

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

Let's Help Bo Final Project Report

Version 1.0

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

Revision History

Date	Version	Description	Author
06.01.2013	0.1	Initial draft	Aleksandar Nikodinovski
20.01.2013	1.0	Release version	Aleksandar Nikodinovski, Hrvoje Novak, Antonio Gallucci

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

Table of contents

1	Introduction	5
1.1	Purpose of this document	5
1.2	Intended Audience	5
1.3	Scope	5
2	Definitions and acronyms	5
2.1	Definitions.....	5
2.2	Acronyms and abbreviations.....	5
3	References.....	5
4	Background and Objectives	6
5	Organization.....	8
5.1	Project Group	8
5.2	Customer.....	9
5.3	Supervisor	9
5.4	Others.....	9
6	Development process	10
6.1	Description	10
6.2	Project phases.....	11
6.2.1	Requirements gathering phase	11
6.2.2	Design	11
6.2.3	Implementation.....	11
6.2.4	Integration and Testing	11
6.2.5	Verification.....	11
6.3	Roles description.....	12
6.3.1	Project leader	12
6.3.2	Team leader	12
6.3.3	Requirements manager.....	12
6.3.4	Design manager	12
6.3.5	Database manager.....	12
6.3.6	User interface manager.....	12
6.3.7	Usability manager.....	13
6.3.8	Documentation manager	13
6.3.9	SVN manager.....	13
6.3.10	Testing manager	13
6.3.11	Meeting manager	13
6.3.12	Implementation manager	13
6.4	Quality assurance	13

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

6.4.1	Organization	13
6.4.2	Planning	14
6.4.3	Software Quality	14
7	Milestones	14
8	Project Results	15
8.1	Requirements	15
8.1.1	Requirement Compliance Matrix	15
8.1.2	Requirements Compliance Summary	17
8.2	Deliverables	17
8.2.1	Remarks	19
8.3	Risks	19
8.4	Project Experiences	20
8.4.1	Positive Experiences	20
8.4.2	Improvement Possibilities	20
8.5	Metrics	21
8.5.1	Work per Member	21
8.5.2	Milestone Metrics	21
8.5.3	Effort Metrics	22

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

1 Introduction

1.1 Purpose of this document

The purpose of this document is to give an overall summary of the development progress of the project, the achieved goals and delivered items. This document is defined in the last stage of the project and as such is not up for revision. Since this document is mostly a summary of other artifacts and documents in the project, it is not provided as basis for any other document.

1.2 Intended Audience

This document is intended for the supervisor and the steering group as a report on what has been done during the course of this project and what the results are. It is also intended for the customer as an insight into what he will get from/out of this project.

1.3 Scope

This document provides the description of the project, its objectives and the development process, planned milestones and deliverables and project organization. It describes the results of this project and the compliance with the planned progression. Results consist of acquired requirements and their completion status, final set of delivered items, what risks were encountered, project experiences, possible improvements and work metrics. This document does not provide detailed instructions on how the software was developed nor does it give any kind of user manual.

2 Definitions and acronyms

2.1 Definitions

Keyword	Definitions
ABB CRC	The Corporate research center of ABB, a multinational corporation operating in robotics and mainly in the power and automation technology areas

2.2 Acronyms and abbreviations

Acronym or abbreviation	Definitions
LHB	Let's Help Bo
FER	Faculty of Electrical Engineering and Computing
MDH	Mälardalen University

3 References

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

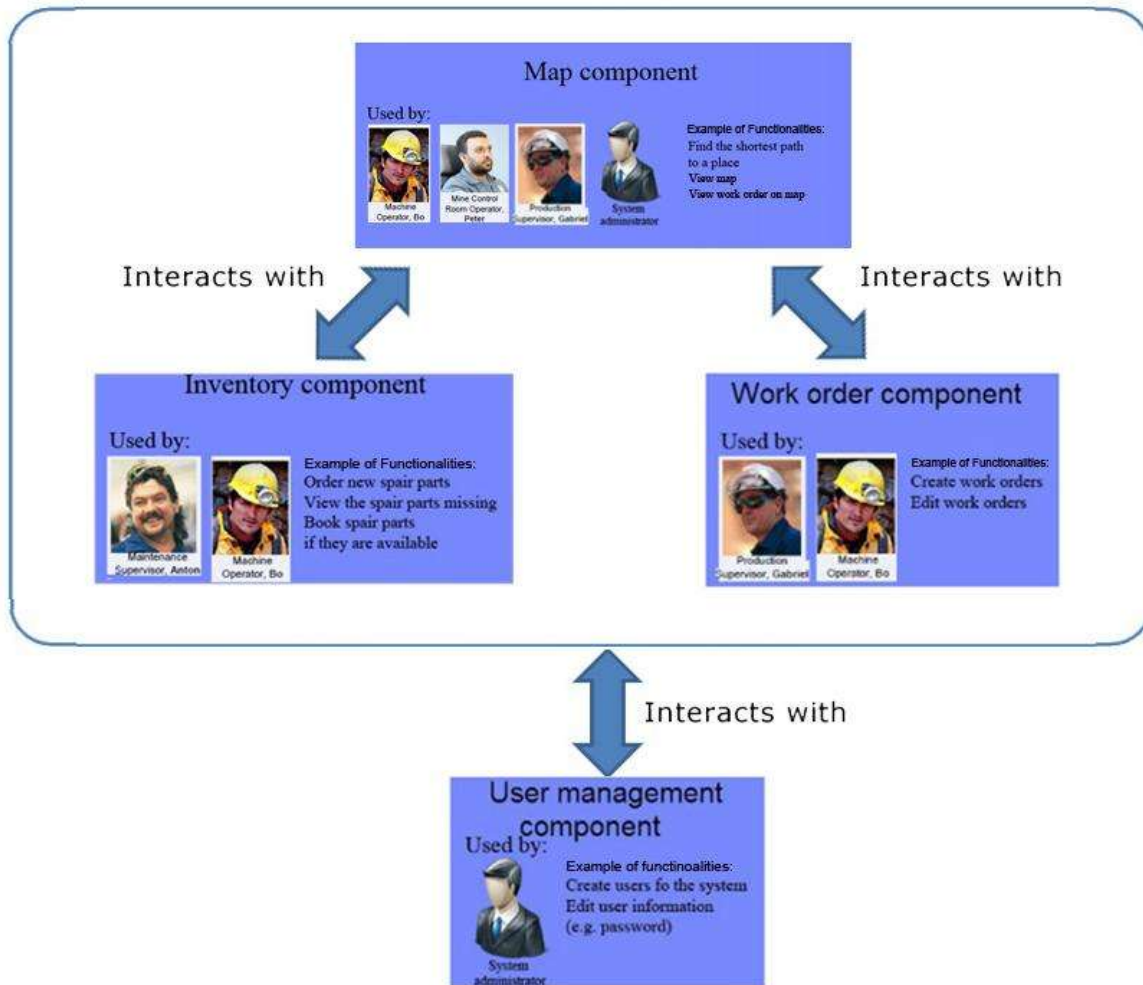
- Project DSD home page
http://www.fer.unizg.hr/rasip/dsd/projects/lets_help_bo
- Project solution desktop web page
<http://161.53.67.226:8080/Desktop>
- Project solution mobile web page
<http://161.53.67.226:8080/Mobile>

4 Background and Objectives

The main subject of the project is to develop an inventory support system for future mines. The main goal is to help machine operators in their everyday work. Since the machines and equipment used in the mine are subjected to malfunction, spare parts need to be acquired from warehouses. In normal circumstances, this uses up quite some time from the workers. This project resolves that issue by automating the whole process. Machine operators can access the central booking system via an application to order the necessary spare parts. They also get a notification upon completion of the order and directions to the location where to pick it up. Part of this project is also the feature to manage the work orders that needs to be changed during the pickup of the spare parts.

The project offers extra functionalities that aim to help in the daily activities of other people that work in the mine, too, beside machine operators. Below are the components that will be part of the system, and the actors that will interact with them.

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013



One of the main requirements is to develop a software that is easy to use considering the environment in which it will be used. Detailed description of all requirements is defined in the requirements definition document.

General milestones are:

- Project vision
- Project plan
- Requirements Definition
- System Architecture
- Alpha prototype
- Beta prototype
- Final product

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

Deliverables include:

- Project plan document
- Requirements Definition document
- Design Description document
- Summary Week Reports
- Minutes of Meeting
- Alpha Mockups
- Coding policy
- SVN policy
- Traceability matrix
- Acceptance test plan
- Test report
- Instructions videos of the system for the customers
- Installation manual
- Final Project Report
- Final Product

To assure that the system is working and to remove bugs from the product, the software needs to be tested. Unit testing should be performed on each individual module, to detect early problematic system modules, and to avoid problems to spread over the entire system. Additionally, integration tests should be performed to assure the good combination and interaction between all the developed modules, as one unique piece of software to be delivered to the user. Usability tests are also advisable, in order to maximize the performance of the users of this system during a work day.

Final product with final project report should be delivered on 20.01.2012.

5 Organization

5.1 Project Group

Name	Initials	Responsibility (roles)
Aleksandar Nikodinovski	AN	Project leader
Hrvoje Novak	HN	Team Leader
Nives Bučić	NB	Requirement manager Documentation manager
Rasul Niyazimbetov	RN	User interface manager Usability manager
Petar Stojanac	PS	Meeting manager SVN manager

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

Antonio Gallucci	AG	Requirement manager Documentation manager
Gonçalo Filipe Silva	GS	Testing manager
Danijel Jambrečina	DJ	Database manager
Niklas Gilström	NG	Design manager Implementation manager
Robert Pofuk	RP	Implementation manager

5.2 Customer

Our external customers are the ABB Corporate Research Center represented by

- Isak Savo
- Petra Björndal
- Aneta Vulgarakis
- Anton Jansen

Aneta is also the project supervisor. The part of the team that is based in MDH in Sweden can talk to her personally, while the rest of the team uses Skype to hold the meetings and gather the information. With Isak Savo Anton Jansen and Petra Björndal we communicate through Aneta and also via email. From them we can learn a lot about the functionalities and behaviour of the application.

5.3 Supervisor

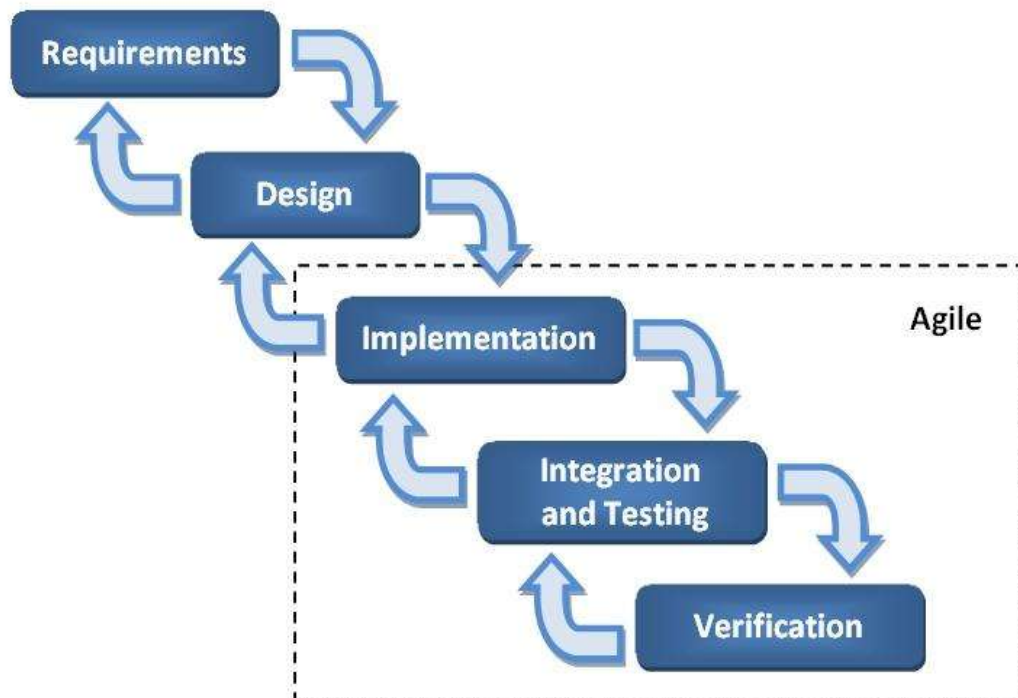
Aneta Vulgarakis is the project supervisor but she also has the roles of a customer and a steering group member.

5.4 Others

The steering group is composed of professor Ivica Crnković from MDH and professor Mario Žagar from FER which hold the DSD course and monitor how the teams are progressing, and Aneta Vulgarakis, the project supervisor.

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

6 Development process



6.1 Description

The methodology we chose to adapt on this project is an iterative waterfall model with a mix of agile methodology. In the early stages of the project we will use the waterfall methodology and in the later stages of the development, such as implementation and testing, we will switch to agile. The main reason for this is that our team is a fairly large team, it has 10 members and they are located in two different countries. Because of that obstacle, the communication is not as good as it would be if the team was working locally and the pure agile approach wouldn't be as efficient.

Another important reason is that our customers insist on user centered design and high usability. In order to achieve that, requirements need to be gathered carefully and they need to be well documented. Waterfall model allows the customers to clearly state their requirements and it allows the designers to design a solid architecture that will respond to them.

Finally, the documentation we need to produce and deliver during this course matches pretty well with the documentation that is made during the initial phases of the waterfall methodology and the later agile approach to implementation is well suited for developing a component based system.

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

6.2 Project phases

6.2.1 Requirements gathering phase

During this phase there is a lot of communication inside the project team as well as with the supervisor and with the customers. The goal is to crystallize well defined requirements, to document them extensively and to understand them so every team member gets the general idea of what the application is going to be about.

6.2.2 Design

The project team focuses on designing the architecture and the graphical user interface of the application so it can meet all the requirements. Meeting the requirements is the most important part but there is also a matter of extensibility and scalability. It's important to design the architecture in such a way that sometime in the future extra features can be introduced on top of the existing ones and that is what our team will strive to.

6.2.3 Implementation

The implementation phase consists mainly of coding and unit testing. It's an iterative model so the product of the phase will be a working and tested component of the system. During this phase the managing skills of the managers are tested, especially in a big team of 10 people. Every team member has different skills and it's up to the managers to use them most efficiently. This is where we will implement code sharing via Subversion repository to speed up the development.

6.2.4 Integration and Testing

In this phase the components that were developed by separate team members are integrated together to make the system whole. Testing must be done to ensure the success of the integration phase which means that the system has no errors or failures and can be started. In the meantime, individual components are also tested before integration with the system is done. The product of this phase is the core of the application (together) with some of the components tested and ensured that they all work as a whole.

6.2.5 Verification

This is the last phase of project development. When the system integration is complete, before it can be delivered to the customer, it must be verified that it meets all the defined requirements. Also, in this phase, the system is tested for eventual bugs so they can be fixed.

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

6.3 Roles description

Our project team consists of 10 members. This is considered to be a big team and to properly manage it we have assigned manager roles that alleviates and speeds up the development process. The managers are responsible for coordination of specific aspects of the development process.

6.3.1 Project leader

The project leader distributes tasks amongst the team members. He has constant communication with all the team members as well as with the project supervisor and customers. Managing and minimizing risk. Follow the work progress of each team member and assign help if needed. Urging team members to meet their deadlines. Gathering week reports from everyone and summarizing them in a summary week report.

6.3.2 Team leader

Communicating intensely with the project leader and local team members. Distributing tasks amongst the local team. Keep track of the work progress of the local team. Arranging local team meetings and preparing topics for them.

6.3.3 Requirements manager

In charge of documenting, analyzing, tracing, prioritizing and agreeing on requirements and then controlling change and communicating to relevant stakeholders.

6.3.4 Design manager

Design manager coordinates work on application design and further divides the tasks given by the project leader and team leader into sub tasks and distributes them to the rest of the team members. He will also advise the rest of the team on the proper practices when designing the architecture of the application and resolve any arising questions and problems.

6.3.5 Database manager

In charge of setting up the virtual machine for the database server. Coordinate team members and give advice on how to build a correctly designed database. Implement the designed database and fill in the initial data. Responsible for regular maintenance of the database.

6.3.6 User interface manager

Responsible for organizing and distributing tasks when it comes to implementation of the user interface. Needs to coordinate with the usability manager.

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

6.3.7 Usability manager

Specific role created because of the requirement for user centered design. The task of the manager is to keep track of the graphical interface design and make sure that it is in accordance to the principles of user centered design. It is also to give advice to the designers on the right approach to designing such interface.

6.3.8 Documentation manager

Supervising the writing of the software documentation. Reviewing all the documents before delivery.

6.3.9 SVN manager

Needs to keep a regular backup of all the files that are on SVN. Uploads the final version of deliverable documents to SVN. When a conflict appears he works together with the team member that is in conflict to resolve it.

6.3.10 Testing manager

Manages the testing of the code during the implementation and coordinates the writing of the test report document.

6.3.11 Meeting manager

Writes Minutes of meeting document where he summarizes everything that was said on the meeting including who needs to do what and when. Uploads the document to the DSD course website.

6.3.12 Implementation manager

Coordinates the activities related with the implementation. Responsible for the integration of the code. Needs to be in close contact with user interface manager and usability manager. He is responsible for determining the code policy to prevent the conflicts that could arise from distributed development.

6.4 Quality assurance

Ensures that the final product meets the requirements of customers and passes validation. Manages the quality of software and its development process.

6.4.1 Organization

Aiming to create a quality culture in the team by encouraging the team members to meet the deadlines. Improving quality by using a well known software development methodology that produces documentation for each phase of the development process and using templates for those documents. The work products are sent for confirmation to the project supervisor and the customers.

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

6.4.2 Planning

Improving quality by identifying the potential risks and by planning appropriate counter measures.

6.4.3 Software Quality

Team members with more experience mentor other on how to produce well-defined documents. After a document is finished it needs to pass a review done by other team members. Writing documents like code policy and SVN policy that establish a set of rules and guidelines on how to code and use Subversion.

7 Milestones

ID	Milestone description	Responsible dept./initials	Planned week	Promised week	Late +/-	Delivered week	Metr	Rem
M-01	Project vision	AN, HN	43	43	0	43	0	
M-02	Project plan	AN, PS	44	44	0	44	0	
M-03	Requirements Definition	AG, NB	44	44	0	44	0	
M-04	System Architecture	NG	45	45	0	45	0	
M-05	Alpha prototype	All team members	48	48	0	48	0	
M-06	Beta prototype	All team members	51	51	0	51	0	
M-07	Final project	All team members	3	3	0	3	0	

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

8 Project Results

8.1 Requirements

8.1.1 Requirement Compliance Matrix

Id	Requirement Description	completed	Rem
	User management component		
UM-1	Logging		
UM-1.1	Logging in	<i>Yes</i>	
UM-1.2	Logging out	<i>Yes</i>	
UM-2	Managing users		
UM-2.1	Creating user	<i>Yes</i>	
UM-2.2	Viewing user details	<i>Yes</i>	
UM-2.3	Editing user	<i>Yes</i>	
UM-2.3.1	Recovering password	<i>Dropped</i>	
UM-2.4	Deleting user	<i>Yes</i>	
	Work order component		
WO-1	Viewing work orders		
WO-1.1	Viewing assigned work orders	<i>Yes</i>	
WO-1.1.1	Viewing updated work orders	<i>Yes</i>	
WO-1.2	Viewing all work orders	<i>Yes</i>	
WO-2	Managing work orders		
WO-2.1	Creating work order	<i>Yes</i>	
WO-2.2	Editing work orders		
WO-2.2.1	Machine operator editing	<i>Yes</i>	
WO-2.2.2	Production supervisor editing	<i>Yes</i>	
WO-3	Viewing work order pool	<i>Yes</i>	
WO-4	Getting work order from pool	<i>Yes</i>	

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

WO-5	Viewing work order templates	<i>Yes</i>	
WO-6	Managing work order templates		
WO-6.1	Creating work order template	<i>Yes</i>	
WO-6.2	Editing work order template	<i>Yes</i>	
WO-6.3	Deleting work order template	<i>Yes</i>	
	Schedule component		
SC-1	Viewing schedules		
SC-1.1	Viewing own schedule	<i>Dropped</i>	
SC-1.2	Viewing all schedules	<i>Dropped</i>	
SC-2	Managing schedules		
SC-2.1	Creating schedule	<i>Dropped</i>	
SC-2.2	Editing schedule	<i>Dropped</i>	
	Inventory component		
IC-1	Booking spare part	<i>Yes</i>	
IC-2	Viewing bookings		
IC-2.1	Viewing own bookings	<i>Yes</i>	
IC-2.2	Viewing all bookings	<i>Yes</i>	
IC-3	Searching spare part	<i>Yes</i>	
IC-3.1	Viewing spare part details	<i>Dropped</i>	
IC-4	Managing spare parts		
IC-4.1	Creating spare parts	<i>Yes</i>	
IC-4.2	Editing spare parts	<i>Yes</i>	
IC-4.3	Deleting spare parts	<i>Yes</i>	
	Map component		
MC-1	Managing map		
MC-1.1	Managing tunnels		
MC-1.1.1	Creating tunnel	<i>Yes</i>	

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

MC-1.1.2	Editing tunnel	Yes	
MC-1.1.3	Deleting tunnel	Yes	
MC-1.2	Managing hotspots		
MC-1.2.1	Creating hotspot	Yes	
MC-1.2.2	Editing hotspot	Yes	
MC-1.2.3	Deleting hotspot	Yes	
MC-2	Viewing map	Yes	
MC-2.1	Finding shortest route	Yes	
	Nonfunctional requirements		
NR-1	Usability	Yes	
NR-2	Security	Yes	
NR-5	Portability and compatibility	Yes	

8.1.2 Requirements Compliance Summary

Total number of requirements	42
Number of requirements implemented	36
Requirements partially fulfilled	0
Requirements not fulfilled	0
Requirements dropped	6

8.2 Deliverables

To	Output	Planned week	Promised week	Late +/-	Delivered week	Remarks
All stakeholders	Project plan	w43	w43	-	w43	-
All stakeholders	Alpha GUI Mockups	w44	w44	-	w44	-

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

All stakeholders	Requirement definition	w44	w44	-	w44	-
All stakeholders	Design Description	w44	w44	-	w44	-
All stakeholders	Code policy	-	-	-	w47	-
All stakeholders	SVN policy	-	-	-	w47	-
All stakeholders	Alpha prototype	w48	w48	-	w48	-
All stakeholders	Beta prototype	w51	w51	-	w51	-
All stakeholders	Acceptance test plan	w52	w52	-	w52	-
All stakeholders	Test report	w3	w3	-	w3	-
All stakeholders	Final project report	w3	w3	-	w3	-
All stakeholders	Final product	w3	w3	-	w3	-
All stakeholders	Summary Week Reports	-	-	-	-	R_01
All stakeholders	Minutes of Meeting	-	-	-	-	R_02
All stakeholders	Traceability matrix	w3	w3	-	w3	-
All stakeholders	Instructions videos of the system for the customers	-	-	-	-	-
All stakeholders	Installation manual	w3	w3	-	w3	-

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

8.2.1 Remarks

Remarks ID	Description
R_01	Delivered every Monday before 23.59 during the whole course of the project
R_02	Delivered after every team meeting and after the meetings with the customers during the whole course of the project

8.3 Risks

Possibility	Risk	Preventive action	Appearance
high	Members have other courses to attend	Divide work according to member possibility's	Appeared but their impact was low because of preventive actions
medium	Members being late with their work and missing deadlines	Internal deadlines earlier than official. Redistribution of work not done to other members.	Avoided
medium	Impossible to meet schedules	Have more working hours or exclude some features	Avoided
medium	Some members have no experience with some technology	Members with experience provide assistance and tutorials	Appeared but their impact was low because of preventive actions
low	Members are not reachable	Have many communication channels.	Appeared but their impact was low because of preventive actions
low	Misunderstandings	Discuss and write all things that could lead to misunderstanding	Avoided
low	Conflicts	Try to resolve on meetings	Avoided

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

low	Losing database	Backup of database	Avoided
low	Corrupt database	Database manager should fix corruptions, backups	Avoided
low	Hardware malfunctions	Regular SVN commits	Avoided

8.4 Project Experiences

8.4.1 Positive Experiences

During the course of the project, the team evolved into a well-organized working group. They learned to work in a distributed multi-cultural environment and acquired useful experience from it. Everyone knew their assignments and finished them on schedule with some spare time for fine tuning, making the project constantly on track. Communication with the customer was well established, the team was in contact with all necessary personas involved in the project. The requirements were well analyzed and verified with the customer and the acquired feedback was successfully assimilated so the project wasn't lacking in constructive ideas which made the software a quality product.

8.4.2 Improvement Possibilities

One possible improvement during the project work could have been having a smaller team size. Lower amount of deadlines during exam periods could improve the quality of work. Finally, less documentation would give more time for implementation.

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

8.5 Metrics

8.5.1 Work per Member

Member	W43	W44	W45	W46	W47	W48	W49	W50	W51	W52	W01	W02	W03	Total
AN	26	18	18	22	30	12	21	36	21	7	5	20	22	258
HN	14	12	17	17	24	19	13	23	8	5	3	8	10	173
AG	25	26	27	20	30	24	20	42	22	6	2	6	21	271
RN	16	16	23	19	30	13	17	12	17	16	26	16	4	225
NG	28	25	16	25	40	28	30	45	60	26	17	25	15	380
NB	20	20	17	26	23	21	33	35	13	13	10	11	14	256
PS	17	17	10	20	12	25	14	27	24	0	3	6	4	179
RP	6	6	13	16	10	8	8	10	15	0	2	9	0	103
DJ	14	11	19	19	16	13	21	32	9	6	5	12	10	187
GS	13	1	6	8	23	5	18	16	14	2	0	1	11	118
Total	179	152	166	192	238	168	195	278	203	81	73	114	111	2150

The total amount of hours is approximately the same with the forecasted 2150/2038. The average number of hours by team member is: 215

There is a difference in the number of work hours between the team members, because some team members were motivated more to be more proactive and do extra work and improve the project.

8.5.2 Milestone Metrics

Completed as planned or earlier	Total	Timeliness
7	7	100%

Let's Help Bo	Version: 1.0
Final Project Report	Date: 20.01.2013

8.5.3 Effort Metrics

ID	Activity	Actual Effort	Planned Effort	Deviation (%)
1	Requirement gathering and analysis	10	11	-9%
2	Design	44	42	5%
3	Alpha version implementation	83	73	14%
4	Alpha version testing	21	14	50%
5	Beta version implementation	92	95	-3%
6	Beta version testing	42	25	68%
7	Final version implementation	21	20	5%
8	Final version testing	15	15	0%
9	Preparing and rehearsing presentations	41	45	-9%
10	Writing documentation	79	95	-17%
11	Meetings	70	68	3%
12	Contacting customers	8	3	167%
13	Setting up and managing VM, Database and SVN	11	4	175%

Remark: In the planned effort the number of man-days is displayed. One man-day equals 4 work hours

Deviation is calculated with the formula:

Deviation % = $100 \cdot (\text{Actual} - \text{Planned}) / \text{Planned}$

Effort estimation accuracy (%) <i>(100*(1 - abs(Actual - Planned)/Actual))</i>	95%
------------------------------------------------------------------------------------------	-----