Final Report

Company 10

Martin Ruzicka
Version 0.2

Status

Reviewed	Martin Ruzicka, Hans Sanell, Jonas Mattsson, Henrik Nilsson, Vamsi Seshabhattaru	2010-03-03
Approved	Sebastian Rehnby (PM)	2010-03-02

Project identity

Company 10, 2010
Chalmers University of Technology, MPSEN

Name	Responsibility	Phone	E-mail
Martin Ruzicka	Software Architect	070 219 20 81	ruzicka@student.chalmers.se
Jonas Mattsson	Database Designer	070 58 18 095	emattsso@student.chalmers.se
Hans Sanell	Software Architect	073 53 55 210	hansn@student.chalmers.se
Sebastian Rehnby	Project Manager	0735 080 850	
Vamsi Seshabhattaru	Quality Manager	0739 190 703	vamsi@student.chalmers.se
Henrik Schulze Nilsson	Graphical Designer	0733 643 501	henrnil@student.chalmers.se

suit-group-10@googlegroups.com

Client: Central Operative & Independent Retailers Acquirer, 9 Downing St, Westminster, London SW1A, UK
Client Contact: Peter Arch, +44 333 757589, peter.arch@coira.com
Supervisor: Per Zaring

Contents

Document History

Version	Date	Changes	Responsible	Approved
0.2	2010-03-03	Reviewed and fixed minor issues.	MR/SR	SR
0.1	2010-03-02	First version.	MR	SR

Abstract

Company 10 is a reputed international organization with a lot of satisfied customers and an increasing list of services provided. The company has been approached by Professor Per Zaring at Chalmers University of Technology, Gothenburg, for helping him out with a project, The Unlimited Well, for the company Coira. The project is part of a course Per is in charge of "Developing Computer-Based Systems in Teams". The project has been taken up by Company 10 who has provided a working solution for the system specification provided by COIRA and the option to continue work on the solution provided.

This project report is part of the project and describes the various development work carried out by Company 10 for the successful completion of the project. The document starts with the introduction of the project and the background of the project. The following sections explain the development plans implemented including the Project Objectives, Directives and a summary of the various technical details of the project.

1. Introduction

This report describes the fully functional e-commerce system, The Unlimited Well. The content will include description of the development process, analysis of the project proposal, as well as an analysis and discussion of the result. The report will be finalized with a conclusion of the whole project.

1.1. Background

Central Operative & Independent Retailers Acquirer (Coira) is an enterprise with a number of retailers in large parts of the western world. Coira deals in a number of unspecified products and wishes to expand to the rest of the world through a web-based portal, called The Unlimited Well (UW). To achieve this, Coira wants to create systems for handling retail, customer purchases and logistics, all of which should utilize existing systems, such as Coira's financial system. Coira has contacted Company 10 to design and produce these systems.

1.1.1. Client

Central Operative & Independent Retailers Acquirer (Coira).

1.1.2. Overall Project Description

Coira wants a web-based portal where customers from all corners of the world can purchase products. This portal should consist of three parts. The portal itself, called Mall Unlimited Well (MUW) that contains the interface towards the customer and the database of available products. A system for handling the suppliers and the purchases, called Purchase Unlimited Well (PUW), which contains the list of approved suppliers and their products. Also, a system for transporters and logistics is needed, called Logistics Unlimited Well (LUW) that contains all approved transporters and their routes. These systems cooperate so that when a customer orders something through MUW, a request is sent to PUW to order the goods from the supplier and a separate request is sent to LUW to book a transport for the purchased goods from the supplier to the customer. All systems should be fault tolerant and provide high availability. All systems must be secure and the integrity of customers, suppliers and transporters must have the highest priority.

MUW handles all interaction with the customer. MUW displays all available products; their details, price and how long it will take to deliver. MUW also handles the customers orders, displays information and options about orders, such as where they are and if there is any problems, and notifies the customer if there is any problems along the way. MUW's graphical interface will provide context-sensitive help at all levels of the interface. At purchase, MUW should find an optimal combination of supplier and transporter. MUW contains the database of available products, their status and suppliers, as well as the customer register. MUW must support several languages and it must be easy to extend the number of languages supported. MUW will also collect statistics about purchases and use these to make predictions about demands and ask PUW to make sure that there is enough supply to satisfy the demand.

PUW is, upon purchase, to contact the chosen supplier and book the products specified. PUW will also keep track of suppliers and update their status in MUW's database. PUW will select suppliers that should undergo a quality review given certain conditions. PUW also has to notify Coira employees if there is not enough suppliers in the system to meet the demand as predicted by MUW, so that new suppliers may be acquired. PUW will also handle delivery notes and send all required data to Coira's financial system.

LUW will be responsible for handling logistics and transportation, and keeps track of transporters and their routes. Upon purchase, LUW will book suitable transportation and produce all the needed papers, such as bills of lading and import licenses. LUW is responsible for updating the status of orders and where a particular order is physically. To be able to do this, LUW must accept input from a number of different sources so that the companies that handle the actual transport can update regardless of internal system used. LUW shall utilize a GIS company to compute distances and time consumption. LUW must also make sure that each transport is as profitable as possible with very few, but existing, exceptions. LUW must also allow independent transporters to use the system to plan transports and routes. This includes transports that have nothing to do with Coira. In this case the system is called Logistics at Transporters (LaT). In case the system is used this way, the content in the transporters area must be exclusive to that transporter. LUW should notify MUW and Coira personnel in case any hiccups occur.

1.2. Purpose

For Coira to be able to reach and service worldwide customers and to streamline and enhance the process from placed order to delivery, in a way that is as cost efficient and fast as possible.

1.3. Project Objectives and Deliverables

The main objective of the project is to build a system conforming to specification provided by Coira. To achieve this the following project objectives and deliverables has been set up.

Project objectives:

- Apply an incremental V-model development process
- Make a Software Architecture Description of the system
- Use a well-structured Project Plan and measure progress
- Perform and document testing using applicable methods
- Ensure system and development quality throughout the process

Project deliverables:

- Project Directives
- Project Plan
- Development plan
- Software Requirements
- Test Plan
- Test Report
- Project Report

- Final software system
- System documentation

1.4. Project Organization

The organization of the team will be consistent throughout the project. Depending on workload and phase, team members should offer assistance to the person responsible for the highest prioritized task/deliverable. If nothing else is specified, the responsibility for each activity should be determined by the roles specification.

1.4.1. Organization per Phase

The different phases will not change the responsibilities for a given task, but assistance should be offered to the member responsible based on the nature of the tasks in each phase. A visualization of the organizational structure is shown in Figure 1.

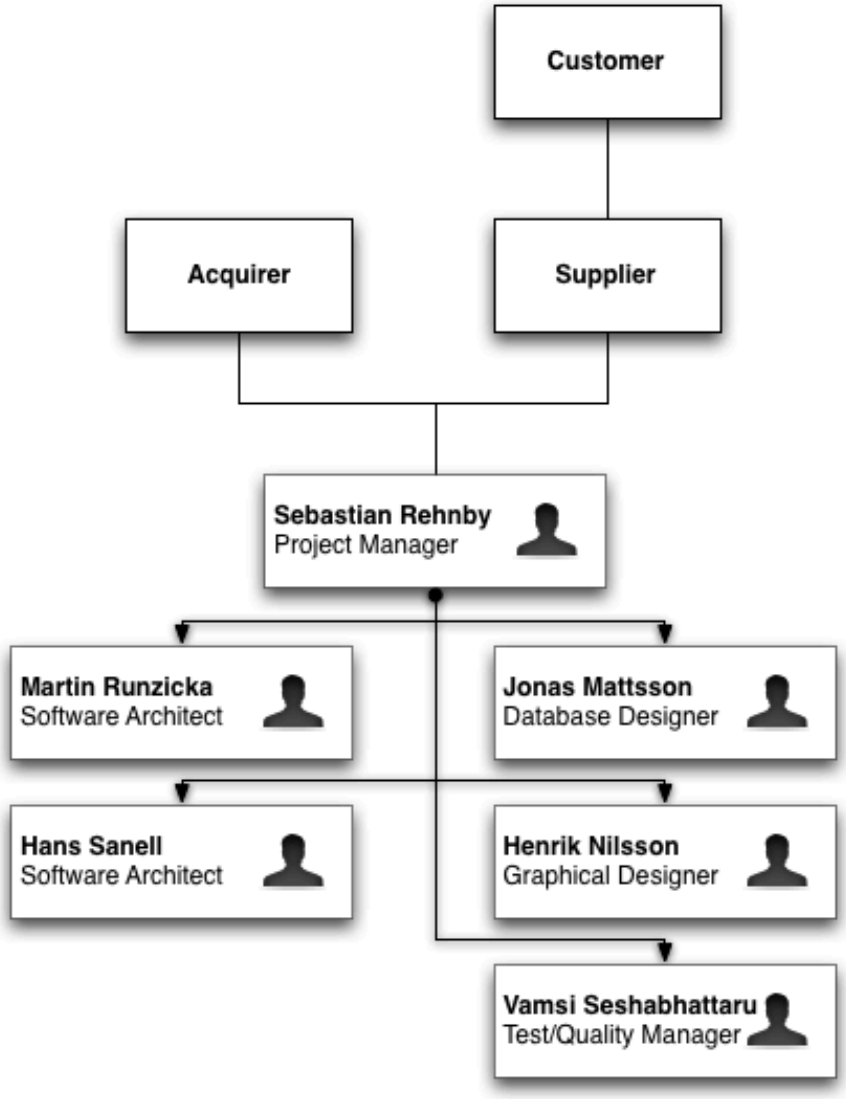


Figure 1 - Organizational Structure

1.4.2. Work Breakdown Structure and Responsibility Areas

The project has been divided into various key role areas and the tasks have been allocated to the team members accordingly. Each team member has been given a primary role and a back up or a secondary role as a risk mitigation step.

The main key roles and the team members for each of the phases has been distributed as follows:

Role	Lead	Second
Project Manager	Sebastian Rehnby	Martin Ruzicka
Software Architecture	Hans Sanell	Martin Ruzicka
Database Designer	Jonas Mattsson	Henrik Schulze Nilsson
Test/Quality Manager	Vamsi Seshabhattaru	Hans Sanell
Graphical Designer	Henrik Schulze Nilsson	Jonas Mattsson
Clerical Role	The clerical role is to be distributed to all the team members on a rotational time bound schedule.	

Table 1 – Responsibility areas

1.5. Definitions, Acronyms and Abbreviations

The following definitions, acronyms and abbreviations are used throughout the document:

Abbr.	Explanation
C10	Company 10
CMS	Content Management System
Coira	Central Operative & Independent Retailers Acquirer
LaT	Logistics at Transporters
LUW	Logistics Unlimited Well
MUW	Mall Unlimited Well
P0	Highest prioritized set of requirements
PUW	Purchase Unlimited Well
P*X*	Set of requirements prioritized to level *X*
PM	Project Manager
SA	Software Architecture
SRS	Software Requirements Specification
TM	Test/Quality Manager
UW	Unlimited Well

Table 2 – Definitions, acronyms and abbreviations

2. Technical Summary

This section describes the technical aspects of the system.

2.1. System To Be Developed

The system will be based on a number of core subsystems; MUW, PUW and LUW. These will work together to ensure that the system meets the usability properties specified. This means, not only that the system developed will have to provide a cost efficient and fast way of placing orders, it also has to be easy to deploy and scale in order to handle an increasing number of customers. The system user interface will be a web-portal, making the service easy to access and use worldwide. The web-portal turns to customers as well as logisticians.

2.2. Purpose for the System

To serve the original purpose of making it possible for Coira to *"be able to reach and service worldwide customers and to streamline and enhance the process from placed order to delivery, in a way that is as cost efficient and fast as possible"*.

2.3. Intended Users

Table 3 describes the stakeholders of the system.

Coira	Holder of the system. Responsible for keeping the system running and making sure it works well
Customers	Main target for the system. Orders products using the system
Suppliers	Ensuring products availability. Sells their products via the system
Logisticians	Delivering products. Uses the LaT part of the system for logistics

Table 3 - Stakeholders

3. Methodology

The project has been divided into various phases and the team members are given individual tasks during the various phases of the project.

3.1. Development Planning

This section describes the development planning.

3.1.1. Terminology

The development process will consist of a number of phases. Each phase is an individual development cycle, consisting of a number of activities. Each activity (e.g. software design or implementation) consists of a number of tasks as defined in the Time Plan (Appendix A). Each task consists of a number of features, a form of work package resulting in a document or implementation unit. Features usually refer to a specific requirement defined in the SRS.

3.1.2. Phases

The development process will be divided into phases. These phases are incremental development cycles, each containing a number of development activates, tasks and features. The phases of the project is defined in the SRS

document, by the prioritization of the specified requirements. Requirements are prioritized from P0 (Highest) and on all the way to P*X* (lowest prioritized set of requirements). This way, our process will adapt well to changing requirements and the phase tasks are clearly defined by the content of the SRS.

3.1.3. Process

The project will be developed using an incremental development approach, where each phase (iteration) is a variation of the V-model development process. This means that each phase will itself contain activities such as design, implementation and testing. The big difference from the complete V-model process in each phase is that the requirements analysis will be completed before the first phase of development. Figure 2 shows a visualization of the development process.

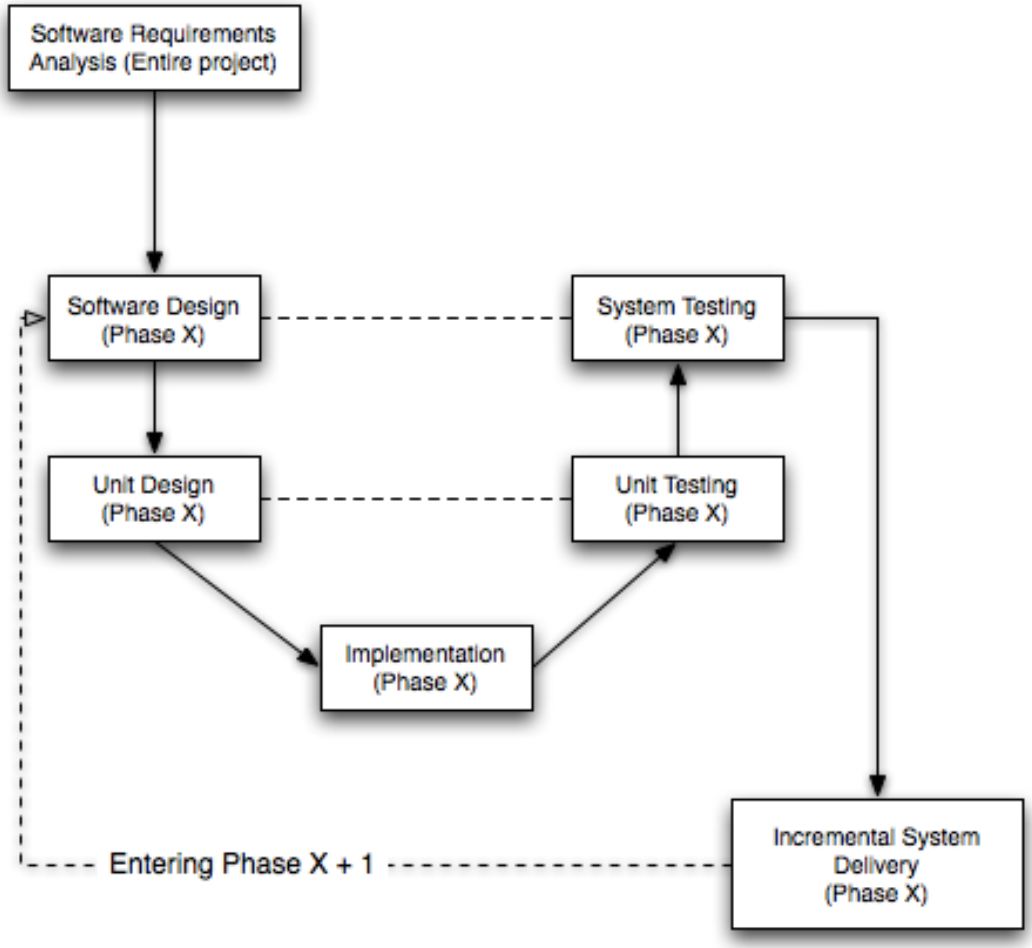


Figure 2 - The development process

3.1.4. Delivery

Each phase will yield an incremental official delivery to the client, including the corresponding test plan and test report. The system delivered should be thoroughly tested according to the test plan. Transparency of test results and

possible issues with a delivery should be clearly communicated to the client.

3.1.5. Roles and Responsibilities

The roles and responsibilities are the same throughout the project, as shown in Table 1.

3.2. Development Monitoring

A set of mechanisms will be used to monitor and assess the overall project progress, both during and after the completion of the individual phases.

3.2.1. Progress Assessment

Project progress will be measures by the milestones and time estimations defined in the Development Time Plan (Appendix B).

3.2.2. Reporting

Reporting will continue as usual during the development phases, but the team members will refer the specific feature(s) they have been working in their weekly time reports. The reason for this is to be able to do a post-phase evaluation of the time estimations projected in the Development Time Plan (Appendix B).

3.2.3. Documentation

Each development cycle (Phase) will yield a number of phase specific documents. The implementation proposals will be used as a design reference during implementation, while the test related documents would be subject to quality evaluation to assess the extent to which they comply with the SRS.

Document	Approved by	Purpose	Distributed to
Implementation Proposal	PM/SA	To define and motivate design choices. To be used as a reference during implementation	Team
Test Plan	TM	Define the level of testing and test cases for the system	Team
Test Report	TM	Analyzing test results and feedback to developers, and also to prove the quality of the system to client	Client/Team

Table 4 – Table of documents produced

4. Development Execution

This section describes how the development was executed from the perspective of the respective role.

4.1. Project Management

The first part of planning development of the system was to consider the time constraints put on the project, and determine the amount of time that could be spent on actual development. We estimated that we would have about two

weeks of time to spend on developing the first iteration of the system, out of the 6 weeks we were given.

Given this time constraint, we decided that we would use an incremental development process, and deliver the system in multiple incremental releases, with a pre-defined set of functionality. The development increments, also referred to as phases (P0-P*X*), would result in a new system delivery. This way, we were able to only include the highest prioritized requirements in the initial release (P0), which was the only release targeted for this 6 week project.

To be able to tell what needed to go in each of the planned deliveries, we started an elaborate requirements elicitation process in collaboration with the customer, to accurately identify core functionality and estimate the workload for P0. Once the requirements were defined and approved by the customer, they were prioritized. Since the time constraint of 2 weeks of development was known, it was a fairly straightforward process to define which of the requirements would be included in the P0 release.

Development started in project week 5 with system design, resulting in an implementation proposal document. This document was then used to implement each of the features and requirements targeted for P0. Testing was also a part of the development, and was continuously carried out in parallel with the implementation to assure requirements were met.

4.2. Software Architecture

The development process from an architectural viewpoint started with an evaluation of available open source commerce systems contra in-house development to fulfill the high priority requirements in a satisfactorily way. The solution chosen was the open source CMS (Content Management System), Drupal with the third party module Ubercart which makes it a full featured e-commerce system. Also, a language module with translations was added to localize the site. The communications between the shop and supplier/transporter modules was implemented by SOAP services. These modules only have basic functionality, e.g. logging due to the decision to only implement the MUW system. However, these modules prove that the communication channel works and exists.

4.3. Database Design

The database is an integral part of the system that drives The Unlimited Well, and as such, it was created early in the process using the systems domain model as a base. The original idea was to make a lightweight database that could be implemented on any SQL-service, so that any command that was not SQL-standard was out. The database services available were MySQL and Oracle, both available for free to C10, and the first choice was MySQL since none in C10 liked the Oracle interface. Still, the idea was to follow SQL-standard so that UW would not become dependent upon one specific service. The choice of Drupal, and later its extension Ubercart, as the projects base cemented the use of MySQL and the requirement to use only SQL-standards was softened. Since Drupal/Ubercart almost completely lacked documentation on its database, a considerable amount

of time was spent to analyze what was already there and what needed to be added.

See Appendix C for the database schema.

4.4. Graphical Design

The graphical user interface was designed during Phase 0 in the low-level design sub-phase. It was clearly intended to be both aesthetically attractive and to aid the fulfillment of the demands on usability. We used the common way of creating the graphical design by first producing a mock-up. It was made by reflecting on necessary features and, only after evaluating what features and functionality was really needed, where they should be placed etcetera, the finer aesthetically design parts was taken in account.

Due to a later change in the use of CMS from osCommerce to Drupal, the first mockup was disregarded in favor of a second one, which also fulfilled the requirements. This turned out being the easiest way to do it since it allowed an implementation more suited for the new system.

4.5. Testing

The testing of the project The Unlimited Well adheres to the project plan and software development life cycle selected for this project. In the modified V-model life cycle testing takes place in each phase of development and also at the end of the project. For each phase, unit testing and integration testing would be done for the features implemented in the particular phase. The test plan document gives the overall test methodology implemented in this project including the test cases for each individual feature and also for the overall integrated system. The test report document for the implemented phase 0 has been done and updated with the test results for the functional features of phase 0.

5. Result

This section describes the actual results of the development process.

5.1. Project Management

From a managerial point of view, project progress was measured by a number of milestones and decision gates (DG), as well as completed implementation tasks. Since these tasks had been prioritized and separated into phases, it was fairly easy to analyze and estimate how far the development had come at any given time. Decision gates were used to measure the documentation progress, and we were able stay on track with the planned DGs for the most part.

The distribution of work was clearly defined from the start, although people helped out where help was needed, since some areas required more work than others. This is an obvious advantage of working in a small development team.

At the end of every meeting during development we looked at what needed to be done, both in terms of development as well as for documentation. When this work had been distributed, everyone was able to work relatively independently,

giving continuous progress updated to the Project Manager. This worked very well, and communication was handled through instant messaging or e-mail for the most part.

The main challenge of the project was the short amount of time allocated for actual development, and the complexity of this kind of system. The use of the open source system Drupal helped with the rapid development of the P0 delivery.

5.2. Software Architecture

The e-commerce site is implemented using an open source system, Drupal. The actual shop is implemented by using a third party module, Ubercart. These two systems together provide the functionality needed for a working e-shopping system. To communicate between the shop and suppliers/transportation, two simple SOAP services was implemented and provides logging for testing and backtracking. This solution fulfills all the P0 requirements, found in the Requirements Specification document.

5.3. Database Design

The Unlimited Well's database is built as an extension to the database used in the module Ubercart. This gives The Unlimited Well a plethora of possibilities that can be utilized if Coira should decide that they want to further develop the system. Currently, it allows multiple suppliers to provide the same or different items. It also allows for transporters to store routes and coordinate the transportation of goods between suppliers and customers. The suppliers are also able to specify individual transporter preferences. The database used is MySQL, simply because of the choice to use Drupal as the base for The Unlimited Well.

5.4. Graphical Design

The result of the graphical design allowed the website to contain features such as:

- Login at any time
- Customer registration
- Change customer details
- Display products and details
- Cancel orders

These features were also stated in the requirements of development phase 0. The creating of the graphical design was straight forward after having straighten out what features was needed what the basic layout would look like. This also allowed for an easy remake of the finish of the design a little bit later, which we made due to the change of CMS from osCommerce to Drupal.

Drupal is built up by components and uses themes to allow different front-end designs. We created a new theme and customized it accordingly to the mockup.

5.5. Testing

The test report document for Phase 0 has been developed and updated with test results carried out for the phase 0. During the testing of phase 0 it has been

shown that the all the functional requirements have been implemented and the web-portal, www.theunlimitedwell.com, is found to be working with all the functional features of phase 0 and all the functional requirements have been implemented. The testing has been carried out in two levels - unit and integrated testing. The working of systems PUW and LUW has been checked and tested with the system and database log to see the flow of events happening when there is activity in the user interface level i.e. MUW. Overall the results for phase 0 have been found satisfactory given the various limitations the project has come across and a clearance for starting Phase 1 can be given at this point of time.

6. Discussions

This section describes discussions, encountered problems and decisions during the development process.

6.1. Project Management

Given the time constraint of this project, careful planning was needed and a lot of time was spent on just that, planning. Since we were a team of six people, almost immediately after the project started the Software Architects started investigating different possibilities and ways of implementing the system as fast as possible. The information they came up with helped greatly with planning, since it allows us to approximate the work needed for implementation.

The team members showed commitment throughout the project, and very few issues or conflicts came up. By having meetings almost every other day, everybody was very aware of what was going on in the project and what needed to be done. Any individual misunderstandings or uncertainties where quickly discovered and diverted as a result.

A project of this magnitude needs a lot more time to be executed properly, as the element of stress is introduced immediate, and might affect the overall productivity. By limiting the functionality we were able to handle this, but there was no time for reviewing or correcting changes or decisions throughout the project. Pressure and stress rarely results in high quality software, and this should be considered for future development phases of The Unlimited Well.

6.2. Software Architecture

In the investigation phase of the development, the open source project osCommerce was considered as a platform for the implementation. Due to the poor code structure and documentation, this alternative was omitted. The second idea was to build the solutions from scratch. This option would provide total control and understanding of the system. However, after the requirements elicitation, this option was infeasible. Fulfilling security, availability, portability requirements among others would be hard to accomplish given the timeline. The third idea of using the open source project Drupal as a core was chosen. Drupal provides modularity and is extensively documented and widely used. The community around Drupal is very extensive and there is a large contribution of third party modules to integrate with the system. One of these modules is Ubercart that a fully integrated e-commerce portal to Drupal. No additional

configurations of e.g. databases were needed, provided this solution. Naturally this solution was chosen due to the time limitation and the extensive functionality included.

To communicate with suppliers and transportation a Drupal SOAP module was considered. However, this module was more extensive than the current need. Instead, a decision was made to implement in-house lightweight SOAP-services, to provide the communication channel. These services can be called from the e-commerce site, and provide logging of the call chain.

To ensure global accessibility for customer testing, the domain www.theunlimitedwell.com was purchased.

6.3. Database Design

The design of the database was a pretty straightforward matter. The biggest choice was what database server that UW should use. Given time and money constraints, two systems that were easily available where considered. These were Oracle and MySQL. Since C10's employees hold a deep dislike for the interface and implementation of the Oracle service available, the decision was leaning in MySQL's favor. Then when the decision to use Drupal and Ubercart was made, the choice of MySQL was cemented since that where the database of Drupal/Ubercart. The choice of Drupal/Ubercart also led to some trouble, since they, upon install, creates a database for internal use. This database schema is poorly documented and a lot of effort was spent trying to map the structure. This was worth the time spent, since the mapping uncovered that almost half of the tables required by the design was already there. Thus, the implementation became a lot smaller as well as gaining some free functionality.

Since a database is an integral part of the system and a redesign is costly in terms of data loss if the system is running, even if the system is only in alpha-stages of its incremental development. Thus, the entire database was required to be finished for the first delivery.

6.4. Graphical Design

Doing a mockup seems to be the best way of creating the design for implementation. In our case we had decided on beforehand to use a modifiable CMS system. Event though these systems are normally possible to modify in any preferred way, they normally come with some type of customizable template which means that taking the implementation of the design in account while creating the design can really ease the workload. In that way you do not need to implement every part of it from scratch.

This way of changing the design finish after CMS system cannot be said to have affected the aesthetical quality of the site. Of course this is always in the eye of the beholder and the second system just happened to allow for a different design finish. The basic layout and the placement of features could be kept thus fulfilling the same requirements as before.

It was mainly due to the splitting of the design in "features and layout" and "design finish" that the redesign was relatively easy make.

6.5. Testing

The testing of this project has gone through various discussions and analysis before the final methodology was finalized. The first and foremost of issues was the timing of the testing. The issue was sorted out after the project life cycle model was developed and finalized. The selection of life cycle model meant that the testing would be a continuous process, which occurs in every phase of the project and runs through out the project. The next issue to be considered was the features to be tested as the limited resources mainly time constraints meant that the testing would be carried out only for the functional features to be implemented in Phase 0 of the project. Although the test plan document was prepared for the overall project, including the testing to be carried out in the future phases of the project. Also the test report document for the Phase 0 was developed and the unit testing of individual modules and also the integrated module was carried out in the Phase 0. And the results were found to be satisfactory as the web-portal was functional with all necessary features implemented and in a state to upgrade with a short notice. This was good news for the team as the team was able to produce a good quality product in the short time available for the development and implementation.

7. Conclusions

The project The Unlimited Well has been a great learning experience for the whole team. The initial resource constraints put on the project included work force, time and financing. It was a challenge for Company 10 to satisfy the customer specifications with the limited resources available. But the company came up with a project plan and a life cycle model, which enabled the company and the client to continue the project again in the future when the resources and new requirements are provided.

Careful planning was a major factor in the success of the project. The project always had a high risk factor to begin with as there were a lot of constraints involved. But the extra efforts in the project planning phase and the dedication of the team ensured that the project did not encounter any major setbacks and that the milestones were met with good accuracy resulting in the successful completion of the project.

Company 10 has delivered a fully working system, fulfilling the P0 requirements as agreed upon by the client. The P1 phase has been planned for in terms of requirements and development tasks, but the time planning still remains. It is up to client to decide and inform Company 10 about the future of the project.

Throughout the project, all team members have shown commitment and been able to work independently and efficiently.

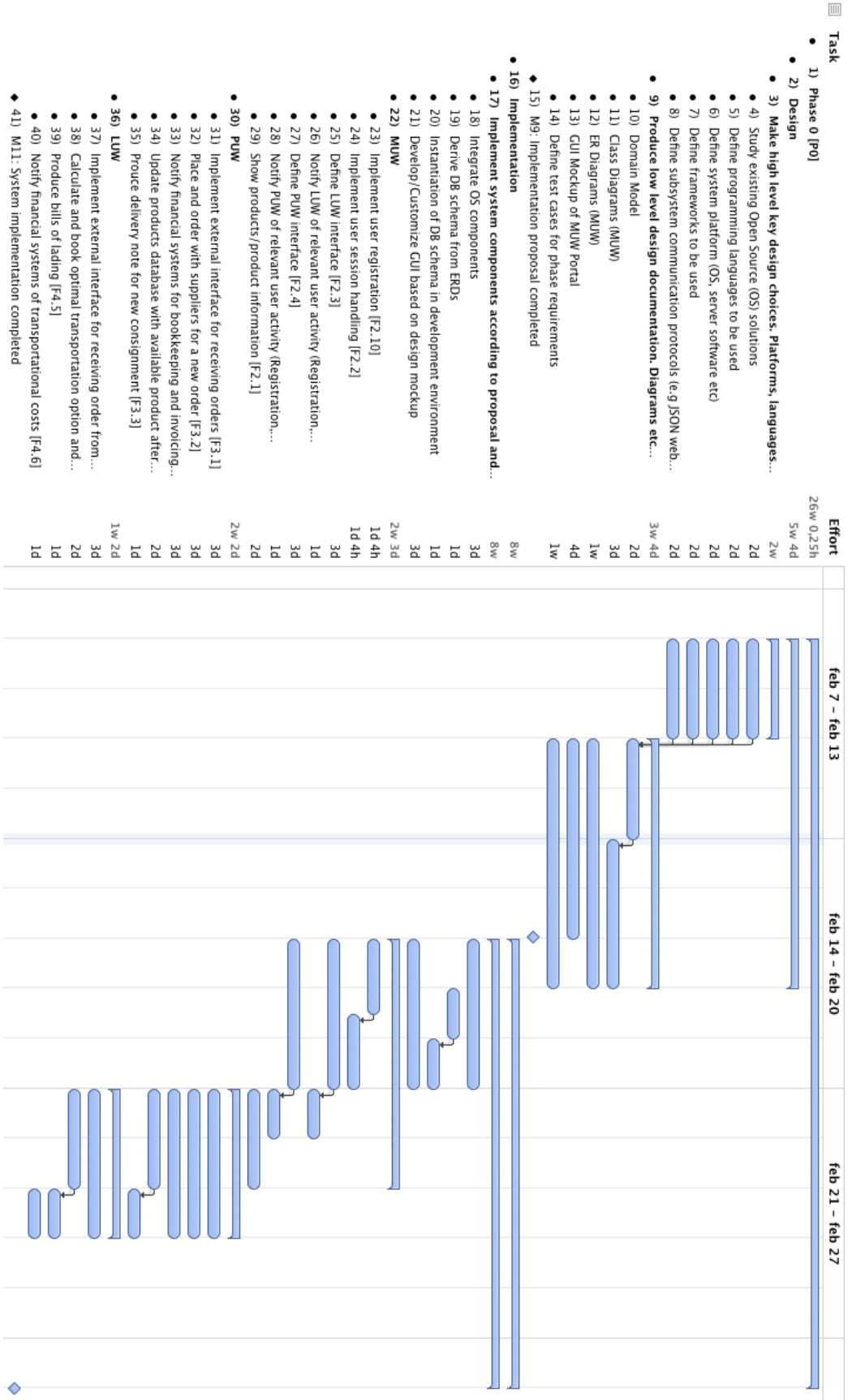
Appendix A. Project Time Plan

Timeplan

Project:	The Unlimited Well
Project group:	Company 10
Orderer:	Central Operative & Independent Retailers Acquirer (Coira)
Course:	EDA382
Date:	2010-02-02
Version:	1
Responsible:	Sebastian Rehnby
Approved:	

	ACTIVTIES AND MILESTONES	TIME	WHO	WHEN (week numbers)							
#	Description	Hours	Initials	3	4	5	6	7	8	9	
1.	**Organise project team**	28									0
1.1	Assign project manager	12	All	12							12
1.2	Implement group agreement	12	All	12							12
1.3	Detail project organisational structure	4		4							4
	M1: Project organisation defined and approved										0
2.	**Analyse project background and goals**	48									0
2.1	Create background description	16	VS/JM/HN	16							16
2.2	Describe aims and goals of the project	20	VS/JM/HN	20							20
2.3	Identify decision gates and deadlines	6	SR	6							6
2.4	Identify project deliverables	6	SR	6							6
	M2: Project Directives verified and approved										0
3.	**Project planning**	60									0
3.1	Perform resource and time allocation	14	SR/VS		14						14
3.2	Define meeting and coordination plan	12	SR/VS		12						12
3.3	Define project activities	6	SR/VS		6						6
	M3: Project activities defined										0
3.4	Define areas of responsibilites throughout project	8	SR	4	4						8
3.5	Define and verify milestones	20	SR/VS		20						20
	M4: Project plan verified and approved										0
4.	**Analyse system requirements**	178									0
4.1	Analyse project description	44	MR/HS/JM	20	24						44
4.2	Collect functional system requirements	38	MR/HS/JM		8	30					38
4.3	Collect functional subsystem requirements	24	MR/HS/JM		8	16					24
4.4	Collect non-functional system requirements	24	MR/HS/JM		8	16					24
4.5	Define use cases	24	HN		8	16					24
4.6	Prioritize requirements	24	MR/HS/Client		4	20					24
	M5: Completed Software Requirements Specification										0
5.	**Development planning**	64									0
5.1	Define development approach	8	SR/MR		4	4					8
5.2	Define development activities	20	SR/MR			4	16				20
	M6: Development process defined and approved										0
5.3	Define tasks for each development activity	14	SR/MR			4	10				14
5.4	Allocate and estimate time resources for tasks	8	SR				8				8
5.5	Prioritize tasks int o phases according to requirements P0-Px	10	SR/MR				10				10
5.6	Verify development timeline and deliverables	4	SR/MR				4				4
	M7: Development Plan verified and approved										0
6.	**System design**	82									0
6.1	Make high level key design choices. Platforms, languages etc	20	MR/HS			10	10				20
6.2	Produce low level design documentation. Diagrams etc	20	MR/HS				20				20
6.3	Identify implementational phases according to system requirements	10	MR/HS				10				10
	M8: Design specification completed and approved										0
6.4	Identify system dependencies within the design	6	MR/HS				6				6
6.5	Define order of implementation for system/subsystem components	6	MR/HS				6				6
	M9: Implementation proposal completed										0
6.6	Define test cases based on implementation proposal	8	VS				8				8
6.7	Prioritize test cases according to implementational phases in design specification	12	VS/MR/HS				12				12
	M10: Test plan completed and approved										0
7.	**System implementation & testing**	202									0
7.1	Analyse implementation proposal	16	MR/HS/JM/HN					16			16
7.2	Implement system components according to proposal and phases defined, Phase X	140	MR/HS/JM/HN/SR					80	60		140
7.3	Develop system tests for phase X	20	VS					10	10		20
7.4	Execute Phase X test suite	8	VS					4	4		8
7.5	Analyse test result	4	VS					2	2		4
7.6	Produce test reports and feedback to developers	10	VS					6	4		10
7.7	Verify successful testing for Phase X	4	VS					2	2		4
	M11: System implementation completed										0
	M12: System testing and verification successfully completed										0
8.	**System demonstation**	52									0
8.1	Deploy system in demo environment	12	MR/HS/JM/HN							12	12
8.2	Prepare promotional material & documentation	32	SR/MR						12	20	32
8.3	Demo project for client and additional stakeholders	8	SR							8	8
	M13: System demonstation completed										0
9.	**Documentation**	46									0
9.1	Collect and package system documentation	4	HN/VS/JM						4		4
9.2	Produce final Project Report	22	All						22		22
9.3	Deliver required documentation to client and stakeholders	4	MR/HS							4	4
	M14: System documentation finalized and approved										0
9.4	Perform postmortem project analysis and evaluation	16	All							16	16
	M15: Project completed, entering maintenance mode										0
		760		100	120	120	120	120	120	60	760

Appendix B. Development Time Plan



42) Testing and Verification 12w 1...
43) Produce phase Test Plan 1w 0,25h
44) M10: Test plan completed and approved
45) MUW 7w 1d
46) Is System working in different platforms—Linux, Windows,... 3d
47) Is the system working in various browsers—IE, Chrome,... 3d
48) Is the system globally access—system will be run from... 3d
49) Are products visible in the site [F2.1] 3d
50) Are product details available [F2.1] 3d
51) Are products classified properly [F2.1] 3d
52) Access to the systems at all times [F2.2] 3d
53) Access to home page at all times[F2.2] 3d
54) User activities tracked in the system[F2.3] 3d
55) Is Purchase order of User tracked by LUW[F2.3] 3d
56) Is a registration option available for the customer on the... 3d
57) Is the web portal available in all local languages including... 3d
58) LUW 2w 2d
59) Is an order placed in MUW received/accepted by MUW... 2d
60) Is Placing of orders with the supplier working.[F3.2] 2d
61) Is a delivery note produced for an consignment [F3.3] 2d
62) Is a delivery note produced for every consignment[F3.3] 2d
63) Is financial system updated with data of every transaction... 2d
64) Is database updated with product information.[F3.7] 2d
65) PUW 1w 3d
66) Is a transporter booked for each order placed[F4.1] 2d
67) Is a transporter booking done based on the order delivery... 2d
68) Is a transporter booking done based on the cheapest... 2d
69) Is delivery of the product reported correctly. 2d
70) M12: System testing and verification successfully completed

Appendix C. Database Schema