Functional Design

Stenden Support Desk

13.12.18

—

IT1C
Van Schaikenweg
7811 KL Emmen
Netherlands

Table of content

1. Introduction

The functional design also known as a system design, contains a complete and precise design for a new information system that was described in the project plan. This information system will be made by one or more programmers based on the functional design. A functional design report is written by the functional designer and is made for the client and programmers.

The functional design report must be easy to read for the principal that the information system must be described as precisely as possible to help the technical designer and the programmer to make it. The functional designer must be aware of this when he writes the report.

The purpose of this information system is to remove the use of calls and replace it with a ticket handling system that customers with maintenance licences have access to.

Stenden produces financial management software and provides a support desk for clients with maintenance contracts. The managing director, Raymond Blankesteijn, oversees all projects at Stenden. The company's primary product, financial management software is produced by different branches of the Stenden organization. The E-Help department and its team leader René Laan, oversee the collection and resolution of submitted incidents of clients with an active maintenance contract.

The current system for addressing client’s incidents is a system where clients must call the support desk and report the problem, it is then recorded on a single spread sheet, managed by one of the three members of the E-Help department at one moment. The purpose of the new ticket handling system is to address the failings and complications of the old system and streamline the efficiency at which work is done.

This project is of no relation or importance to any other project that is currently being undertaken by the Stenden organisation, nor is it a sequel or continuation of a current or past project.

2. Description of the new system

The ticket handling web application developed for Stenden E-Help by IT_1C will consist of a web-based user interface connected to a database. The system works on a hierarchical layer system that requires users to have the right permissions to access and edit content.

The permissions of users are divided into the following levels:

1. **Clients (Stenden E-Help customers)**

   - ❏ A company/client with an active license may view the FAQ page.
   - ❏ A company/client with an active maintenance licence may submit a ticket to the support desk.
   - ❏ A company/client with an active maintenance licence may up-vote and down-vote on the FAQ page on what questions they usually encounter.

2. **Employees**

   - ❏ Able to view and work on tickets.
   - ❏ Able to communicate with client.
   - ❏ Able to set ticket status.

3. **Administrator (René Laan)**

   - ❏ Assign tickets to employees.
   - ❏ View ticket information.
   - ❏ Effect employee registration and profiles.

Pages and functions

The front end of the web application will be broken up into the following pages:

1. Login
    - ❏ Users will be prompted to login with their email and password

2. FAQ
    - ❏ Displays the top issues
    - ❏ After users have viewed the listed issues they may vote on how helpful the solution was

3. Employee overview
    - ❏ Accessible by only Administrator. Displays a list of all current employees at Stenden E-Help, with the ability to add a new employee. An employee can be selected and then viewed in more detail in the following page

4. Employee page
    - ❏ Displays employee information and can be edited by the administrator

5. Client information overview
    - ❏ Displays saved information of clients with the ability to update old information
    - ❏ Clients can view and edit the list of their registered employees with access to the client account

6. Client ticket history
    - ❏ An interactive table of all the tickets submitted by the client. The table displays and is sortable by the headings: Ticket ID, Ticket Name, and

Submitted by, Date Submitted, Employee Assigned and Ticket Status. From this page clients can choose a ticket to expand and view in more detail

7. Ticket submission
    - ❏ Clients with an active Maintenance License can submit new tickets. Entering a Name of the ticket, employee submitting the ticket, and description of the problem. Clients without an active Maintenance License will be notified to buy the maintenance licence before continuing

8. Employee ticket overview
    - ❏ The page presented to Employees upon login. An interactive table showing all submitted tickets. The table will display tickets and can be navigated with a search bar, headings of tickets can be used to sort the displayed information. From here employees can also select a ticket to access

9. Employee ticket view
    - ❏ This is where employees have access to tickets and can provide solutions and solutions to the ticket problem.

Ticket handling procedure

1. The user fills in ticket information. The user receives a confirmation email. The “TicketStatus” field is now set to 1, “Open”

2. After the employee opens the ticket, “TicketStatus” is set to 2, “Pending”.

3. After a solution has been found and added to the solution/feedback field the ticket’s “TicketStatus” is set to 3 and the client receives an email.

4. The client then tries the posted solution, if it is successful, they mark it as so, “TicketStatus” is set to 4, “Confirmed”. If it does not work, they mark it so, “TicketStatus” returns to 2, “Pending”

5. The employee will either close the ticket, setting “TicketStatus” to 5 “Closed” or continue to work on the problem.

3. Data model

Tables

USER

Name	User	
Description	Project Leader and Employees of Stenden Support Desk	
Number of records	3	
Attributes	**Description** (plus possible requirements set)	**Type**
UserID	Unique key to identify each user	**INT** (10) **AI PK**
FirstName	First name of user	**VARCHAR** (50)
LastName	Last name of user	**VARCHAR** (50)
EMail	E-Mail address of user	**VARCHAR** (50)
Password	Argon2i encrypted Password of user	**VARCHAR** (100)
PermLevel	Sets the user permission (0 = Employee, 1 = Project Leader)	**INT** (1)
Picture	Path to the image of employee	**BLOB**

CUSTOMER

Name	Customer	
Description	Customer records from the Stenden company	
Number of records	Approx. 2000, increasing by 200 annually	
Attributes	**Description** (plus possible requirements set)	**Type**
CustomerID	Unique key to identify each customer	**INT** (10) **AI PK**
CustomerNumber	Unique surrogate key to identify each customer from the customer registration system	**INT** (10) **SK**
Company	Name of the company	**VARCHAR** (50)
Email	E-Mail of the company	**VARCHAR (**50**)**
License	License type (0 = user license, 1 = maint. license)	**TINYINT** (1)
MaintValidUntil	Until when the maintenance license is valid	**DATE**

CONTACT_PERSON

Name	Contact Person	
Description	Contact Person from the specific company	
Number of records	-	
Attributes	**Description** (plus possible requirements set)	**Type**
ContactID	Unique key to identify each contact person from each company	**INT** (10) **AI PK**
CustomerID	Foreign key to identify the company of the contact person [**Table:** CUSTOMER]	**INT** (10) **FK**
FirstName	First name of the Representative for the company	**VARCHAR** (50)
LastName	Last name of the Representative for the company	**VARCHAR** (50)
Role	Role of the contact person in his company	**VARCHAR** (50)
EMail	E-Mail address of the representative	**VARCHAR** (50)
Password	Argon2i encrypted Password of contact person	**VARCHAR** (100)
Phone	Phone number of the representative	**VARCHAR** (30)

TICKET

Name	Ticket	
Description	Records of the incidents	
Number of records	-	
Attributes	**Description** (plus possible requirements set)	**Type**
TicketID	Unique key to identify each Ticket	**INT** (10) **AI PK**
ContactID	Foreign key to identify the contact person of the company [**Table:** CUSTOMER]	**INT** (10) **FK**
UserID	Foreign key to identify the employee [**Table:** USER]	**INT** (10) **FK**
TStatusID	Foreign key to identify the status of the incident [**Table:** TICKET_STATUS]	**INT** (10) **FK**
Title	Title of the incident	**VARCHAR** (100)
Description	Description of the incident	**VARCHAR** (500)
Date	Date of the ticket creation	**DATE**
Time	Time of the ticket creation	**TIME**
ForwardDate	Date assigned to employee	**DATE**

Priority	Urgency of incident (enum('Low', 'Medium', 'High'))	**ENUM(...)**
SolutionID	Foreign key to identify the message which solved the incident [**Table:** MESSAGE]	**INT(1)**
Tags	Keywords for ticket	**VARCHAR(50)**
Type	Type of the incident (0 = query, 1 = wish, 2 = crash, 3 = functional problem, 4 = technical problem)	**INT** (1)

TICKET_STATUS

Name	Ticket Status	
Description	Status of each incident	
Number of records	-	
Attributes	**Description** (plus possible requirements set)	**Type**
TStatusID	Unique key to identify status of each ticket	**INT** (10) **AI PK**
StatusID	Foreign key to identify the status [**Table:** STATUS]	**INT** (10) **FK**
TicketID	Foreign key to identify the ticket it is assigned to [**Table:** TICKET]	**INT** (10) **FK**
Date	Last change in status	**DATE**

STATUS

Name	Status	
Description	List of possible status	
Number of records	-	
Attributes	**Description** (plus possible requirements set)	**Type**
StatusID	Unique key to identify each status	**INT** (10) **AI PK**
StatusName	Name of status	**VARCHAR** (10)

MESSAGE

Name	Message	
Description	Messages within a ticket between employee and customer	
Number of records	-	
Attributes	**Description** (plus possible requirements set)	**Type**
MessageID	Unique key to identify each message	**INT** (10) **AI PK**
SolutionID	Unique key copy of MessageID for solution message	**INT(10) UNIQUE**
TicketID	Foreign key to identify the ticket [**Table:** TICKET]	**INT** (10) **FK**
UserID	Foreign key to identify the Employee [**Table:** EMPLOYEE]	**INT** (10) **FK**
ContactID	Foreign key to identify the contact person [**Table:** CONTACT_PERSON]	**INT** (10) **FK**
Description	Message of the user	**VARCHAR** (300)
Date	Date of the message	**DATE**
Time	Time of the message	**TIME**

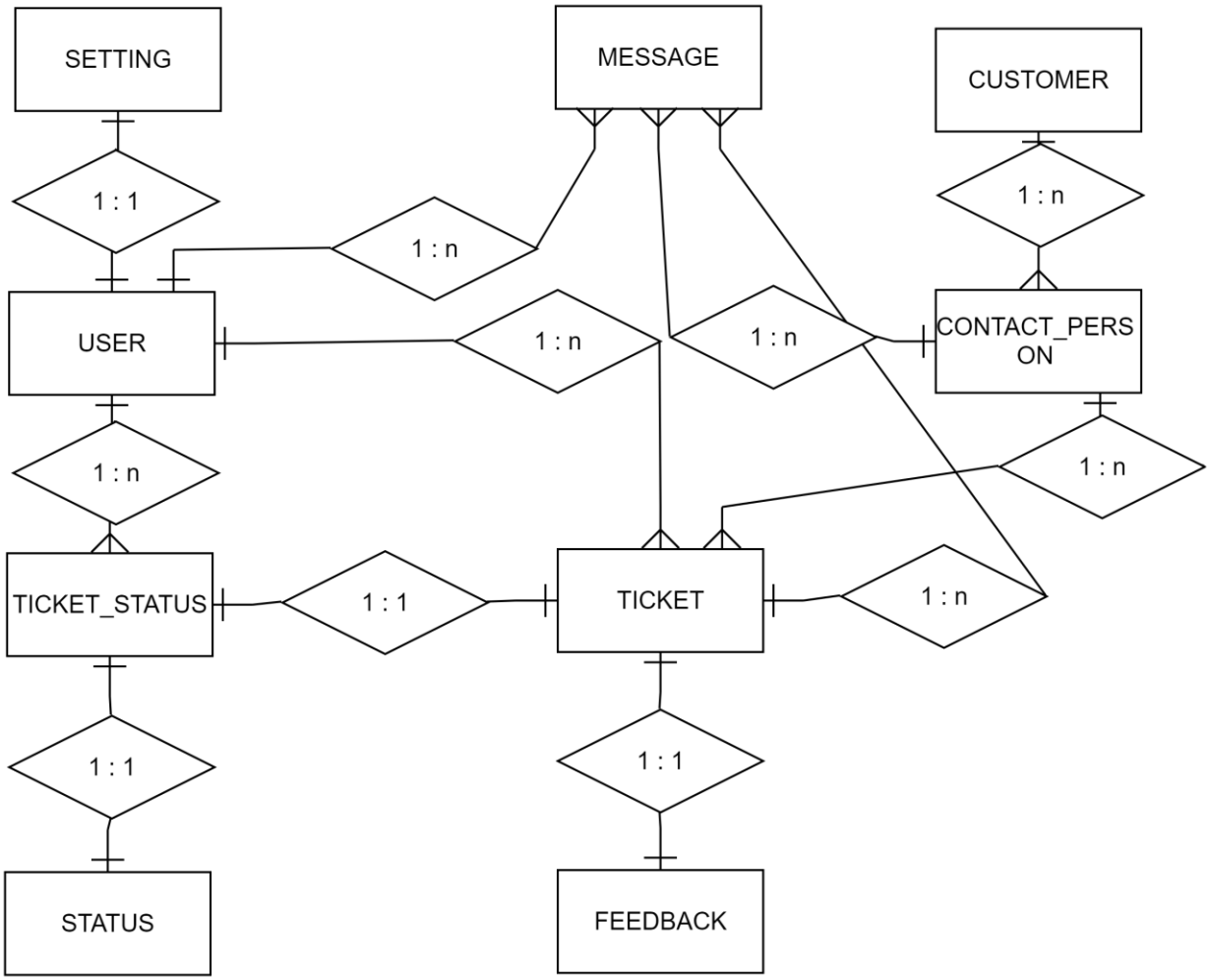
FEEDBACK

Name	Feedback	
Description	Feedback from a customer to a ticket/employee	
Number of records	-	
Attributes	**Description** (plus possible requirements set)	**Type**
FeedbackID	Unique key to identify each Feedback	**INT** (10) **AI PK**
TicketID	Foreign key to identify the ticket [**Table:** Ticket]	**INT** (10) **FK**
Rating	Rating from the customer for the ticket	**INT** (1)
Description	Description of the Feedback	**VARCHAR** (500)

SETTING

Name	Setting	
Description	Saves the unique setting of each user (e. g. Enable/Disable Guide...)	
Number of records	-	
Attributes	**Description** (plus possible requirements set)	**Type**
UserID	Unique key to identify each user [**Table:** USER]	**INT** (10) **PK**
Guide	Enables / Disables Guide upon login (0 = disabled, 1 = enabled)	**INT** (1)
DarkMode	Enables / Disables dark mode (0 = disabled, 1 = enabled)	**INT** (1)

ERD


SETTING
MESSAGE
CUSTOMER
1 : 1
1 : n
1 : n
USER
1 : n
1 : n
CONTACT_PERSON
1 : n
1 : n
TICKET_STATUS
1 : 1
TICKET
1 : n
1 : 1
1 : 1
STATUS
FEEDBACK

4. Data output

Name	Problem report
User	Admin
Objective	Provide an overview of any and all problems and their frequency.
Frequency	Weekly, Monthly, and annually
Sorting	Incident Occurrence number
Selection	Ticket.TicketID, Ticket
Data to be printed	**A:** count(TicketID) **B:** TicketID **C:** ContactID **D:** UserID **E:** TStatusID **F:** count(Type)

Incident report Overview (Administrator)

Problem ID	Problem Type	Frequency
ExampleID_1	Example problem	90
ExampleID_2	Example problem	85

Name	Employee overview
User	Admin
Objective	Display all employees
Frequency	N/A
Sorting	UserID
Selection	UserID,FirstName,LastName,EMail
Data to be printed	**A**: UserID **B**: FirstName **C**: LastName **D**: Email **E**: TStatusID

Employee Accounts

Registered employee accounts. Select one to view more details.

Employee ID	Employee Name	Employee Email	Number of unsolved tickets
100	John Smith	johnsmith@email.com	8
101	Jane Doe	janedoe@email.com	6

Name	Employee View
User	Employee, Admin
Objective	Employee page
Frequency	N/A
Sorting	N/A
Selection	UserID,FirstName,LastName,EMail
Data to be printed	**A**: UserID **B**: FirstName **C**: LastName **D**: Email

Account details of: __EMPLOYEENAME__

Here you may view information relevant to your account. Certain account details may be edited by you, but grayed out fields are unable to be manually changed.

Registered ID:

Password:

Employee email:

Database privilege:

Current employee image:

Upload new image:

Name	Tickets overview
User	Employee, Admin
Objective	Display all tickets
Frequency	N/A
Sorting	Completely dynamic. Can be sorted by any filter
Selection:	Table: Ticket
Data to be printed	Table: Ticket

Name	Employee Ticket View
User	Employees
Objective	Employee ticket working page
Frequency	N/A
Sorting	N/A
Selection	Ticket.CustomerID, Ticket.ContactID, Ticket.Title, Ticket.Description, Ticket.TStatusID, Ticket.TicketID, Ticket.Date, Ticket.Priority, Ticket.SolutionID, Ticket.Tags, Ticket.Type, Ticket.TStatusID, Message.MessageID, Message.SolutionID,

Data to be printed	**A**: Customer.Company **B**: Contact_Person.Email **C**: Contact_Person.Role **D**: Contact_Person.FirstName **E:** Contact_Person.LastName **F**: Ticket.TicketID **G**: Ticket.Title **H**: Ticket.Date **I**: Ticket.Time **J**: Ticket_Status.Date **K**: Ticket.ForwardDate **L**: Ticket.Tags **M**: Message.Description

Details for ticket: __TICKETID__, __TICKETNAME__

Client Name: Employees Assigned:

Submitter Name: Date Submitted:

Submitter Role: Last update:

Submitter Email: Forwarded:

Keywords:

Problem description:

Messages:

Name	Frequently Asked Questions
User	Customer
Objective	Provide an overview on the most asked questions.
Frequency	N/A
Sorting	N/A
Selection	Ticket.Title, Ticket.TicketID, Message.SolutionID, Message.Description
Data to be printed	**A**: Ticket.Title **B**: Ticket.TicketID **C**: Question **D**: CompanyName

Frequently Asked Questions

On this page you will find help for the 3 most common problems.

You may view the solutions by selecting a question. The answer will then be presented.

Question_1:

Answer_1

Question_2:

Answer_2

Question_3:

Answer_3

Name	Customer Ticket History
User	Customer
Objective	Overview of submitted tickets
Frequency	N/A
Sorting	Date Submitted
Selection	Ticket.Title, Ticket.TicketID, CompanyID, Ticket.Date, Ticket_Status.TSatusID, Ticket_Status.Date, User.FirstName, User.LastName, Contact_Person.LastName
Data to be printed	**A**: Ticket.Title **B**: Contact_Person.LastName **C**: User.LastName **D**: User.FirstName **E**: Ticket_Status.Date **F**: Ticket.Date **G**: Ticket_Status.TStatusID

Tickets Submitted

Here is a list of all tickets submitted by your account.

Ticket Name	Submitter	Date Submitted	Date Updated	Employee	Status
Ticket_1	Ralf	05/04/2018	06/04/2018	Tim Fletcher	Open
Ticket_2	Emily	10/05/2018	12/05/2018	Tom Tom	Closed

Name	Customer Ticket View
User	Customer
Objective	Information of a submitted ticket
Frequency	N/A
Sorting	N/A
Selection	Ticket.CustomerID, Ticket.ContactID, Ticket.Title, Ticket.Description, Ticket.TStatusID, Ticket.TicketID, Ticket.Date, Ticket.Priority, Ticket.SolutionID, Ticket.Tags, Ticket.Type, Ticket.TStatusID, Message.MessageID, Message.SolutionID,

Data to be printed	**A**: Customer.Company **B**: Contact_Person.Email **C**: Contact_Person.Role **D**: Contact_Person.FirstName **E**: Contact_Person.LastName **F**: Ticket.TicketID **G**: Ticket.Title **H**: Ticket.Date **I**: Ticket.Time **J**: Ticket_Status.Date **K**: Ticket.ForwardDate **L**: Ticket.Tags **M**: Message.Description

Details for ticket: __TICKETID__, __TICKETNAME__

Client Name: Employees Assigned:

Submitter Name: Date Submitted:

Submitter Role: Date Edited:

Submitter Email: Forwarded:

Problem description:

Messages:

Name	Customer overview
User	Customer, Administrator
Objective	Display and editing account information
Frequency	N/A
Sorting	N/A
Selection	Customer.CustomerID, Customer.Company, Customer.EMail, Customer.MaintValidUntil
Data to be printed	**A**: Customer.Company **B**: Customer.EMail **C**: Customer.MaintValidUntil

Account details of: __COMPANYNAME__

Here you may view information relevant to your account. Certain account details may be edited by you, but grayed out fields are unable to be manually changed.

Password:

Company Email:

Maintenance License Valid:

5. Required input

Name	Input ticket issue
User	Customer
Objective:	Ability to save the input
Description	Ability for user to add information in the ticket
Frequency	Often
Files	Customer.CustomerNumber, Customer.Company, Customer License, Contact_Person.Name, Contact_Person.Email, Contact_Person.Role, Contact_Person.Phone, Ticket.Title, Ticket.Description, Ticket.Date, Ticket.Type
Screens Used	Ticket Input

Name	Input ticket priority
User	Employee
Objective	Level of priority
Description	To add each ticket a level of priority (Low, Mid, High)
Frequency	Often
Files	Ticket.Priority, Ticket.TicketID
Screens Used	Ticket Detailed View, Ticket Overview

Name	Worker Assignment
User	Admin
Objective	Assign a specific ticket to a specific employee
Description	The ability to assign a ticket to an employee.
Frequency	Often
Files	Ticket.TicketID, Ticket.ContactID, Ticket.UserID, Status
Screens Used	Ticket Detailed View, Ticket Overview

Name	Solution Input
User	Employee
Objective	Answering the ticket
Description	The ability to add the solution for a ticket.
Frequency	Often
Files	Ticket.TicketID, Ticket. ContactID, Ticket.UserID, Ticket.SolutionID
Screens Used	Solution View

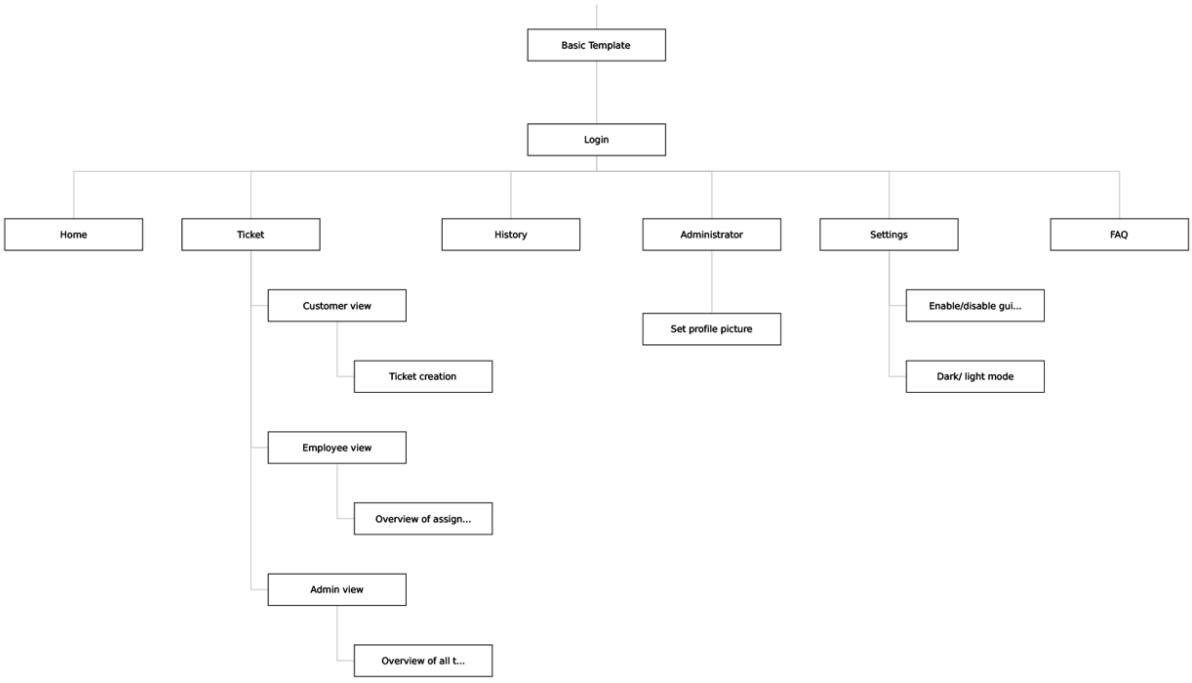
Name:	Feedback Input
User	Customer / Contact_Person
Objective	Feedback for the ticket
Description	The ability to give feedback to the solution of the ticket.
Frequency	Often
Files	Feedback.FeedbackID, Feedback.TicketID, Feedback.Rating, Feedback.Description
Screens Used	Feedback Screen

6. Authorization & Menu structure diagram

Authorization

Pages	Customer (with maint. License)	Employee	Administrator
Login	yes	yes	yes
Home	yes	yes	yes
Ticket create page	yes	yes	yes
Client Ticket history	yes	yes	yes
Employee ticket view	no	yes	yes
All ticket history	no	no	yes
Employee overview page	no	no	yes
Client overview page	no	yes	yes
Guide	no	yes	yes

Menu structure diagram



7. Organizational consequences

The developed system is going to be tested by the group IT1 C on the course of 1 week in which tasks are going to be divided to the group members. The group is going to allow 3 days for bug testing, 3 days for security tasking, so that the system will not be able to be hacked and 1 day for other minor inconveniences.

When the product is going to reach its final form, the interface will be user-friendly, so that every new user is going to be able to easily access it. There will be no problem for them to use the system, so that there will be no assistance or guide available for the users.

A database will be created on the server so that the employees and users are going to be able to access it. There can be some conversion problems expected such as updating new information to the new database. In order to solve this problems the employees are going to use the guide implemented by the group IT1 C on the webpage to solve this problem. In the guide available there will be presented a tutorial step by step on how the application's features work, how to manage and use the web application, so that the employees of the company are going to be well equipped with the sufficient knowledge in order to solve any occurring incident.

8. Technical consequences

The technical equipment required for running the software is a computer or a laptop equipped with a mouse and a keyboard. There is also a stable internet connection required, such as a strong wireless internet signal. The requirements of the application will not be very high, so that any computer younger than 10 years is going to be able to run it without a problem. The employees are going to be prepared with the necessary equipment, in order to manage everything smoothly and to make sure that there cannot occur any inconveniences.

A printing system will not be needed, due to the fact, that there will be no need of a printed version of the document. If a user is really going to need a printed version, a simple printer will get the job done without a problem. The backup of the software is going to be done both online and offline, so that the company will not use any data if technical problems occur. There will be a second server needed with the data available in the original one, just for security and safety reasons.

The employees are going to need the email and the phone number of the customers or even their social media information such as Facebook or WhatsApp, these being the ways of communication between the both sides. The email address is the main way of interaction, but if the customer is not checking his email constantly, there have to be replacements available for this such as the examples mentioned before.