retrieved from <http://netbeans.org/>