

D'oh!

Test Specification Report



D'OH! PICTURES PRESENTS IN ASSOCIATION WITH **DOSTLUGUN BOYLESI MOVIES** A **NOTHER** PRODUCTION A **RONTGEN**
FILM "TEST SPECIFICATION REPORT" **BUGRA OKTAY GİZEM KILIÇ ÖZKAN AKMAN GÜVEN İŞCAN** CASTING BY
GİZEM KILIÇ EDITED BY **GÜVEN İŞCAN** MUSIC BY **ÖZKAN AKMAN** DIRECTED BY **BUGRA OKTAY**

IN THEATRES ON MAY 31, 2009

TABLE OF CONTENTS

1 INTRODUCTION	1
1.1 Purpose of the Report	1
2. STRATEGIES	
2.1. Unit Testing	1
2.2. Integration Testing	1
2.3. System Testing	2
2.4. Performance Testing	2
3. PROCESS	
3.1. Unit Testing	
3.1.1 TheDevice	2
3.1.2 Bridge it Jones	3
3.1.2.1 GUI Unit	3
3.1.2.2 Bluetooth Unit	3
3.1.2.3 GPRS Unit	4
3.1.2.4 G-Unit	4
3.1.3 The Webber	4
3.1.3.1 Database Unit	4
3.1.3.2 Security Unit	5
3.1.3.3 GUI Unit	6
3.2. Integration Testing	7
3.3. System Testing	7
3.4. Performance Testing	7
4. TESTING TOOLS	
5. REFERENCES	

1. INTRODUCTION

The purpose of this report is to indicate the approach we are going to implement while testing our project, to achieve a final product. Testing is an important part of software developing, it has an additional importance on our project since health of patients are on the nail. Briefly speaking about the project, it consists of three parts, namely TheDevice (embedded part), Bridge it, Jones (mobile application part), The Webber (Database-Web part). We are going to follow complete different ways of testing each of them, as their development follows different ways. Our purpose is to minimize the number of bugs in the whole system, building a compact, reliable way for the users.

2. STRATEGIES

This section explains the strategies that will be used for testing BePatient Remote Patient Monitoring System. The strategies can be listed (starting from the most specific to most general) as follows:

- Unit Testing
- Integration Testing
- System Testing
- Performance Testing

Detailed descriptions of these testing strategies are given below.

2.1 Unit Testing

It is the testing of the each individual component of the project separately where we may gain confidence that individual units of source code are fit for use and find out errors arising from these units. This testing should be handled before the integration testing. It should be certain that errors arising in integration testing are not about individual components. Moreover, there will be one scenario to test each component. It is a white-box type of testing.

2.2 Integration Testing

Integration testing is the activity of software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before system testing.

Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

The purpose of integration testing is to verify functional, performance and reliability requirements placed on major design items. Test cases are constructed to test that all components within design items interact correctly, for example across procedure calls or process activations, and this is done after unit testing.

2.3 System Testing

System testing is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic. It occurs after Integration Testing.

2.4 Performance Testing

Performance Testing is the testing conducted to evaluate the compliance of a system or component with specified performance requirements. In this part, our aim is to test our system for detecting its limits when it comes to speed (of data flow) and overall performance.

3. PROCESS

3.1 Unit Testing

3.1.1 The Device

This part contains the purpose of testing the device and the results. As you know, the device gets data from patients' body and sends it to the mobile phone via Bluetooth. We have only two sensors and we simulate other three sensors. For the sensor data we have to determine correctness of sensor data. We know that sensors are electronic devices and they might have some problems while measuring the data or they may be some problem while getting from the sensors. Beside the data which we get from the sensor, we simulate some data just because we don't have sensors. For that data we have to determine reliability of simulated data. We are simulating the data so it must be very close to real sensor values.

The purposes of this unit are:

- Getting data from the sensors
- Simulating sensor data
- Transfer data via Bluetooth

The device Unit testing goals are:

- Determining the correctness of sensor data
- Determining the reliability of simulated data

3.1.2 Bridge it, Jones

In this part, the testing process related to the mobile application part of BePatient (Jones) will be explained in detail.

3.1.2.1 GUI Unit

The GUI Unit includes the front-end part of Jones. The main screen, options screen, command buttons and splash screen form the entire GUI Unit. User interface related code is written with modularity and as a part of the MIDlet application and this makes it easy to shut down the functionality behind while testing the GUI Unit.

The purposes of this unit are:

- Serving as an intermediate layer between the user and low level functions
- Making it possible for even a non-technical user to configure the application easily

GUI Unit testing goals are:

- Detecting bugs related to GUI Design
- Providing error-free interface for the user

3.1.2.2 Bluetooth Unit

Bluetooth Unit is composed of everything Bluetooth: Device Discovery, Device Matching, Input / Output Streaming and Authorization. Bluetooth related functions are normally called using GUI. However, for testing purposes a standalone testing application (actually two: one for the server simulator and one for the client) will be designed and all of the functionality will be tested over there.

The purposes of this unit are:

- Making Bluetooth device discovery
- Establishing Bluetooth connection between devices
- Handling Bluetooth authorization problems
- Controlling data flow between devices over streams

Bluetooth Unit testing goals are:

- Detecting coding errors related to Bluetooth Unit
- Ensuring the transfer of data from TheDevice to the mobile phone completely
- Providing correct device matching each and every time
- Keeping user away from dealing with authorization related issues

3.1.2.3 GPRS Unit

GPRS Unit is where GPRS connection is established and data received over Bluetooth is transmitted to the web. Just like the Bluetooth Unit, GPRS Unit testing will be carried out with the help of a standalone testing application.

The purposes of this unit are:

- Establishing HTTP Connection with the web server/website over GPRS
- Handling GPRS authorization problems
- Controlling data flow
- Warning the user when GPRS Service Provider is unavailable

GPRS Unit testing goals are:

- Detecting coding errors related to GPRS Unit
- Ensuring synchronous data transfer from the mobile phone to the web
- Keeping user away from dealing with authorization related issues

3.1.2.4 G-Unit

G-Unit (Garbage Unit) is the part of the code which deals with the dirty business.

It is expected from this unit to:

- make the necessary calculations
- store and access the local files (emergency numbers, logs, images etc.)
- send emergency SMSs
- play an emergency voice record
- keep the logs
- parse information (when needed)

G-Unit testing goals are:

- Ensuring error-free calculations and data manipulation
- Providing a flawless emergency situation handling

3.1.3 THE WEBBER

In this part, the testing process related to the web application part will be explained in detail.

3.1.3.1 Database Unit:

Database part includes the database connection of the web site and the operations done in the tables which lay in the database such as querying, insertion, deletion, update. Database

part is the functional unit of The Webber, since all the data operations are carried through it and web site is just a way to visualize it.

The purposes of this unit are:

- Holding the main information about all parts of the user.

- *these parts are, namely ; doctor, relative, and the patient

- *the main information consists of the user info (name, surname, tel no, e-mail address etc.), patient health variables which come from the sensors

- *the doctor's main information

- *all the things that are needed in the sign up and login phases

- Enabling the reachability of a values or information in the web site

- Insertion, deletion and editing of the values.

The goals for the database unit testing are:

- Catching the errors that can take place because of the lacking points in connection, and if found fixing it.

- Organizing the tables in an efficient way to hold less place, and reach maximum speed. Since we are dealing with the values which arrive in a little time periods, and these values should be organized.

- Defining the security holes reasoning by the database connection (i.e. direct data access) and if there is any, fixing it.

3.1.3.2 Security Unit:

Security is one of the most important things in a web site. Users share some of their information in log up section, or during the time that they spend in the web site. If a site is not secure enough, it can cause undesired share of information.

The WEBBER part of the project prepared by considering

the TOP 10 SECURITY TIPS [1]

- 1 Define and assess the security risks

- 2 Take a holistic view

- 3 Don't trust anyone else's data (or your own)

- 4 Enforce review and approval at each milestone

- 5 Help the development team code securely

- 6 Integrate security into the testing programme

- 7 Build in audit, logging and alerting

- 8 Deploy the website securely

- 9 Include security in every contract and service level agreement

- 10 Consider disaster recovery (and business continuity)

and the TOP 10 SECURITY ISSUES [2] articles

- 1 Validation of input and output data
- 2 Direct data access (and theft)
- 3 Data poisoning
- 4 Malicious file execution
- 5 Authentication and session management
- 6 System architecture and configuration
- 7 Phishing
- 8 Denial of service
- 9 System information leakage
- 10 Error handling

In our project, we try to compose the maximum security by applying some of these tips which are available for us.

the goals for the security unit testing are:

- Testing the session management and preventing the data access of the users to the places which they don't have permission (for instance, preventing the access of the relatives to the doctor session)
- Testing the direct data access possibility, and if any, preventing that.
- Testing the validation of input and output data. If there is any missing data, placing that.

3.1.3.3 Design Unit

The design unit consists of the visual parts and the functionality of the webpage. So, this unit is more important for the users.

The goals for the design unit testing are:

- Page/Element Testing: Testing the pages, elements, links and approving their correctness.
- Template Testing: Homepage template and the special pages template testings.

3.2 Integration Testing

Our project consists of different modules, so integrating testing has crucial importance in this project. In integrating testing first we test the Bluetooth connection between the device and the bridge it Jones. We tested the correctness of the data. Second part of the unit testing is testing the GPRS connection between bridge it Jones and Webber. We controlled data which come from bridge it Jones. And finally we test the whole system which includes both Bluetooth and GPRS connection.

3.3 System Testing

System testing is done to ensure if the final product meets the functional requirements. In project definition, functional requirements are mainly defined as transferring data from one point to another. Thus our aim is to monitor the measurements in the web site.

3.4 Performance Testing

The data we are carrying all the way from sensors to web server is vital. Hence constructing a reliable way must be dealt carefully. We logged time of reception in mobile phone and controlled their intervals with the measurement times to ensure correctness in timing. We did the same procedure between the mobile application and web server parts. Even with small periods between measurements there were no problems transferring data between systems.

4. TESTING TOOLS

We are going to utilize the same tools we used for developing to test our product.

-Pic C Compiler

-NetBeans

-Visual Studio

5. REFERENCES

- http://en.wikipedia.org/wiki/Unit_testing
- http://en.wikipedia.org/wiki/Integration_testing
- *IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries; IEEE; New York, NY.; 1990.*
- <http://www.watsonhall.com/methodology/top10-website-design-security-tips.pl>
- <http://www.watsonhall.com/methodology/top10-website-security-issues.pl>