
<Company Name>

PoS Systems Software Requirements Specification

Version 1.2

| | |
|-------------------------------------|----------------|
| PoS Systems | Version: 1.2 |
| Software Requirements Specification | Date: 20.10.16 |
| pos_rup_srs | |

Revision History

| Date | Version | Description | Author |
|----------|---------|---|--|
| 20.10.16 | 1.0 | Creation of Document | Sven Baumann, Dominik Schneider, Sandra Kramlich |
| 26.10.16 | 1.1 | - replaced blue text passages - reworked "Mean Time to Repair" | Sven Baumann |
| 01.11.16 | 1.2 | -added references | Sven Baumann |
| | | | |

| | |
|-------------------------------------|----------------|
| PoS Systems | Version: 1.2 |
| Software Requirements Specification | Date: 20.10.16 |
| pos_rup_srs | |

Table of Contents

| | | |
|-------|---|---|
| 1. | Introduction | 4 |
| 1.1 | Purpose | 4 |
| 1.2 | Scope | 4 |
| 1.3 | Definitions, Acronyms, and Abbreviations | 4 |
| 1.4 | References | 4 |
| 1.5 | Overview | 4 |
| 2. | Overall Description | 4 |
| 3. | Specific Requirements | 4 |
| 3.1 | Functionality | 4 |
| 3.1.1 | Selection between Cash- and Management System Area | 4 |
| 3.1.2 | Management System | 4 |
| 3.1.3 | Cash | 5 |
| 3.2 | Usability | 5 |
| 3.2.1 | Instruction | 5 |
| 3.3 | Reliability | 6 |
| 3.3.1 | Availability | 6 |
| 3.3.2 | Mean Time Between Failures | 6 |
| 3.3.3 | Mean Time to Repair | 6 |
| 3.3.4 | Accuracy | 6 |
| 3.3.5 | Bugs | 6 |
| 3.4 | Performance | 6 |
| 3.4.1 | Response Time | 6 |
| 3.4.2 | Capacity | 6 |
| 3.5 | Supportability | 6 |
| 3.6 | Design Constraints | 6 |
| 3.6.1 | Software Languages | 6 |
| 3.6.2 | Process Requirements | 6 |
| 3.6.3 | Architecture | 6 |
| 3.6.4 | Class Libraries | 6 |
| 3.7 | On-line User Documentation and Help System Requirements | 6 |
| 3.8 | Purchased Components | 6 |
| 3.9 | Interfaces | 7 |
| 3.9.1 | User Interfaces | 7 |
| 3.9.2 | Hardware Interfaces | 7 |
| 3.9.3 | Software Interfaces | 7 |
| 3.9.4 | Communications Interfaces | 7 |
| 3.10 | Licensing Requirements | 7 |
| 3.11 | Legal, Copyright, and Other Notices | 7 |
| 3.12 | Applicable Standards | 7 |
| 4. | Supporting Information | 7 |

| | |
|-------------------------------------|----------------|
| PoS Systems | Version: 1.2 |
| Software Requirements Specification | Date: 20.10.16 |
| pos_rup_srs | |

1. Introduction

1.1 Purpose

This SRS will give an overview for the requirements of this project. Our application is a product group based cash for retailers, connected to a small management system. It should be user-friendly, fulfill all legal requirements and offer all functions needed in a small retail.

1.2 Scope

This software specification applies to the whole project and implies the cash and the management system.

1.3 Definitions, Acronyms, and Abbreviations

| | |
|-----|------------------------------------|
| SRS | Software Requirement Specification |
| UCD | Use-Case-Diagram |
| POS | Point-of-sale |
| n/a | not available |
| tbd | to be determined |

1.4 References

| Title | Date |
|---|----------|
| UCS: Manage Product Groups | 01.11.16 |
| UCS: Select between Management- & Cash-Area | 01.11.16 |

1.5 Overview

The following chapters are about our vision and the software requirements.

2. Overall Description

Our ambition is to create an application, which fulfills all legal requirements and still is user-friendly. It should deliver all functions needed in a small retail. Therefore the user needs the possibility to create and manage product groups and employee numbers, which are used as identifier in the system. There should also be the possibility to create some statistics with the old selling processes. During a selling process the user can select a product group and the price. After all products are added the user selects the payment methods and finishes the sales process. Different roles decide the access rights for employees. Depending on the rights, a user will be directly redirected to the cash or can decide rather he wants to use the cash or the management system.

3. Specific Requirements

3.1 Functionality

3.1.1 *Selection between Cash- and Management System Area*

The user has the possibility to choose between the Cash-Area for selling processes and the Management-Area to manage the master data, depending on user rights.

3.1.2 *Management System*

3.1.2.1 *Employee Management*

Manage employee data like cashier numbers, rights and passwords.

3.1.2.2 *Costumer Management*

Manage costumer data like name, surname, address, e-mail, etc.

| | |
|-------------------------------------|----------------|
| PoS Systems | Version: 1.2 |
| Software Requirements Specification | Date: 20.10.16 |
| pos_rup_srs | |

3.1.2.3 *Product Group Management*

Manage product group date, like number, name and profit margin.

3.1.2.4 *Statistics*

Shows different statistics in a specific period of time.

3.1.2.4.1 *Sales*

Shows all sold articles with date, receipt number, cashier number, price of sale, sales discount and transaction volume.

3.1.2.4.2 *Product Groups*

Shows transaction volume separated by product group.

3.1.2.4.3 *Cashier*

Shows transaction volume separated by cashier.

3.1.3 *Cash*

3.1.3.1 *Cash Opening Entry*

After starting the cash, the user has to enter the actual money paid in.

3.1.3.2 *Enter Customer Number*

If a costumer already has a costumer ID, the cashier can enter it for statistic purpose.

3.1.3.3 *Add New Costumer*

If a costumer hasn't a costumer ID, the cashier can assign a new one.

3.1.3.4 *Enter Product Group*

The cashier can choose between multiple product groups. After selecting one, he has to enter the price and the entry is shown in a shopping basket list.

3.1.3.5 *Sales Discount*

Every entry in the shopping basket list can be selected to give sales discount.

3.1.3.6 *Finish Sales Process*

After every article is entered, the cashier can finish the sales process by asking the costumer to pay and printing the receipt.

3.1.3.7 *Cash Check*

The cashier is shown the sum of the transaction volume since the last cash opening.

3.1.3.8 *Log out*

A cashier can log out, to make way for another cashier.

3.1.3.9 *Closing the Cash Accounts*

At the end of a day of sales, the cashier has to close the cash accounts.

3.2 **Usability**

3.2.1 *Instruction*

The application should be as clearly and as easy to use as possible and shouldn't allow a wrong handling. Therefore the instruction time should be below 15 minutes.

3.3 **Reliability**

3.3.1 *Availability*

The application should be available 99% of time that retailers aren't restricted to a specific time, but the

| | |
|-------------------------------------|----------------|
| PoS Systems | Version: 1.2 |
| Software Requirements Specification | Date: 20.10.16 |
| pos_rup_srs | |

main utilisation time will be between 7am and 9pm.

3.3.2 *Mean Time Between Failures*

The mean time between failures need to be at least one month.

3.3.3 *Mean Time to Repair*

The mean time to repair depends on the failure. Small failures need to be fixed in about 30 minutes, bigger ones, including corrupt data in the database, should be done within two hours.

3.3.4 *Accuracy*

Since our application is handling financial data, there need to be a 100% accuracy.

3.3.5 *Bugs*

There must not be any critical bugs, which lead to complete loss of data, corrupt data or wrong financial data. These need to be found during the tests. Significant bugs, like an inability to use certain parts of the application, also need to be prevented through extensive tests.

3.4 Performance

3.4.1 *Response Time*

To prevent slowing down the selling process, the transactions need to be completed in less than a second.

3.4.2 *Capacity*

At the start there should be only one running system of the application, but it should be scalable for future use.

3.5 Supportability

3.5.1 *IDE and Coding Conventions*

All Developers will be using Eclipse with the Spring Tool Suit (STS) plugin, as there are a lot of Tutorials for the Spring MVC Framework written for this IDE and it is a direct recommendation of the website "spring.io". Another benefit of STS is, that it does not require the purchase of a usage license and can be used for commercial purposes for free. The use of a singular IDE will make it easier to use a uniform coding style by using the same auto-format settings in the IDE.

3.6 Design Constraints

3.6.1 *Software Languages*

- Java EE 8
- HTML
- CSS

3.6.2 *Process Requirements*

tbd

3.6.3 *Architecture*

Spring MVC framework.

3.6.4 *Class Libraries*

tbd

3.7 On-line User Documentation and Help System Requirements

The whole application will be built with an intuitive design, so there shouldn't be need for help, but an instruction with relevant processes with follow.

| | |
|-------------------------------------|----------------|
| PoS Systems | Version: 1.2 |
| Software Requirements Specification | Date: 20.10.16 |
| pos_rup_srs | |

3.8 Purchased Components

tbd

3.9 Interfaces

3.9.1 User Interfaces

tbd

3.9.2 Hardware Interfaces

tbd

3.9.3 Software Interfaces

n/a

3.9.4 Communications Interfaces

tbd

3.10 Licensing Requirements

n/a

3.11 Legal, Copyright, and Other Notices

n/a

3.12 Applicable Standards

tbd

4. Supporting Information

n/a