**Software Requirements Specification (SRS)**

**Case Study 3**

**accomplished at the Business Informatics study program**

**of the University of Applied Sciences Technikum**

**Version 0.2**

**Project manager: Patrick Homm (ph)**

**Project team: Martin Wollner (mw), Leonardo Fisic (lf), Andreas Burger (ab)**

**Date: 2013-07-10**

Change Log

|  |  |  |  |
| --- | --- | --- | --- |
| **Version number** | **Date** | **Change** | **Person** |
| 0.1 | 2013-23-09 | First draft, basic informations | mw, ph |
| 0.2 | 2013-07-10 | Adding informations from meeting on 2013-24-09 | mw, ph |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table 1: Change log

|  |  |  |
| --- | --- | --- |
| **Legend** | | |
| **State** | **W**  **R**  **F** | In work  In review  Released, freed version |
| **Versioning** | **0.1, 0.2**  **1.0**  **1.1, 1.2**  **2.0** | Not approved version  First released version  Extended first version  Second released version |

Contact Persons

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Organizational unit** | **Project role** | **Telephone** | **e-mail** |
| Patrick Homm | Project Team | Project Manager | 0699/11055260 | patrick.homm@technikum-wien.at |
| Martin Wollner | Project Team | Team Member |  | Martin.wollner@technikum-wien.at |
| Leonardo Fisic | Project Team | Team Member |  | Leonardo.fisic@technikum-wien.at |
| Andreas Burger | Project Team | Team Member |  | Andreas.burger@technikum-wien.at |
| Philipp Schiedauf | Customer |  |  | schiedau@technikum-wien.at |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 2: Contact persons

**Table of Content**

[Software Requirements Specification (SRS) 1](#_Toc368948626)

[1 Introduction 6](#_Toc368948627)

[1.1 Purpose 6](#_Toc368948628)

[1.2 Scope 6](#_Toc368948629)

[1.3 Definitions, Acronyms, and Abbreviations. 6](#_Toc368948630)

[1.4 References 6](#_Toc368948631)

[1.5 Overview 6](#_Toc368948632)

[2 The Overall Description 7](#_Toc368948633)

[2.1 Product Perspective 7](#_Toc368948634)

[2.1.1 System Interfaces 7](#_Toc368948635)

[2.1.2 Interfaces 7](#_Toc368948636)

[2.1.3 Software Interfaces 7](#_Toc368948637)

[2.1.4 Communications Interfaces 7](#_Toc368948638)

[2.1.5 Site Adaptation Requirements 7](#_Toc368948639)

[2.2 Product Functions 8](#_Toc368948640)

[2.3 User Characteristics 9](#_Toc368948641)

[2.4 Constraints 9](#_Toc368948642)

[3 Specific Requirements 10](#_Toc368948643)

[3.1 External Interfaces 10](#_Toc368948644)

[3.1.1 E-Mail Interface 10](#_Toc368948645)

[3.1.2 Payment Interface 10](#_Toc368948646)

[3.2 Functions 10](#_Toc368948647)

[3.2.1 Registration 10](#_Toc368948648)

[3.2.2 Maintain personal information 10](#_Toc368948649)

[3.2.3 Web Shop 11](#_Toc368948650)

[3.2.4 Chocolate Editor 12](#_Toc368948651)

[3.2.5 Static Contents 12](#_Toc368948652)

[3.2.6 User administration 12](#_Toc368948653)

[3.2.7 Product administration 12](#_Toc368948654)

[3.2.8 Sale administration 13](#_Toc368948655)

[3.2.9 Database Synchronization Interface 13](#_Toc368948656)

[3.3 Performance Requirements 13](#_Toc368948657)

[3.4 Logical Database Requirements 14](#_Toc368948658)

[3.5 Design Constraints 14](#_Toc368948659)

[3.5.1 Standards Compliance 14](#_Toc368948660)

[3.6 Software System Attributes 14](#_Toc368948661)

[3.6.1 Reliability 14](#_Toc368948662)

[3.6.2 Availability 15](#_Toc368948663)

[3.6.3 Security 15](#_Toc368948664)

[3.6.4 Maintainability 15](#_Toc368948665)

[3.6.5 Portability 15](#_Toc368948666)

[4 Figures 16](#_Toc368948667)

[5 Tables 16](#_Toc368948668)

[6 Glossary 16](#_Toc368948669)

# Introduction

## Purpose

This document describes all requirements belonging to the project „chocolate customizer“.

With this project, people all over the world shall be able to buy their favourite chocolate creations again and again.

The following features must be realised:

* Visitor registration
* Searching and browsing through products
* Buying existing products (chocolate products, vouchers, gifts for corporate customers) using credit card
* Rating products
* Create or customize chocolate products and gifts (for corporate customers)

## Scope

The project produces a web application called “chocolate customizer” (will be registered using a similar domain name). The web application will be reachable all over the world and allow users to buy existing or to create new chocolate creations. Additionally vouchers and gifts will be sold.

This web application is not intended to sell anything else than chocolate products, or products related to this chocolate products (vouchers, gifts).

## Definitions, Acronyms, and Abbreviations.

RDP Remote Desktop Protocoll, a Protocol used to connect to a server

HTTP Hypertext Transfer Protocoll, foundation protocol for data communications in the

WWW

## References

* http/https: <http://tools.ietf.org/html/rfc2616>
* E-Mail: <http://tools.ietf.org/html/rfc5322>
* Telephone Number Syntax: <http://www.itu.int/rec/T-REC-E.123-200102-I/e>

## Overview

This document is created after the IEEE 830-1998 standard. Some of the following points are not specified in detail (in cause of missing information). Chapter 1 (Introduction) contains important information about purpose, scope, definition and references.

In chapter 2 is a common (overall) description defined. It contains information about the functionality of the web application, a description about the affected users and a list of constraints for the project.

Chapter 3 is the most incomplete part of this document, in cause of missing information. It contains (and will contain in a more detailed way) specific requirements for this web application like functional requirements, external interfaces, design contraints, performance requirements and quality requirements.

# The Overall Description

## Product Perspective

The project “chocolate customizer” can be compared with many other existing web application for customizing standard products, like for example “mymuesli”, “myparfum”, et cetera.

### System Interfaces

Payment System interface

Because the web shop of the application allows only credit cards, the interface of the credit card institute has to be used.

Printer

The on the server installed printer shall be used for printing chocolate packages to the production department.

### Interfaces

The web application will be presented to it’s users with a nice looking and clear GUI, which has to be handicapped accessible. It must use an efficient design, so that the click count is as low as possible. Designers have to make sure, that the GUI is easy to understand and follows a clear structure.

### Software Interfaces

E-Mail Server

For sending information and confirmation mails a HTTP Get Mail Interface provided by the sellers of this Project will be used.

Database

For the web frontend, a MySQL Database will be used.

The backend will use a Sybase Database.

Ingredients Supplier

Ingredients orders will be automatically delivered to the supplier by an interface.

### Communications Interfaces

As the project produces a web application, the HTT-Protocol (RFC 2616) is used. To make sure that payment information and personal information is save, HTTPS (RFC 2818) is used. RDP will be used to administrate the server (ITU-T T.128).

### Site Adaptation Requirements

Needed Software components:

* Operating System

Windows Server 2008 R2 will be used.

* Webserver

The Microsoft IIS Webserver will be used to provice web content

* PHP

The Middleware PHP will be used. The version depends on the required version by the CMS Typo3

* MySQL Database
* CMS Typo3

Hardware Requirements:

* Intel Core i7 Processor
* 16GB of RAM needed for the live system (1GB during development)
* 128GB Hard Disk Storage

Network Requirements:

* 100Mbit synchronous
* Open Ports 80, 443 and 60143 (mapped to RDP for security reasons)

## Product Functions

The following pages will list all major functions for this web application. Please note that the second section of this list is just a placeholder for the backend, which will be implemented in “case study 4”, so it will be described there.

**Major functions:**

* **Web shop:**the web shop is one of the two major functions. It will provide all products (in this case it’s chocolate products only) and provides the possibility to search, mark and buy those products. It has a state-of-the-art stock shopping cart.  
  Normal customers can buy standard chocolate articles, their own created or customized articles, and vouches. If a customer is registered as corporate customer he is also able to order three different types of chocolate gifts. These chocolates can either be standardized or individually created. The packaging of those gifts can be created individually too.  
  The order process will be finished by paying via credit card (independent from the customer type).  
  Individually created products or gifts will automatically be added to the public web shop (so any customer can buy those products) and can be rated by every buyer.
* **Chocolate editor:**the chocolate editor (called “chocoladitor”) is the second of the two major functions. It gives the possibility to personalize existing products of the web shop or to create new products from the scratch. It contains tools to specify the size of the chocolate bar, the cocao ratio in the chocolate, to add additional ingrededients (e.g. nuts, strawberry, et cetera) and to define the grid of pieces (like chessboard, triangles …).

**Standard functions:**

* **Registration:**visitors are able to use both of the major functions, but they have to register themselves to finish the buying process. So a registration formular is needed to take personal, address and buying information from the future customer. A customer also has to decide whether he is normal “private” customer or a corporate customer.
* **Maintain personal information:**logged in users can maintain their personal information (in case of change – e.g. marriage)
* **History:**logged-in users can take a look at their already bought products and reorder them directly.

**Common functions:**

* About us
* FAQ
* Contact formular

**Backend functions (detail description in “case study 4”):**

* **User administration**

An Administrator will get the possibility to add, delete or edit users and all their properties.

* **Product administration**

The product administration will provide views to edit the product category list (add, delete, edit properties) and products itself (add, delete, edit properties, hide/show). It shall also give the possibility to order ingredients which run low on stock.

There shall also be the possibility to make standard products out of customly created chocolate creations made by customers.

Statistics (sell volume, viw count) shall be provided for all products and custom chocolate creations.

* **Sale administration**

Employees of the chocolate company shall get the possibility to take care of ordered chocolate vouchers or the packaging of chocolate gift packages. After he has finished a creation of the chocolate voucher, it shall be sent via email to the customer. The chocolate packaging will be printed out by the printer and automatically sent to the production department.

**Other Functions**

* Database synchronization

The Frontend and backend database will be synchronized to ensure consistent data on both sides.

## User Characteristics

Personalized chocolate is a perfect gift, so most of the customers will be people that are looking for those. Chocolate is also something that everybody loves, so we cannot define a clear list of customers. The UI of the web application will make sure, that every user, independently from his educational level or technical skills, can use the shop and the chocolate editor. It’s handling will be easy and clear.

## Constraints

* Not other frontend languages than english will be supported
* The Payment interface will not be implemented, an existing will be used
* No chocolate production processes or interfaces will be implemented they will only be used
* No supply processes or interfaces will be implemented, they will only be used

# Specific Requirements

## External Interfaces

Most interfaces aren’t defined yet, therefore there are no technical informations about external interfaces available.

### E-Mail Interface

A self implemented PHP Interface will serve as E-Mail Interface for both, Front- and Backend. It will be called via HTTP GET Request containing 5 Parameters:

* From E-Mail Address
* To E-Mail Address(es) (Semicolon separated String)
* CC E-Mail Address(es) (Semicolon separated String)
* Subject
* Content

Basic authentification will be used for protecting the interface against abusage.

### Payment Interface

A HTTPS Webservice shall be called. This webservice has 2 methods:

* Paying
* Payment Confirmation

A detailed description of the payment interface will be given by the credit card companies when they are needed. The usage is documented in the how-to-use documentation of the credit card companies.

## Functions

### Registration

The registration shall be done with an HTML Form and has to contain following fields:

1. First Name (required)
2. Middle Name (optional)
3. Last Name (required)
4. Birth Date (required, dd.mm.yyyy)
5. E-Mail Address (required, for syntax refer to RFC5322 section 3.2.3 and 3.4.1)
6. Passwort (required, has to be given twice and equal each other, min 8 char no furtur complexity requirements)
7. Address (Country, ZIP Code, City, Street, Housenumber) (optional)
8. Telephone Number (optional, for syntax refer to E.123)

The E-Mail Address has to be verified after the Registration. For this the external E-Mail Interface described in 3.1.1 will be needed.

Every field shall be validated. If any field is wrong, the user shall get the exact information wich value failed the validation. The error message shall be positioned over the HTML Form.

### Maintain personal information

The System has to give the user the possibility to maintain the personal informations. This Page shall be done with an HTML Form and has to contain the same fields as the registration.

### Web Shop

#### Search

A searchfield for searching Products has to be available. Further search requirements aren’t declared yet.

If the search fails because there are no matches available, a message has to be shown that there arent’t any matches available.

If results are given, the total count of results shall be shown at the top.

#### Category List

The category levels have to be defined in cooperation with the customer.

The user will get a navigation field in which all Categories and possible subcategories are listet. If he clicks on one, a further product list shall be shown.

One category has to be “Customized” in which all choclolates created by users will be visible. This shall give the possibility to search for chocolate creations before creating an own.

#### Product List

The product list shall be shown in the content area of the webpage. The title, a picture, the price, stock status and a short descrition shall be given for each entry.

The customer shall be able to choose how many entries he wants to have displayed on one page in a combo box. Possible values ar 10, 25, 50 and 100. If the user changes the value of the selected entries, the webpage shall automatically start regenerating itself.

The current and number available pages shall be shown at the top and at the bottom of the list. Navigation thru the pages shall be also available at the top and at the bottom of the product list.

The user shall have the possibility to add products to the cart right away from the product list by pressing a “add to cart” button.

#### Product Details

In the product details page, every available attribute of the selected product shall be displayed. The definition of these attributes isn’t set yet and will be specified in the next version.

The user shall have to possibility to add products to the cart by pressing a “add to cart” button.

A user shall also get the possibility to rate this product by selecting 0-5 stars and adding an optional comment if he has bought this item in the past.

#### Cart

All selected products of this user shall be listed (picture, title, short description, quantity) in the cart. For corporate customers a possibility to create gift packages individually shall be given. Further details about the packages have to be defined.

#### Order

After the user has proceeded to the order page, the selected products (picture, title, short description, quantity) shall be shown read only.

Additional costs for packages shall be listed at the bottom of the list. The user has to fill out an HTML Form for shipping address. If an address is given in the personal details, it shall prefill the form. After he proceeds, a pop up for payment will open (further details refer to 3.1.2 Payment interface). After the payment succeeded, a final confirmation which includes all ordered items, the total price and the estimated shipping date shall be displayed. After each successful order a confirmation email (refer 3.1.1 E-Mail Interfache) which includes a list of all ordered products as well as the total price and the shipping informations shall be sent to the customer.

If the payment fails, an error message will be shown and the customer shall be asked to contact the company.

#### History

The user can review all his orders. A summarize of all orders shall be given in a list which contains total price, total number of items shipped, shipping date and shipping address. After selecting a single order, a detail view which contains every information (list of ordered products, package informations, total price, shipping informations) about the selected order.

### Chocolate Editor

In the Chocolate Editor a user can create his own chocolate. He can choose from predefined ingredients, which will be administered in the Chocolate Editor administration (see 3.2.8). If any by the user the selected ingredients aren’t on stock, an information shall be shown to the user, that the shipment could will be delayed if the product would be ordered right now.

The selection of ingredients shall be limited. It shall not be possible to choose unlimited ingredients for the chocolated. The maximum amount of ingredients used in one chocolate shall be 3.

After the user has finished creating his own chocolate, the system shall check if there is an already existing chocolate which matches the exact same attributes and mixture. If there aren’t, the website shall search for standard chocolates which almost match the same mixture (flavours shall be exact the same, toppings can differ by one). If there is any exact matching chocolate creation, the system shall automatically select the already existing one to avoid duplicates.

The user shall now get the possibility to save his creation and add it to the basket.

### Static Contents

Static pages for About us, FAQ and General Terms and Conditions and Contact informations are needed. There shall be a hyperlink to each static page in the bottom bar of the website. The text for these pages will be given in a separate document

### User administration

The user administration will be done in the backend and should be for both, front- and backend users. After switching to the user administration, a list of all available users shall be shown. It should be possible to filter the list by user account type:

1. Frontend User
2. Customer
3. Corporate Customer
4. Backend User

It shall be possible to to delete and edit Users. In the edit view, all account details shall be shown and editable.

### Product administration

The product administration is part of the backend functionality. It is for both, standard products and customly created chocolate creations made by customers. Following navigation tree shall be implemented:

1. Product administration
2. Product list
3. Product details
4. Chocolate Editor administration
5. Chocolate Creations list
6. Chocolate Editor settings

In the Product administration, a list of all available standard products shall be shown by their category. In here, the user gets the possibility to edit details of a product. Statistics of the product (sales volume, view count) shall be visible in the product details section, but the administrator shall also get the possibility to order the product list by their sales volume. Following actions shall be given to products:

1. Delete a product
2. Hide/show a product from/to the frontend
3. Edit product details

In the chocolate editor section, the User shall be able to add or remove ingredients to the chocolate editor.

In another menu a list of all customer chocolate creations shall be listed. The User has now the possibility to see statistics about different chocolate creations (sales volume, view count). It shall be possible to create a new standard product from a customer’s creation.

### Sale administration

At first a list of all Sales shall be showed in a list. This is a first come – first serve process, which means, the user can only work on the earliest request.

The detail view shall provide informations about gift packages and products. The user must create a voucher. After closing the case a confirmation e-mail shall be sent to the customer and the packaging of the chocolate gift shall be printed to the production department using the internal default printer driver.

### Database Synchronization Interface

The Web-Frontend Database MySQL and the backend Database Sybase have to be synchronized. A Windows Service for synchronizing the databases shall be implemented. There are already existing C# .net APIs to use Sybase and MySQL Databases. The synchronization algorithm has to be written. Both databases shall be compared.

User data will be synchronized two way (users register and edit their private data, administrators can also edit, add or remove users).

Customer chocolate creations will be also synchronized two way (users create chocolate creations in the frontend, administrators can make standard products out of custom creations).

Product detail synchronization is complex. Only sales volumne and view count will be synchronized from the frontend database to the backend database. All other values will only be synchronized from the backend to the frontend database.

The synchronization intervals shall be adjustable by an argument. Default value shall be 10 minutes.

## Performance Requirements

In general 95% of the HTTP Response times shall be under 5 seconds in a timespan of 1 year.

In 1 year 95% of all actions on the frontend (including SQL Statements, HTTP Responses, Paying Transactions) shall respond (=only 5% timeouts allowed).

In a Survey (carried out a year after going online) asking a 1000 persons on the frontend and all users on the backend about their oppinion of performance, 95% of the asked users shall answer with "it performs nice".

Alle performance requirements shall be measured on a modern pc (not older than 2 years), a modern and commonly used browser (Internet Explorer 9, Google Chrome, Firefox, Safari and Opera, each in the newest version) and with an internet speed of at least 4Mbit Download and 512Kbit Upload rate.

## Logical Database Requirements

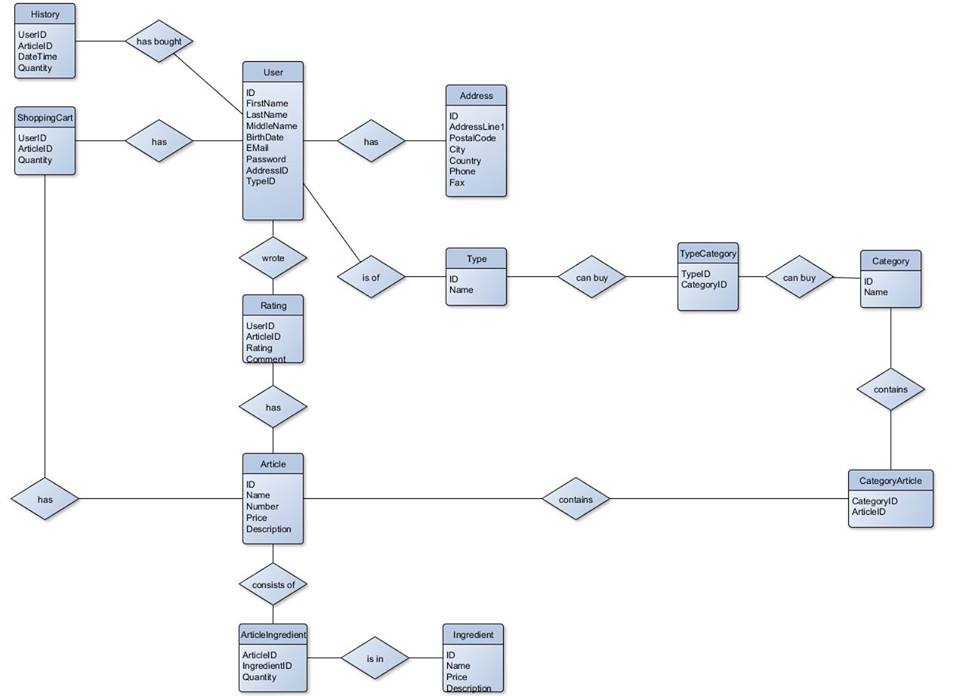


Figure 1: ERD

## Design Constraints

### Standards Compliance

The frontend web software shall follow the guidelines given by W3C to provide a widely and barrier-free usage of the content.

## Software System Attributes

### Reliability

The reliability of the Software depends on the reliability of the used components. The latest versions of all components will be used, which automatically prevents the base and most commonly reasons for leaks of reliablility.

### Availability

All ressources (database, web frontend) shall be available and online at least 99% of the year without system errors.

### Security

All passwords will be stored md5 hashed in all databases. The passwords also need to be complex (3 out of 4 requirements (at least: 1 Upper Letter, 1 Lower Letter, 1 number, 1 Special Character)) and contain at least 6 characters.

No account informations like credit card data will be stored to protect customers from abuse of data. All private data will be only visible for administrators to protect the customer’s privacy.

All systems will be updated and backuped regularly. All unneeded network ports except for port 80 (HTTP) and port 60143 (RDP) shall be blocked.

The RDP Port shall be set to 60143 to avoid the usage of default ports (will reduce random brute force attempts to a minimum)

Regular backups of the database and the webserver must be done.

### Maintainability

Updates for the Operating System and the Webserver will be done automatically by Windows Server 2008 R2. Updates to Typo 3 or any other components have to be done manually but shall only be done if serious security issues will be detected.

### Portability

The portability of the system depends on the used components. We guarantee 100% portability to a system with the same attributes as the one we install, which means the same version and patchlevel of the Operating System. All components needed to run Typo3 (PHP, MySQL) have to run the same version and configuration as on the provided system.

In general we guarantee the portability to almost Sofware cloned systems, but none other.

# Figures

[Figure 1: ERD 14](#_Toc368948221)

# Tables

[Table 1: Change log 2](#_Toc364426454)

[Table 2: Contact persons 3](#_Toc364426455)

# Glossary

|  |  |  |
| --- | --- | --- |
| **Key concept or abbreviation** | **Explanation** | **Additional Information** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |