# B3Albania_OSMS Requirements Specification

## Version 1.0

## April 2, 2019

# Table of Contents

## *Executive Summary*

### *1.1 Project Overview*

This project is about creating and building a software, specifically a web-based application that manages the activity within a company. The software is called OSMS (Online Shop Management System). OSMS can be used both by the clients/costumers and staff. It covers different procedures from online ordering on the client site to inventory, products, price, and orders management on staff site (admin or storekeeper).

The client can make the orders directly from a web browser and also can check the status of its own order. This is a very efficient way of buying, as it saves time and money (going to the shop, finding a parking place which may cost). Basically, the app has a special user interface for each user.

Another task being managed by the app is sales management. The lists of products are updated every time a new product is added in a specific category. This lists are accessible by the costumers, who can see the prices, product description and photo. The staff users can see the orders, the status of the order, the sale's progress. Also they could ask the system for daily, monthly, yearly reports (ex. How many orders, feedback for a specific product etc.) .

### *1.2 Purpose and Scope of this Specification*

The purpose of this software is to provide a set of solutions to the company that depend on sales management and user management. In the current state, a full lifecycle of B3Albania when it comes to sales is very unorganized and unprofessional. The administrator has no way of get ready automatic reports, and the employees have a very frustrated way of managing the orders. Additionally the clients could not buy without going in the shop or at least by calling the company to make an order and this is so uncomfortable because imagine one hundred people that want to buy products by a call at the same time. Orders are often overlooked and this leads to major problems in the future, which may result in bankruptcy. This software will provide a very organized, hybrid and comprehensive solution for these issues, which will increase the cooperation between administrators, employees and customers. It will make the life inside the company easier and save needed time and energy for both parts that go through the process (staff and costumers).

## *2. Product/Service Description*

B3Albania is a new company in the market, founded in December 2017. Is started by trading perfumes for kids and after that continued with the skin care (mostly concentrated in sun protection) and recently going on with nails care. The company is mainly focused in children stuff. As it is a sale based company the main goal is to have higher sale rates in a shorter time and simpler way.

OSMS (Online Shop Management System) is a software which aims to redesign the lifecycle of the company and provide new technological solutions to every step of the process. This software will be used by the administrator, storekeeper and the customer. Our mission is to create a dynamic software, which will prove useful to every user of the company (staff and clients).

The software aims to help every step of the company job, starting from simple order receiving to order status tracking, report generation, dynamic form filling and much more. If we divide the company in two major subsections we are able to give a more detailed description of what this software provides for both parties.

Company Staff (Administrator, Storekeeper, Economists)

- Real time Statistics, Sales Performance, Charts and Graphs
- Live Feeds
- Order Receiving and Tracking, Dynamic Forms, Reports and much more.

Company Costumers

- Order Making
- Order Status Checking
- Offers Providing
- Chat Support

### *2.1 Product Context*

The context of this software is related to the B3Albania, even though our scope is to provide a solution for every branch of the company. This solution will be used by every office employee of the company and also by the costumers in order to enhance their cooperation between staff and clients, make their jobs/purchases easier and make sure they make a full use of their time and energy.

### *2.2 User Characteristics*

There are three user types that will take advantage and use the software:

- Administrator
- Manager
- Costumer

a) Administrator
   - This user will be responsible for fulfilling the product quantity, adding new products, categories. This user will be provided an administrator level account that will be able to CRUD product, categories, orders, users etc. These user is the backbone of the system, making sure that data is generated in the system by continuously observing the reports.

b) Manager
   - This user will be responsible for order receiving from all costumers, making them ready, send them to the costumer, checking the order status, and the product availability. This user will be provided a manager level account that will be able to Read Update Send orders. These users are the engine of the system, making sure that orders are successfully processed in all their lifecycle.

c) Costumer

- This user will be responsible for **only** its orders, chart, and purchases. This user will be provided a classic level account that will be able to CRUD only its own orders. This user is the king of the system (company) by purchasing the products.

### *2.3 Assumptions*

- It is assumed that the B3Albania has the right to go through all the data generated from every other branch of company according to law.
- It is assumed that the data generated from the system will be fully confidential and only available to the inspectorate and/or higher state institutions which govern the inspectorate.
- It is assumed that every manager is equipped with a smart device from where they will perform every operation they are required to. If not, the company is assumed responsible for equipping every manager with the required device. These smart devices have to be capable of receiving information from web browsers. The smart devices are assumed to have an IOS, Android, Windows, LINUX, Ubuntu etc. Operating System.
- It is assumed further that the managers are able to use smart devices and especially the OSMS web application effectively and efficiently.
- It is assumed that every costumer should be limited to watching only his orders and not interfere with other costumers.
- It is assumed that office employees (administrator, manager) have a web browser and an active internet connection.
- It is assumed that every completed order needs to be stored in the system for documentation and report purposes.
- It is assumed that the manager is responsible for the progress of the orders.
- It is assumed that the administrator is responsible for updating stock in real time.

### *2.4 Constraints*

This system will be potentially constrained by:

- The fact that employees have to be equipped at all times with smart devices
- The need of a fast internet connection and real time database update.
- Having every employees understand the way the system works and training them to use the web application correctly and efficiently.
- The need of a browser in every device to run the application.

### *2.5 Dependencies*

List dependencies that affect the requirements.

- Users need to have a strong internet connection in order to run effectively.

- Customer should be logged-in in case of making an order.
- The order needs to satisfy product stock availability.
- The costumer need to have at least 1 product in cart in case to make an order.
- The system is also dependent on the browser performance.
- Products are on stock but are expired and cannot be ordered.

## *3. Requirements*

**Priority Definitions**

The following definitions are intended as a guideline to prioritize requirements.

- Priority 1 – The requirement is a "must have" as outlined by policy/law
- Priority 2 – The requirement is needed for improved processing, and the fulfillment of the requirement will create immediate benefits
- Priority 3 – The requirement is a "nice to have" which may include new functionality

It may be helpful to phrase the requirement in terms of its priority, e.g., "The value of the employee status sent to DIS **must be** either A or I" or "It **would be nice** if the application warned the user that the expiration date was 3 business days away". Another approach would be to group requirements by priority category.

### *3.1 Functional Requirements*

a. I want you to build for me a website for B3 from where people can make online purchases. (Original version: Dua te me ndertosh per B3 nje website nga ku njerezit mund te bejne blerje online.)

   i. The software must be a web-based application.

   ii. The service that the system will offer will be based on online sales and purchases.

   iii. The application must have a specific UI (user interface) for client.

   iv. Client should register in order to gain access to online transactions.

b. Also I want to have the opportunity to check these orders together with the manager for making them ready to deliver to respective clients. (Original version: Gjithashtu une te kem mundesi ti kontrolloj keto porosi se bashku me magazinieren qe ti bejme gati per ti bere dergesat neper addresat perkatese.)

   i. The application must have two other user interfaces administrator and manager.

   ii. The manager should provide the access of reading orders and updating their status.

   iii. The administrator must also provide access on orders.

c. I also want to have the opportunity to easily add new products or new lines that we can add later in company. (Original version: Po ashtu te kem mundesi dhe une te shtoj produkte te reja lehtesisht ose linja te reja qe mund te fusim me vone.)

   i. The administrator must have the highest level of user access.

   ii. The administrator must have access to CRUD categories, products, orders, users.

d. And if you can make it that at the end of each month to have the opportunity to print the monthly sales performance to see how the online store is going on. ( Original version: Dhe nese ke mundesi ne fund te muajit te kem mundesi te printoj ecurin mujore te shitjeve per te pare sesi po ecen dyqani online.)

   i. The system must generate automatically every last day of the month the sales report.

   ii. The reports must be accessed only by the administrator.

   iii. The statistical data will be displayed by charts or graphs in order to review and compare easily to each other.

   iv. The admin must have the ability to filter the reports depended on his needs.

   v. The system must convert the reports in PDF format in order to be printed in a correct format.

   vi. The system should provide a print service.

   vii. The user orders and data will be saved by the system for statistical issues.

e. I am open for any suggestions that may come out during the way, on what we can add or remove. (Original version: Per cdo sygjerim qe mund te dali rruges se cfare mund te shtojme ose te heqim i mirepres.)

   i. The Software Development Life Cycle will be in an iterative process flow.

f. Attached I am sending you the lists of the products with the relevant descriptions. (Original version: Bashkengjitur po te coj dhe listen e produkteve me pershkrimet perkatese.)

   i. In user interface a specific description will be displayed for every product.

   ii. In user interface the price will be displayed for every product.

| Req# | Requirement | Comments | Priority | Date Rvwd | Written/ Reviewed & Approved |
|---|---|---|---|---|---|
| BR_01 | The software must be a web-based application. | | 1 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_02 | The service that the system will offer will be based on online sales and purchases. | | 2 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_03 | The application must have a specific UI (user interface) for client. | | 1 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_04 | Client should register in order to gain access to online transactions. | | 1 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_05 | The application must have two other user interfaces administrator and manager. | | 1 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_06 | The manager should provide the access of reading orders and updating their status. | | 1 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_07 | The administrator must also provide access on orders. | | 1 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_08 | The administrator must have the highest level of user access. | | 1 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_09 | The administrator must have access to CRUD categories, products, orders, users. | | 1 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_10 | The system must generate automatically every last day of the month the sales report. | | 2 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_11 | The reports must be accessed only by the administrator. | | 1 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_12 | The statistical data will be displayed by charts or graphs in order to review and compare easily to each other. | | 3 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_13 | The admin must have the ability to filter the reports depended on his needs. | | 2 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_14 | The system must convert the reports in PDF format in order to be printed in a correct format. | | 2 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_15 | The system should provide a print service. | | 2 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_16 | The user orders and data will be saved by the system for statistical issues. | | 1 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_17 | The Software Development Life Cycle will be in an iterative process flow. | | 2 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_18 | In user interface a specific description will be displayed for every product. | | 2 | 29/03/2019 | Greta Daci, Tajda Kumbulla |
| BR_19 | In user interface the price will be displayed for every product. | | 2 | 29/03/2019 | Greta Daci, Tajda Kumbulla |

## *3.2 Non-Functional Requirements*

### 3.2.1 User Interface Requirements

The main application shall be a web application, which can be used either with Mozilla, Internet Explorer, Chrome or Safari.

The main page of web application for the customer side will display the menu, the categories, and products with images, descriptions and prices. The costumer can filter the products, add them to wish list, add them to card or order them to buy. Costumers must create accounts in order to purchase. In their accounts they could see costumer info, addresses (only their), orders made by them, their wish list and shopping card.

The main page of web application for staff side shall display a simple login interface, where it will ask the user for its Username and Password. Based on the given credentials, the necessary system constraints will apply. The user will gain access to the system, in case of proven authenticity otherwise, an error message of invalid credentials will be displayed.

As part of the system structure different system modules are included in staff interfaces, such modules are, dashboard module, user module, orders module, report module and the settings module. Once the user is logged in the web application, he/she shall have access to specific modules of the system. The will redirect the user to the dashboard module. Thus the page displayed by the system after the login interface, shall provide additional information, regarding the sales overall performance.

The last module provided by the system, is the reports model. This model, just like in the dashboard model will provided statistical and useful information, regarding the sales, order and products.

The format of such reports shall be mainly in pdf in case of printing them and excel format in case of downloading them.

### 3.2.2 Usability

Include any specific usability requirements, for example,

- Learnability
    - The application is simple to use and understand.
    - The web application will come together with a PDF manual, providing a step by step information on how to effectively use the system.
    - Specific error messages will be displayed, by also identifying the specific action, that caused the error.
    - The application is specified for certain users, thus the system will know, when a certain action is not allowed.
- Accessibility
    - The software shall be easy to access remotely and at all times, since both customers and staff will use the application on their devices (laptop, computer, smartphone etc.).
- Responsiveness

- The software shall be responsive both in design and data transactions.

- Flexibility
  - The software shall be easy to update in order to accommodate new requirements.
  - The software shall be designed in such a way that the isolation and management of errors is possible

- Effectiveness
  - The software shall provide both staff and clients with practical tools of managing their data and with a convenient way of communicating their needs across the platform.

- Efficiency
  - The software will provide users and administrators with a fast and reliable way of accomplishing their goals such as updating products information in little time at their own convenience.

### 3.2.3 Performance

The application, being a web application, will be stored in a web server.

The application's time of execution will depend on:

- The algorithm's efficiency for fetching data from the database.
- The users Internet connection strength.
- The server hardware capabilities.
- The operating system installed on a server.
- Third party library dependencies that need to be installed.
- The number of active user accessing the website.

#### 3.2.3.1 Capacity

- Database writes are limited to 3000 per second, which will be more than enough for our use case.
- Maximum concurrent connections web clients are limited to 150000 per database.
- Maximum number of documents that can be passed to a Commit operation in a transaction is 500
- Maximum number of composite indexes for a database is 200

#### 3.2.3.2 Availability

Include specific and measurable requirements for:

- The web application will be available for use 24/7.
- The web application will work in an optimal manner during the working hours of the day.
- The application can be accessed and used in any geographical area, as long as the user has an active Internet connection.
- By creating separate user sessions, their overall work efficiency and productivity will not decrease by much, while using the application.
- Specific error messages will be provided, in case an action would cause systems fatal error.

#### 3.2.3.3 Latency

The latency of the web application will mainly depend on:

- The Internet connection strength.
- The efficiency of the specific algorithm for fetching the data from the database.
- The size of the database.

Specific modules are expected to load within such time line:

- The login module should load within the first 100 ms
- The dashboard module should load within the first 300ms
- The reports module should load within the 1500 ms
- The setting module should load with the first 300 ms

### 3.2.4 Manageability/Maintainability

#### 3.2.4.1 Monitoring

The application user interface, will be easy and it will not provide cases that would crash the system. Necessary actions for any of error will be taken. The login interface needs a Username (1) and Password (2) as input. These two input data must be valid input data. The user will log in the system, in case the user has entered valid credentials, otherwise an error message of "Invalid Credentials" will be displayed.

The user module will order the registered users (separated clients from staff). Within the module, a search bar is included, in order to fetch a user having some specific credential. Users can be searched by: name (1), email (2) and username (3). Placing other type of data information will result in an empty table. Also within the user module, a failure message is included, for modifying the user data. A failure condition is included whenever the administrator tries to change the password. If the administrator tries to place information which is not compatible with the data in the database, then a message alert will be shown. If the changed password does not match with the confirmed password then no changes are made and an alert message will be shown.

The settings module will hold the user personal information. Within this module the users can change their credentials. If the new data is not the same as the data type specified by the database, then an error message alert will be displayed. If the changed password does not match with the confirmed password then no changes are made and an alert message will be shown.

#### 3.2.4.2 Maintenance

The software will be developed working on Laravel framework. Laravel uses MVC pattern for software developing.The Model–view–controller shortly known as MVC is a software architectural design for implementing user interfaces on computers. Also the database will be built using MySQL.
Advantages of using MVC & MySQL:

- Faster development process
- Ability to provide multiple views

- Support for asynchronous technique
- Modification does not affect the entire model
- MVC model returns the data without formatting
- SEO friendly Development platform
- Data Security
- On-Demand Scalability
- High Performance
- Round-the-Clock Uptime
- Comprehensive Transactional Support
- Complete Workflow Control

#### 3.2.4.3 Operations

Operations required by the user:

- Weekly back up.
- No data sharing is allowed.
- Data security.
- Users will have the authority to change his/her own credential data.
- CRUD functionality for users.
- Admin will have the authority to add or remove new users (from staff).
- Admin will have the authority of checking and evaluating performance.

### 3.2.5 System Interface/Integration

#### 3.2.5.1 Network and Hardware Interfaces

TCP (transfer control protocol) and IP (internet protocol) define as how computers should connect with each other through the internet and how they share information. Common TCP/IP protocols are HTTPS, HTTP and FTP. We use these protocols to access the website without even realizing it. FTP (file transfer protocol) used to download files from a browser.

HTTPS (where the “S” stands for security) responsible for the secure communication between a server and a browser like an account information or personal information.

SSL (secure sockets layer) and TLS (transport layer security) are the protocols associated with trust and security.

We are also using SMTP (google SMTP) for email verification.

Ports allow one device to connect with the other through a different and unique IP address. A device can have more than one port.

Port 25 (SMTP) is usually reserved for email. It is used to transmit data from remote email servers. If that port is blocked no emails can be sent. Some other port numbers are 80(HTTP), 443 (HTTPS), 21 (FTP). Firewalls can often block a port to tighten the security.

#### 3.2.5.2 Systems Interfaces

System to system interface
Email using google SMTP service:
SMTP server (i.e., outgoing mail): smtp.gmail.com
SMTP username: be3albania@gmail.com
SMTP password: email password
SMTP port: 465
SMTP TLS/SSL required: yes
In order to store a copy of outgoing emails in your Gmail or Google Apps Sent folder, log into your Gmail or Google Apps email Settings and:
Click on the Forwarding/IMAP tab and scroll down to the IMAP Access section: IMAP must be enabled in order for emails to be properly copied to your sent folder.

### 3.2.6 Security

#### 3.2.6.1 Protection

Specify the factors that will protect the system from malicious or accidental access, modification, disclosure, destruction, or misuse. For example:

- The encryption of customer data (personal information, credit card information) as well as inventories, company financial information, etc.
- Communication between modules is done via the exchange of named objects with the event and object manager. Modules therefore do not depend on specific other modules being available in memory or having been run before, but only on whether the necessary data objects have been stored in the event or the object manager. This allows us to completely specify a module by defining which objects or containers it expects as input from the event and object manager and which objects are produced as output in event and object manager. Any two objects using and producing the same objects should be completely interchangeable in running macros. The names of input and output objects for each module should be set to a default value in the modules constructor. They can be overwritten at any time, allowing the same module to be run on the same event with different tuning parameters producing different output containers. Using special tags in the comment fields of the module data member declaration allows the programmer to identify the names of input and output containers. This will be used to automatically check the availability of the containers needed by the module, check whether a particular chain of modules is likely to be consistent and provide automated module documentation. As a rule of thumb, modules should exit gracefully and with a meaningful error message if one of necessary input objects is not found in the event or object manager.
- Data integrity we are going to use in the database system by a series of integrity constraints. The integrity used will be:

- Entity integrity concerns the concept of a primary key. Entity integrity is an integrity rule which states that every table must have a primary key and that the column or columns chosen to be the primary key should be unique and not null.
- Referential integrity concerns the concept of a foreign key. The referential integrity rule states that any foreign-key value can only be in one of two states. The usual state of affairs is that the foreign-key value refers to a primary key value of some table in the database. Occasionally, and this will depend on the rules of the data owner, a foreign-key value can be null. In this case, we are explicitly saying that either there is no relationship between the objects represented in the database or that this relationship is unknown.
- Domain integrity specifies that all columns in a relational database must be declared upon a defined domain.
- User-defined integrity refers to a set of rules specified by a user, which do not belong to the entity, domain and referential integrity categories.

#### 3.2.6.2 Authorization and Authentication

Authentication:

- Used by a server when the server needs to know exactly who is accessing their information or site.
- Used by a client when the client needs to know that the server is system it claims to be.
- The user or computer has to prove its identity to the server or client.
- By a server entails the use of a user name and password.
- By a client usually involves the server giving a certificate to the client in which a trusted third party states that the server belongs to the entity (such as a bank) that the client expects it to.
- Identifies and verifies who the person or system is.

Authorization:

- A process by which a server determines if the client has permission to use a resource or access a file.
- Coupled with authentication so that the server has some concept of who the client is that is requesting access.
- The type of authentication required for authorization may vary; passwords may be required in some cases but not in others.

### 3.2.7 Data Management

The database that the application will be using, will contain this kind of possible information:

- Users (id, email, password, full_name, phone, birth_date, user_level, image, address, city, region, postal_code, verification_token, verified, created_at, updated_at)
- Orders ( id, user_id, status_code, note, created_at, updated_at )
- Products (id, code, price, title, description, in_stock, created_at, updated_at)
- Reviews (user_id, product_id, description, image, rating, created_at, updated_at)

- Categories (id, title, description)
- Carts (user_id, product_id, quantity)
- Favorites (user_id, product_id)

### 3.2.8 Standards Compliance

Specify the requirements derived from existing standards, policies, regulations, or laws (e.g., report format, data naming, accounting procedures, audit tracing). For example, this could specify the requirement for software to trace processing activity. Such traces are needed for some applications to meet minimum regulatory or financial standards. An audit trace requirement may, for example, state that all changes to a payroll database must be recorded in a trace file with before and after values.

### 3.2.9 Portability

Portability it is not an issue. The application will be accessed as long as you have internet connection by using either a computer or mobile phone.

### 3.2.10 Other Non-Functional Requirements

Please provide all necessary non-functional requirements, similar to the requirements explained in the lesson slides or in the textbook.

## *3.3 Domain Requirements*

- OSMS can be used both by the clients/costumers and staff.
- It covers different procedures from online ordering on the client site to inventory, products, price and orders management on staff site (admin or storekeeper).
- This business application does not communicate with any third party entity.
- Different system users have different privileges that should be respected and take full responsibility by the account holder.

## *4. User Scenarios/Use Cases*

1. Story name: Successful Login
   a. The user is asked to enter his username
   b. The user is asked to enter his password
   c. The user presses “Sign In” button
   d. If his credentials are matched in the database, he is authorized to be redirected
   e. The user gets redirected to the website (front page)

2. Story name: Failed Login
   a. The user is asked to enter his username
   b. The user is asked to enter his password
   c. The user presses “Sign In” button
   d. The credentials that were entered are not found in the database
   e. The user is displayed an error message, “You credentials were not correct, please try again”
   f. The user tries to enter the credentials again’

3. Story name: First time client register successfully
   a. The customer wants to add a product to wish list/card

b. The costumer is asked to register first
c. The customer fill the registration form
d. The data entered are valid
e. The costumer is displayed a message, “You have been registered successfully”
f. The costumer become a user

4. Story name: First time client register failed
   a. The customer wants to add a product to wish list/card
   b. The costumer is asked to register first
   c. The customer fill the registration form
   d. The data entered are ambiguous(not valid)/ not existent email / one of the required fields left empty
   e. The costumer is displayed an error message, “Your data are not valid or you have left any required field empty ”
   f. The costumer retries to fill the form

5. Story name : Costumer successful order
   a. The customer wants to add a product to card
   b. The costumer is asked to enter his username
   c. The costumer is asked to enter his password
   d. The costumer presses “Sign In” button
   e. If his credentials are matched in the database, he is authorized to be redirected
   f. The costumer gets redirected to the chosen product link
   g. The costumer add the product to card
   h. The costumer check-out the product
   i. The costumer is asked to fill the check-out form
   j. The order is made

6. Story name : Costumer failed order
   a. The customer wants to add a product to card
   b. The costumer is asked to enter his username
   c. The costumer is asked to enter his password
   d. The costumer presses “Sign In” button
   e. His credentials are not matched in the database
   f. The costumer is displayed an error message, “You credentials were not correct, please try again”
   g. The costumer reenter the credentials

7. Story name : Costumer feedback of a product (when he is logged in)
   a. The costumer wants to review a product
   b. The costumer press the number of stars to give to a product
   c. A message is shown to the costumer, “Do you want to give ‘nrOfStars’ stars to this product”
   d. The costumer presses “Yes” button

e. The feedback is given

8. Story name : Costumer feedback of a product (when he is NOT logged in)
   a. The costumer wants to review a product
   b. The costumer is asked to log-in first.
   c. The costumer enters the username and password
   d. The credentials are valid
   e. The costumer logged-in
   f. The costumer press the number of stars to give to a product
   g. A message is shown to the costumer, "Do you want to give 'nrOfStars' stars to this product"
   h. The costumer presses "Yes" button
   i. The feedback is given

9. Story name: Costumer check his orders (when is logged in )
   a. The costumer scroll down to the end of the page
   b. The costumer press "My Account"
   c. The costumer press "Orders"
   d. His orders are shown to him

10. Story name: Costumer check his order status (when is logged in )
    a. The costumer scroll down to the end of the page
    b. The costumer press "My Account"
    c. The costumer press "Orders"
    d. His orders are shown to him
    e. The costumer goes to the specific order
    f. The costumer press "Status"
    g. The costumer can see the status of his order

11. Story name: Costumer change the credentials successfully (when is logged in )
    a. The costumer scroll down to the end of the page
    b. The costumer press "My Account"
    c. The costumer press "Personal Info"
    d. The costumer press "Edit data"
    e. The system asks for reentering password
    f. The costumer enters the password
    g. Password is valid
    h. The costumer change the data
    i. The costumer press "Update" button
    j. The data are updated

12. Story name: Costumer change the credentials failed (when is logged in )
    a. The costumer scroll down to the end of the page
    b. The costumer press "My Account"

c. The costumer press “Personal Info”
d. The costumer press “Edit data”
e. The system asks for reentering password
f. The costumer enters the password
g. Password is NOT valid
h. A message is displayed to the costumer, “The password you entered is incorrect, please reenter the password!”
i. The costumer reenter the password

13. Story name: Admin delete a user (when is logged in )
    a. Admin goes to users module
    b. Admin press “Users”
    c. The users are displayed to the admin
    d. Admin press “Delete” button to a specific user
    e. If the user level is admin an error is shown, “ERROR You are not allowed to delete a admin user!”
    f. If the user level is everything else despite admin the user will be deleted

14. Story name: Admin create a user (when is logged in )
    a. Admin goes to users module
    b. Admin press “Users”
    c. The users are displayed to the admin
    d. Admin press “Create New” button
    e. A form is displayed to admin for filling the new user credentials and user level
    f. Admin fill the form and press create
    g. The system check if the user data are valid or existent before
    h. If the user exist an error message is displayed, “This user exists already!”
    i. Else the user is created

15. Story name: Admin check reports (when is logged in )
    a. Admin goes to reports module
    b. Admin press “Reports”
    c. The filtering bar is displayed to the admin
    d. After putting the filtering conditions admin press the button “Filter”
    e. The report is generated based on the admin filters
    f. Admin can view report / download report/ print report

16. Story name: Admin update product description (when is logged in )
    a. Admin goes to products module
    b. Admin press “Products”
    c. The table of products data is displayed
    d. At the end of the row admin press edit
    e. The product information form is displayed with the actual data

f. Admin changes the description
g. Admin press “SAVE” button
h. The product description is updated

17. Story name: Admin update product image (when is logged in )
    a. Admin goes to products module
    b. Admin press “Products”
    c. The table of products data is displayed
    d. At the end of the row admin press edit
    e. The product information form is displayed with the actual data
    f. Admin remove photo
    g. Admin re upload new photo
    h. The system check the format of the file uploaded
    i. If the file is not jpg, jpeg, png, or pdf the system will print an error message, “The format of the file is not supported, try another format!”
    j. Else the file is uploaded
    k. Admin press “SAVE” button
    l. The product image is updated

18. Story name: Manager update order status (when is logged in )
    a. The manager goes to orders
    b. The manager choose new orders
    c. The manager makes the orders ready
    d. The manager sends orders in the post office
    e. After the orders are sent in post office the manager press change status
    f. The manager change the status from received to send
    g. The status is updated

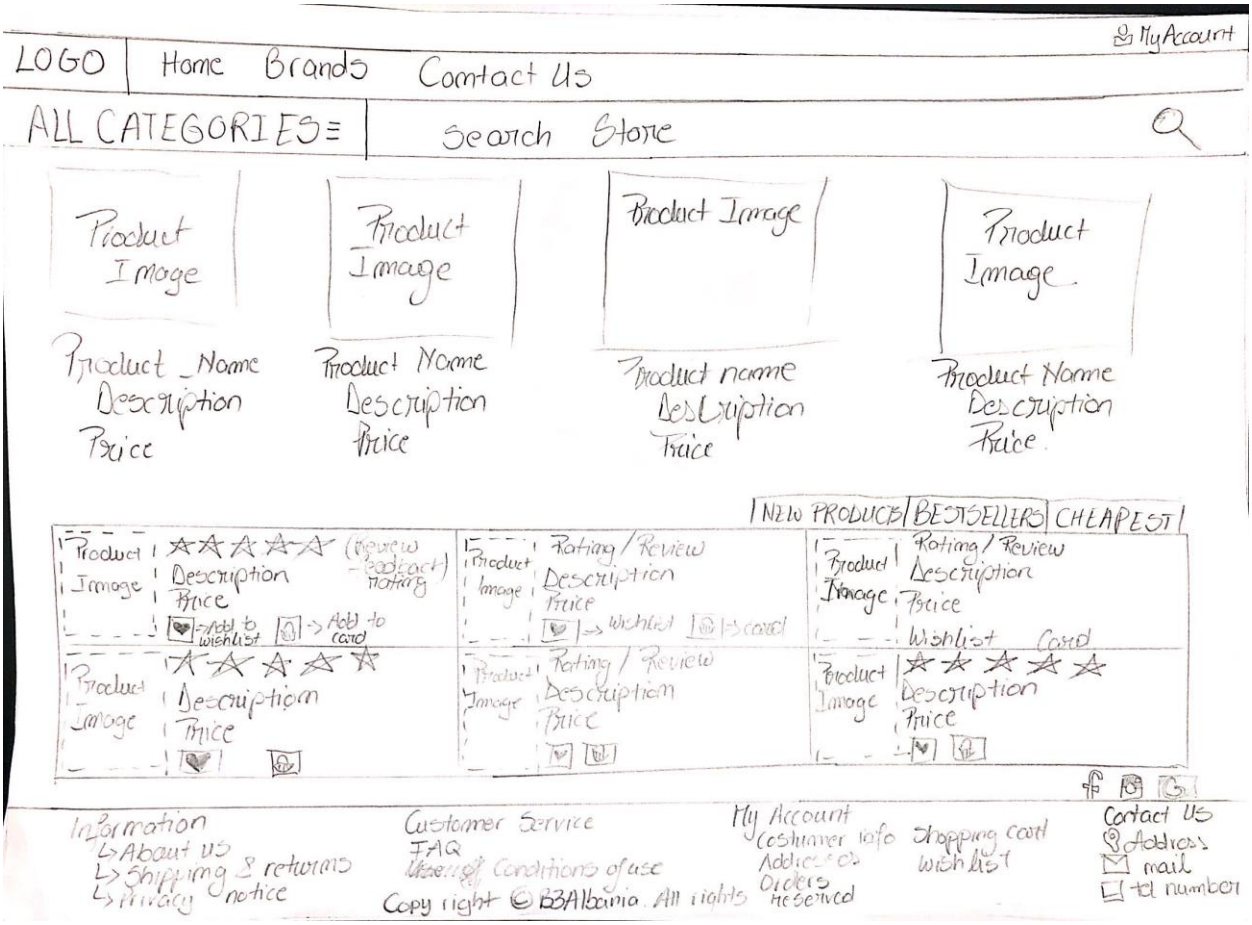
APPENDIX

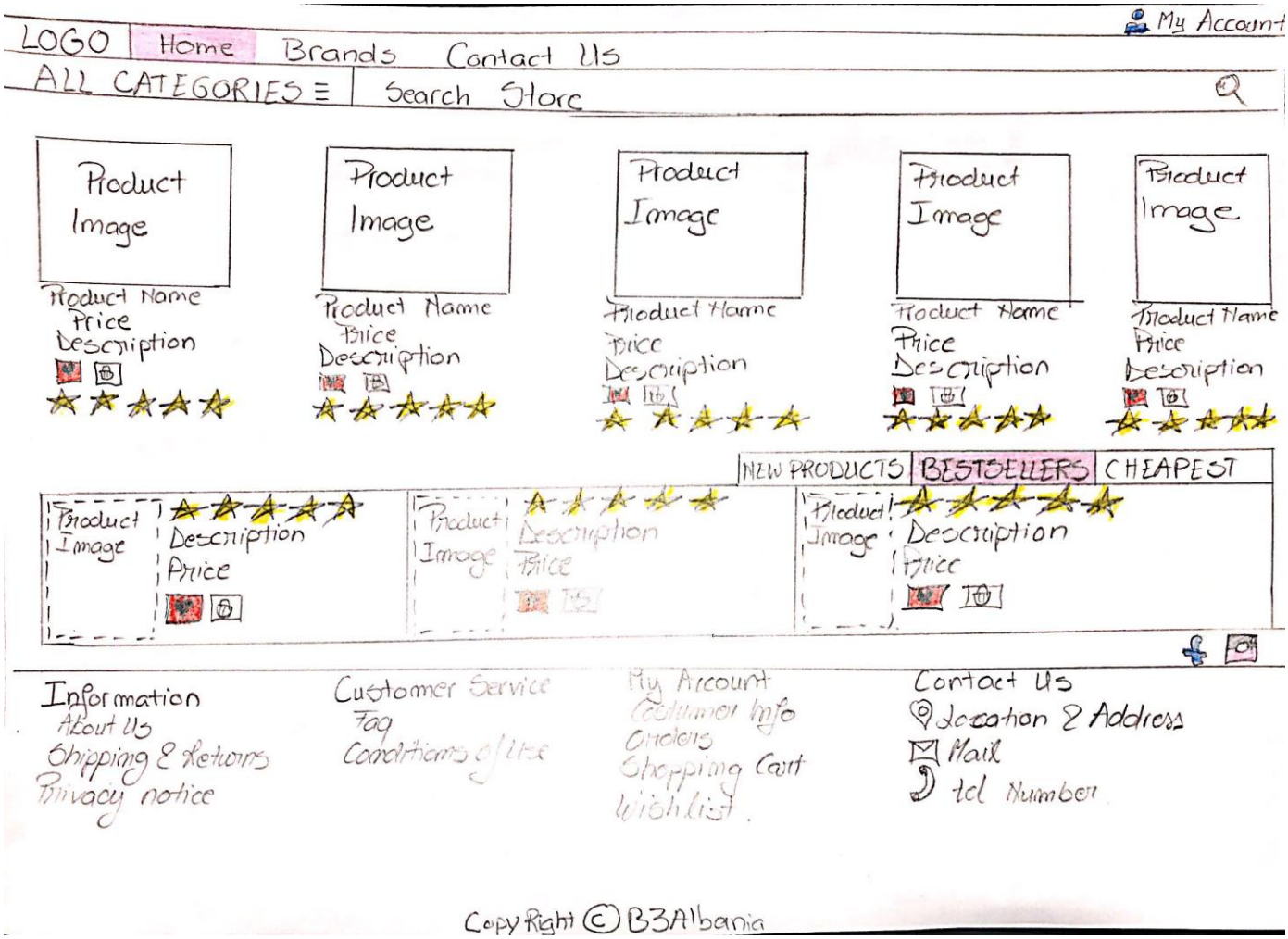
The appendixes are not always considered part of the actual Requirements Specification and are not always necessary. They may include

- Sample input/output formats, descriptions of cost analysis studies, or results of user surveys;
- Supporting or background information that can help the readers of the Requirements Specification;
- A description of the problems to be solved by the system;
- Special packaging instructions for the code and the media to meet security, export, initial loading, or other requirements.

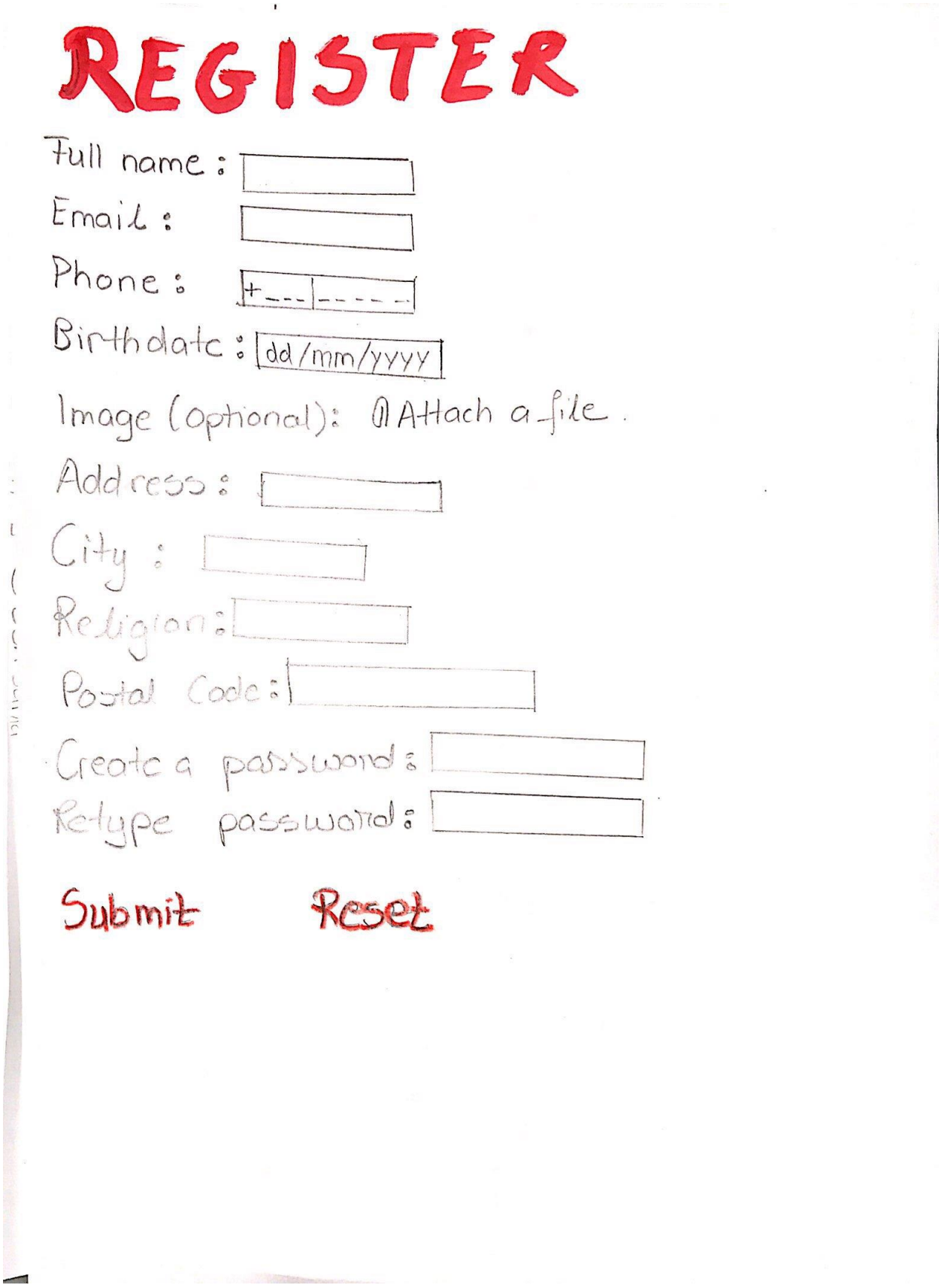
When appendixes are included, the Requirements Specification should explicitly state whether or not the appendixes are to be considered part of the requirements.

## Appendix A. Schetches




My Account
LOGO
Home
Brands
Contact Us
ALL CATEGORIES
Search Store
Product Image
Product Name
Price
Description
NEW PRODUCTS
BESTSELLERS
CHEAPEST
Information
About Us
Shipping & Returns
Privacy notice
Customer Service
Faq
Conditions of Use
My Account
Customer Info
Orders
Shopping Cart
Wishlist
Contact Us
location & Address
Mail
tel Number
Copy Right © B3Albania


My Account
Username:
Password:
LOG IN
REGISTER


REGISTER
Full name :
Email :
Phone :
+ - - -
- - - - -
Birthdate :
dd/mm/yyyy
Image (optional):
Attach a file.
Address :
City :
Religion :
Postal Code :
Create a password :
Retype password :
Submit
Reset

My Account
User Picture
Welcome User_Name!
Personal Info's
MY ORDERS
SHOPPING CART
WISHLIST
Edit

## Appendix B. References

List all the documents and other materials referenced in this document.

## Appendix C. Requirements Traceability Matrix

The following trace matrix examples show one possible use of naming standards for deliverables (FunctionalArea-DocType-NN). The number has no other meaning than to keep the documents unique. For example, the Bargaining Unit Assignment Process Flow would be BUA-PF-01.

For example (1):

| Business Requirement | Area | Deliverables | Status |
|---|---|---|---|
| BR_LR_01<br>The system should validate the relationship between Bargaining Unit/Location and Job Class.---Comments: Business Process = "Assigning a Bargaining Unit to an Appointment" (Priority 1) | BUA | BUA-CD-01<br>Assign BU Conceptual Design | Accepted |
| | | BUA-PF-01<br>Derive Bargaining Unit-Process Flow Diagram | Accepted |
| | | BUA-PF-01<br>Derive Bargaining Unit-Process Flow Diagram | Accepted |
| BR_LR_09<br>The system should provide the capability for the Labor Relations Office to maintain the job class/union relationship.---Comments: Business Process = "Maintenance" (Priority 1) | BUA | BUA-CD-01<br>Assign BU Conceptual Design | Accepted |
| | | BUA-PF-02<br>BU Assignment Rules Maint Process Flow Diagram | ReadyForReview |

For example (2):

| BizReqID | Pri | Major Area | DevTstItems DelivID | Deliv Name | Status |
|---|---|---|---|---|---|
| BR_LR_01 | 1 | BUA | BUA-CD-01 | Assign BU Conceptual Design | Accepted |
| BR_LR_01 | 1 | BUA | BUA-DS-02 | Bargaining Unit Assignment DB Modification Description | Accepted |
| BR_LR_01 | 1 | BUA | BUA-PF-01 | Derive Bargaining Unit-Process Flow Diagram | Accepted |
| BR_LR_01 | 1 | BUA | BUA-UCD-01 | BU Assign LR UseCase Diagram | ReadyForReview |
| BR_LR_01 | 1 | BUA | BUA-UCT-001 | BU Assignment by PC UseCase - Add Appointment and Derive UBU | Reviewed |
| BR_LR_01 | 1 | BUA | BUA-UCT-002 | BU Assignment by PC UseCase - Add Appointment (UBU Not Found) | Reviewed |
| BR_LR_01 | 1 | BUA | BUA-UCT-006 | BU Assignment by PC UseCase - Modify Appointment (Removed UBU) | Reviewed |
| BR_LR_09 | 1 | BUA | BUA-CD-01 | Assign BU Conceptual Design | Accepted |
| BR_LR_09 | 1 | BUA | BUA-DS-02 | Bargaining Unit Assignment DB Modification Description | Accepted |
| BR_LR_09 | 1 | BUA | BUA-PF-02 | BU Assignment Rules Maint Process Flow Diagram | Accepted |
| BR_LR_09 | 1 | BUA | BUA-UCD-03 | BU Assign Rules Maint UseCase Diagram | Reviewed |

| BizReqID | Pri | Major Area | DevTstItems DelivID | Deliv Name | Status |
|---|---|---|---|---|---|
| BR_LR_09 | 1 | BUA | BUA-UCT-045 | BU Assignment Rules Maint: Successfully Add New Assignment Rule | Reviewed |
| BR_LR_09 | 1 | BUA | BUA-UCT-051 | BU Assignment Rules MaintUseCase: Modify Rule | Reviewed |
| BR_LR_09 | 1 | BUA | BUA-UCT-053 | BU Assignment Rules MaintUseCase - Review Assignment Rules | Reviewed |
| BR_LR_09 | 1 | BUA | BUA-UCT-057 | BU Assignment Rules MaintUseCase: Inactivate Last Rule for a BU | Reviewed |
| BR_LR_09 | 1 | BUA | BUA-UI-02 | BU AssignRules Maint UI Mockups | ReadyForReview |
| BR_LR_09 | 1 | BUA | BUA-TC-021 | BU Assignment Rules Maint TestCase: Add New Rule (Associated Job Class Does Not Exist) - Success | ReadyForReview |
| BR_LR_09 | 1 | BUA | BUA-TC-027 | BU Assignment Rules Maint TestCase: Modify Rule - Success | ReadyForReview |
| BR_LR_09 | 1 | BUA | BUA-TC-035 | BU Assignment Rules Maint TestCase: Add New Rule (Associated Job Class Does Not Exist) - Error Condition | ReadyForReview |
| BR_LR_09 | 1 | BUA | BUA-TC-049 | BU Assignment Rules Maint TestCase: Modify Rule - Error Condition | ReadyForReview |

For example (3):

| BizReqID | CD01 | CD02 | CD03 | CD04 | UI01 | UI02 | UCT01 | UCT02 | UCT03 | TC01 | TC02 | TC03 | TC04 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| BR_LR_01 | | | X | | X | | X | | | X | | X | |
| BR_LR_09 | X | | | X | | X | | | X | | X | | X |
| BR_LR_10 | X | | | X | | | | | X | | X | | |
| BR_LR_11 | | X | | | | | | | | | | | |

# Appendix D. Organizing the Requirements

This section is for information only as an aid in preparing the requirements document.

Detailed requirements tend to be extensive. Give careful consideration to your organization scheme. Some examples of organization schemes are described below:

**By System Mode**
Some systems behave quite differently depending on the mode of operation. For example, a control system may have different sets of functions depending on its mode: training, normal, or emergency.

**By User Class**
Some systems provide different sets of functions to different classes of users. For example, an elevator control system presents different capabilities to passengers, maintenance workers, and fire fighters.

**By Objects**
Objects are real-world entities that have a counterpart within the system. For example, in a patient monitoring system, objects include patients, sensors, nurses, rooms, physicians, medicines, etc. Associated with each object is a set of attributes (of that object) and functions (performed by that object). These functions are also called services, methods, or processes. Note that sets of objects may share attributes and services. These are grouped together as classes.

**By Feature**
A feature is an externally desired service by the system that may require a sequence of inputs to affect the desired result. For example, in a telephone system, features include local call, call forwarding, and conference call. Each feature is generally described in a sequence of stimulus-response pairs, and may include validity checks on inputs, exact sequencing of operations, responses to abnormal situations, including error handling and recovery, effects of parameters, relationships of inputs to outputs, including input/output sequences and formulas for input to output.

**By Stimulus**
Some systems can be best organized by describing their functions in terms of stimuli. For example, the functions of an automatic aircraft landing system may be organized into sections for loss of power, wind shear, sudden change in roll, vertical velocity excessive, etc.

**By Response**
Some systems can be best organized by describing all the functions in support of the generation of a response. For example, the functions of a personnel system may be organized into sections corresponding to all functions associated with generating paychecks, all functions associated with generating a current list of employees, etc.

**By Functional Hierarchy**
When none of the above organizational schemes prove helpful, the overall functionality can be organized into a hierarchy of functions organized by common inputs, common outputs, or common internal data access. Data flow diagrams and data dictionaries can be used to show the relationships between and among the functions and data.

**Additional Comments**
Whenever a new Requirements Specification is contemplated, more than one of the organizational techniques given above may be appropriate. In such cases, organize the specific requirements for multiple hierarchies tailored to the specific needs of the system under specification.

There are many notations, methods, and automated support tools available to aid in the documentation of requirements. For the most part, their usefulness is a function of organization. For example, when organizing by mode, finite state machines or state charts may prove helpful; when organizing by object, object-oriented analysis may prove helpful; when organizing by feature, stimulus-response sequences may prove helpful; and when organizing by functional hierarchy, data flow diagrams and data dictionaries may prove helpful.