Developing a Reclaim Information System
Case: Lemminkäinen Betonituote

Kolehmainen, Jukka

2009 Laurea Kerava

Laurea University of Applied Sciences
Laurea Kerava

Developing a Reclaim Information System
Case: Lemminkäinen Betonituote

Jukka Kolehmainen
Information Technology
Thesis
April, 2009

Laurea-ammattikorkeakoulu **Tiivistelmä**
Kerava
Tietojenkäsittelyn koulutusohjelma
Digitaalinen media

Jukka Kolehmainen

Reklamaatiotietojärjestelmän kehittäminen Lemminkäinen Betonituotteelle

Vuosi 2009 Sivumäärä 37

Opinnäytetyön päämääränä oli kehittää Lemminkäinen Betonituote Oy:lle tietojärjestelmä reklamaatiotietojen tehokasta ja rakentavaa käsittelyä varten. Lemminkäinen Betonituote Oy kuuluu Lemminkäinen Oyj:n Rakennustuotteet toimialaan. Lemminkäinen Betonituotteella on kaksi tuoteryhmää. Elemento-tuoteryhmään kuuluu porras- ja mosaiikkibetonituotteet ja Formento-tuoteryhmään ympäristötuotteet kuten pihakivet.

Yhtiön reklamaatiotietoja on tähän saakka kerätty yksinkertaiselle Word-lomakkeelle joko sähköisessä tai paperimuodossa. Tietoja on hallinnoitu järjestämällä lomakkeita eri kansioihin. Vaikka välittömät toimenpiteet asiakkaalle saadaankin suoritettua, ennalta-ehkäiseviä, kehittäviä toimenpiteitä ei ole seurattu ja suoritettu järjestelmällisesti. Reklamaatiotiedoille on toivottu tehokkaampaa käsittelytapaa, jota sertifikaatti edellyttää.

Lemminkäinen Betonituotteella on toistaiseksi vielä käytössään yksinkertainen asiakashallintajärjestelmä, joka ei sisällä työkalua reklamaatiotietojen käsittelyä varten. Lemminkäinen Betonituote tarvitsi alan ja yrityksen tarpeisiin tarkasti räätälöidyn järjestelmän. Uusi järjestelmä kehitettiin yhteistyössä tulevien käyttäjiensä kanssa. Opinnäytetyössä esitellään ohjelmiston kehityksen määrittely- ja suunnitteluvaiheet. Ohjelmiston toteutukseen ja käyttöönottoon liittyvät dokumentit kuten ohjelmakoodi ja käyttöohje on jätetty työn ulkopuolelle. Järjestelmä toteutettiin PHP-ohjelmointikielellä verkkopalveluksi ja se tallentaa reklamaatiotiedot XML–tiedostoina. Käyttöönotto on aloitettu vaiheittain pilottiryhmästä alkaen.

Asiasanat: ohjelmistokehitys, tietojärjestelmät, reklamaatio

Laurea University of Applied Sciences
Kerava
Information Technology
Digital Media

Abstract

Jukka Kolehmainen

Developing a reclaim information system, Case: Lemminkäinen Betonituote

Year 2009 Pages 37

The goal of this thesis was to develop an information system for the client, Lemminkäinen Betonituote Inc., to efficiently and productively process their reclaim information. Lemminkäinen Betonituote Inc. is a part of Lemminkäinen Ltd.'s Construction Products branch. Lemminkäinen Betonituote has two product groups. The Elemento product group consists of concrete stair and mosaic products. Formento consists of environment products such as garden stone.

Until now Lemminkäinen Betonituote's reclaim information has been collected on a simple Word-form either on an electronic form or on paper. The information has been maintained by moving the forms into a number of different folders on a network hard drive. Even though the immediate compensations or other actions towards the client have been carried out properly the important preventative or progressive actions have not been followed and accomplished systematically enough. There is a need for a more effective way to process reclaim information and it is also demanded by the certification.

Lemminkäinen Betonituote utilises a simple customer management system which does not include a tool to process reclaim information. The company needs a system directly customised to meet the demands of the company and the branch. The new system was developed in co-operation with its future users. The definition and design phases of developing the system are presented in this thesis. Documents related to the implementation and the deployment of the system have been left out. The system was developed as a web service using PHP programming language and it stores reclaim information as XML files. Deployment is carried out phase to phase starting with a pilot group.

Keywords: software engineering, information systems, reclaim

Contents

Introduction

Lemminkäinen Betonituote Inc. produces and sells concrete and granite products and it is a part of Lemminkäinen Ltd. Construction Material Group. Their subsidiaries are Suonenjoen Betonituote Inc. and Elemento Inc. Savonlinna. The company’s head office is located at the Sammonmäki industrial area in Tuusula.

Lemminkäinen Betonituote has two product groups: Elemento and Formento. The Elemento products consist of stair elements and mosaic concrete products. The Formento products are environment products such as garden stone, garden tile, edge stone, wall stone and walls out of concrete and granite.

Lemminkäinen Betonituote needs a tool for handling their reclaim information. Earlier the reclaim forms were filled based on telephone calls and electronic mail. The forms were archived and it did not necessarily emerge whether the reclaim issues had been solved or not and thus the possible needs for further development were not necessarily being systematically discussed within the organisation. The customer relationship management system used by Lemminkäinen Betonituote does not include a feature for handling reclaim information.

The reclaim form has to be available in a suitable ready to use form for all the office personnel in the company. It was twined together as a part of the Lemminkäinen Betonituote’s intranet. This way reclaim information can be collected as efficiently as possible and without intermediates and the documents can be classified and arranged depending on the stage which their processing has been proceeded to.

The personnel of both product groups Elemento and Formento had their own way to store and process the reclaim information concerning their products and contracts. Reclaim information on Formento products was filled on electronic forms which were then distributed via electronic mail among the personnel concerned. Occasionally a reclaim form was received from a contracting site filled in by a member of Lemminkäinen Betonituote personnel. The paper forms were distributed and archived in their original paper form. Elemento personnel had their own Excel form on which the reclaim information is stored.

The names of the persons in charge of satisfying the customers were written on the reclaim forms. It had also been a custom to include the actions to prevent similar trouble in the future and to include the names of the persons who are in charge of seeing that those actions are accomplished.

The main objective of development and the reason for launching the project was specifically to create a tool to help systematically control the accomplishment of the preventive actions for example on the production plants and the contract worksites. Controlling the accomplishments is had not been organised systematically enough.

The actions to prevent bias were not always properly written into the bias report. They were meant to be written down in the monthly meetings. Matters were discussed and preventive actions were taken. Still in a number of cases clear and consistent notes were not documented. The lack concerned the organisation as a whole including administration, product group management, contracts and production management.

Minor and common reclaim issues such as colour defects or small fractions on the products or minor delivery defects are not handled one by one. For more severe reclaim issues and groups of reclaims preventive actions have to be scheduled and development discussions must be carefully documented. Reclaims will have to be classified in a reasonable manner. Through the classifications of the reclaim forms must emerge the state of processing of the reclaim issues. Both the immediate actions to be taken on the client's side and the preventing actions on the organisation's side must be carefully documented. There has also been lack in documenting the negotiations with the client.

Developing an application to help process reclaims is a part of the intranet development project at Lemminkäinen Betonituote. The intranet project concerns the company's intranet as a whole and especially the way how the information is presented on the intranet. The company's intranet has been in use since 2006. There have not been enough resources to begin with to efficiently manage and update it. During the past year more resources have been concentrated its way and slowly the intranet development project has started to advance.

During the reclaim information system project my PHP skills and knowledge have developed – especially working with and creating XML documents. The project work demanded using the PHP Simplexml functions and the more complicated to use PHP DOM library to control the XML documents. The end product of the project the reclaim information system has been produced in a professional manner and the project has consisted of the main phases of software engineering, each phase proceeded carefully.

1 Glossary

1.1 Customer Relationship Management System (CRM)

Lemminkäinen Betonituote Inc. utilises a simple CRM called iActivate Interactive Business Solution to manage their customer information. The company has lately obtained a new CRM called Vineyard. At the moment the new system is in its implementation phase. It is far more sophisticated and wider than iActivate which is actually only a list of customer information.

Vineyard supports the transforming of the operation of an enterprise or an organisation into a more customer friendly manner. Together with the Vineyard experts the demands and goals of the client organisation are studied. The client's business and status quo will act as a baseline for the particular customised solution. (http://www.vineyard.fi)

It is possible to integrate Vineyard with the client's other database solutions such as billing. This makes possible that the information on customers and transactions only have to be stored in one system. (http://www.vineyard.fi)

1.2 Reclaim Notification Form

The Lemminkäinen Betonituote Inc.'s reclaim notification form which was used both as an electric version and as a paper version until the new information system was set up. The Formento product group reclaim form in its electric form was simply a Word document. The Elemento product group personnel use an Excel table to store their reclaim information.

1.3 Software Engineering

Software engineering stands for all the work that the developing of complete software requires. Software engineering may consist of following steps which are program specification, program design, implementation, testing and documentation. By iterating through these five stages of software engineering a project can be completed systematically. This helps in maintaining high quality. (Peltomäki & Silander. 2003.)

Program specification means presenting the needs and wishes targeted to the program in oral, written or in graphic form. Specification includes defining the forms of the program's input and output data. Input data means the human-inputted data to the program. The program conducts procedures based on this data. Output data means presenting the program's actions in a human understandable form. (Peltomäki & Silander. 2003.)

Program design stands for designing the algorithms as in procedures and data structures as in presenting forms of the data used in the program. (Peltomäki & Silander. 2003.)

In the implementation phase the algorithms and the data structures are presented with the chosen programming language. (Peltomäki & Silander. 2003.)

After the program code is completed its working order has to be tested with some programming system. In practise testing is a simultaneous work phase alongside with the implementation. The program is tested piece by piece during the implementation. After reaching a certain milestone the program will be tested as a whole. (Peltomäki & Silander. 2003.)

Documentation is the most crucial part of software engineering. The completed and tested program has to be documented for the end users and maintenance. All the solution procedures are documented, a user's guide is written and the testing phase with the appropriate testing material is documented as well. In the documentation phase a maintenance guide is produced to describe how to further develop the program and which things must be taken into consideration while making changes to the program. (Peltomäki & Silander. 2003.)

In practical software engineering each phase of programming produces a group of documents. In fact the program code produced in the implementation phase is merely a document among the others. It is the most up to detail and accurate of all the documents. It includes specific instructions on how the tasks defined in the specification phase are completed by programming. (Peltomäki & Silander. 2003.)

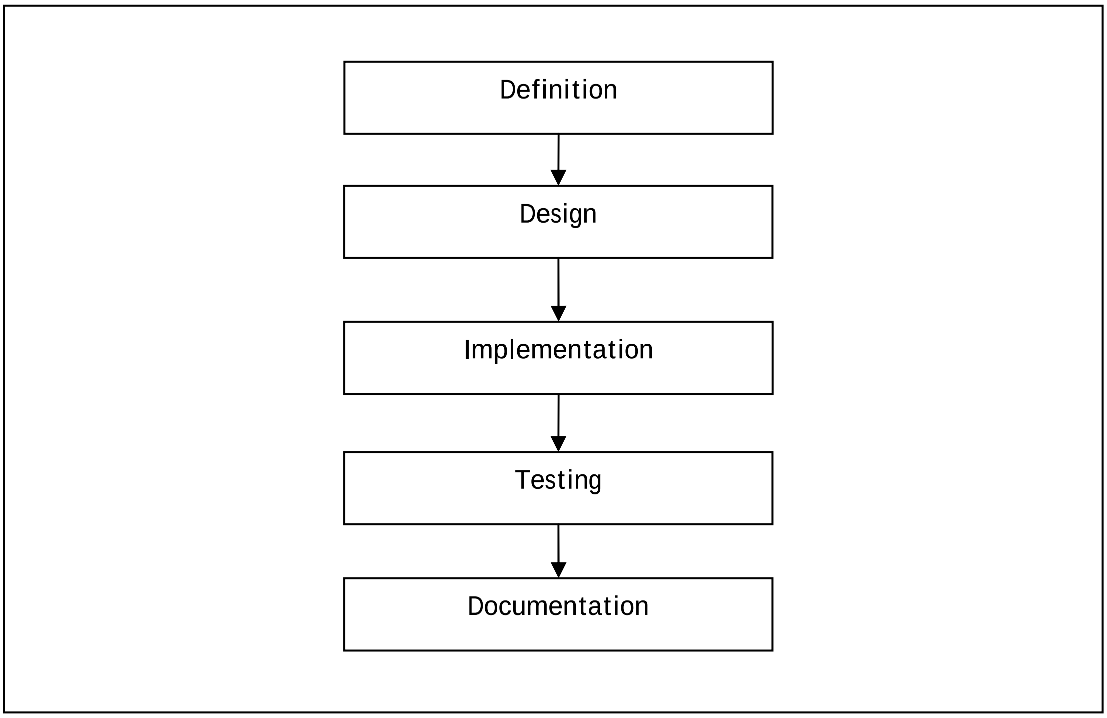


Figure 1: The five stages of software engineering (Peltomäki & Silander. 2003)

1.4 UML

UML is an abbreviation of Unified Modelling Language. It is a graphic modelling language consisting of 13 different diagrams standardised by Object Management Group (OMG) in the year 1997. Six of the diagrams are used to describe structure, three diagrams to describe behaviour and four to describe interaction. UML was originally developed for system and software development. It was generated by joining three of the leading object modelling techniques (OMT, Booch, Oose). The latest UML standard is the 2.0 from the year 2004. (http://atlas.kennesaw.edu)

1.5 UML Diagrams

The Behaviour diagrams included in the UML are Activity diagram, Use Case diagram and the State (Machine) diagram. The UML Interaction diagrams are Timing diagram, Interaction Overview diagram, Communication diagram and the Sequence diagram. The UML Structure diagrams are Component diagram, Composite Structure diagram, Class diagram, Object diagram, Package diagram and the Deployment diagram. It is not necessary to use all the diagrams in every project but only the indispensable ones. (http://atlas.kennesaw.edu)

Behaviour diagrams	
	Activity diagram
	Use Case diagram
	State (Machine) diagram
Interaction diagrams	
	Timing diagram
	Interaction Overview diagram
	Communication diagram
	Sequence diagram
Structure diagrams	
	Component diagram
	Composite Structure diagram
	Class diagram
	Object diagram
	Package diagram
	Deployment diagram

Table 1: There are 13 different UML diagrams (http://atlas.kennesaw.edu)

1.6 The Use Case Diagram

A set of scenarios describing an interaction between a user and a system is called a use case. A use case diagram is used in nearly every project to display the relationship among actors and use cases. (http://atlas.kennesaw.edu)

Use cases are quite facile to draw. Drawing a use case diagram may start with making a list of steps a user might take in order to complete an action. For example placing an order with a sales company might follow steps such as: browsing a catalogue and selecting items, calling a sales representative, supplying shipping and payment information and receiving a confirmation number from the sales person. Drawing the user, the sales person and the appropriate interactions between the actors would form a use case diagram. From such diagram the requirements of an ordering system can be derived easily. (http://atlas.kennesaw.edu)

1.7 The Class Diagram

The class diagram is used to model the class structure and contents. Design elements such as classes, packages and objects are used in class diagrams. The class diagram also displays relationships such as containment, inheritance and associations. Designing a system is described in three different perspectives which are conceptual, specification and implementation. (http://atlas.kennesaw.edu)

1.8 The Sequence Diagram

The sequence diagram is an interaction diagram used to display the sequence of the objects participating in the interaction. The vertical dimension represents time and the different objects are drawn in the horizontal dimension. Sequence diagrams are drawn in a way familiar from piano roll editors in MIDI sequencer software used by musicians. The sequence is drawn as timelines underneath the different objects. (http://atlas.kennesaw.edu)

1.9 The State Machine Diagram

The state diagram or the state machine diagram displays the sequences that an object of an interaction passes through during its life cycle in response to received stimuli. Also its responses and actions are displayed. (http://atlas.kennesaw.edu)

State diagrams are used to demonstrate the behaviour of an object through many use cases of the system. They should only be used for classes where it is necessary to understand the behaviour of the object through the entire system. (http://atlas.kennesaw.edu)

1.10 The Activity Diagram

The activity diagram focuses on flows driven by internal processing and is a special state diagram where most of the states are action states and most of the transitions are triggered by completion of the actions in the source states. (http://atlas.kennesaw.edu)

1.11 The Combined Deployment and Component Diagram

Deployment and component diagrams are called physical diagrams. The component diagram is used to display the high level packaged structure of the code itself. The Deployment diagram is used to display the configuration of runtime processing elements and the software components, processes and objects which live on them. (http://atlas.kennesaw.edu)

1.12 PHP (Hypertext Preprocessor)

PHP is a server side scripting language usually written in an HTML context. A PHP script is only sent to the client after it is parsed by the PHP binary or module which is server side installed. HTML elements in the script are sent as such. PHP code is interpreted and executed. The possibilities of use for PHP are endless. After being combined the PHP output and the HTML are sent to the user's web browser. (http://www.softwareprojects.org)

1.13 XML

XML_is an abbreviation of Extensible Mark-up Language. XML is designed to describe data and to focus on the actual contents of the data. Since the XML tags are not predefined own tags must be defined. To describe the data XML uses a Document Type Definition (DTD) or an XML Schema. (http://www.w3schools.com)

1.14 Apache Server Software

In order to execute scripts written with PHP must the script files be uploaded to a server equipped with PHP support. PHP scripts can be executed on a home computer if an appropriate PHP supporting server software is installed and the service is turned on. Another possibility is to use a PHP emulator such as Nusphere for example. The Lemminkäinen Ltd. server in Pasila is equipped with the latest version of PHP.

2 Project

The developing of the reclaim information handling system was launched as a project and a proper project plan was composed to help maintain the actions and resources concerned.

2.1 The Task of the Project

The task of the project was to develop a tool to handle reclaim information for Lemminkäinen Betonituote Inc. The reclaim information handling system or reclaim information system will act as a part of the company's intranet. The goal was to create a developable tool of which the data input and output is based on accurate facts as much as possible. It is also important that the program will not increase anyone's work load but decreases it and makes working with the company's reclaim information more efficient.

The first version of the tool was built only for handling reclaim information. The client has already made queries of many more features to be added in the future such as handling customer feedback information from clients. All the extra features are left out of this thesis. This thesis concentrates only on developing the basic vital features to handle reclaim information.

The project includes defining, designing, implementing, testing and the deployment of the reclaim information system with all the appropriate documents included. In this thesis the definition and design phases are presented. The implementation, testing and deployment phases including the documentations are left out.

2.2 Division of Responsibilities

The project organisation is small and consists of only myself and the representative of the client Lemminkäinen Betonituote Inc. I represent the executing party and act also as a project manager. The representative of the client Eeva-Liisa Hannu acts as a development manager in the company and is responsible of providing the necessary information for the project. The co-operation in practise has consisted of informal interviews and provision of material for the project by Eeva-Liisa Hannu.

The decision of developing a reclaim information handling system was first found while working on Lemminkäinen Betonituote Intranet development project. The project group that met approximately six times during last year time consists of the following persons: Eeva-Liisa Hannu (development manager), Virpi Juutilainen (company secretary), Jukka Kolehmainen (system specialist), Ilkka Kuusi (designer), Jorma Lamminmäki (contracting), Pirkko Merisalo (marketing manager), Reijo Oksanen (security manager), Antti Sirén (development engineer), Jaani Turunen (landscape designer/sales representative) and Markku Virtanen (controller).

Eeva-Liisa Hannu, development manager
Virpi Juutilainen, company secretary
Jukka Kolehmainen, system specialist
Ilkka Kuusi, designer
Jorma Lamminmäki, contracting
Pirkko Merisalo, marketing manager
Reijo Oksanen, security manager
Antti Sirén, development engineer
Jaani Turunen, landscape designer/sales representative
Markku Virtanen, controller

Table 2: The members of the intranet development group

2.3 Tasks and Partial Tasks

Presenting the tasks of the implementation, testing and deployment phases is necessary since the project plan itself is one of the early documents of the definition phase.

2.4 Specification and Design

The development of the system began with exploring the needs and backgrounds of the client's procedures. Based on this information a process diagram was produced to describe the features of the new system.

A UML model of the system was produced with all the necessary diagrams included. The model was iterated with the client's representative until a satisfactory level was reached.

2.5 Implementation

During the implementation phase the algorithms and data structures vital for the system were written following PHP and XML techniques as close to as designed as possible.

2.6 Testing

The PHP scripts and the XML documents were tested continuously while writing them. When most of the writing was completed the system was tested as a whole and as well with the representatives of the client to receive more wishes and specifications concerning improvements to usability and added functionalities.

2.7 Documentation

The documenting was proceeded continuously through the whole project. The first documents of the project were the subject analysis and the project plan followed by the process diagram and the documents produced by UML modelling, PHP scripts, HTML code, XML structures, commentary lines alongside the scripts, meeting log sheets and memorandums. Manuals describing how to use the system were written for users and administrators and tutorial lessons were arranged. Testing material was created by inputting old reclaim information into the system.

Commentary lines were written on most of the lines in the PHP scripts to avoid obscurity and to help further develop the system possibly by a third person. The comments were written regularly and neatly.

2.8 Project Schedule

The practical work on the project started with the subject analysis on September the 3rd 2007. The informal discussions and requirement specification began earlier. The project plan was completed on schedule on October the 8th 2007. The first deadline for the project was set with Lemminkäinen Betonituote Inc. and it was on December the 31st 2007. As a result of the tight schedule and difficulties in writing some parts of the PHP scripts as was predicted in the risk analysis the schedule was not met. After a brief review of the system with the company’s Intranet development group a new deadline was set to the end of March 2008. The first rough and still rather unstable beta version of the program was delivered to Eeva-Liisa Hannu and Jaani Turunen for further evaluation on January the 25th 2008.

The script writing was done during working hours along with other tasks and the English writing tasks on private time. The intensity of writing varied depending on other work tasks and varying priorities.

After the project plan was completed on October the 8th 2007 the actual work started with specifying and designing the future system. Originally the time reserved for specifying and designing was 60 percent of the 84 days reserved for the entire project and 40 percent for testing and programming the system into working order. The programming phase began instantly after the specifying and designing had been completed into a satisfactory level. The programming work can and has been done side to side with the specifying and designing iterations. The specifying and designing phase was in a satisfactory level and the programming phase started with producing a prototype on October the 23rd 2007. The first specifications and designing took only fifteen days to comply which is 17.9 percent of the reserved 84 days.

2.9 Resources and Funding

The main human resource of the project was me as an employee of Lemminkäinen Betonituote. The company’s development manager Eeva-Liisa Hannu and my mentor, lector Merja Jalava, acted as supporters and providers of information.

Important information was received also from the company’s intranet development group and from all the office personnel with sales personnel in particular.

The project did not oblige large amounts of material or articles to accomplish. Two personal computers were at my disposal. They were the one I use at work at Lemminkäinen Betonituote and my own one which I use at home. There was no need to acquire new hardware for

the project. For designing the UML models I used Visual Paradigm Suite. To write the program codes and scripts I used Adobe Dreamweaver 8, Programmer's Notepad 2 and XAMPP for Windows. The XAMPP package includes the Apache server software with PHP support. I tested the scripts with the Lemminkäinen Ltd.'s server used by Lemminkäinen Betonituote.
I produced the project documents mainly using Microsoft Office products.

The main costs of the project are the total of wages of the personnel involved. The accurate sum would be difficult to calculate since all the personnel have worked continuously on other tasks.

One of the computers is the property of Lemminkäinen Betonituote Inc. and the other one is mine. Both pieces of hardware have been taken from existing resources. No additional funding has been necessary. I used a free student copy of Visual Paradigm Suite to draw the UML diagrams. XAMPP and Programmer's Notepad 2 are free of charge. Adobe Dreamweaver 8 and Microsoft Office licenses are property of Lemminkäinen Betonituote and they were included in the existing resources.

2.10 Risk Analysis

The main risks of the project were programming difficulties that could have and at some extent did emerge. Difficulties with programming could have affected the schedule and delay the project especially if one or more of the programming techniques or scripting languages would have had to be changed during the project.

A risk looming in the future is the new CMS for the Lemminkäinen Betonituote Inc.'s Internet and intranet that will be implemented in one year's time. According to the latest information it will take years before the CMS is changed on the intranet side and it will still have at least some kind of interface that can be used to integrate the system together with the new CMS.

After some reconnaissance it appeared that the biggest extra work would be with the layout of the system rather than the technical side. The layout has to be alike with the company's common intranet layout.

Since the software is mainly based on PHP technique a potential risk may be ceasing or limiting of the PHP support on the Lemminkäinen Ltd. server. This is not current at the moment still. The server is updated with the most recent PHP version on a regular basis and there are a number of people for example in the Lemminkäinen Ltd. IT-management who continuously develop PHP applications.

2.11 Reporting

The news on progression of the project were reported to the Lemminkäinen Betonituote Inc. intranet development project group in their meetings and to their interest groups through memorandums. The company's development manager was briefed on the subject on a daily basis.

When the system was completed and was in its deployment phase it was presented in the Lemminkäinen Betonituote Inc.'s intranet and by electronic mail to the personnel who will be using it.

2.12 Quality Control

The software was tested continuously while building it. When the program code was completed to a certain level it was tested more systematically utilising a pilot group and in co-operation with its other future users. Since the system had not been widely tested a number of problems emerged during deployment and the system was repaired on the fly.

3 Description

Actual work on the project began with making a simple process diagram followed by drawing a group of UML diagrams.

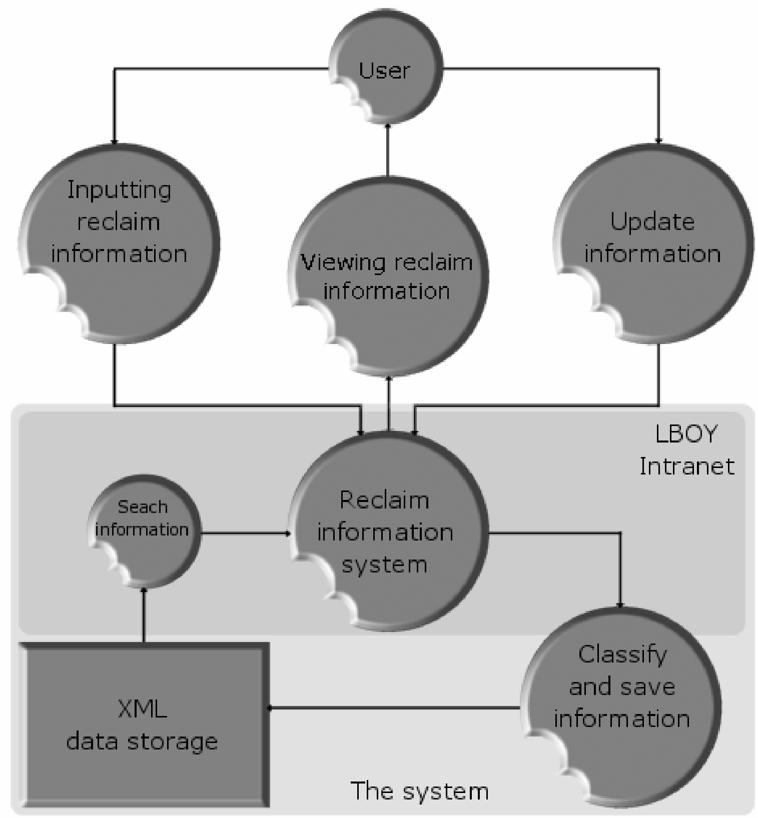

User
Inputting reclaim information
Viewing reclaim information
Update information
LBOY Intranet
Seach information
Reclaim information system
XML data storage
Classify and save information
The system

Figure 2: Process diagram

3.1 Altering Reclaim and Classification Information (Use Case Diagram)

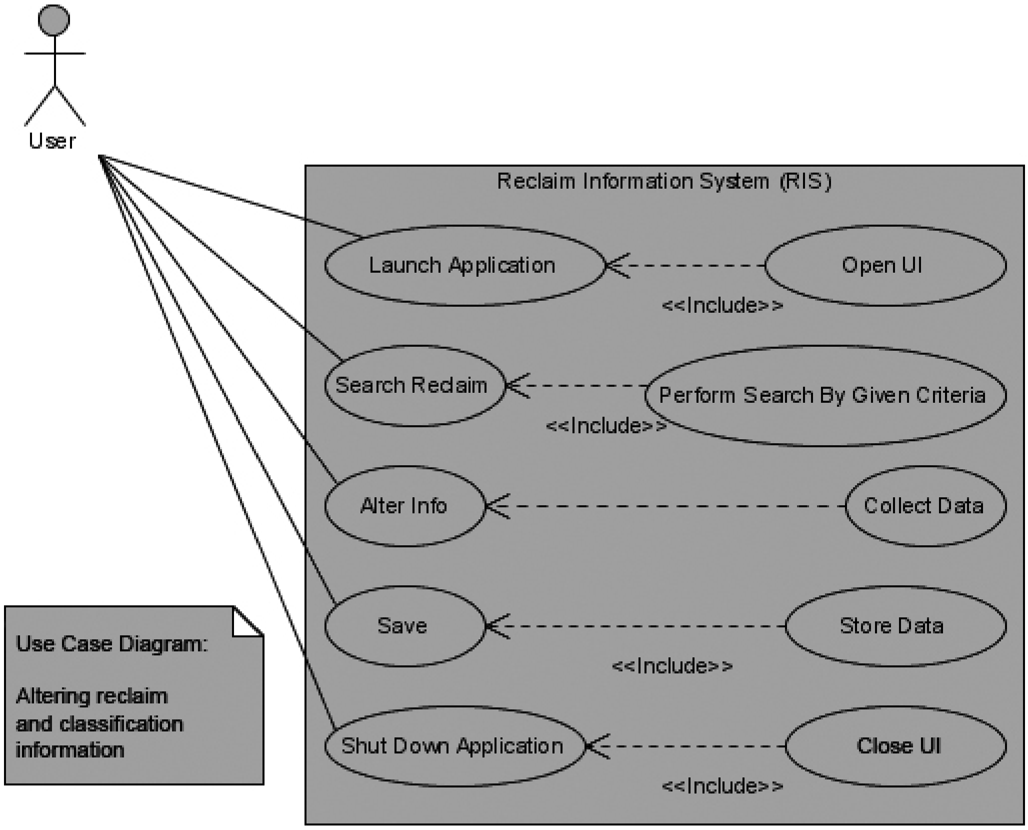


Figure 3: Use Case diagram: altering reclaim and classification information

The user launches the application and chooses to alter reclaim and classification information. First she or he has to either input criteria to the system to search the file wanted or to choose the desired file from the list. After finding the desired file she or he changes for example the state of the reclaim process which is one of the classification criteria. The user then saves and quits the application or navigates to the beginning to perform more tasks.

3.2 Inputting Reclaim Information (Use Case Diagram)

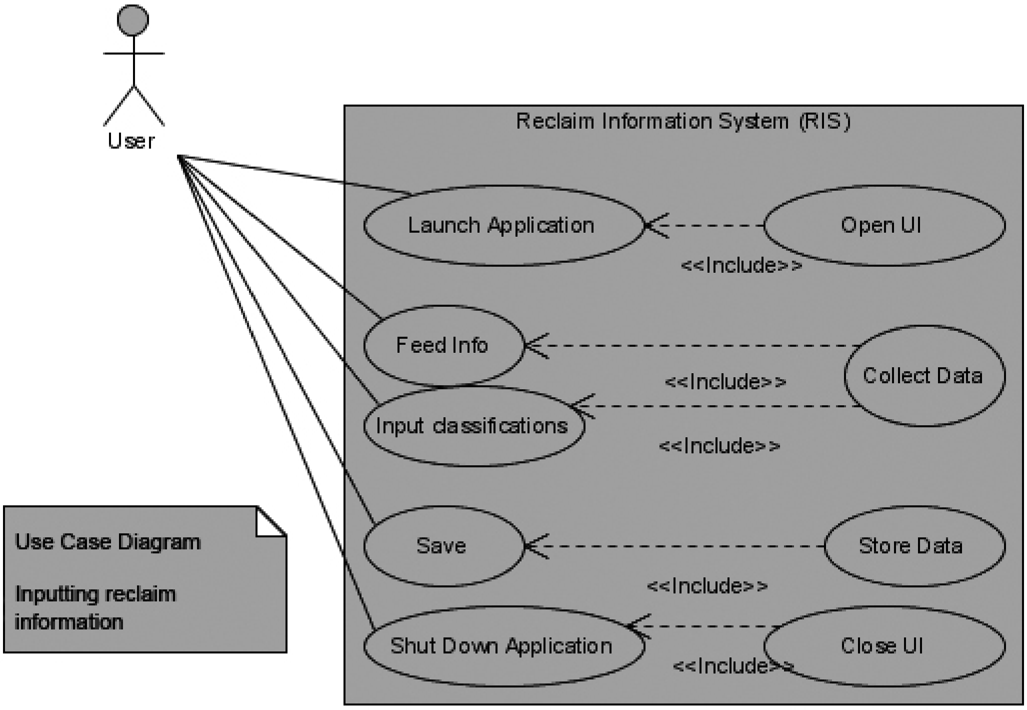


Figure 4: Use case diagram: inputting reclaim information

There are two parties in the diagram which are the user and the reclaim information system. This use case begins with the user launching the tool by activating a link button in the Lemminkäinen Betonituote Inc.'s intranet. Activating the link button launches the main program from Lemminkäinen Ltd. server in Pasila. The program outputs the systems user interface on the screen. The user chooses to input new reclaim information into the system. The system outputs a blank form on the screen and allows the user to type in the information. After finishing the user chooses to save the new information and quit the application. Additionally it is possible to choose to input a new reclaim after saving the first one or simply clear the form and start over again.

3.3 The Class Diagram

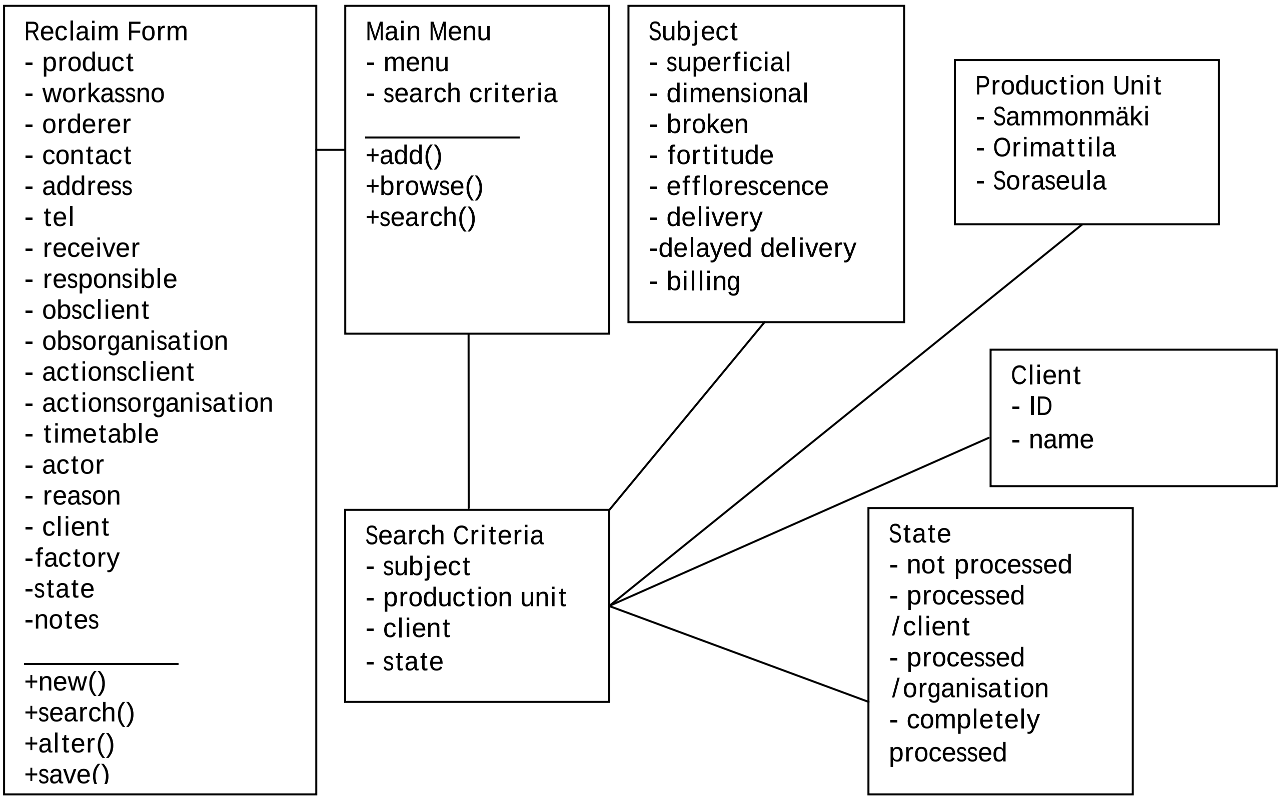


Figure 5: The Class diagram

The Class diagram consists of the reclaim form class and the main menu class which includes the four part search criteria.

The subject class of the search criteria contains the defects or the reasons why a reclaim has been posted. The reasons may be superficial, dimensional, efflorescence, fortitude, delivery or billing defects or a reclaim may concern a broken product or a delayed delivery. More defects and other reasons can be added to the system easily if necessary.

superficial defect
dimensional defect
broken product
efflorescence defect
fortitude defect
delivery defect
delayed delivery
billing defect

Table 3: The reasons for posting a reclaim

The clients are presented by a name and a client ID. Using client IDs will make it possible to integrate with other systems easier since the client IDs will be the same ones that are used for example in the company's customer relationship management system. The Production Unit class contains the Lemminkäinen Betonituote Inc.'s Formento production units: Sammonmäki, Orimattila and Soraseula.

Sammonmäki (concrete stair elements, concrete environment products)
Orimattila (concrete environment products)
Soraseula (concrete environment products, light weight gravel bullions)

Table 4: The Formento production units

There are four different statuses of a reclaim. The first is "reclaim not processed" when the reclaim information is stored into the system but no actions have been taken. "Reclaim processed with customer" is when the customer has been satisfied by for example replacing the invalid goods. "Reclaim processed in the organisation" is when matters concerning the reclaim have been processed in the organisation and the possible preventative actions have been taken. Generally the customer's needs are satisfied first. "Reclaim completely processed" is the status when the reclaim has been processed both with the customer and in the Lemminkäinen Betonituote Inc.'s organisation.

reclaim not processed
reclaim processed with customer
reclaim processed in organisation
reclaim completely processed

Table 5: The four reclaim statuses

It is possible to search and alter reclaim information or fill in a blank notification and save new reclaim information. The form attributes include “product” which is reserved for the name of the product or service in question. “Workassno” is for the current work assignment number. There can be several reclaim notifications related to one work assignment number. “Buyer” is for the name of the company or person who´s representative has filed the reclaim. “Contact” is for the name of the person representing the client (“buyer”). “Receiver” is for the receiver of the reclaim notification. The receiver inputs the reclaim information into the system. The receiver can be for example a member of the Lemminkäinen Betonituote Inc.’s sales staff. “Incharge” is for the name of the person accountable for the reclaim as well as the future preventative and developmental actions concerning it.

“Obsclient” is for the client’s notes. It can be a report from the worksite or a description of damaged goods. “Obsorganisation” is for the notes made on the organisation’s side. “Actionsclient” is for describing the possible compensatory actions towards the client. “Actionsorganisation” is for describing the future preventative and developmental actions concerning the reclaim. “Timetable” is for the deadline for the actions to be taken in the organisation. Actions towards the client are to be taken immediately. “Actor” is for the name of the actor on the preventative and developmental actions in the organisation. “Subject” is for the subject of the reclaim or the reason for posting the reclaim. The subject on the notification form is to be chosen from a drop down menu so the reclaim notifications can be classified according to the subject. “Factory” is for the name of the production unit accountable for the reclaim. “State” is for the current state of processing the reclaim. It can be “not processed”, “processed on the client’s side”, “processed in the organisation” or “completely processed”. “Clientid” is for the client number. “Notes” is reserved for extra notes.

product	product in question
workassno	work assignment number
buyer	buyer/client
contact	contact person representing the client
receiver	receiver of the reclaim
incharge	person in charge/organisation
obsclient	notes/client
obsorganisation	notes/organisation
actionsclient	actions/client
actionsorganisation	actions, precautions
timetable	actions' timetable
actor	actor on actions and precautions
subject	subject/reason for posting the reclaim
factory	the production unit
state	progression state of processing
clientid	client number
notes	a space reserved for notes

Table 6: The form attributes

3.4 Viewing Reclaim Information (Sequence Diagram)

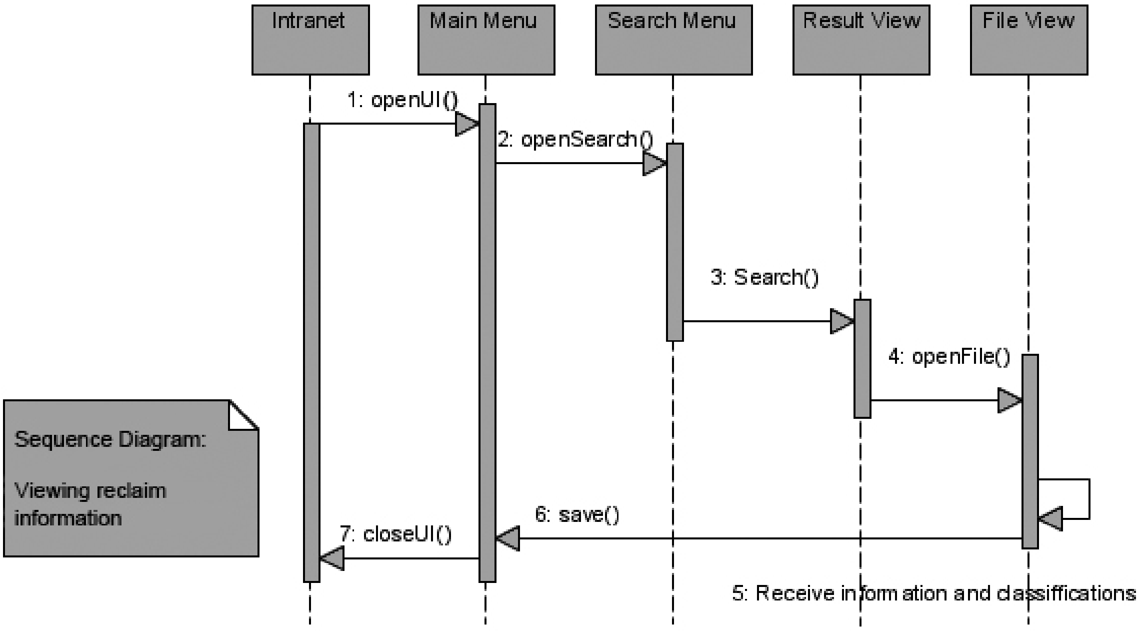


Figure 6: Sequence diagram: viewing reclaim information

The sequence starts in the Lemminkäinen Betonituote Inc.'s intranet when the user activates the link which starts the reclaim information system. The system's user interface is launched and displayed. The first screen is the "Main Menu". In this case the user chooses to navigate to the "Search Menu" to find a reclaim form. Then she or he chooses the search criteria and launches the search. After the search is performed the "Result View" is displayed. It outputs a files list on the screen.

The user is then prompted to choose a reclaim for closer investigation. This happens in the "File View". In the "File View" the user is offered the possibility to alter the information on the particular reclaim case by giving it a new classification tag or adding notes. When the user chooses to save the altered reclaim form the application processes the given information and stores it into the system. When quitting the application the user interface is closed and the user is returned to the Lemminkäinen Betonituote Inc.'s intranet.

3.5 Receiving Reclaim Data (Sequence Diagram)

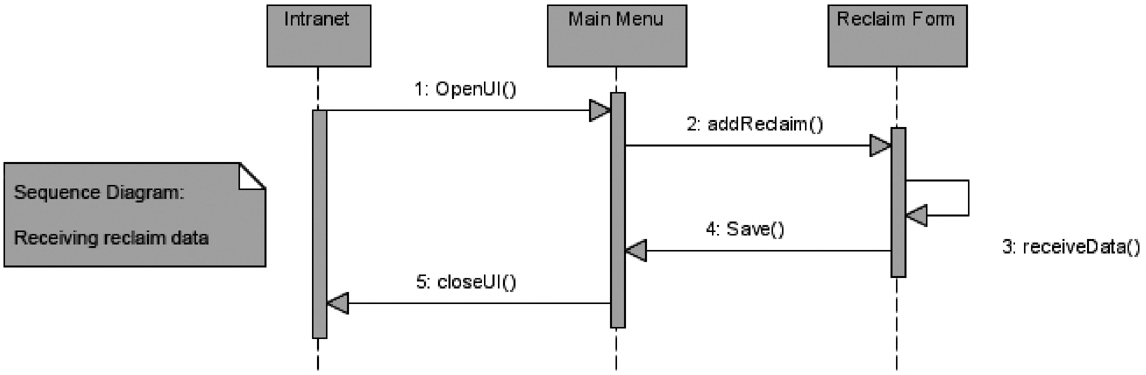


Figure 7: Sequence diagram: receiving reclaim data

This sequence describes the actions taking place in the second use case which is on inputting reclaim data. The user begins by opening Lemminkäinen Betonituote Inc.'s intranet on a web browser and launching the application. In the "Main Menu" she or he chooses to add a new reclaim. A blank reclaim form with appropriate directions is displayed and the user is prompted to input the reclaim data. After the user has chosen to save the new reclaim information the application processes and stores the data. The user quits the application and returns to the company's intranet.

3.6 Altering Reclaim and Classification Information (State Machine Diagram)

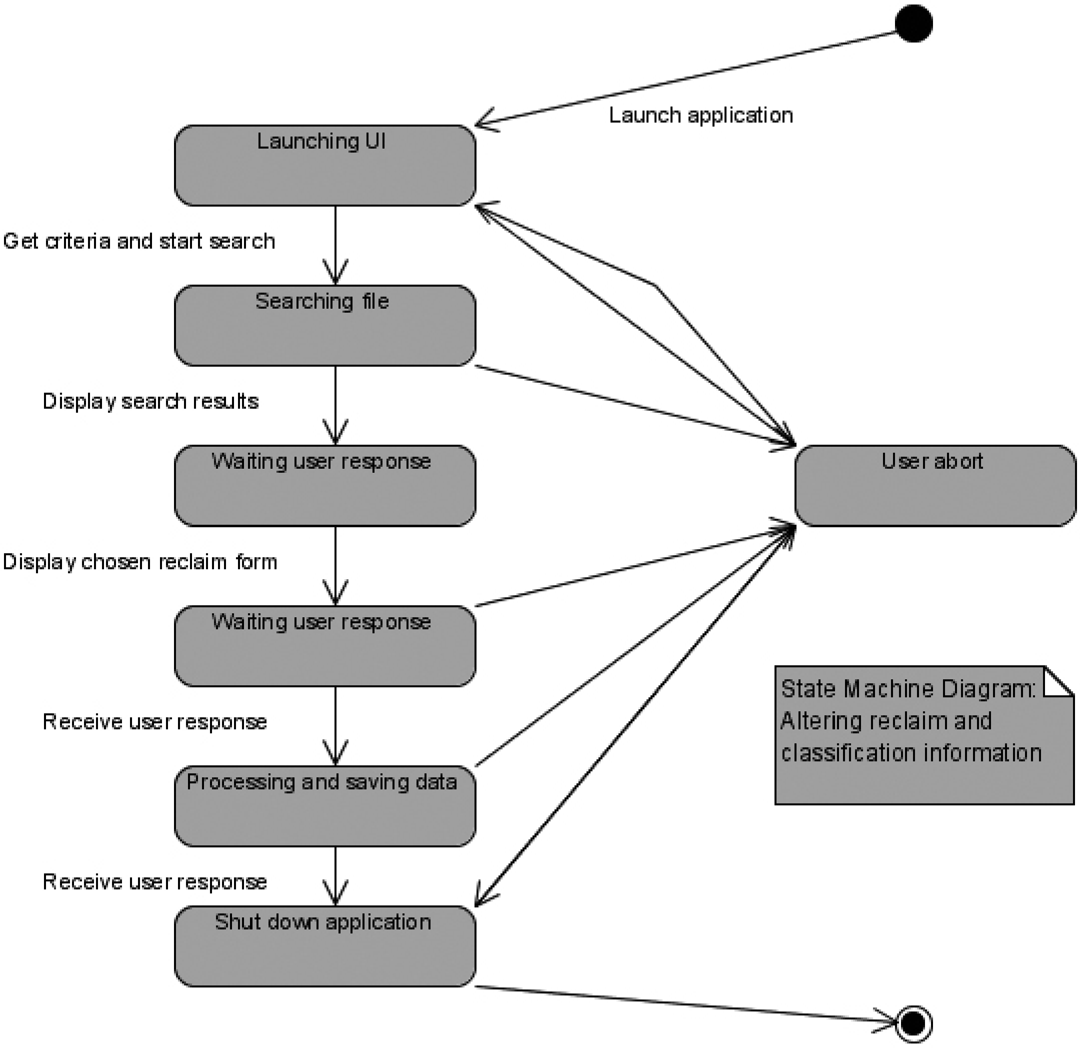
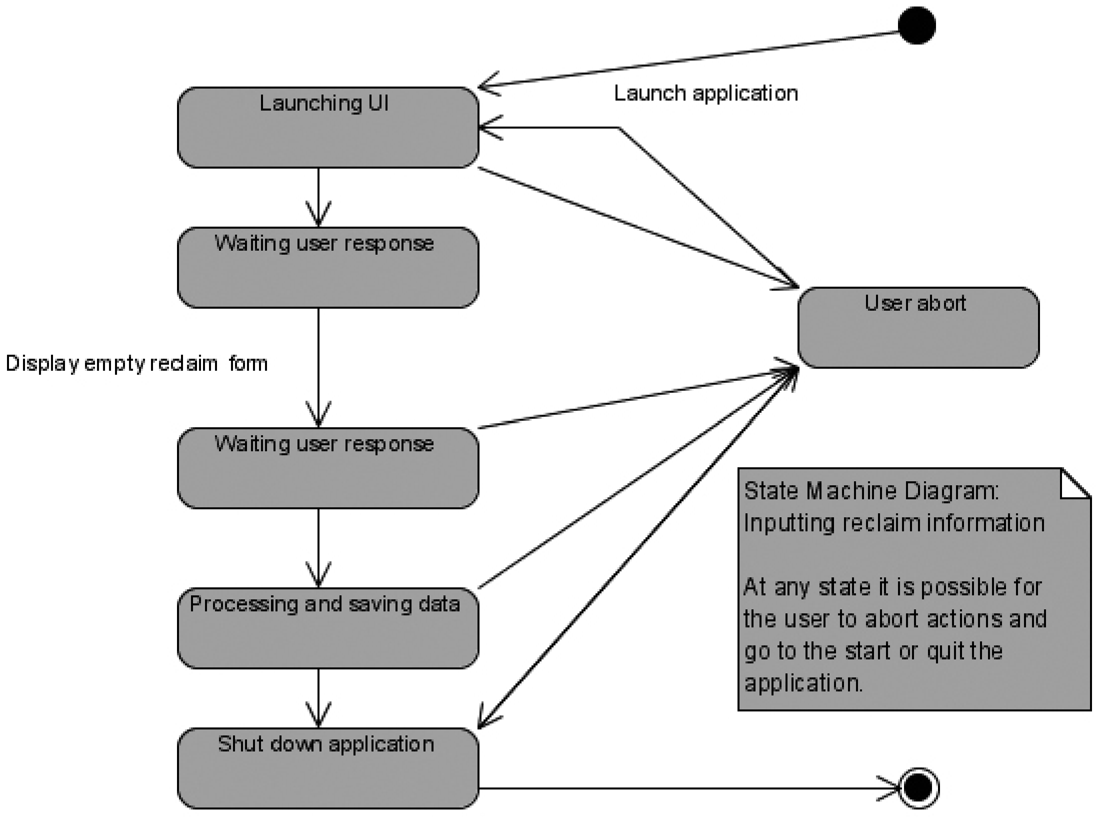


Figure 8: State Machine diagram: altering reclaim and classification information

This State Machine diagram describes the different states that occur in the first use case which is on altering reclaim and classification information. After the user interface is displayed the system stops to wait for the user's response or decision of the next action to be taken by the system. In this case the response leads to the system displaying the "Search Menu" and prompting the user to give the necessary criteria and to start the search. The system searches the file or files and displays them in the "Result View" and prompts the user to choose one of the files.

After outputting the information on the screen the system again waits for the user to view and alter the information for example by giving the reclaim a new classification. When the user chooses to save the altered information the system processes and stores the data. At any part of the process it is possible for the user to abort and quit the application or return to the start.

3.7 Inputting Reclaim Information (State Machine Diagram)



State Machine Diagram: Inputting reclaim information

At any state it is possible for the user to abort actions and go to the start or quit the application.

Figure 9: State Machine diagram: inputting reclaim information

This State Machine diagram describes the different states that occur in the second use case which is on inputting reclaim information. After the user interface is displayed the system stops to wait for the users response or decision of the next action to be taken by the system. This time the “Add reclaim” is chosen and the blank reclaim form is displayed. The system stops again to wait for the user’s response. When the user decides to save the information the system processes and stores the data. The user may now either quit the application or go to the start. At any part of the process it is possible for the user to abort and quit the application or return to the start. It is also possible to save partial information and return to add more at a later point in time.

3.8 An Activity Diagram with Both Use Cases at One

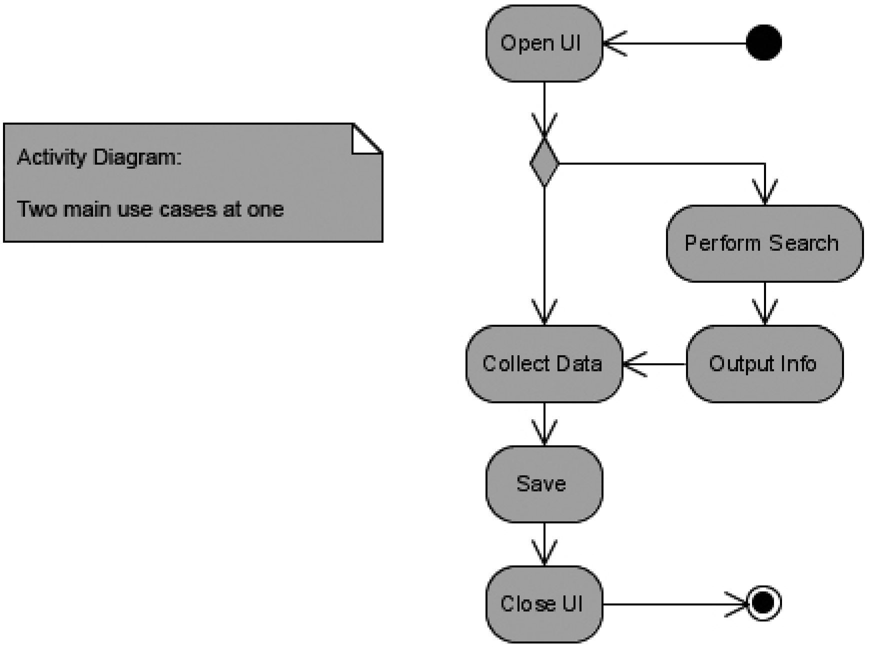


Figure 10: Activity diagram: two main use cases at one

After opening the reclaim information system user interface two separate paths will branch on the diagram. One branch processes the activities of performing a search using criteria given by the user and outputting the information chosen by the user. After outputting the information the system waits for user response and then proceeds to processing and storing the data. This is not done automatically. The user has to accept saving the data first. She or he can also choose to quit, go the start or clear the form. The other branch, after opening the user interface (UI), proceeds straight to the activity of collecting and processing new data.

3.9 Combined Deployment and Component Diagram

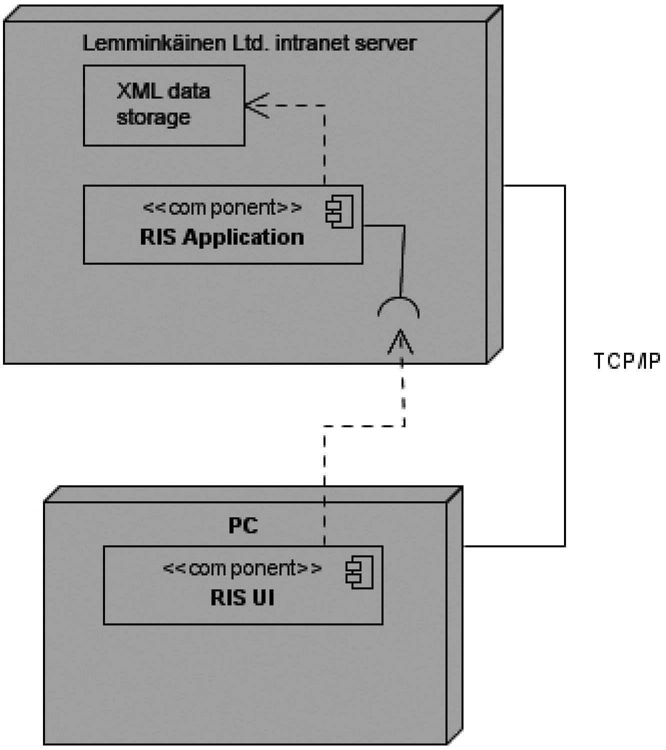


Figure 11: Combined Deployment and Component diagram

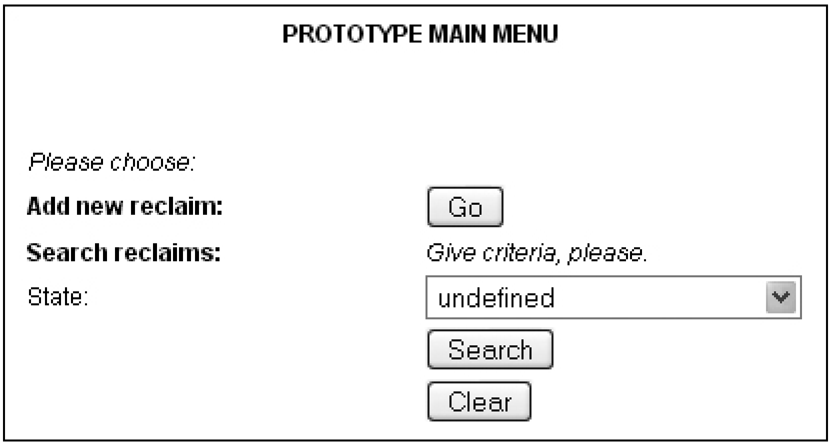
There are two nodes connected by TCP/IP which are the Lemminkäinen Ltd. intranet server and the user's personal computer equipped with an Internet browser. The main component is the reclaim information system application which produces the application's user interface which is executed by the PHP server and the results sent as HTML to the user's Internet browser. The XML files act as a database. Although the XML technique is used it is still possible to automatically convert the data into SQL form when needed.

3.10 The Prototype

After the UML modelling phase a simplified but functional prototype of the system was made. The prototype was made to learn and to test all the techniques related to developing the reclaim information system.

The system prototype has two functionalities which are adding reclaims and searching reclaims. The search criteria are limited to searching a reclaim in a certain state. Browsing reclaims is also available if the search criteria are undefined. In the complete version there is a specific feature for browsing reclaims. The prototype was made based on the two main use cases on altering reclaim and classification information and inputting reclaim information.

3.11 Searching, Viewing and Altering Reclaim Information



PROTOTYPE MAIN MENU

Please choose:

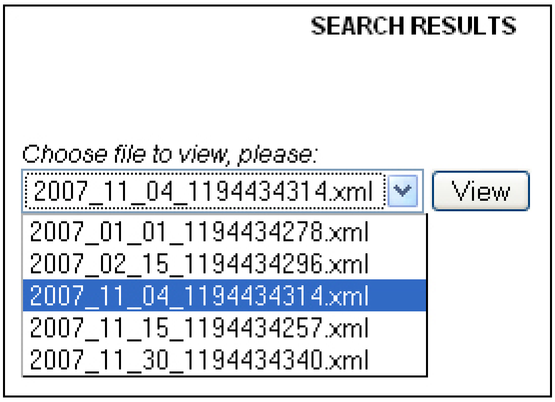
Add new reclaim:

Search reclaims: *Give criteria, please.*

State: undefined

Figure 12: Prototype of the ”Main Menu”

In the main menu and the search section the desired search criteria is defined by using drop down menus. After defining the criteria the search button is pressed to launch the search. It is also possible to clear the search criteria and start again.



SEARCH RESULTS

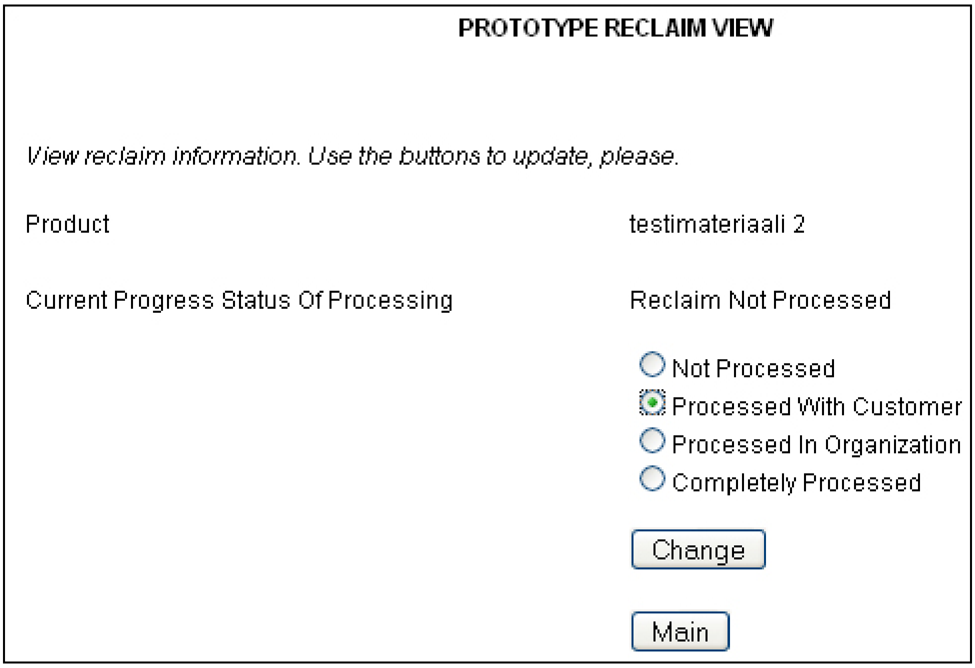
Choose file to view, please:

2007_11_04_1194434314.xml

View
2007_01_01_1194434278.xml
2007_02_15_1194434296.xml
2007_11_04_1194434314.xml
2007_11_15_1194434257.xml
2007_11_30_1194434340.xml

Figure 13: Prototype of the ”Search Results” view

The search results are presented in a drop down menu where the wanted file is chosen from. The filenames are individualised by user chosen dates and Unix timestamps supplied by PHP. This naming procedure makes the filenames unique. In addition all the filenames are checked for duplicates.



PROTOTYPE RECLAIM VIEW

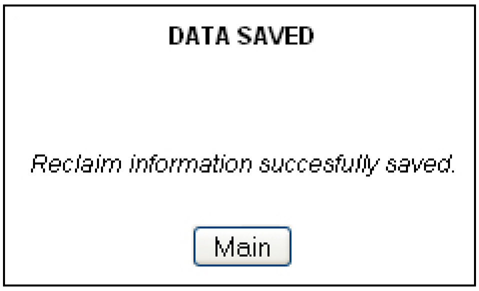
View reclaim information. Use the buttons to update, please.

Product	testimateriaali 2
Current Progress Status Of Processing	Reclaim Not Processed

○ Not Processed
● Processed With Customer
○ Processed In Organization
○ Completely Processed

Figure 14: Prototype of the ”Reclaim View”

After loading the XML file the product name and the current progress status of processing the reclaim are displayed. Also the options for altering reclaim information are presented. It is possible to change parts of the information or return to main menu. For the complete version a “back” button for returning to previous page will be added.



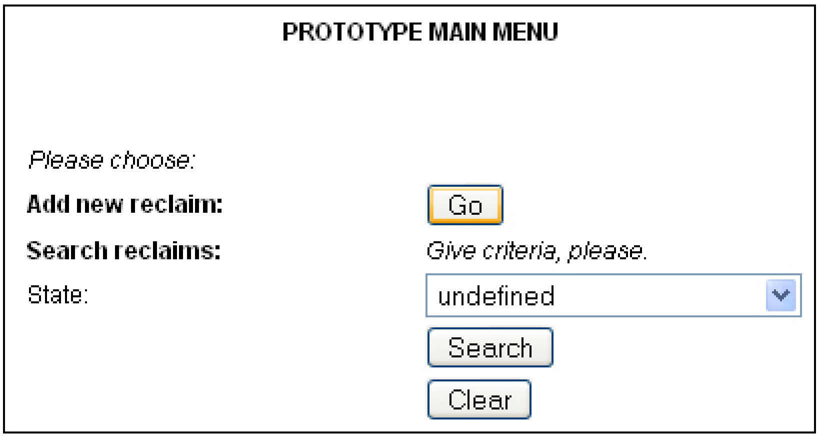
DATA SAVED

Reclaim information succesfully saved.

Figure 15: Prototype of the message to the user

After saving the altered information an announcement offering possibilities for further actions is presented. In this prototype there is only a button leading back to the main menu.

3.12 Inputting Reclaim Information



PROTOTYPE MAIN MENU

Please choose:

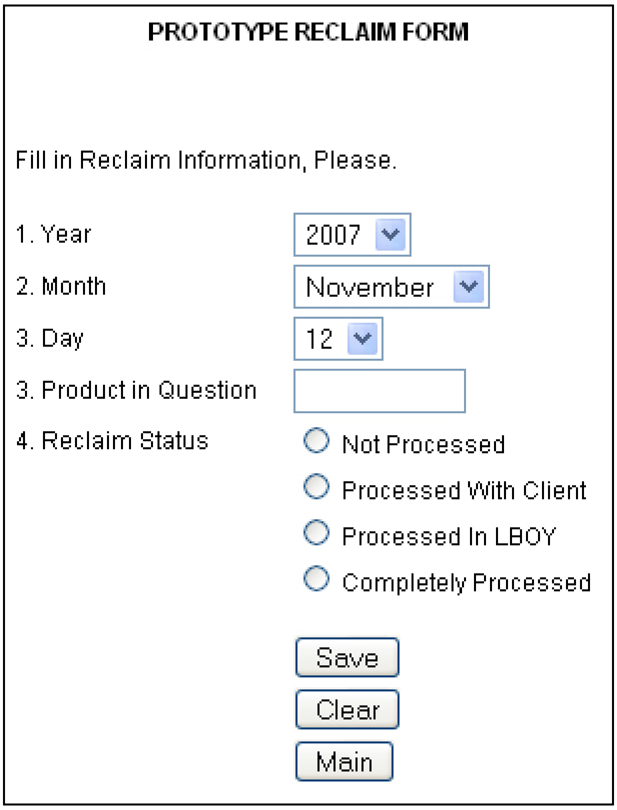
Add new reclaim:

Search reclaims: *Give criteria, please.*

State: undefined

Figure 16: Click ”Go” to add new reclaim information

To add a new reclaim into the system the “Go” button is pressed.



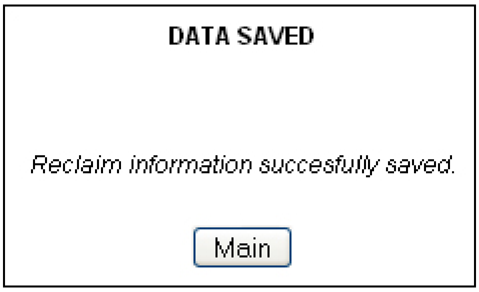
PROTOTYPE RECLAIM FORM

Fill in Reclaim Information, Please.

1. Year 2007
2. Month November
3. Day 12
3. Product in Question
4. Reclaim Status
   - ○ Not Processed
   - ○ Processed With Client
   - ○ Processed In LBOY
   - ○ Completely Processed

Figure 17: Prototype of the ”Reclaim Form”

A blank reclaim form is displayed. There are only five sections on it in this simplified prototype. The sections are “year”, “month”, “day”, “product” and “reclaim status”. The user can always clear the form or return to main menu at any phase.



DATA SAVED

Reclaim information succesfully saved.

Figure 18: Prototype of the ”data saved” message

After saving the altered information an announcement offering possibilities for further actions is presented. In this prototype there is only a button leading back to the main menu.

4 Evaluation

The project plan related to the thesis was made with adequate accuracy although the project schedule could have been of a higher resolution. The schedule planning still would have been difficult since I had limited experience of projects such as this. The project schedule was based on a percentage and it was formed according to how much time it would be reasonable to use with the different phases of software engineering.

Main part of the time was planned to be used with the definition and design phases. The phases were completed early on schedule. When looking into it at this later point of time the definition and the design should have been more accurate and so more time should have been spent with them. Due to my limited PHP experience there was pressure towards launching the implementation phase and the software engineering continued with making a functional prototype of the system. Making of the prototype was a convenient way to test techniques and skills demanded by the project and a way to become aware and solve most of the programming issues before actually beginning the implementation of the entire system.

The programming phase took more time than it was planned to and it will continue in form of updates and programming custom additional features. Further specification, design, testing and documentation will continue with the help of a pilot group. Plenty of requests and suggestions have been received after the first version of the system was released which in a way stretches the project timeline and it cannot be told when the project will be completely finished. In the beginning of April 2008 the first genuine reclaim information has been inputted to the system. A new meeting with the project group has been scheduled and the deployment of the system on the organisation level is at sight.

List of References

Peltomäki J. & Silander S. 2003. Java 2 ohjelmoinnin peruskirja. Porvoo: Docendo

<http://www.php.net> 10.12.2007

<http://atlas.kennesaw.edu/~dbraun/csis4650/A&D/UML_tutorial/> 25.9.2007

<http://www.w3schools.com/css/default.asp> 10.12.2007

<http://www.w3schools.com/html/> 10.12.2007

<http://www.w3schools.com/xml/xml_whatis.asp> 10.12.2007

<http://dn.codegear.com/article/images/31863/usecase.html> 8.10.2007

<http://www.softwareprojects.org/php-what-is-01.htm> 10.12.2007

<http://atlas.kennesaw.edu/~dbraun/csis4650/A&D/UML_tutorial/diagrams.htm> 10.12.2007

<http://www.vineyard.fi/fi/solutions.jsp> 20.5.2007

List of Captions