GroupBarkod

Mobile Medical Management

Requirement Analysis Report

Umut Kiremitçi
Ferdi Gündüz
Ali Mert Ertuğrul
Tolgahan Hancı
1 Contents

2 Introduction ........................................................................................................................................ 3

2.1 Background Information ............................................................................................................. 3

2.2 Description and Scope ............................................................................................................... 3

2.2.1 General Use ......................................................................................................................... 4

2.2.2 Member Use ......................................................................................................................... 4

2.3 Team Summary .......................................................................................................................... 4

3 Research ........................................................................................................................................ 5

3.1 Marketing Research .................................................................................................................... 5

3.1.1 Online Appointment System ............................................................................................... 5

3.1.2 Online Test Results .............................................................................................................. 6

3.1.3 Sponsor ................................................................................................................................. 7

3.2 Technology Research .................................................................................................................. 7

3.2.1 Development Platforms ...................................................................................................... 7

3.2.2 Programming Languages .................................................................................................... 10

3.2.3 Database Servers ................................................................................................................ 10

4 Requirements ................................................................................................................................ 10

4.1 Functional Requirements ......................................................................................................... 10

4.1.1 Data Flow and Interfaces .................................................................................................... 10

4.1.2 Use Cases ........................................................................................................................... 16

4.2 Non-Functional Requirements .................................................................................................. 22

4.2.1 Runtime Requirements ....................................................................................................... 22

4.2.2 Development Requirements ............................................................................................... 23

4.3 System Requirements ................................................................................................................. 23

4.3.1 Client Side Requirements .................................................................................................. 23

4.3.2 Server Side Requirements ................................................................................................. 23

5 Project Schedule............................................................................................................................... 24

6 Conclusion...................................................................................................................................... 25
INTRODUCTION

This report is a requirement analysis report of the project Mobile Medical Management (3M). The following a few pages contain a background information and a detailed definition and introduction of the report. Firstly background information will be stated and then a wide definition and scope of the project will take place.

2.1 Background Information

Today health care of a person is the most important issue. People spend lots of money, time and effort for their health problems. Especially in the developing countries like Turkey people are having big problems to get over their health problems. For example waiting to be examined by a doctor in hospitals is a very big problem for patients. Sometimes it takes two or three hours for a patient to be examined and also sometimes patients need to wait till afternoon since doctors take their noon breaks. Moreover, a patient can not know how many patients are waiting to be examined before going to hospital. So, it does not matter that a patient should go to the hospital early in the morning because there may not be too many people.

With the rapid development of information era, the number of technological innovations especially in the cell phone area offer lots of beneficial alternatives to their customers. Therefore people are getting used to take advantage of their mobile devices for their daily experiences. Therefore people may handle with their all daily and long term problems easily.

So with this project the aim is to handle this important issue by using the most used technological device cell phones. 3M is going to be designed so that it will let patients know which hospital is the most suitable for them concerning the time and distance. As a second feature 3M will help patients with a membership by reminding their appointments who have to see their doctors frequently like having dental problems, kidney sicknesses etc. Also as a third feature sending members results of some tests like blood, urine or sperm tests is planned.

2.2 Description and Scope

The project aims to offer solutions for the stated problems at the background part above. Group Barkod is going to develop a database dependent, SMS connected, shortest pathfinder system in minimum time. Secondly a membership system will be implemented that reminds patients their appointments and sends patients test results. The main goal of this project is to make sick people's hospital lives easier. The system will enable their users to know which hospital has the least patients in the waiting queue before going to the hospital and to learn the test results so that they may show the results to another doctors/hospitals. Additionally
the system is going to show the shortest path to the hospital from the current location of the user. Also the system will remind the appointments of the patients. And the system is going to achieve this by using geological information of the user that the GSM operator provides and triangulation which comes from the main hospital database.

This project aims to create an algorithm which will rely on two main concepts that are time and distance. By looking at the priority of the patient the system will reply by advising the most suitable one in the database. This database that includes hospitals, positions of the hospitals, number of patients in these hospitals' polyclinics, estimated number of patients that will come up until the user reaches to the hospital, estimated examination time of a patient with respect to the department is also going to be created by the team. Of course the system waits which is more important for the patient the closest one or the least crowded one. The system is going to take this information also with SMS. With a second database the members data will be stored. With this database members' appointments information and the requested test results will be sent to the users.

The project will provide the users with following advantages:

2.2.1 General Use
• Learn the most suitable hospital: the closest and has the least patients in the waiting queue.
• See the shortest map to the hospital by using the 3G technology.

2.2.2 Member Use
• Be informed with the news related to the hospital like opening of new polyclinic increase or decrease in the number of doctors.
• Get the results of the tests.
• Be reminded for the appointments.

2.3 Team Summary

Group Barkod is composed of four senior Computer Engineering students. The members of the team are Umut Kiremitçi, Tolgahan Hancı, Ferdi Gündüz and Ali Mert Ertuğrul which aim to create an efficient system that help people and make patients lives easier.

➔ Umut Kiremitçi : Initiator, Optimist
➔ Tolgahan Hancı : Devil's Advocate, Time Keeper
3 Research

3.1 Marketing Research

3.1.1 Online Appointment System

There is such a system already in use in Turkey. People can also have appointments by using their cell phones. But this system is only to give appointments related to the specific hospitals. In other words people can not choose which hospital they should go. But this system is not efficient for daily use. Minister of Health determined five pilot regions in order to start to apply this system in Turkey in 2008. Nowadays, most of the hospitals in Turkey are applying online appointment system. Since patients can not choose which hospital they should go by using the system, the design will enable to have more options, so it will be more user friendly.¹

Example of an Online Appointment System Interface

¹-Ref: http://212.175.55.200/randevu_web/
3.1.2 Online Test Results

There are such online systems in Turkey. When patients open the several websites, they can have test results. The another system about this topic is that people are informed by mobile phones, when their test results are ready. However, this system serves people only information. In the design which is planned by the team, members can also reach whole test results by using short message service. The figure below shows an example of online tool related to test results.²

2-Ref: http://www.tahlilsonucu.com/
3.1.3 Sponsor

Another market research is about the market share of GSM operators in Turkey. The figure below shows market shares of these operators in the end of 2008.3

Health care issue is an important area for GSM operators. They try to give mobile solutions to this issue. Since Turkcell has the most market share in Turkey and this project is sponsored by this company, the team desired to work on this field.

3.2 Technology Research

3.2.1 Development Platforms

Some tools will be used to develop the design are:

3.2.1.1 Weka

In the design it will be tried to determine which days or hours are more crowded in hospitals according to previous data, therefore data mining tool is needed to use.

The Weka workbench contains a collection of visualization tools and algorithms for data analysis and predictive modeling, together with graphical user interfaces for easy access to this functionality. The original non-Java version of Weka was a TCL/LK front-end to (mostly third-party) modeling algorithms implemented in other programming languages, plus data preprocessing utilities in

3-Ref: www.donanimhaber.com
C, and a Makefile-based system for running machine learning experiments. This original version was primarily designed as a tool for analyzing data from agricultural domains, but the more recent fully Java-based version (Weka 3), for which development started in 1997, is now used in many different application areas, in particular for educational purposes and research. The main strengths of Weka are that it is:

- freely available under the,
- very portable because it is fully implemented in the Java programming language and thus runs on almost any modern computing platform,
- contains a comprehensive collection of data preprocessing and modeling techniques, and
- is easy to use by a novice due to the graphical user interfaces it contains.

Weka supports several standard data mining tasks, more specifically, data preprocessing, clustering, classification, regression, visualization, and feature selection. All of Weka's techniques are predicated on the assumption that the data is available as a single flat file or relation, where each data point is described by a fixed number of attributes (normally, numeric or nominal attributes, but some other attribute types are also supported). Weka provides access to SQL databases using Java Database Connectivity and can process the result returned by a database query. It is not capable of multi-relational data mining, but there is separate software for converting a collection of linked database tables into a single table that is suitable for processing using Weka. Another important area that is currently not covered by the algorithms included in the Weka distribution is sequence modeling.

Weka's main user interface is the Explorer, but essentially the same functionality can be accessed through the component-based Knowledge Flow interface and from the command line. There is also the Experimenter, which allows the systematic comparison of the predictive performance of Weka's machine learning algorithms on a collection of datasets.

The Explorer interface has several panels that give access to the main components of the workbench. The Preprocess panel has facilities for importing data from a database, a CSV file, etc., and for preprocessing this data using a so-called filtering algorithm. These filters can be used to transform the data (e.g., turning numeric attributes into discrete ones) and make it possible to delete instances and attributes according to specific criteria. The Classify panel enables the user to apply classification and regression algorithms (indiscriminately called classifiers in Weka) to the resulting dataset, to estimate the accuracy of the
resulting predictive model, and to visualize erroneous predictions, ROC curves, etc., or the model itself (if the model is amenable to visualization like, e.g., a decision).

The Associate panel provides access to association rule learners that attempt to identify all important interrelationships between attributes in the data. The Cluster panel gives access to the clustering techniques in Weka, e.g., the simple k-means algorithm. There is also an implementation of the expectation maximization algorithms for learning a mixture of normal distributions. The next panel, Select attributes provides algorithms for identifying the most predictive attributes in a dataset. The last panel, Visualize, shows a scatter plot matrix, where individual scatter plots can be selected and enlarged, and analyzed further using various selection operators.

### 3.2.1.2 ASP.NET

It is planned to use ASP.NET in order to do interface.

ASP offered the efficiency of ISAPI applications along with a new level of simplicity that made it easy to understand and use. However, ASP script was an interpreted script and consisted unstructured code and was difficult to debug and maintain. As the web consists of many different technologies, software integration for Web development was complicated and required to understand many different technologies. Also, as applications grew bigger in size and became more complex, the number of lines of source code in ASP applications increased dramatically and was hard to maintain. Therefore, an architecture was needed that would allow development of Web applications in a structured and consistent way.

The .NET Framework was introduced with a vision to create globally distributed software with Internet functionality and interoperability. The .NET Framework consists of many class libraries, includes multiple language support and a common execution platform. It's a very flexible foundation on which many different types of top class applications can be developed that do different things. Developing Internet applications with the .NET Framework is very easy. ASP.NET is built into this framework, we can create ASP.NET applications using any of the built-in languages.

Unlike ASP, ASP.NET uses the Common Language Runtime (CLR) provided by the .NET Framework. This CLR manages execution of the code we write. ASP.NET code is a compiled CLR code instead of interpreted code (ASP). CLR also allows objects written in different languages to interact with each other. The CLR makes development of Web applications simple.
3.2.1.3 Microsoft SQL Server

Since the design will store and process data, SQL server is appropriate to use. Microsoft SQL Server is a database server software. It provides to develop and manage the databases. This server is also known the most powerful and reliable database server. It enables to reach data on applications of SQL Server client.

3.2.2 Programming Languages

3.2.2.1 Java

“Java is both a programming language and a platform. Java programs are written by using Java programming language and they run on Java Platforms.”

Also Java provides robust technologies for server side application development. It is generally the first choice for business application developers because there are lots of solid frameworks that are prepared for this goal.

Java programming language is preferred since it is more appropriate for the design.

3.2.3 Database Servers

Since the design is based on a database, the team is planning to use a database programming language. There are alternatives such as MySQL, MsSql, DB2, PostgreSQL and Oracle. The choice of team is now MySQL. It is easy to host because lots of hosting firms offer MySQL services.

3.2.3.1 PHP

“PHP is a widely-used general purpose scripting language that is especially suited for Web development and can be embedded into HTML.” Although PHP application development is very easy and it is easy to find hosting firm, its structure is less suitable for big scaled projects.

4 Requirements

4.1 Functional Requirements

4.1.1 Data Flow and Interfaces

The data flow within the system and the external interfaces are illustrated in the data flow diagrams below. As it can be seen from the diagrams, the application is mainly interacts with user as a mobile application. The system also gets location information from mobile devices. This data can be gathered or other location services provided by mobile phone operator.
Figure 1 - Context Level DFD
4.1.1.2 Level 1: Mobile Medical Management

![Diagram: Mobile Medical Management Level 1 DFD]

Figure 2 - Mobile Medical Management Level 1 DFD
4.1.1.3 Level 2: Process Command

Figure 3 - Process Command Level 2 DFD
Figure 4- Context Level DFD
4.1.1.4 Level 1: Membership System

Figure 5: Membership System Level 1 DFD
4.1.2 Use Cases

The main features and functional requirements of the Mobile Medical Management (3M) application are described as use cases in this section.
4.1.2.1 Creating an account

- **Description**: User must have an account to use the specific features of the application.

- **Stimulus/Response Sequences:
  
  **Basic Data Flow:**
  1. Users send SMS to the service which contains the information:
     - Name, surname, city, gender, age and Identity number.
  2. System receives the SMS.
  3. System adds the new user to the members table in the database.
  4. Finally, if the membership process is completed correctly, system sends an SMS to the new user which contains the verification message and membership ID of the new user.

  **Alternative Data Flow:**
  1. Users send SMS to the service which contains the information:
  2. If the information that the user sent to the service is wrong, system detects it.
     - E.g. The city name that the user typed is “Anka” instead of “Ankara”.
  3. The system will send a warning message to the user.

4.1.2.2 Requesting the most suitable hospital

- **Description**: A user can search the most suitable hospitals via the system. This search can be done according to two different parameters which are time and distance.

- **Stimulus/Response Sequences:
  
  **Basic Data Flow 1:**
  1. Users send SMS to the related service which contains the information:
     - Department that the user will be examined, the parameter “T” (T for time or D for distance) and the request for map (optional).
  2. The system receives the SMS.
  3. The system firstly get the parameter.
  4. The system gets the location of the user with the help of “Turkcell”.
  5. The system communicates with the hospitals' database then applies algorithms to the data and uses data mining tools like “WEKA” and gives the most suitable hospital name.
     - E.g. There is a hospital which is 20 km away from the user and the examination time of the user is 60 minutes. And another hospital which is 10 km away from the user has an examination time 90 minutes. Hence, the system will offer the 20 km away hospital by concerning the time.
6. After finding the most suitable hospital the system sends the hospital name via SMS, and will predict the optimum time that the patient will be examined.

7. Finally, if the user requested for the map, system will communicate with the related systems like “google map” and send the path image of the hospital to the user via MMS.

**Alternative Data Flow:**
1. Users send SMS to the service which contains the information:
2. If the information that the user sent to the service is wrong or inappropriate format, the system detects it.
   E.g. The city name that the user typed is “Anka” instead of “Ankara”. SMS that the user sent to the service is “T,Kardioloji” instead of “Kardioloji,T”
3. The system will send a warning message to the user.

**Basic Data Flow 2:**
1. Users send SMS to the related service which contains the Information:
   • Department that the user will be examined, the parameter “D” (T for time or D for distance) and the request for map.
2. The system receives the SMS.
3. The system firstly gets the parameter.
4. The system gets the location of the user with the help of Turkcell.
5. Then, the system communicates with the hospitals' databases then applies algorithms to the data and uses data mining tools like WEKA and gives the most suitable hospital name.
   E.g. There is a hospital which is 20 km away from the user and the examination time of the user is 60 minutes. And another hospital which is 10 km away from the user has an examination time 90 minutes. So the system will offer the 10 km away hospital by concerning the distance.
6. After finding the most suitable hospital, the system sends the hospital name via SMS, and will predict the optimum time that the patient will be examined.
7. Finally, if the user requested for the map, system will communicate with the related systems like “google map” and send the path of the hospital to the user via MMS.

**Alternative Data Flow:**
1. Users send SMS to the service which contains the information:
2. If the information that the user sent to the service is wrong or inappropriate format, the system detects it.
   E.g. The city name that the user typed is “Anka” instead of “Ankara”.
SMS that the user sent to the service is “T, Kardioloji” instead of “Kardioloj.T”
3. The system will send a warning message to the user.

4.1.2.3 Take test results:
• Description: Test results like blood test, urine test, sperm test, elisa test can be received by the users. But users have to become a member to the system to able to use this service.

• Stimulus/Response Sequences:
Basic Data Flow:
1. Every day the system controls the test results in the hospital database.
2. Then the system retrieves the patients ID's from the database.
3. The system controls the ID's and eliminate the IDs which do not belong the member users.
4. Finally, the system sends the appropriate test results to the appropriate users.

4.1.2.4 Getting users locations Info:
• Description: When a user searches the most suitable hospitals via the system. System has to get the location of the user at that time to be able to give the most suitable hospital.

Basic Data Flow:
1. The system receives SMS from the user.
2. The system communicates with the service provider “Turkcell”.
3. Then the system gets the location information.

4.1.2.5 Edit Hospital Database
• Description: Administrators of the system can change the hospital entries in the database.

• Assumption: Users are administrators.

• Stimulus/Response Sequences:
Basic Data Flow 1:
1. User enters his account name and password, and enters to the database.
2. User selects to edit the database.
3. User adds a new hospital name to the database.
4. User adds the needed information like address of the hospital, which departments are currently working, how many doctors the hospital has.
5. User selects OK.
6. The new entry is saved to the database.

**Basic Data Flow 2:**
1. User enters his account name and password, and enters to the database.
2. User selects to edit the database.
3. User deletes a hospital name from the database and all the related fields about this hospital.
4. User selects OK.
5. The hospital is deleted from the database.

**Basic Data Flow 3:**
1. User enters his account name and password, and enters to the database.
2. User selects to edit the database.
3. User changes a hospital name, address, department information, number of the doctor from the database.
4. User selects OK.
5. The hospital information is updated in the database.

**Alternative Data Flows**

**Alternative Data Flow 1**
1a. User tries to add an existing hospital to the database.
1b. An error message appears “the entered hospital is already in the database”
1c. User changes the hospital name

**Alternative Data Flow 2**
2a. User tries to delete a non-existing hospital
2b. An error message appears “the selected hospital cannot be deleted”
2c. User changes the hospital name

**Alternative Data Flow 3**
3a. User tries to change a non-existing hospital’s address or department information
3b. An error message appears “the selected hospital is not in the database”
3c. User changes the hospital name

**Alternative Data Flow 4**
4a. User logs out without saving the changes
4b. The system asks to the user “Do you want to save the changes?”

**Alternative Data Flow 5**
5a. User chooses cancel
5b. Exit from database

4.1.2.6 Edit Member Database

• **Description**: Administrators of the system can change the member entries in the database.

• **Assumption**: Users are administrators.

• **Stimulus/Response Sequences:**

**Basic Data Flow 1:**
1. User enters his account name and password, and enters to the database.
2. User selects to edit the database.
3. User adds a new member id to the database.
4. User adds the required information like name of the member, address, gender and age
5. User selects OK.
6. The new entry is saved to the database.

**Basic Data Flow 2:**
1. User enters his account name and password, and enters to the database.
2. User selects to edit the database.
3. User deletes a member id from the database.
4. User selects OK.
5. The member is deleted from the database.

**Basic Data Flow 3:**
1. User enters his account name and password, and enters to the database.
2. User selects to edit the database.
3. User changes a member id, name, address or personal information of the member.
4. User selects OK.
5. The member information is updated in the database.

**Alternative Data Flows**

**Alternative Data Flow 1**
1a. User tries to add an existing member to the database.
1b. An error message appears “the entered member is already in the database”
1c. User changes the member id

**Alternative Data Flow 2**
2a. User tries to delete a non-existing member
2b. An error message appears “the selected member cannot be deleted”
2c. User changes the member id

**Alternative Data Flow 3**
3a. User tries to change a non-existing member's address or personal information
3b. An error message appears “the selected member is not in the database”
3c. User changes the hospital name

**Alternative Data Flow 4**
4a. User logs out without saving the changes
4b. The system asks to the user “Do you want to save the changes?”

**Alternative Data Flow 5**
5a. User chooses cancel
5b. Exit from database

### 4.2 Non-Functional Requirements

#### 4.2.1 Runtime Requirements

1. **Usability**: The system should have easy user interfaces. It should have minimum number of interface steps for any facility that it supports to user at client side.

2. **Documentation**: The system should include a client side tutorial to make it easy for user’s. Also a more detailed online documentation about the system’s usage should be prepared.

3. **Availability/Reliability**: The system should be available to its users almost every time. When it is not possible to reach the system’s server side, the client side application should record the events of the user for further push to server.

4. **Scalability**: The system should be able to be used when it have 1 – 100.000 clients. So, an automated logic behind the system is mandatory. (i.e. An approach where administrators control data legibility will most probably fail.) Also the facilities supported to users should not depend on much user crowd. Almost all facilities should be meaningful when the system has less or more users.

5. **Security**: The system’s client and server side should communicate with each other through a secure way, like https, since client side will push personal information about the user.

6. **Quality of Service Requirements**: System should be able to process at least 100000 transactions between server-client per minute.
4.2.2 Development Requirements

1. Localizability: The system does not have to be localizable necessarily. But a localizability approach in the future could be beneficial.

2. Modifiability/Extensibility: The system should have enough modularity so that it could be extended by new features/facilities for users in the future.

3. Portability: The system’s server-client communication should be platform-independent. i.e. It should allow GroupBarkod to develop new client-side applications to other platform’s.

4.3 System Requirements

4.3.1 Client Side Requirements

1. Hardware Requirements: The minimal requirements are 32 megabytes of RAM, 32 megabytes of flash, and a 400-Mhz online processor.

2. Software Requirements: There is no restriction on OS at client side.

4.3.2 Server Side Requirements

1. Hardware Requirements: The minimal requirements are Celeron® 2.0GHz Processor with 1x180 GB disk drive, 1 GB RAM and 5 Mbps Broadband Internet Connection.

2. Software Requirements: Any modern OS with Java Virtual Machine and a Database Server.
# 5 Project Schedule

The roadmap of the Mobile Medical Management project is illustrated below.

<table>
<thead>
<tr>
<th>Phase</th>
<th>Duration</th>
<th>Start Date</th>
<th>End Date</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Planning</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Problem Definition</td>
<td>5 days</td>
<td>05.10.09</td>
<td>09.10.09</td>
</tr>
<tr>
<td>Project Survey</td>
<td>5 days</td>
<td>10.09.09</td>
<td>14.10.09</td>
</tr>
<tr>
<td>Decision of Roles of Group</td>
<td>3 days</td>
<td>15.10.09</td>
<td>17.10.09</td>
</tr>
<tr>
<td>Quality and Configuration</td>
<td>3 days</td>
<td>18.10.09</td>
<td>20.10.09</td>
</tr>
<tr>
<td>Project Proposal</td>
<td>6 days</td>
<td>21.10.09</td>
<td>26.10.09</td>
</tr>
<tr>
<td><strong>Analysis</strong></td>
<td>21 days</td>
<td>27.10.09</td>
<td>17.11.09</td>
</tr>
<tr>
<td>Analysing Requirements</td>
<td>3 days</td>
<td>27.10.09</td>
<td>29.10.09</td>
</tr>
<tr>
<td>Determining the Problem</td>
<td>4 days</td>
<td>30.10.09</td>
<td>03.11.09</td>
</tr>
<tr>
<td>Data Modeling</td>
<td>10 days</td>
<td>04.11.09</td>
<td>13.11.09</td>
</tr>
<tr>
<td>Functional Modeling</td>
<td>10 days</td>
<td>04.11.09</td>
<td>13.11.09</td>
</tr>
<tr>
<td>Behavioral Modeling</td>
<td>10 days</td>
<td>04.11.09</td>
<td>13.11.09</td>
</tr>
<tr>
<td>Data Dictionary</td>
<td>10 days</td>
<td>04.11.09</td>
<td>13.11.09</td>
</tr>
<tr>
<td>Writing the Analysis Report</td>
<td>8 days</td>
<td>10.11.09</td>
<td>17.11.09</td>
</tr>
<tr>
<td><strong>Design</strong></td>
<td>18 days</td>
<td>18.11.09</td>
<td>05.12.09</td>
</tr>
<tr>
<td>Architectural Design</td>
<td>2 days</td>
<td>18.11.09</td>
<td>19.11.09</td>
</tr>
<tr>
<td>User-Interface Design</td>
<td>3 days</td>
<td>20.11.09</td>
<td>22.11.09</td>
</tr>
<tr>
<td>Component Level Design</td>
<td>2 days</td>
<td>23.11.09</td>
<td>24.11.09</td>
</tr>
<tr>
<td>Initial Design Report</td>
<td>2 days</td>
<td>25.11.09</td>
<td>26.11.09</td>
</tr>
<tr>
<td>Architectural Design-2</td>
<td>2 days</td>
<td>27.11.09</td>
<td>28.11.09</td>
</tr>
<tr>
<td>User-Interface Design-2</td>
<td>2 days</td>
<td>29.11.09</td>
<td>30.11.09</td>
</tr>
<tr>
<td>Component Level Design-2</td>
<td>2 days</td>
<td>01.12.09</td>
<td>02.12.09</td>
</tr>
<tr>
<td>Final Design Report</td>
<td>3 days</td>
<td>03.12.09</td>
<td>05.12.09</td>
</tr>
<tr>
<td><strong>Prototype Development</strong></td>
<td>31 days</td>
<td>06.12.09</td>
<td>07.01.10</td>
</tr>
<tr>
<td>Coding Prototype</td>
<td>12 days</td>
<td>06.12.09</td>
<td>17.12.09</td>
</tr>
<tr>
<td>Prototype Demo</td>
<td>19 days</td>
<td>18.12.09</td>
<td>07.01.10</td>
</tr>
<tr>
<td><strong>Implementation</strong></td>
<td>88 days</td>
<td>21.02.10</td>
<td>25.02.10</td>
</tr>
<tr>
<td>Coding</td>
<td>3 days</td>
<td>21.02.10</td>
<td>23.02.10</td>
</tr>
<tr>
<td>User Manual</td>
<td>2 days</td>
<td>24.02.10</td>
<td>25.02.10</td>
</tr>
<tr>
<td><strong>Testing</strong></td>
<td>8 days</td>
<td>26.02.10</td>
<td>03.03.10</td>
</tr>
<tr>
<td>Unit Testing</td>
<td>8 days</td>
<td>04.03.10</td>
<td>11.03.10</td>
</tr>
<tr>
<td>Integration Testing</td>
<td>6 days</td>
<td>12.03.10</td>
<td>17.03.10</td>
</tr>
<tr>
<td>Validation Testing</td>
<td>8 days</td>
<td>18.03.10</td>
<td>25.03.10</td>
</tr>
<tr>
<td>Recovery Testing</td>
<td>12 days</td>
<td>26.03.10</td>
<td>08.04.10</td>
</tr>
<tr>
<td>Security Testing</td>
<td>12 days</td>
<td>09.04.10</td>
<td>20.04.10</td>
</tr>
<tr>
<td>Performance Testing</td>
<td>11 days</td>
<td>21.04.10</td>
<td>02.05.10</td>
</tr>
<tr>
<td>Test Report</td>
<td>8 days</td>
<td>03.05.10</td>
<td>10.05.10</td>
</tr>
<tr>
<td>Software Demo</td>
<td>18 days</td>
<td>11.05.10</td>
<td>28.05.10</td>
</tr>
</tbody>
</table>
6 Conclusion

This report is prepared to show Mobile Medical Management project's requirement details in terms of several aspects. Firstly, a brief description of “Mobile Medical Management” is introduced. Then, a marketing and technology research is carried out and the results are established. And at the body part, the requirement details of the project are described. As a last work, project’s schedule is presented. Thus, this report focused on the aspects which are seemed to be important.