

Software Requirements Specifications  
for  
Project “BookExpress”

Marc A. Harnos  
mharnos@gmail.com

Joscha Rapp  
jraxxo@gmail.com

Christian Schulz  
crs.s@gmx.net

October 2012

# Contents

<b>1</b>	<b>Product Purpose</b>	<b>4</b>
1.1	Obligatory Requirements (“must have”) . . . . .	4
1.1.1	Communication with Publishers . . . . .	4
1.1.2	Communication with Customers . . . . .	4
1.1.3	Internal Administration . . . . .	5
1.1.4	Support . . . . .	5
1.2	Optional Requirements (“nice to have”) . . . . .	6
1.3	Non-Requirements (“need not have”) . . . . .	6
<b>2</b>	<b>Product Environment</b>	<b>7</b>
2.1	Application Area . . . . .	7
2.2	User Groups . . . . .	7
2.3	Operating Conditions . . . . .	7
<b>3</b>	<b>Product Overview</b>	<b>8</b>
<b>4</b>	<b>Product Functions</b>	<b>9</b>
4.1	Internal . . . . .	9
4.1.1	Employees of “BookExpress” . . . . .	9
4.2	External . . . . .	16
4.2.1	Book Stores (“Clients”) . . . . .	16
4.2.2	Book Stores and Publishers . . . . .	16
4.2.3	Book Stores (“Clients”) . . . . .	17
4.2.4	Publishers . . . . .	20
<b>5</b>	<b>Product Data</b>	<b>21</b>
5.1	User Data . . . . .	21
5.1.1	Book Stores . . . . .	21
5.1.2	Publishers . . . . .	21
5.2	Order Data . . . . .	21
5.3	Internal Data . . . . .	22
5.3.1	Book Stock . . . . .	22
5.3.2	Truck Information . . . . .	22

<b>6</b>	<b>Product Performance</b>	<b>23</b>
6.1	System . . . . .	23
6.2	Database . . . . .	23
6.3	Security . . . . .	24
<b>7</b>	<b>Quality Requirements</b>	<b>25</b>
<b>8</b>	<b>User Interface</b>	<b>26</b>
8.1	Client Application (for clerks) . . . . .	26
8.2	Web Interface (for customers) . . . . .	26
<b>9</b>	<b>Non-Functional Requirements</b>	<b>27</b>
<b>10</b>	<b>Technical Product Environment</b>	<b>28</b>
10.1	Software . . . . .	28
10.2	Hardware . . . . .	28
10.3	Orgware . . . . .	28
10.4	Interfaces (product) . . . . .	28
<b>11</b>	<b>Special Requirements for the Development Environment</b>	<b>29</b>
11.1	Software . . . . .	29
11.2	Hardware . . . . .	29
11.3	Orgware . . . . .	29
11.4	Interfaces (development) . . . . .	29
<b>12</b>	<b>Subproducts and Subsystems</b>	<b>30</b>
12.1	Server . . . . .	30
12.2	Web Interface / Web Service . . . . .	30
12.3	Client Application . . . . .	30
<b>13</b>	<b>Additional Specifications and Stipulations</b>	<b>31</b>
<b>14</b>	<b>Appendices</b>	<b>32</b>

# Document History

Editor(s)	Date	Purpose of Editing	Version
Harnos, Rapp, Schulz	2012-10-01	Initial Document Creation	v0.01

# Chapter 1

## Product Purpose

To handle the addition of new books and the distribution in a better way, "BookExpress" asked us to develop an IT solution to optimize the idle time and labour usage by getting rid of the current, deprecated system. Furthermore one important goal of the software solution is to be very user friendly and easy to use; also there should be remote access implemented, so several customers can access the software at once and distribution partners can access and keep their book stock up to date.

### 1.1 Obligatory Requirements (“must have”)

#### 1.1.1 Communication with Publishers

The direct connection between the **publishers** and BookExpress shall be implemented by providing a web interface for the publishers.

- As each publisher also has a PIN, it will be used as the account name for the interface.
- The web interface has to provide the following features for the publisher: maintain the book database (add books, remove books, edit book data), edit their personal data, view statistics on how often which books are bought
- The interface has to be comfortably accessible from mobile devices as well, the data that is transmitted should be minimal

#### 1.1.2 Communication with Customers

The direct connection between the **customers** and BookExpress shall also be implemented by providing a web interface for the customers.

- As each customer already has a unique identifier, the PIN, we will use it as an account name for the login.
- The web interface has to provide the following features: view catalogue and stock, submit orders, view the current state of their orders and cancel them (if they haven't been shipped yet)
- The system has to assign a unique ID to each order, ideally incorporating the PIN of the customer
- The interface has to be comfortably accessible from mobile devices as well, the data that is transmitted should be minimal

### **1.1.3 Internal Administration**

The employees of BookExpress shall be provided a web interface as well.

- Each employee will get a unique user ID to log into the web interface. Ideally, this will be formatted along the lines of surname.name or something similar.
- The web interface must provide the following features for the employees: processing, editing and cancellation of submitted orders, directly contact the customers and publishers as well as the truck fleet and the warehouse
- Furthermore, the underlying system shall generate logfiles with login times, error codes and statistics (e.g. which books are ordered the most and by whom) for the publishers and the accounting and IT department / tech support.
- The interface has to be comfortably accessible from mobile devices as well, the data that is transmitted should be minimal

### **1.1.4 Support**

- The system should be as automatic as possible - we need a good, user-oriented manual.
- Additionally, a context-sensitive help tool has to be implemented in the interface itself.
- Moreover, As we want to make sure that the system is as reliable as possible, we will need a dedicated support team that is available 24/7 and can be dispatched in less than 2 hours to fix critical issues.

## **1.2 Optional Requirements (“nice to have”)**

- Book Stores should be able to save recurring orders
- Book Stores should be notified when the delivery of their order is imminent

## **1.3 Non-Requirements (“need not have”)**

- The software should not allow final customers to buy books. It is only intended for retailers.
- Moreover, invoices are generated elsewhere, as well as financial reports, so the system should not interfere with that.

## **Chapter 2**

# **Product Environment**

The Product takes the orders from the book stores and files them into the system, so that they can be prepared for shipping. For different sized book stores there should be several account types with separate functionality and different options for packaging and ordering. The targeting groups of the software solution are the book store owners(our consumers), the assistants of "BookExpress" and the distribution partners.

### **2.1 Application Area**

The new IT system should replace the old system and provide administrative and commercial functions to relieve the employees of unnecessarily complex administrative tasks. Certain processes can be handled without any interaction at all, e.g. if a book runs low in stock, the system will automatically order new copies from the publisher.

### **2.2 User Groups**

- BookExpress commercial employees
- BookExpress IT employees
- Publishers
- Book Stores

### **2.3 Operating Conditions**



## Chapter 3

# Product Overview

The products environmental diagram.

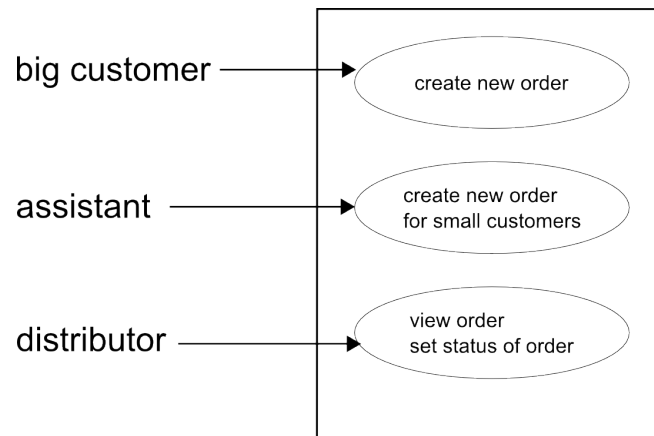


Figure 3.1: The environmental view of the product

## Chapter 4

# Product Functions

### 4.1 Internal

#### 4.1.1 Employees of “BookExpress”

/PF10/	<b>Process</b>	Register Publisher or Book Shop (client)
	<b>Actor(s)</b>	BookExpress employee
	<b>Category</b>	Must have
	<b>Description</b>	A Book Express employee creates an account for a book shop or publisher after both business partners have signed a contract handling the business relations.
	<b>Goal</b>	The contract partner should have an account to log into the Software. After registration through the Book Express employee, the contract partner will be send a verification and its information via email.
	<b>Preconditions</b>	Contract signed
	<b>Postcondition success</b>	Successfully created account
	<b>Postcondition failure</b>	Unsuccessful in creating an account
	<b>Trigger</b>	Menu option "register new client" (Software)
	<b>Sequence</b>	Input relevant data and contractor details into the managements system Save the signed contract on the server Register a new client by creating a new PIN in the database Send verification and account information to the contractor

/PF20/	<b>Process</b>	Edit client information
	<b>Actor(s)</b>	BookExpress employee
	<b>Category</b>	
	<b>Description</b>	The clients data is outdated and needs to be updated
	<b>Goal</b>	Client data is updated
	<b>Preconditions</b>	Client exists in database
	<b>Postcondition success</b>	Client data is updated and a notification mail is sent to the client.
	<b>Postcondition failure</b>	Client data is not updated but rolled back to the previous state. Notification with problem report is shown to the employee.
	<b>Trigger</b>	Menu option "edit client data" (Software)
	<b>Sequence</b>	Menu option "edit client data" Edit specific data Submit edited data to the server
/PF02/	<b>Process</b>	Delete account
	<b>Actor(s)</b>	BookExpress Employee
	<b>Category</b>	
	<b>Description</b>	Client gets deleted due to resignation of the contract
	<b>Goal</b>	Delete client from the system
	<b>Preconditions</b>	Client exists in database Contract is being resigned by one of the contractors
	<b>Postcondition success</b>	Client is deleted from the system
	<b>Postcondition failure</b>	Client could not be deleted because of existing orders
	<b>Trigger</b>	Menu option "delete client" (Software)
	<b>Sequence</b>	Choose client Check if client is deletable (i.e. no existing orders) Send notification mail to client Delete client

/PF/	<b>Process</b>	Client places order
	<b>Category</b>	
	<b>Actor(s)</b>	BookExpress employee, client
	<b>Description</b>	A client orders book via telephone
	<b>Goal</b>	Order is placed in an order queue
	<b>Preconditions</b>	Client exists in database Client is authorized
	<b>Postcondition success</b>	Order is placed in an order queue
	<b>Postcondition failure</b>	-
	<b>Trigger</b>	Client initiates order process by calling an BookExpress employee BookExpress employee triggers manual order process
	<b>Sequence</b>	BookExpress employee selects client in Software BookExpress employee enters ordered book into the system Order is placed in order queue Order notification containing detailed report is sent to client
/PF/	<b>Process</b>	Validate order
	<b>Category</b>	
	<b>Actor(s)</b>	System
	<b>Description</b>	Order is checked for validation
	<b>Goal</b>	The software checks every order for validation and confirms it to the client. BookExpress knows about every orders validation status.
	<b>Preconditions</b>	Order is in order queue
	<b>Postcondition success</b>	Client receives a confirmation email
	<b>Postcondition failure</b>	Client receives a problem report about invalid order
	<b>Trigger</b>	Order was placed and exists in order queue
	<b>Sequence</b>	System gathers details and checks whether order is valid or not System sends mail to client

/PF/	<b>Process</b>	Stock Update
	<b>Category</b>	Must have
	<b>Actor(s)</b>	System/BookExpress employee
	<b>Description</b>	The employee in charge of the stock checks the availability of all books and updates the stock accordingly.
	<b>Goal</b>	Check succeeds
	<b>Preconditions</b>	Order is placed or BookExpress employee initiates check
	<b>Postcondition success</b>	Books which are nearly out of stock get displayed as a list
	<b>Postcondition failure</b>	-
	<b>Trigger</b>	Stock check is initiated
	<b>Sequence</b>	Check if a book is available in an sufficient amount Make a list of all books which need to be ordered Submit list to system
/PF/	<b>Process</b>	Forward order to Logistics
	<b>Category</b>	
	<b>Actor(s)</b>	System
	<b>Description</b>	A validated order gets forwarded to Logistics
	<b>Goal</b>	Logistics proceeds processing the order
	<b>Preconditions</b>	Order is valid
	<b>Postcondition success</b>	Logistics processes delivery
	<b>Postcondition failure</b>	Order could not be forwarded to Logistics
	<b>Trigger</b>	Order is validated
	<b>Sequence</b>	Send order information/details to Logistics

/PF/	<b>Process</b>	Deliver order
	<b>Actor(s)</b>	System
	<b>Category</b>	
	<b>Description</b>	The order is send to the book store
	<b>Goal</b>	Deliver ordered books
	<b>Preconditions</b>	Order is valid
		Books are available in stock
	<b>Postcondition success</b>	Order is marked as sent
	<b>Postcondition failure</b>	Order could not be marked as sent
	<b>Trigger</b>	Forward order to Logistics
	<b>Sequence</b>	Check if books are in stock
Receive book store information		
Sent order to book store		
Mark order as sent		
/PF/	<b>Process</b>	Confirm delivery
	<b>Actor(s)</b>	Employee (Logistics)
	<b>Category</b>	
	<b>Description</b>	Logistics confirms the correct delivery of an order
	<b>Goal</b>	Delivery is successful
	<b>Preconditions</b>	Order is marked as sent
	<b>Postcondition success</b>	Order is marked as delivered
	<b>Postcondition failure</b>	Order could not be marked as delivered
	<b>Trigger</b>	Deliver order
	<b>Sequence</b>	Get delivery status
		Mark order as delivered if delivery status is "successfully delivered"

/PF/	<b>Process</b>	BookExpress Order
	<b>Actor(s)</b>	BookExpress employee
	<b>Category</b>	
	<b>Description</b>	The Employee orders a number of books from the publisher
	<b>Goal</b>	Order books from a publisher
	<b>Preconditions</b>	need for new books
	<b>Postcondition success</b>	Books are ordered and publisher gets an notification mail
	<b>Postcondition failure</b>	-
	<b>Trigger</b>	BookExpress employee initiates an order
	<b>Sequence</b>	Select books from list or search TODO /PF/ Submit list to publisher
/PF/	<b>Process</b>	Create an invoice
	<b>Actor(s)</b>	System
	<b>Category</b>	
	<b>Description</b>	The system creates an invoice after TODO /PF/ order is valid
	<b>Goal</b>	Documenting order and deliver it with invoice
	<b>Preconditions</b>	Order is valid
	<b>Postcondition success</b>	Invoice successfully created
	<b>Postcondition failure</b>	Invoice could not be created
	<b>Trigger</b>	Check information about order
	<b>Sequence</b>	Order is valid TODO /PF/ Create an entry in database List the ordered books with price List tax to invoice

/PF/	<b>Process</b>	Send invoice
	<b>Actor(s)</b>	System
	<b>Category</b>	
	<b>Description</b>	The invoice is send to the book store via mail
	<b>Goal</b>	Inform the book shop
	<b>Preconditions</b>	Successful created invoice
	<b>Postcondition success</b>	Book store receives a notification mail
	<b>Postcondition failure</b>	-
	<b>Trigger</b>	Creation of an invoice
	<b>Sequence</b>	Get invoice
		Send invoice to book store
/PF/	<b>Process</b>	Backup
	<b>Actor(s)</b>	System
	<b>Category</b>	
	<b>Description</b>	A backup of the complete databases must be done. This should happen periodically and save all data on a server.
	<b>Goal</b>	In case of a crash, every data must be saves as a preventative backup
	<b>Preconditions</b>	Server for the backup
	<b>Postcondition success</b>	Successful backup in periodical intervals
	<b>Postcondition failure</b>	A notification will be send to the administrator of the system
	<b>Trigger</b>	Configured time to start a backup
	<b>Sequence</b>	Start Backup
		Place backup on the backup server



## 4.2 External

### 4.2.1 Book Stores (“Clients”)

#### 4.2.2 Book Stores and Publishers

/PF/	<b>Process</b>	Login
	<b>Actor(s)</b>	Book store/publisher
	<b>Category</b>	Must have
	<b>Description</b>	The user logs on with his PIN and password
	<b>Goal</b>	Provide a fully functional system to the user
	<b>Preconditions</b>	Existing account
	<b>Postcondition success</b>	Successful login
	<b>Postcondition failure</b>	Unsuccessful login caused by a wrong PIN or password
	<b>Trigger</b>	Menu Option "Login"
	<b>Sequence</b>	Enter PIN Enter password Validity of data is checked by the System Login successful or display an error message in case of unsuccessful login
/PF/	<b>Process</b>	Update account information
	<b>Actor(s)</b>	Book store/publisher
	<b>Category</b>	Must have
	<b>Description</b>	The user is able to edit/update its account information (e.g. address, correspondent etc.)
	<b>Goal</b>	User is able to update changes
	<b>Preconditions</b>	User is logged on
	<b>Postcondition success</b>	Updating the account succeeded
	<b>Postcondition failure</b>	Updating the account failed
	<b>Trigger</b>	Menu option "edit"
	<b>Sequence</b>	Edit specific data  System checks correctness of data (e.g. email format etc.) Submit edited data to the server

/PF/	<b>Process</b>	Logout
	<b>Actor(s)</b>	Book store/publisher
	<b>Category</b>	Must have
	<b>Description</b>	User logs out
	<b>Goal</b>	User is logged out
	<b>Preconditions</b>	User is logged in
	<b>Postcondition success</b>	Successful logout
	<b>Postcondition failure</b>	Unsuccessful logout caused by an abort of the user
	<b>Trigger</b>	Menu option "Logout"
	<b>Sequence</b>	User confirms to log out The system terminates the current session

#### 4.2.3 Book Stores (“Clients”)

/PF/	<b>Process</b>	Search for a book/show catalogue
	<b>Category</b>	Must have
	<b>Actor(s)</b>	Client
	<b>Description</b>	Client searches for a book to add it to his order
	<b>Goal</b>	Searching for a book succeeded
	<b>Preconditions</b>	Client is logged into the system
	<b>Postcondition success</b>	Book is being found
	<b>Postcondition failure</b>	Display items and options Book could not be found
	<b>Trigger</b>	Display notification
	<b>Sequence</b>	Menu option "find books" (Software) Enter ISBN/title/author/genre in search bar Hit the "Search"-Button Choose book from list to add it to the order

/PF/	<b>Process</b>	Book Store Order
	<b>Category</b>	
	<b>Actor(s)</b>	Book Store Employee
	<b>Description</b>	The Book Store orders a book from BookExpress. This can either happen through a constant connection where every order is individually processed or the store can choose to collect orders internally and send them as one big order every day.
	<b>Goal</b>	Order is placed
	<b>Preconditions</b>	Client is successfully logged into the web interface Connection to BookExpress server is established
	<b>Postcondition success</b>	Order is delivered to BookExpress and added to the order queue
	<b>Postcondition failure</b>	Order is saved as draft and client gets a notification
	<b>Trigger</b>	Menu Option "create new order" (Web interface)
	<b>Sequence</b>	Client adds books to order (via book search or ISBN) Client specifies amount for every book Client submits order to server Order is in the order queue

/PF/	<b>Process</b>	Change order
	<b>Actor(s)</b>	Book store
	<b>Category</b>	
	<b>Description</b>	As long as the order is not validated, the user can change information
	<b>Goal</b>	User can correct an order
	<b>Preconditions</b>	Order exists and user is logged in
	<b>Postcondition success</b>	Order information successfully changed
	<b>Postcondition failure</b>	Order information could not be changed because the order is already being validated
	<b>Trigger</b>	Menu option "Change order"
	<b>Sequence</b>	Select order to be changed
		System checks validity status of the order If not validated, user can change order information Submit changes to the System User receives a confirmation mail
/PF/	<b>Process</b>	Cancel Order
	<b>Actor(s)</b>	Book Store Employee
	<b>Category</b>	
	<b>Description</b>	The employee at the book store can cancel the order if it hasn't been shipped yet
	<b>Goal</b>	Current orders can be cancelled
	<b>Preconditions</b>	Order exists
	<b>Postcondition success</b>	Order successfully cancelled. Book Store receives a confirmation mail
	<b>Postcondition failure</b>	Book Store receives a problem report
	<b>Trigger</b>	Menu option "cancel order"
	<b>Sequence</b>	Select order to cancel
		Check whether order isn't shipped yet Cancel order

/PF/	<b>Process</b>	Tracking System
	<b>Actor(s)</b>	Book Store Employee
	<b>Category</b>	
	<b>Description</b>	The employee at the book store can look up at which stage his order is right now, e.g. processing or shipping to BookExpress/the book store.
	<b>Goal</b>	Look up order information/status
	<b>Preconditions</b>	Order exists
	<b>Postcondition success</b>	Employee receives information/status of order
	<b>Postcondition failure</b>	-
	<b>Trigger</b>	Menu option "tracking system"
	<b>Sequence</b>	Select order to track

#### 4.2.4 Publishers

/PF10/	<b>Process</b>	Inventory maintenance
	<b>Actor(s)</b>	Publisher
	<b>Category</b>	Must have
	<b>Description</b>	Publishers can update their inventory list (i.e. the list of available books) either via mail or using the direct connection over the web-based interface.
	<b>Goal</b>	Updated inventory list
	<b>Preconditions</b>	Publisher exists
	<b>Postcondition success</b>	Send success mail to publisher
	<b>Postcondition failure</b>	Send problem report to publisher
	<b>Trigger</b>	Menu option "inventory maintenance" (Software) or via mail
	<b>Sequence</b>	Edit inventory list (i.e with TODO book search) Submit list to server

## Chapter 5

# Product Data

### 5.1 User Data

#### 5.1.1 Book Stores

/PD20/	<b>Data</b>	Book Store Data
	<b>Composition</b>	Name, PIN, Address, Correspondent, E-Mail Address, Phone Number, Contract Data
	<b>Size</b>	approx 10,000
	<b>Traffic</b>	Low

#### 5.1.2 Publishers

/PD30/	<b>Data</b>	Publisher Data
	<b>Composition</b>	Name, PIN, Address, Correspondent, E-Mail Address, Phone Number, Contract Data
	<b>Size</b>	approx 15,000
	<b>Traffic</b>	Low

### 5.2 Order Data

/PD40/	<b>Data</b>	Order Data
	<b>Composition</b>	PIN, ISBN, Amount, Price, Date, Current Ordering State
	<b>Size</b>	500,000
	<b>Traffic</b>	Very High

/PD41/	<b>Data</b>	Order Data History
	<b>Composition</b>	PIN, ISBN, Amount, Price, Date
	<b>Size</b>	2,500,000
	<b>Traffic</b>	High

## 5.3 Internal Data

### 5.3.1 Book Stock

/PD10/	<b>Data</b>	Book Data
	<b>Composition</b>	ISBN, UPC/EAN, Title, Author, Publisher, Availability
	<b>Size</b>	approx 1,500,000
	<b>Traffic</b>	Average

### 5.3.2 Truck Information

/PD50/	<b>Data</b>	Truck Fleet Data
	<b>Composition</b>	Driver, Registration Number, Status
	<b>Size</b>	1,000
	<b>Traffic</b>	Average

## Chapter 6

# Product Performance

### 6.1 System

/PP10/	<b>Type</b>	System (Web Interface)
	<b>Description</b>	The Web Interface should be always reachable during business hours, but also 24/7 for quick changes. The availability outside business hours can be less than 100% but should not fall below 99.7%.
/PP30/	<b>Type</b>	System (Order Processing)
	<b>Description</b>	The order processing should always be reachable, confirming (or informing about, in case of error) any order.
/PP40/	<b>Type</b>	System
	<b>Description</b>	The System in general should be reliable and responsive all the time, taking not longer than 1-3 seconds for any request.
/PP50/	<b>Type</b>	System
	<b>Description</b>	The response time criteria should also be met, when the processing limit of max. 2.000 simultaneous orders is reached.

### 6.2 Database

/PP20/	<b>Type</b>	Database (DB Connection)
	<b>Description</b>	A Connection to the database should never be lost, an average uptime of 99.99% has to be reached to satisfy any part of the system, currently needing data.



/PP21/	<b>Type</b>	Database (DB Backup)
	<b>Description</b>	A database backup has to be initiated meeting the guidelines on general backup, also ensuring a consistent and complete history of any processed orders and any critical data.

### 6.3 Security

/PP60/	<b>Type</b>	Security (Stock Availability)
	<b>Description</b>	Stock items should always be able to be ordered, ensuring fast delivery on order.

/PP61/	<b>Type</b>	Security (Connection)
	<b>Description</b>	Any connection issued to the system and between system components should be at least encrypted with 256 bit SSL.

/PP62/	<b>Type</b>	Security (Database)
	<b>Description</b>	The database should be on an encrypted device, securing personal customer data.

## Chapter 7

# Quality Requirements

Product Quality	excellent	good	normal	not applicable
<b>Functionality</b>				
suitability		✓		
correctness	✓			
interoperability		✓		
conformity		✓		
security	✓			
<b>Reliability</b>				
maturity		✓		
error tolerance		✓		
re-invoking	✓			
<b>Usability</b>				
comprehensibility		✓		
learnability			✓	
ease of use		✓		
<b>Efficiency</b>				
time behaviour	✓			
resource use			✓	
<b>Maintainability</b>				
easy to analyse		✓		
easy to modify			✓	
stability	✓			
testability				✓
<b>Installability</b>				
adaptability			✓	
easy to install			✓	
conformity		✓		
exchangeability				✓

## **Chapter 8**

# **User Interface**

**8.1 Client Application (for clerks)**

**8.2 Web Interface (for customers)**

## **Chapter 9**

# **Non-Functional Requirements**

## **Chapter 10**

# **Technical Product Environment**

**10.1 Software**

**10.2 Hardware**

**10.3 Orgware**

**10.4 Interfaces (product)**

## **Chapter 11**

# **Special Requirements for the Development Environment**

**11.1 Software**

**11.2 Hardware**

**11.3 Orgware**

**11.4 Interfaces (development)**

## **Chapter 12**

# **Subproducts and Subsystems**

**12.1 Server**

**12.2 Web Interface / Web Service**

**12.3 Client Application**

## **Chapter 13**

# **Additional Specifications and Stipulations**



## **Chapter 14**

# **Appendices**