TABLE OF CONTENTS

1. INTRODUCTION
   1.1. Project Title
   1.2. Motivation
   1.3. Project Definition
   1.4. Project Scope
   1.5. Team Organization
       1.5.1. Team Structure
       1.5.2. Technical Work Distribution

2. THE PROCESS MODEL

3. MARKET RESEARCH
   3.1. Literature Survey
   3.2. Customer Survey
4. REQUIREMENT ANALYSIS
   4.1. System Requirements
      4.1.1. Hardware Requirements
      4.1.2. Software Requirements
      4.1.3. Development Environment Requirements
   4.2. Functional Requirements

5. USE CASE ANALYSIS
   5.1. Use Case Diagrams
   5.2. Use Case Scenarios

6. MODELING
   6.1. Data Modeling
      6.1.1. Entity Relationship Diagrams
   6.2. Functional Modeling
      6.2.1. Data Flow Diagrams
      6.2.2. Data Dictionary of Data Flow Diagrams
7. PROJECT SCHEDULE

7.1. Project Milestones

7.2. Gantt Chart

8. RISK MANAGEMENT

8.1. Risk Mitigation, Monitoring and Management Plan

8.2. Risk Table

9. REFERENCES
1. INTRODUCTION

1.1. Project Title

IMBO

1.2. Motivation

In accordance with the developments in technology, the communication methods people chose have altered. Internet; becoming widespread during recent years; is used by 1.244 billion people according to the Usage and World Population Statistics for September 30, 2007\(^1\). A result of the widespread usage of Internet, Web applications became very popular. The ability to update and maintain Web applications without distributing and installing software on potentially thousands of client computers is a key reason for their popularity. Recently, it is adopted in principle to make web applications faster and more practical for real-world applications.

Instant messaging; a form of real-time communication between two or more people based on typed text; is a kind of web application. Modern, Internet-wide, GUI-based instant messengers began to appear in the mid 1990s with ICQ (1996); followed by followed by AOL Instant Messenger, Yahoo, MSN, Excite, Ubique,
IBM, MSN, AIM and Google Talk. Nowadays, the number of web sites providing their users instant messaging is increasing. For example, Quick Contacts in Gmail and Windows Web Messenger are web applications on web sites. Some web applications provide an integrated environment among instant messaging systems. For example; Meebo is an in-browser instant messaging program which supports multiple IM services, including Yahoo! Messenger, Windows Live Messenger, Google Talk, AIM, ICQ by using Jabber protocol. Other examples are ebuddy (supporting MSN, Yahoo, AIM, and GTalk) and Communication Tube (supporting ICQ, MSN, IRC, GTalk).

1.3. Project Definition

IMBO will enable users to reach their multiple IM services through IMBO IM service contemporaneously. So IMBO will be an online, browser based instant messaging (IM) environment which supports multiple popular IM services; MSN, GTalk, ICQ, Yahoo. IMBO will use Jabber (XMPP) as a technological base for instant messaging purposes. Following figure shows the context diagram of IMBO:
1.4. Project Goals

Here are some general features that will be in IMBO:

- Visitor is the user of IMBO selecting one of the IM services from MSN, GTalk, ICQ, Yahoo and requesting to sign up his/her chosen IM service. Visitor will be able to chat with his/her contacts from his/her chosen IM service. If his/her chosen IM service enables file transfers, he/she will be able to transfer files with his/her contacts.

- Signed-Up User of IMBO is the user subscribed to IMBO IM service and has more privileges than the Visitor has. The contacts of Signed-Up User of
IMBO will consist of the contacts from all IM services the user has. IMBO will enable this kind of user to chat, transfer files, search other users with profile information, subscribe to offline tracking, constitute groups, enrol as a member of groups and send offline messages with his/her contacts. The ability of constituting and subscribing groups will enable the user to socialize; and this property will provide IMBO to have extendable features like the ones in Facebook.

1.5. Team Organization

1.5.1. Team Structure

Our team has a controlled decentralized team organization. We preferred it because the software knowledge level of each member team is not equal. Considering experience, we chose our team leader with unanimity. Our team leader assigned subtasks to each group member in the team. Being a controlled decentralized team, we know the importance of information exchange and we have been trying to get our levels of knowledge closer.

1.5.2. Technical Work Distribution

Erdem Karahan: Development of Jabber Plug-in
Esra Abacıoğlu: Development of Jabber Plug-in
Taha Bekir Eren: Development of the Web Application
2. **THE PROCESS MODEL**

Our team is supposed to progress through analysis, initial design, detailed design, release of prototype, implementation, testing and maintenance phases in a limited time. Although we are going to try our best not to encounter problems during the project, there is always a possibility of not being able to complete the project owing to the existing risks. Therefore, the first goal of our team is being successful in the release stage of the prototype. We agreed on that linear sequential model (Waterfall Model) is the most suitable process model for our project up to the release of the prototype. Our first reason is that it is a systematic step-by-step approach that begins at the system level and progresses through analysis, design, coding, testing and maintenance; and these progress phases are the ones our team is supposed to follow. Our second reason is that a schedule can be set with deadlines for each stage of development in Waterfall Model and considering our strict deadlines, scheduling is an important issue for us. Our third reason is that it is documentation driven, that is, documentation is produced at every stage. Therefore, the preparation for writing reports will not be problem for us; as we anyway
produce the documentation needed in the reports according to our choice of process model.

As Waterfall Model does not allow for much reflection or revision, we are planning to change our process model after release stage of the prototype until the release stage of the final version of the product. We will adopt the spiral process model since then, because the idea behind the development of the prototype into the final version of the product is collinear with the spiral process model. Another reason for preferring spiral process model during these stages is that it is more able to cope with the changes; and our project can undergo changes according to the suggestions of our teaching assistant and our supervisor Mr. Güven Fidan.

3. MARKET RESEARCH

People have been using a variety of different methods for communication and instant messaging (IM) is one of those methods, which has become more popular recently. Especially web based IM services have become more preferable since they require no additional software download. Besides, these services can be reached from anywhere that has Internet connection. This popularity has given rise to the development of many different such services. Almost all of these services have some features in common. We can name these features as the basic requirements an IM service should satisfy. We can list these requirements briefly as follows:
➢ There should be no need to download any additional software in order to use the service.

➢ An IM service should support popular IM networks like MSN Messenger, Google Talk, or ICQ.

➢ The service should provide enough security.

In addition to these basic requirements, different IM services provide different facilities like chat rooms, invisible sign on, or mobile messaging. After an investigation about the current web based IM services in the market, we decided to include facilities like profile searching and offline messaging in our product IMBO.

In the following section, you will find brief information about some web based IM services we have examined.

3.1 LITERATURE REVIEW

3.1.1 Communication Tube:
Communication Tube is not the most popular but one of the most powerful web based IM service. It supports many popular IM networks like ICQ, Google Talk, MSN Messenger, and IRC. We can list the advantages and disadvantages of this service as follows:

**Advantages:**

- Messages appear instantly using AJAX technology.
- It works through CGI proxy services. Cookies and Java Script is enabled. For this reason, you can use your favourite IM client even though its server is not reachable.
- 1024 bit RSA encryption is used to sent usernames and passwords.
- It is compatible with the following web browsers: Microsoft IE 5.0+, Mozilla Firefox 0.8+, Mozilla 1.4 +, Netscape Navigator 6+, Opera 7+, Konquerer etc.
- It supports mobile messaging such as PDAs, smart phones and other mobile devices. For these devices, it provides compatibility with the following browsers: Opera Mobile Browser, Safari for iPhone, BlackBerry.

**Disadvantages:**

- Conversation history is not saved.
Profile searching is not included.

No support for chat rooms.

3.1.2 Meebo:

Meebo is one of the most common IM service. It is still being developed and in the future, it will provide mobile support as well as ring notifications. Currently it supports IM networks lie MSN Messenger, AIM, Google Talk, and Yahoo Messenger. We can group the features of Meebo:

Advantages:

- Invisible sign on.
- Users are allowed to create user accounts so that they can use a single login for multiple services.
- Ability to store chat logs.
- Ability to rename contacts.
- Ability to send messages to users not in your contact list.
- Chat room support.
- Users can host their own chat rooms.

Disadvantages:
- No support for profile searching.
- No offline messaging.

3.1.3 eBuddy:
eBuddy is another web-based messenger that enables people to chat with their MSN Messenger, Yahoo Messenger, MySpace, Google Talk, and AIM friends.

**Advantages:**

- Invisible sign on.
- Support for chat rooms.
- Supports multiple IM networks in one interface.
- Supports mobile IM. It works on devices (cell phones, PDAs, PlayStation Portables) that support WAP or xHTML. It has a version for iPhone, which is based on AJAX platforms.

**Disadvantages:**

- No support for profile searching.

### 3.2 CUSTOMER SURVEY:

While defining the features of IMBO, there were two main things we took into consideration. One of them was examining the current IM services in the market. Brief information about some of these services has been given in the
previous section. The other thing we took into consideration was the opinions of potential customers. For this purpose, we conducted a small survey among our instant messaging contacts. As a result of this survey, we realized that people seek for some important features in an IM service like offline messaging, profile search and contact tracking. This is why we decided to implement these features in IMBO.

4. REQUIREMENT ANALYSIS

4.1. System Requirements

4.1.1. Hardware Requirements

Considering our project, strictly defining hardware requirements is not wise. By the time as our number of users increase, our hardware requirements will change. As a baseline, we need a single Pentium 4 computer with 1 GB of ram. Nevertheless, as the number of users increase some enhancements on hardware configurations must be considered:

- Moving to a multi-processor and vast memory (both ram and durable storage) server
- Creating a clustered server architecture
- Separating DBMS server from the Web Server
- Installing a RAID system on the DBMS server
- Using a distributed file system.
4.1.2. Software Requirements

- Windows as the operating system (because of ASP.NET application)
- IIS (Internet Information Services)
- Openfire (Jabber Server)
- MS SQL Server as DBMS

4.1.3. Development Environment Requirements

- Microsoft Visual Studio 2005
- Eclipse
- SQL Server Management Studio

4.2. Functional Requirements

Web Application:

Our web application is the only part of the project, which is directly in touch with the user. Other components of our system serve to our web application, either directly or indirectly. We determined our requirements based on our market research.
- Users can login with their existing instant messaging (IM) account without any subscription. This can be achieved by making a connection to the provided jabber server (e.g. if the user provides a Gmail address, connect to talk.google.com). Obviously, no database activity is needed for this.

- Users can create an IMBO account, under which they can combine multiple other IM accounts. This relation will be saved in our database.

- Users can create groups, and become members of existing groups. They can search for groups or invite their contacts to become a member of an existing group.

- Users will be able to send and receive instant messages with their IMBO account, even if they do not have any other IM accounts. In this case, our jabber server will be used to send and receive instant messages.

- If a user has an IMBO account, he/she will be able to create and maintain a user profile. If the user gives authorization, other users will be able to make searches using profile criteria. Otherwise, the user will be excluded from the search.
Users will be able to send messages even when the recipient is offline (some current IM systems provide this function, as GTalk and Windows Live Messenger, some do not).

If the user combines multiple IM accounts less than one IMBO account, when he/she signs in, all his/her accounts will be automatically signed in. All contacts; he/she has in various IM accounts; can be seen within a single list.

A curious user may track his/her contact(s)'s status while he/she is offline. Changes in status will be reported when the user signs in next time.

Profile Web Services:

Profile searching will be available to third party applications via xml web services. The web service will query the database by using database classes.

Jabber Plug-in:

The plug-in will send the offline messages to the recipients when they become online. Sender's login information may be needed to send the messages. Another alternative is sending the messages in the name of sender with another address (such as admin@imbo.com).
The plug-in will save contact tracking information to the database to be used by the web application. In addition, contacts to be tracked information will be read from the database.

5. USE CASE ANALYSIS

5.1. Use Case Diagrams

Use Case Diagram For Visitor:
Use Case Diagram For Candidate User Of IMBO:

Use Case Diagram For Signed-Up User Of IMBO:
5.2. Use Case Scenarios

**Visitor:**

- **Login:** The visitor will choose one of the IM services from MSN, GTalk, ICQ, and Yahoo; and will login to that IM service. After the validation of login information, the visitor will be able to make operations that are allowed by the IM service he/she has login.

- **Operations Done From MSN:** The visitor will have the ability to chat and transfer files with the contacts he/she has in MSN after he/she signs in MSN.

- **Operations Done From GTalk:** The visitor will have the ability to chat and transfer files with the contacts he/she has in GTalk after he/she signs in GTalk.

- **Operations Done From ICQ:** The visitor will have the ability to chat and transfer files with the contacts he/she has in ICQ after he/she signs in ICQ.

- **Operations Done From Yahoo:** The visitor will have the ability to chat with the contacts he/she has in Yahoo after he/she signs in Yahoo.

**Candidate User of IMBO:**

- **Request Sign-Up From IMBO:** In order to sign up to IMBO, the candidate user will have to submit user name and password to web user interface.
Optionally, the candidate user may submit profile information to be a member of the profile search feature.

**Signed-Up User of IMBO:**

- **Login:** The signed-up user of IMBO will have to login to IMBO IM service in order to be able to make operations that are allowed by IMBO IM service after the validation of login data.

- **Operations Done From IMBO:** The signed-up user of IMBO will have the ability to chat, transfer files, constitute groups, enrol as a member of groups, make profile search and subscribe to offline tracking.

- **Unsubscribe From IMBO:** The signed-up user of IMBO may unsubscribe from IMBO whenever he/she needs.
6. MODELING

6.1. Data Modeling

6.1.1. Entity Relationship Diagrams
6.2. Functional Modeling

6.2.1. Data Flow Diagrams

Level 0 Data Flow Diagram:

[Diagram with nodes and arrows indicating flow of data and processes]

- User
- Group Information
- Profile Information
- Messages
- Database Server
- Crack Track Info
- Off-line messages
- 1.0 Web Application
- 2.0 Jabber Plug-in
- 3.0 Profile Web Services
- Visitor
- 3rd Party Applications
- Profile Info.
Level 1 Data Flow Diagrams:

1.1 User Interaction
- Send Message
- Receive Message
- Search by Profile
- Save Profile
- Search Groups
- Group Membership Request

1.2 Message Operations
- Outgoing Messages
- Incoming Messages

1.3 Profile Operations
- Query Profiles
- Save Profile
- Query Groups

1.4 Group Operations
- Save Membership Info.

Database
6.2.2. Data Dictionary of Data Flow Diagrams

<table>
<thead>
<tr>
<th>Name</th>
<th>Messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input To</td>
<td>User, Visitor, Jabber Server, 1.0 Web Application</td>
</tr>
<tr>
<td>Output From</td>
<td>User, Visitor, Jabber Server, 1.0 Web Application</td>
</tr>
<tr>
<td>Description</td>
<td>These are the messages communicated through user and web application module, visitor and web application module and jabber server and web application.</td>
</tr>
<tr>
<td>Format</td>
<td>Instant text messages</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Group Information</strong></td>
<td>User, Jabber Server, Database Server, 1.0 Web Application</td>
</tr>
<tr>
<td><strong>Profile Information</strong></td>
<td>User, 1.0 Web Application</td>
</tr>
<tr>
<td><strong>Offline Messages</strong></td>
<td>Database Server, 2.0 Jabber Plug-In, Jabber Server</td>
</tr>
<tr>
<td><strong>Contact Tracking Information</strong></td>
<td>2.0 Jabber Plug-In, Jabber Server, Database Server</td>
</tr>
<tr>
<td><strong>Send Message</strong></td>
<td>1.2 Message Operations</td>
</tr>
<tr>
<td><strong>Receive Message</strong></td>
<td>1.5 Visitor Interaction, 1.1 User Interaction</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Search By Profile</td>
<td>Profile searching results obtained by the profile operations</td>
</tr>
<tr>
<td>Save Profile</td>
<td>Saved profile information data</td>
</tr>
<tr>
<td>Search Groups</td>
<td>Resulting data obtained from group searching</td>
</tr>
<tr>
<td>Group Membership Request</td>
<td>Group membership request data</td>
</tr>
<tr>
<td>Save Membership Information</td>
<td>User membership saving information recorded to the database</td>
</tr>
<tr>
<td>Query Groups</td>
<td>Group information queries obtained from the group records in the database</td>
</tr>
</tbody>
</table>
### Project Details

<table>
<thead>
<tr>
<th>Name</th>
<th>Query Profiles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input To</td>
<td>1.3 Profile Operations</td>
</tr>
<tr>
<td>Output From</td>
<td>Database</td>
</tr>
<tr>
<td>Description</td>
<td>User profile information queries obtained from the profile records in the database</td>
</tr>
<tr>
<td>Format</td>
<td>String representing the group queries</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Incoming Messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input To</td>
<td>1.2 Message Operations</td>
</tr>
<tr>
<td>Output From</td>
<td>-</td>
</tr>
<tr>
<td>Description</td>
<td>Branch of the data “Messages” in level 0</td>
</tr>
<tr>
<td>Format</td>
<td>Instant Text Messages</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Outgoing Messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input To</td>
<td>-</td>
</tr>
<tr>
<td>Output From</td>
<td>1.2 Message Operations, 2.1 Offline Message Processor</td>
</tr>
<tr>
<td>Description</td>
<td>Branch of the data “Messages” in level 0</td>
</tr>
<tr>
<td>Format</td>
<td>Instant Text Messages</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Tracking Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input To</td>
<td>Database, 2.2 Tracking Information Processor</td>
</tr>
<tr>
<td>Output From</td>
<td>2.2 Tracking Information Processor</td>
</tr>
<tr>
<td>Description</td>
<td>Online-Offline information data of the user’s contacts, formed by the tracking information processor</td>
</tr>
<tr>
<td>Format</td>
<td>Information text files</td>
</tr>
</tbody>
</table>

### 7. PROJECT SCHEDULE

#### 7.1. Project Milestones

These are the work packages of our project:

**Project Management:**

- Project Proposal
- System Requirements Specification and Analysis Report
- Initial Design Report
- Final Design Report

Web Application & Web Services:
- Database Design
- Database Class Library Implementation (Prototype)
- Web Application Implementation (Prototype)
- Prototype Integration & Testing
- Prototype Release
- Database Class Library Implementation (Complete)
- Web Application Implementation (Complete)
- Profile Web Services Implementation (Complete)
- Integration
- Alpha Testing
- Beta Testing and Debugging
- Web Application & Web Services Release

Jabber Server Plug-in:
- Plug-in Implementation (Prototype)
- Plug-in Implementation (Complete)
- Integration
- Alpha Testing
- Beta Testing and Debugging
- Plug-in Release

### 7.2. Gantt Chart

<table>
<thead>
<tr>
<th>Project Tasks:</th>
<th>Oct</th>
<th>Nov</th>
<th>Dec</th>
<th>Jan</th>
<th>Feb</th>
<th>Mar</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Project Management</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project Proposal</td>
<td>♦</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System Requirements Analysis Report</td>
<td></td>
<td>♦</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Initial Design Report</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Final Design Report</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Web Application and Web Services</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Database Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>♦</td>
<td>♦</td>
</tr>
<tr>
<td>Database Class Library Implementation (Prototype)</td>
<td>♦</td>
<td>♦</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Database Class Library Implementation (Complete)</td>
<td></td>
<td>♦</td>
<td>♦</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Web Application Implementation (Prototype)</td>
<td></td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
</tr>
<tr>
<td>Web Application Implementation (Complete)</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
</tr>
<tr>
<td>Prototype Integration and Testing</td>
<td></td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
</tr>
<tr>
<td>Prototype Release</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
</tr>
<tr>
<td><strong>Jabber Server Plug-in</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Plug-in Implementation (Prototype)</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
</tr>
<tr>
<td>Plug-in Implementation (Complete)</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
</tr>
<tr>
<td>Integration</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
</tr>
<tr>
<td>Alpha Testing</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
</tr>
<tr>
<td>Beta Testing and Debugging</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
</tr>
<tr>
<td>Web Application &amp; Web Services Release</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
<td>♦</td>
</tr>
</tbody>
</table>
8. RISK MANAGEMENT

Risk is the likelihood of threat-source emerging and exposing the project to a particular potential vulnerability. For determining the likelihood of a future problem, our team must analyze the treats to the project concomitancy with the potential vulnerabilities and the regular controls. The term risk management is the process allowing managers to balance the operational and economic costs of protective measures and achieving gains in mission capability by protecting the systems and data. The risk table and risk mitigation, monitoring and management plan we adopt in principle can be exposed to alternation during the development phases.

8.1. Risk Mitigation, Monitoring and Management Plan

- Problems Due to Lack of Knowledge and Experience on Subject

Our team has encountered in an area in which we have no background; therefore, lack of knowledge and experience will probably get each of team members into problems. If the team member is in luck at that moment, this problem will just be equal to loss of time. However, this problem can get the team into a break point as well. Not to encounter this kind of problems we are trying to be well
prepared for weekly progress meetings, as they secure to get us into sufficient knowledge.

➤ **Problems Due to Misunderstood Requirements**

Although our team recorders try hard for understanding the requirements; our team can ever misunderstand the requirements for the project in any design phase; which cause loss of time. To overcome this problem, we agreed on that the most important part of our weekly meetings should be the exchange of ideas.

➤ **Problems Due to Large Project Size and Unrealistic Scheduling**

Our team is supposed to arrange the schedule in an early phase of the project. However, three of us have never participated in such a large project and we do not have enough experience either. Therefore, the probability of arranging an unreasonable schedule is round the corner for us. Later in the project, if we ever feel that our schedule is unreasonable to follow, then we will enhance our schedule according to the situation we will be in.

➤ **Problems Due to Choice of Tools, Libraries**

As a team, we are taking pain over choosing the tools, libraries that will be used in our project. However, wrong decisions about them could already be made.
In such a case, changing these elements after design phase will be probably difficult; at least causing loss of time owing to the extra reading needed.

➢ Problems Due to the Communication Gap Between Team Members

In such a large project without any experience, it is unavoidable for each of team members that in developing process there would be divergence about emphasizing their own crucial ideas. Therefore, this can give birth to hidden competition. Nevertheless, by playing cards open about what everyone wants to perform within in the project, this insidious enemy could be eradicated from the start.

➢ Problems Due to Inauspicious Moment of Team Members

Team members can suffer from illnesses or consistent workloads. These situations can get our Project into a break point as they are unpleasant jar to the nerves and cause loss of time. Unfortunately, we have already experienced these kinds of problems. We are look forward to getting over them. To avoid these kinds of risks, a team member in an inauspicious moment should notify the situation of his/her to other members. If really necessary little adjustments on the schedule can be made.
### 8.2. Risk Table

<table>
<thead>
<tr>
<th>Risks</th>
<th>Probability</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problems Due to Lack of Knowledge and Experience on Subject</td>
<td>%20</td>
<td>3</td>
</tr>
<tr>
<td>Problems Due to Misunderstood Requirements</td>
<td>%10</td>
<td>2</td>
</tr>
<tr>
<td>Problems Due to Large Project Size and Unrealistic Scheduling</td>
<td>%20</td>
<td>1</td>
</tr>
<tr>
<td>Problems Due to Choice of Tools, Libraries</td>
<td>%10</td>
<td>2</td>
</tr>
<tr>
<td>Problems Due to the Communication Gap Between Team Members</td>
<td>%5</td>
<td>3</td>
</tr>
<tr>
<td>Problems Due to Inauspicious Moment of Team Members</td>
<td>%40</td>
<td>2</td>
</tr>
</tbody>
</table>

**Impact Values:**

1- Catastrophic management process  
2- Critical  
3- Marginal  
4- Negligible

### 9. REFERENCES

3- )  http://en.wikipedia.org/wiki/Meebo

4- )  http://www.communicationtube.net

5- )  http://freeonlinesurveys.com