

# **ElectroSpaceX Requirements Specification**

**Version 2.0**

**April 22, 2022**

# Table of Contents

<b>1. EXECUTIVE SUMMARY .....</b>	<b>3</b>
1.1 PROJECT OVERVIEW .....	3
1.2 PURPOSE AND SCOPE OF THIS SPECIFICATION .....	3
<b>2. PRODUCT/SERVICE DESCRIPTION .....</b>	<b>3</b>
2.1 PRODUCT CONTEXT .....	3
2.2 USER CHARACTERISTICS .....	4
2.3 ASSUMPTIONS .....	4
2.4 CONSTRAINTS .....	4
2.5 DEPENDENCIES .....	4
<b>3. REQUIREMENTS .....</b>	<b>4</b>
3.1 FUNCTIONAL REQUIREMENTS .....	5
3.2 NON-FUNCTIONAL REQUIREMENTS .....	7
3.2.1 Product Requirements .....	7
3.2.2 Usability .....	8
3.2.3 Performance .....	<i>Error! Bookmark not defined.</i>
3.2.4 Manageability/Maintainability .....	13
3.2.5 System Interface/Integration .....	11
3.2.6 Security.....	<i>Error! Bookmark not defined.</i>
3.2.7 Data Management .....	13
3.2.8 Standards Compliance .....	14
3.2.9 Portability .....	14
3.2.10 Other Non-Functional Requirements .....	14
3.3 DOMAIN REQUIREMENTS .....	14
<b>4. USER SCENARIOS/USE CASES .....</b>	<b>14</b>
<b>APPENDIX .....</b>	<b>15</b>
APPENDIX A. DEFINITIONS, ACRONYMS, AND ABBREVIATIONS .....	15
APPENDIX B. REFERENCES .....	15
APPENDIX C. REQUIREMENTS TRACEABILITY MATRIX .....	15
APPENDIX D. ORGANIZING THE REQUIREMENTS .....	17

# 1. Executive Summary

## 1.1 Project Overview

Describe this project or product and its intended audience, or provide a link or reference to the project charter.

The project consists of a web application, which resembles an e-commerce and management system of an electronic business. The web application should be user-friendly, convenient and will contain at least three users, such as the client, economist and administrator.

The services the business provides are mainly focused on the sale of electronic means and their successful delivery. In addition to the services mentioned above, the web application will enable users to give their feedback upon different products. Each user will register and login using its credentials, and then will be able to perform different orders, view its historic of purchases and provide feedbacks. On the other hand, the economist will be responsible of the financial states and the company's profits. The administrator can have much more access than a simple user or economist. He can keep track of the purchases made, be aware of the stock of the products, cancel orders and also view specific client data. He is responsible of taking care of different problems or dissatisfaction users might have.

The technologies we are going to use, include PHP programming on the back-end and HTML, CSS, JavaScript, Bootstrap on the front-end, and MySql as a database.

## 1.2 Purpose and Scope of this Specification

Describe the purpose of this specification and its intended audience. Include a description of what is within the scope what is outside of the scope of these specifications.

The purpose of this project is to provide a proper management system of an online electronic store, which currently operates in Albania.

In scope of these requirements:

- Approaching the system as one of the three users, each of them having distinct responsibilities.
- Being in track of the services provided, as a result the business will be aware of the client feedback and manage decisions related to the improvement of its products
- Managing the supply of the electronic products and its trading. Out of scope of these requirements:
- Delivery of the products to its final destination requires a separate management system.

# 2. Product/Service Description

In this section, describe the general factors that affect the product and its requirements. This section should contain background information, not state specific requirements (provide the reasons why certain specific requirements are later specified).

## 2.1 Product Context

How does this product relate to other products? Is it independent and self-contained? Does it interface with a variety of related systems? Describe these relationships or use a diagram to show the major components of the larger system, interconnections, and external interfaces.

## 2.2 User Characteristics

Create general customer profiles for each type of user who will be using the product. Profiles should include:

- Student/faculty/staff/other

- experience
- technical expertise
- other general characteristics that may influence the product

### **2.3 Assumptions**

List any assumptions that affect the requirements, for example, equipment availability, user expertise, etc. For example, a specific operating system is assumed to be available; if the operating system is not available, the Requirements Specification would then have to change accordingly.

### **2.4 Constraints**

Describe any items that will constrain the design options, including

- parallel operation with an old system
- audit functions (audit trail, log files, etc.)
- access, management and security
- criticality of the application
- system resource constraints (e.g., limits on disk space or other hardware limitations)
- other design constraints (e.g., design or other standards, such as programming language or framework)

### **2.5 Dependencies**

List dependencies that affect the requirements. Examples:

- This new product will require a daily download of data from X, □ Module X needs to be completed before this module can be built.

## **3. Requirements**

- Describe all system requirements in enough detail for designers to design a system satisfying the requirements and testers to verify that the system satisfies requirements.
- Organize these requirements in a way that works best for your project. See Appendix D, Organizing the Requirements for different ways to organize these requirements.
- Describe every input into the system, every output from the system, and every function performed by the system in response to an input or in support of an output. (Specify what functions are to be performed on what data to produce what results at what location for whom.)
- Each requirement should be numbered (or uniquely identifiable) and prioritized. See the sample requirements in Functional Requirements, and , as well as these example priority definitions:

#### **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.

- A good requirement is:

## ***ElectroSpaceX Requirements Specification***

- Correct
- Unambiguous (all statements have exactly one interpretation)
- Complete (where TBDs are absolutely necessary, document why the information is unknown, who is responsible for resolution, and the deadline)
- Consistent
- Ranked for importance and/or stability
- Verifiable (avoid soft descriptions like “works well”, “is user friendly”; use concrete terms and specify measurable quantities)
- Modifiable (evolve the Requirements Specification only via a formal change process, preserving a complete audit trail of changes)
- Does not specify any particular design
- Traceable (cross-reference with source documents and spawned documents).

### **3.1 Functional Requirements**

In the example below, the requirement numbering has a scheme - BR\_LR\_0## (BR for Business Requirement, LR for Labor Relations). For small projects simply BR-## would suffice. Keep in mind that if no prefix is used, the traceability matrix may be difficult to create (e.g., no differentiation between '02' as a business requirement vs. a test case)

The following table is an example format for requirements. Choose whatever format works best for your project.

For Example:

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
BR_01	The system is designed as a web application with three users	This will be the main platform for users	1	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_02	The system's service will be based on online sales and purchases, as well as system management.		2	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_03	All privileges will be granted to the system administrator. Will get realtime access to all system and database options for clients and economists.	The admin account has full access to the system's features.	1	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_04	The application must have a specific UI (user interface) for client.		1	4/14/22	Gejsi Dervishi, Kevin Mamaj

Req#	Requirement	Comments	Priority	Date Rvwd	SME Reviewed / Approved
------	-------------	----------	----------	-----------	-------------------------

### ***ElectroSpaceX Requirements Specification***

BR_05	Every user should be able to log in and out and every client has the opportunity to register into the application	Economist and administrator are already in the database	1	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_06	Every user should have access to their own section of the system.		1	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_07	The system should have profiles for each user.		1	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_08	The administrator must have the most privileges of all users.		1	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_09	Client profiles should be private	Data Protection	1	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_10	The administrator must have access to CRUD operations (Create, Read, Update, Delete) categories, products, orders, users, employees.		1	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_11	The data is presented to the administrator in a table format.	This is done because tables and visuals provide a clearer picture of all the clients who have registered.	2	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_12	Administrator should be able to read all orders and decide if the order is ready to be delivered or not.		2	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_13	A supply balance report should be able to be filed by administrator, and he can choose the amount of products that are needed.		2	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_14	The admin should not be able to see any of the user's personal information.		1	4/14/22	Gejsi Dervishi, Kevin Mamaj
<b>Req#</b>	<b>Requirement</b>	<b>Comments</b>	<b>Priority</b>	<b>Date Rvwd</b>	<b>SME Reviewed / Approved</b>

### ***ElectroSpaceX Requirements Specification***

BR_15	The economist's report/balance should be available to the administrator at the end of each month.		1	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_16	The economist should do a monthly report on the profit or loss of the company		2	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_17	The system should have a calculation of total monthly, weekly and daily incomes and the economist has the right to `study them.		2	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_18	At the moment when the stock is from 0-3 products in the administrator page will be shown a warning in order to send an email to the supplier.	The email will have the required details for the product by the supplier.	3	4/14/22	Gejsi Dervishi, Kevin Mamaj
BR_19	When a product needs to be shipped, the economist will print a receipt with buyer details and give it to the transporter		3	4/14/22	Gejsi Dervishi, Kevin Mamaj

## **3.2 Non-Functional Requirements**

### **3.2.1 Product Requirements**

Our software will be based as a web application, consisting of three user interfaces.

The client interface, the economist interface and the administrator interface will appear corresponding to the login credentials.

The header of the website contains the logo of the business, a search bar and the menu with various options, such as products, news, services, about us and the log-in button. While the footer contains social media contact details, address details and privacy policy information.

#### ☐ Client Interface:

- When the client presses the log-in button then he will be directed to log-in window, with an extra option to register, if not yet.
- The client interface allows the user to make **real-time purchases**, by adding products to the cart or deleting existing ones.
- The client can view its **profile details** and also edit its information. The information includes the name, surname, the phone number, the address, the country name and the postal number.
- The client can add products on the wish list, and also delete them.
- View the **purchase historic** and also the products on the **wish list**.
- The client can search a product or view recommended items.
- The user can post feedback as a comment or represented as star review.

### ***ElectroSpaceX Requirements Specification***

-Read flash news about technology and products on the website. -**Log-out** option included, in order to log out of the website.

#### ☐ Admin Interface:

- The admin logs in the website with its credentials, and owns various responsibilities. - Performing **CRUD operations** related to creating, reading, updating and deleting products, categories, economists and other workers.
- List all the users.
- Adding product quantity.
- Set sales for specific products on the website.
- Checking **product availability**.
- Being able **to view orders** and invoices.
- Requesting supplies** which will be visible on the client interface.
- View reports generated by the economist.
- View the feedback posted by the client upon different products. -Log out of the site.

#### ☐ Economist Interface:

- Logging in as an economist.
- Viewing **client purchase invoices**.
- Viewing total expenses.
- Calculating the **daily & monthly profit**.
- Generate reports** related to the financial state of the business throughout the month.
- View supply needs and its required budget.
- Add workers and define their wages.
- Log-out option, to leave the website.

### **3.2.1.1 Usability Requirements**

Include any specific usability requirements, for example,

#### Learnability Requirements

- The application is user-friendly, simple and easy to understand.
- An extensive document will be provided in order to instruct common users about the usage of the system.
- This step by step instruction will be separated into modules, related to the interfaces the system owns.
- The system is easy to remember for common users.

#### Flexibility Requirements

- The software is designed to minimize as much as possible user errors. ☐ It can be updated easily in order to meet new requirements.

#### Accessibility Requirements

- The software can be conveniently accessed in various times and environments, since it requires only a device and Internet connection to be used.



### **3.2.1.2 Efficiency Requirements**

The software will provide a fast and reliable way of managing an electronic shop operating in online sales.

#### Performance Requirements

The software is a web application, as a result it will be stored in a web server.

- The performance depends on the speed and the quality of the Internet connection.
- Its efficiency is highly related to the server hardware capability.
- Number of users accessing the website effects the performance.

#### Space Requirements

Since the business serves as an e-commerce website, there will be a great number of users predicted to access the website and perform purchases or different tasks. It is thought that several hundred clients can interface with the application. As for the staff it has a limited number of users and does not have a major effect in space.

### **3.2.1.3 Dependability Requirements**

#### Availability

- The web application will be available for use all the time.
- The web application can be accessed in any geographical area, as long as the user has a stable Internet connection.
- Error message warnings will be provided in order to prevent system fatal error.
- Different user categories can access specific parts of the web application.
- The system should behave as expected, with very few errors.
- Downtime can affect the user experience, as a result an immediate maintenance should be provided. It should be avoided as much as possible during certain times when the application is highly frequented.
- Scheduled maintenance can be performed in times when the store itself is closed, or at times when the website bears as little traffic as possible.

#### Latency

Latency is the delay between a user's action and the response of the web application, referred as the round trip time it takes a packet of data to travel through the network.

- Connection based latency is not covered by the software, since it is related to the speed and quality of the Internet connection.
- The latency of the web application depends mainly on the strength of the Internet connection, the size of the database etc.
- The linking between interfaces should be optimal, in order to avoid latency.

### **3.2.1.4 Security Requirements**

Security is a property that defines the ability of the system to protect itself from attacks. When it comes to security, the software should provide:

- Confidentiality: Access should be granted only to authorized users.
- Integrity: Only the authorized user can modify the data depending on its role. □ Availability: The data in the system is accessible all the time, online.

#### **3.2.1.4.1 Protection**

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

- Each user is required to log in using its credentials.
- User input filtering is provided, in case the user enters incorrect data.
- The encryption of customer personal data, as well as inventories, company financial information etc.
- SQL filtering, in case of SQL injections.
- Refreshing the session, in order to prevent Session ID hijacking attempts.
- Data integrity is represented by a series of integrity constraints used in the database system. These constraints include entity integrity, which states that the table must have a primary key, defined to be unique. Referential integrity which states the concept of a foreign key. The foreign key of one table refers to the primary key of another. Domain Integrity restricts the attributes or values of a column upon a defined domain. User-defined integrity which are specified according to the needs of specific users.

#### **3.2.1.4.2 Authorization and Authentication Authorization:**

- The process of giving permission to a user to access their information or website.
- Users of a specific role should be able to view a particular part of the web application.
- Authorization works through settings that are implemented and maintained by the organization.
- It is combined with authentication to give a brief information to the server of the client that is requesting access.

Authentication:

- The process of determining the exact information of a user, by comparing the user's credential with the ones in the database of authorized user or in the data authentication server.
- The user has to prove its identity to the server.
- Identifies who the person or the system that intends to access the software is.
- The server can authenticate users by the input of username and password.

### **3.2.2 Organizational Requirements**

Requirements that are a consequence of organizational policies and procedures.

#### **3.2.2.1 Environmental Requirements**

- Users should have access to Internet connection, in order to use the web application and perform various operations.
- Users should be able to access the system by entering their credentials and perform tasks related to their role. If the user is a client, he can purchase items, put products on the wish list, view their purchases etc.
- The users induce changes into the database, by performing operations, such as purchases, modifications of personal information etc. For example the administrator can add/delete products, categories and users. Thus the database is modified.

### **3.2.2.2 Operational Requirements**

- Users should firstly access the web application by entering the correct credentials. As a result a user can approach the website as a client, an economist or an administrator, depending on their role.
- Users that are authorized by the system can proceed with other operations.
- The administrator is informed of the purchases, the clients that access the website, the stock of the products and the company's finances and takes actions according to these reports.

### **3.2.2.3 Development Requirements**

- Communicating with the business representatives, in order to have a preview of their requirements and services the business will provide. This phase consists in gathering information and analysis of the project.
- Making a design of how the product will resemble on its final state.
- Starting the implementation by creating the ER diagrams and then the database of the system.
- Building the web application step by step using front-end and back-end technologies.
- Creating each interface that has a different target group of users and linking them with each other.
- Testing the product in a group of users and checking for minor errors, which need improvements. Final product is then launched.

### **3.2.3 External Requirements**

Requirements that arise from factors which are external to the system and its development process.

#### **3.2.3.1 Regulatory Requirements**

##### **3.2.3.1.1 Network and Hardware Interfaces**

- Persistent TCP connection is required to enable our web application to exchange messages over the network.
- Operating system is not as important, as long as it supports web browsers.
- A server should be up all the time to host the system, except on cases of event maintenance.

##### **3.2.3.1.2 Systems Interfaces**

#### **3.2.3.2 Ethical Requirements**

The web application cannot make decisions on its own in regard to:

- Add or remove a user from the system.
- Modify specific categories of the database.
- Share personal information of the users.
- Make purchases of the products without the approval of the client user.

The software and its services are not in use for anyone under the age of 18. The website does not collect information from anyone under the age of 18, or allow anyone under that age to use the services provided.

### **3.2.3.3 Legislative Requirements**

#### **Data Protection**

Enshrined in Article 35 of the Constitution of the Republic of Albania (only available in Albanian [here](#)) ('the Constitution'), the protection of personal data constitutes a fundamental right.

User personal information should be confidential and private. Data integrity is quite important, and the safety of the users has to be aimed. The system should be safe and as a result the user experience will be pleasant.

ElectroSpaceX website collects immediately the customers' personal data when it is required to sign up for buying online. It protects all the personal data from its customers and only use them for contractual and informative purposes. Using the website, the customer has to give his confirmation for registering himself as a customer, for using his personal data for further procedures and to guarantee that the given information is correct, true and updated.

### **3.2.4 Manageability/Maintainability**

#### **3.2.4.1 Monitoring**

- The application user interface will be easy to comprehend and it will not be prone to system crashes.
- The log-in interface needs a username and password as input, if not then the client user needs to register and have an active account.
- The user will enter the system in case he has typed the correct credentials, if not an error message will appear. Filtering the data is required for this step.

#### **3.2.4.2 Maintenance**

- The software will be developed using PHP scripting language as a back-end technology. As front-end technologies HTML, CSS and JavaScript will be used. The database will be built using MySQL.
- PHP is quite flexible, and enables the developer to make changes even after starting a project. It is compatible with the majority of operating systems, as a result it can easily run on different platforms.
- The web application will be created in a way that new features can be added in the future.

#### **3.2.4.3 Operations**

Some operations that can be taken by the users:

- Users can log-in using their credentials.
- Client user can purchase a product.
- Client user can update their profile.
- Client user can post a feedback about a specific product.
- CRUD functionality for admin.
- Admin can add and remove other users.
- Admin can send a request when a product is low in stock. □ Economist can check the financial state of the company.

### **3.2.5 Data Management**

The database will contain some of these tables and attributes:

- User(id, username, name, surname, email, phone, password, created\_at, active)
- Role(role\_id, role\_desc, user\_id)
- Product(product\_id, product\_name, product\_desc, category\_id, category\_photo, product\_status, product\_price, product\_discount, quantity, brand\_id, date\_created)
- Cart(id, product\_id, brand\_id, user\_id, quantity, adress\_id)

### **3.2.6 Standards Compliance**

The web application is developed in such a way that it will respect the legislative laws related to the user data safety and guarantee a reliable service on the e-commerce field of electric means.

### **3.2.7 Portability**

Portability does not affect the maintenance of our software. Since it is a web application, it can be accessed anywhere, as long as the user has a stable Internet connection and an electronic device, such as a smartphone, computer or laptop.

### **3.2.8 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**

- The ElectroSpaceX is a web application which covers three categories of users, such as economists, clients and admin.
- It provides different services, from the online trading of products to the management of an electronic store.
- Providing a menu of products, which can be modified by the admin.
- The system data is safe, considering the fact that the users need to be authenticated and authorized to enter their profile.

## **4. User Scenarios/Use Cases**

### **4.1 User Scenarios**

1. Scenario title: Successful login
  - a. User opens the ElectroSpaceX page and in the top right there is a button that says Hyr
  - b. He is redirected to the login page where he should log in with his credentials, username and password.
  - c. The username and password are checked in the dedicated database
  - d. The credentials are correct so the user is authorized to be redirected to the first page/ home page of the system
2. Scenario title: Login failed
  - a. User opens the ElectroSpaceX page and in the top right there is a button that says Hyr

### ***ElectroSpaceX Requirements Specification***

- b. He is redirected to the login page where he should log in with his credentials, username and password.
  - c. The username and password are checked in the dedicated database
  - d. The credentials are incorrect so the login is unsuccessful
  - e. There is a message that shows the user that his credentials are not correct so he should try again
3. Scenario title: User has forgotten his password
- a. User opens the ElectroSpaceX page and in the top right there is a button that says Hyr
  - b. He is redirected to the login page where he should log in with his credentials, username and password.
  - c. The username and password are checked in the dedicated database
  - d. The credentials are incorrect so the login is unsuccessful
  - e. There is a message that shows the user that his credentials are incorrect : “Invalid username or password”
  - f. At the bottom there is a “Forgotten password?” button which redirects the user into the reset password page
  - g. He is asked to enter his email and the safe question answer. At this moment an automatic email has been send to his email with a code which will be put in the window that now is open.
  - h. At this moment he can choose another password for his profile.
4. Scenario title: First time client registers successfully
- a. A guest wants to do a purchase or to add a product to the Wishlist
  - b. In order to do this, he needs to have an account first
  - c. There is a button “Regjistrohu tani” at the login page which means register.
  - d. He is asked to enter the registration details which are a username, email, password, reentering the password and a safe question
  - e. The data are valid.
  - f. A message is shown that tells the user that he is registered successfully.
5. Scenario title: First time client registers unsuccessfully
- a. A guest wants to do a purchase or to add a product to the Wishlist
  - b. In order to do this, he needs to have an account first
  - c. There is a button “Regjistrohu tani” at the login page which means register.
  - d. He is asked to enter the registration details which are a username, email, password, reentering the password and a safe question
  - e. The data entered are ambiguous(not valid)/ not existent email / one of the required fields left empty
  - f. The costumer is displayed an error message, “Your data are not valid or you have left any required field empty ” so he reenters its data.
6. Scenario title: Client edits his profile
- a. At the moment the client registers from the first time at his account he will be redirected into his profile
  - b. He presses the button that says “edito”
  - c. He edits name , surname, address, postal code and his telephone number
  - d. He updates the data and presses the save button
  - e. The data are saved at the database
7. Scenario title: View flash news

### ***ElectroSpaceX Requirements Specification***

- a. At the moment a user logs in or a guest visits the ElectroSpaceX page at the top menu there is a button for the latest news.
  - b. User clicks that button and he is redirected to the flash news page
  - c. User can view news for electronics devices
8. Scenario title: Customer adds a product to the shopping cart
- a. At the menu in the top is a button that says products. Also at the same page there are shown best sellers products and on sale products.
  - b. A guest views a product and wants to add it to the shopping cart. Firstly he logs in as in the scenario 1 then he can add his product to the shopping cart.
  - c. Now he has one product at the shopping cart so he can view other products.
9. Scenario title: Order placement
- a. A customer wants to order a specified product so he choose the product and presses the button buy now.
  - b. He can press the button buy from the shopping cart only.
  - c. The total cost is shown at the left of the window.
  - d. At the right of the window the checkout form is shown that will be filled by the client.
  - e. In this way the order is proceed and the client will be informed by an email for the order details.
  - f. A thank you message is shown at the window and the user is redirected to the home page.
10. Scenario title: Feedback on a product
- a. Customer wants to review a product that he has bought.
  - b. He has a panel which is titled Feedback with 5 stars and a panel where he can leave a comment
  - c. The customer presses the number of stars that he wants to review the product
  - d. Then he can leave a comment if he wants
  - e. The customer presses the post button.
11. Scenario title: Logging out
- a. The user wants to log out from the system
  - b. He selects the log out option in the right top
  - c. He logs out of the system
12. Scenario title: Checking the orders
- a. The customer scrolls down to the end of the page
  - b. Then he selects the option My account
  - c. By pressing the "My Orders" option he can view all his orders
13. Scenario title: Checking the order status
- a. The customer scrolls down to the end of the page
  - b. Then he selects the option My account
  - c. By pressing the "My Orders" option he can view all his orders
  - d. He selects the order that he wants to check the status
  - e. Then he selects the Status option
  - f. The status of his order is shown

**Admin Scenarios**

14. Scenario title: Adding a new product
  - a. After logging in with his credentials, admin is redirected to his dashboard
  - b. At the dashboard there are several options and he selects manage products option
  - c. Then he selects the add new product button
  - d. The admin inserts all the attributes needed to create a new product in the appropriate fields
  - e. The administrator confirms the addition of the new product
  - f. Attributes are extracted from the fields, and product is added to the database
15. Scenario title: Delete a product
  - a. After logging in with his credentials, admin is redirected to his dashboard
  - b. At the dashboard there are several options and he selects manage products option
  - c. Then he selects the product that he wants to be deleted because it is not offered
  - d. Presses the delete option
  - e. The product is deleted from the database
16. Scenario title: Update a product
  - a. After logging in with his credentials, admin is redirected to his dashboard
  - b. At the dashboard there are several options and he selects manage products option
  - c. A table with all products is shown in the window
  - d. He selects the product that wants to edit
  - e. The product information form is displayed with the actual data
  - f. He changes the data and then saves it
  - g. The updated data are saved in the database
17. Scenario title: Adding a new category
  - a. After logging in with his credentials, admin is redirected to his dashboard
  - b. At the dashboard there are several options and he selects manage categories option
  - c. Then he selects the add new categories button
  - d. The admin inserts all the attributes needed to create a new category in the appropriate fields
  - e. Admin then adds the category id to the products that fit in it, or updates it properly
  - f. Everything is added to the database
18. Scenario title: Delete a category
  - a. After logging in with his credentials, admin is redirected to his dashboard
  - b. At the dashboard there are several options and he selects manage categories option
  - c. Then he selects the category that he wants to be deleted because it is not offered
  - d. Presses the delete option
  - e. The category is deleted from the database but he should take care about the products that were in this category
19. Scenario title: Update a category
  - a. After logging in with his credentials, admin is redirected to his dashboard
  - b. At the dashboard there are several options and he selects manage categories option
  - c. A table with all categories is shown in the window
  - d. He selects the category that wants to edit
  - e. The category information form is displayed with the actual data
  - f. He changes the data and then saves it



## ***ElectroSpaceX Requirements Specification***

- g. The updated data are saved in the database
20. Scenario title: Delete a user
- a. Admin goes to his dashboard and selects manage user's option
  - b. All the users with their details are displayed to him
  - c. Admin presses the "Delete" option to the desired user
  - d. The user is deleted from the database
21. Scenario title: Create a new user
- a. Admin can create employees, economists and clients
  - b. The button Add a new user at the manage users page provides the admin with the opportunity to add a new use in the system
  - c. A form is displayed to admin for filling the new user credentials and user level
  - d. Admin fill the form and presses add
  - e. The system checks if the user data are valid or existent before
  - f. If the user exist an error message is displayed, "This user exists already!"
  - g. Else the user is created
22. Scenario title: View bills and print them
- a. Admin enters the system by entering his credentials
  - b. He is redirected to his dashboard
  - c. There is an option named bills where he can view bills filtered by the time, price or user.
  - d. He selects one bill and then selects print the bill option
  - e. The bill is printed.
23. Scenario title: The stock is low, requesting supplies
- a. At the moment that one product is low on stock then a warning is generated at the admin screen.
  - b. Admin logs in with his credentials
  - c. Selects the option request supply
  - d. An automatic email is sent to the supplier
24. Scenario title: Creation of daily and monthly report
- a. Economist logs in using his credentials and he is redirected to his dashboard
  - b. The daily and monthly report for all order proceedings (in total of products used and income) is produced.
  - c. This report is sent to the admin for a verification
25. Scenario title: View stock of products
- a. Economist logs in using his credentials and he is redirected to his dashboard
  - b. At the moment that one product is low on stock then a warning is generated at the economist screen
  - c. He selects the option "request supply"
  - d. An automatic email is sent to the supplier
26. Scenario title: Check monthly payments
- a. Economist logs in using his credentials and he is redirected to his dashboard
  - b. He can calculate total earnings, total spendings and their respective taxes
  - c. It is further supplied with wages, to each worker
  - d. The total is written at the end of the report

## **4.2 Use Case**

UC_1.1	<i>Login</i>
Summary	User type his/her credentials to enter the system.
Dependency	
Actors	<i>Admin, Customer, Economist</i>
Preconditions	Customer must have an existing account. Economist account is created by admin.
Description of the Main Sequence	<ul style="list-style-type: none"> <li>• User opens the ElectroSpaceX page and in the top right there is a button that says Hyr</li> <li>• He is redirected to the login page where he should log in with his credentials, username and password.</li> <li>• The credentials are correct so the user is authorized to be redirected to the first page/ home page of the system</li> </ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"> <li>• In the other case, when a user doesn't have an account, he can register by entering some required credentials.</li> <li>• The Admin is created by us</li> <li>• Economist will be added by the Admin</li> </ul>
Non functional requirements	- This is a crucial part of the project because this is where the 'hackers' will try to break through, by using SQL injection. It is made sure that this will be avoided. Also, the password will be hashed and cannot be seen by any person
Postconditions	The user will be directed to his dashboard

UC_1.2	<i>Edit profile</i>
Summary	Each user has the right to edit his profile. (or credentials)
Dependency	
Actors	<i>Admin, Customer, Economist</i>
Preconditions	<ul style="list-style-type: none"> <li>• User must have a valid account first</li> <li>• User must be logged in the system</li> </ul>
Description of the Main Sequence	<ul style="list-style-type: none"> <li>• User logs in the system</li> <li>• In they dashboard they have the option my profile</li> <li>• Each of them can edit their profile like their username, password, address (for customers) photo.</li> </ul>

### ***ElectroSpaceX Requirements Specification***

Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• If by any chance the new data aren't in the correct form then they will not be put in the database and the profile of the user will remain the same.</li><li>• User will be warned.</li></ul>
Non functional requirements	<ul style="list-style-type: none"><li>• The password will be hashed and cannot be seen by anyone.</li></ul>
Postconditions	<ul style="list-style-type: none"><li>• User can now see the updated profile</li></ul>

UC_1.3	<i>View products and their sales</i>
Summary	Each user and guest can view the available products
Dependency	
Actors	<i>Admin, Customer, Economist, Guest</i>
Preconditions	<ul style="list-style-type: none"><li>• Everyone who visits the website can view the available products</li><li>• It is needed internet connection</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• They can view all the available products with their respective details</li><li>• Customers can add them to the Wishlist or to the shopping cart</li></ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• The stock may be low for some products so there won't be any product.</li><li>• Guest can only view products. He hasn't any right to add the products to the Wishlist or shopping cart.</li></ul>
Non functional requirements	<ul style="list-style-type: none"><li>• To meet the Usability Requirements, the application is user-friendly, simple and easy to understand.</li></ul>
Postconditions	<ul style="list-style-type: none"><li>• When the product is selected, user can view the product details</li></ul>

UC_1.4	<i>Purchasing a product</i>
Summary	Each user can buy a product
Dependency	
Actors	<i>Customer</i>
Preconditions	<ul style="list-style-type: none"><li>• The product must be added in the database by the admin</li><li>• Customer must be logged in first</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• A customer wants to order a specified product so he choose the product and presses the button buy now.</li><li>• He can press the button buy from the shopping cart only.</li><li>• The checkout form will be filled by the client.</li></ul>

### ***ElectroSpaceX Requirements Specification***

Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• The stock may be low for some products so there won't be any product.</li><li>• Guest can only view products. He hasn't any right to add the products to the Wishlist or shopping cart.</li></ul>
Non functional requirements	<ul style="list-style-type: none"><li>• To meet the Usability Requirements, the application is user-friendly, simple and easy to understand.</li></ul>
Postconditions	<ul style="list-style-type: none"><li>• A thank you message is shown at the window and the user is redirected to the home page. The client will be informed by an email for the order details.</li></ul>

UC_1.5	<i>Feedback on the products</i>
Summary	Each customer can leave feedback on a product
Dependency	
Actors	<i>Customer</i>
Preconditions	<ul style="list-style-type: none"><li>• The product must be added in the database by the admin</li><li>• Customer must have bought this product</li><li>• Customer must be logged in first</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• He has a panel which is titled Feedback with 5 stars and a panel where he can leave a comment</li><li>• The customer presses the number of stars that he wants to review the product and a comment</li></ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• Customer hasn't bought yet the product</li><li>• Customer hasn't log in in the system.</li></ul>
Non functional requirements	<ul style="list-style-type: none"><li>• To meet the Usability Requirements, the feedback panel is user-friendly, simple and easy to understand.</li></ul>
Postconditions	<ul style="list-style-type: none"><li>• He is redirected in the home page</li></ul>

UC_1.6	<i>Manage products</i>
Summary	Admin can manage all products
Dependency	
Actors	<i>Admin</i>

### *ElectroSpaceX Requirements Specification*

Preconditions	<ul style="list-style-type: none"><li>• Admin must be logged in with his credentials</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• Admin can manage all products. He can add, delete, view, update specific products.</li></ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• Admin hasn't logged in as admin</li><li>• There isn't any product to view, edit or update. The only option that is left is add new product option.</li></ul>
Non functional requirements	<ul style="list-style-type: none"><li>• To meet the Usability Requirements, the application is user-friendly, simple and easy to understand.</li></ul>
Postconditions	<ul style="list-style-type: none"><li>• Admin is redirected to the Manage products page</li></ul>

UC_1.6.1	<i>Add products</i>
Summary	Admin can add new products with their respective details
Dependency	<i>Manage products</i>
Actors	<i>Admin</i>
Preconditions	<ul style="list-style-type: none"><li>• Admin must log in as an admin in the system</li><li>• Selects the option add a product</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• The admin inserts all the attributes needed to create a new product in the appropriate fields</li><li>• The administrator confirms the addition of the new product</li></ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• The attributed doesn't meet the needed requirements or a field is empty so admin is warned about this.</li></ul>
Non functional requirements	<ul style="list-style-type: none"><li>• Security requirements:</li><li>• Confidentiality: Access should be granted only to authorized users.</li></ul> Integrity: Only the authorized user can modify the data depending on its role.
Postconditions	<ul style="list-style-type: none"><li>• Attributes are extracted from the fields, and product is added to the database.</li><li>• Admin is redirected to his dashboard and customers will have a new product in their menu</li></ul>

UC_1.6.2	<i>Delete products</i>
Summary	Admin can delete a selected product
Dependency	<i>Manage products</i>
Actors	<i>Admin</i>

### ***ElectroSpaceX Requirements Specification***

Preconditions	<ul style="list-style-type: none"><li>• <i>Admin must be logged in as admin firstly</i></li><li>• The product must be available in the database</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• He selects the product that he wants to be deleted because it is not offered anymore</li><li>• Presses the delete option</li></ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• Admin hasn't been logged in.</li><li>• The product is on a sale already</li></ul>
Non functional requirements	<ul style="list-style-type: none"><li>• Security requirements:</li><li>• Confidentiality: Access should be granted only to authorized users.</li></ul> Integrity: Only the authorized user can modify the data depending on its role.
Postconditions	<ul style="list-style-type: none"><li>• The product is deleted from the database</li><li>• Customers cannot view or buy anymore this product</li><li>• Admin is redirected to his dashboard</li></ul>

UC_1.6.3	<i>Update products</i>
Summary	Admin can update product attributes
Dependency	<i>Manage products</i>
Actors	<i>Admin</i>
Preconditions	<ul style="list-style-type: none"><li>• <i>Admin must be logged in as admin firstly</i></li><li>• The product must be available in the database</li><li>• A table with all products is shown in the manage products page</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• He selects the product that wants to edit</li><li>• The product information form is displayed with the actual data</li><li>• He can change the price, image, quantity, name and other details.</li></ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• Any of the attributes is put wrongly or a field is empty so the product won't be updated</li></ul>
Non functional requirements	<ul style="list-style-type: none"><li>• Security requirements:</li><li>• Confidentiality: Access should be granted only to authorized users.</li><li>• Integrity: Only the authorized user can modify the data depending on its role.</li></ul>
Postconditions	<ul style="list-style-type: none"><li>• The updated data are saved in the database</li><li>• Customers and guests have now the updated product in their page</li></ul>

### *ElectroSpaceX Requirements Specification*

UC_1.7	<i>Manage categories</i>
Summary	Admin can manage all categories
Dependency	
Actors	<i>Admin</i>
Preconditions	<ul style="list-style-type: none"><li>• Admin must be logged in with his credentials</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• Admin can manage all categories. He has several options like edit, view, delete, update.</li></ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• Admin hasn't logged in as admin</li><li>• There isn't any category to view, edit or update. The only option that is left is add new category option.</li></ul>
Non functional requirements	<ul style="list-style-type: none"><li>• To meet the Usability Requirements, the application is user-friendly, simple and easy to understand.</li></ul>
Postconditions	<ul style="list-style-type: none"><li>• Admin is redirected to the manage categories page</li></ul>

UC_1.7.1	<i>Add categories</i>
Summary	Admin can add new categories with their respective details
Dependency	<i>Manage categories</i>
Actors	<i>Admin</i>
Preconditions	<ul style="list-style-type: none"><li>• Admin must log in as an admin in the system</li><li>• Selects the option add a category</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• The admin inserts all the attributes needed to create a new category in the appropriate fields</li><li>• The administrator confirms the addition of the new category</li></ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• The attributed doesn't meet the needed requirements or a field is empty so admin is warned about this.</li></ul>
Non functional requirements	<ul style="list-style-type: none"><li>• Security requirements:</li><li>• Confidentiality: Access should be granted only to authorized users.</li></ul> Integrity: Only the authorized user can modify the data depending on its role.
Postconditions	<ul style="list-style-type: none"><li>• Attributes are extracted from the fields, and the category is added to the database.</li></ul>

### *ElectroSpaceX Requirements Specification*

	<ul style="list-style-type: none"><li>• Admin is redirected to his dashboard and customers will have a new category in their menu</li></ul>
--	---

UC_1.7.2	<i>Delete categories</i>
Summary	Admin can delete a selected category
Dependency	<i>Manage categories</i>
Actors	<i>Admin</i>
Preconditions	<ul style="list-style-type: none"><li>• Admin must be logged in as admin firstly</li><li>• The product must be available in the database</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• He selects the category that he wants to be deleted because it is not offered anymore</li><li>• Presses the delete option</li></ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• Admin hasn't been logged in.</li><li>• The category mustn't have any product.</li></ul>
Non functional requirements	<ul style="list-style-type: none"><li>• Security requirements:</li><li>• Confidentiality: Access should be granted only to authorized users.</li></ul> Integrity: Only the authorized user can modify the data depending on its role.
Postconditions	<ul style="list-style-type: none"><li>• The category is deleted from the database</li><li>• Customers cannot view anymore this category</li><li>• Admin is redirected to his dashboard</li></ul>

UC_1.6.3	<i>Update categories</i>
Summary	Admin can update categories attributes
Dependency	<i>Manage categories</i>
Actors	<i>Admin</i>
Preconditions	<ul style="list-style-type: none"><li>• Admin must be logged in as admin firstly</li><li>• The category must be available in the database</li><li>• A table with all categories is shown in the manage category page</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• He selects the category that wants to edit</li><li>• The category information form is displayed with the actual data</li><li>• He can change the attributes of this category</li></ul>
Description of the	<ul style="list-style-type: none"><li>• Any of the attributes is put wrongly or a field is empty so the category won't be</li></ul>



### *ElectroSpaceX Requirements Specification*

Alternative Sequence	updated
Non functional requirements	<ul style="list-style-type: none"><li>• Security requirements:</li><li>• Confidentiality: Access should be granted only to authorized users.</li><li>• Integrity: Only the authorized user can modify the data depending on its role.</li></ul>
Postconditions	<ul style="list-style-type: none"><li>• The updated data are saved in the database</li><li>• Customers and guests have now the updated category in their page</li></ul>

UC_1.8	<i>Manage orders</i>
Summary	Admin can manage all orders.
Dependency	
Actors	<i>Admin</i>
Preconditions	<ul style="list-style-type: none"><li>• Admin must be logged in the system as admin</li><li>• Must be an available order</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• He can view the order details, approve or cancel orders.</li></ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• Admin hasn't been logged in as admin</li><li>• There isn't any order available</li></ul>
Non functional requirements	
Postconditions	<ul style="list-style-type: none"><li>• Admin is redirected to the manage orders page</li></ul>

UC_1.8.1	<i>Approve orders</i>
Summary	Admin can approve an order
Dependency	<i>Manage orders</i>
Actors	<i>Admin</i>
Preconditions	<ul style="list-style-type: none"><li>• Admin must be logged in the system as admin</li><li>• Must visit manage orders page</li><li>• Must be a new order</li></ul>
Description of the	<ul style="list-style-type: none"><li>• He can approve the new order by clicking in an approve option</li></ul>

### *ElectroSpaceX Requirements Specification*

Main Sequence	
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• Admin hasn't been logged in as admin</li><li>• The product that has been ordered isn't available for the moment in the warehouse so the admin should email the client.</li></ul>
Non functional requirements	<ul style="list-style-type: none"><li>• The order details are shown only to admin and the client. They are invisible to other users.</li></ul>
Postconditions	<ul style="list-style-type: none"><li>• Admin is redirected to the manage orders page</li><li>• Order is approved.</li></ul>

UC_1.8.1	<i>Cancel orders</i>
Summary	Admin can cancel an order
Dependency	<i>Manage orders</i>
Actors	<i>Admin</i>
Preconditions	<ul style="list-style-type: none"><li>• Admin must be logged in the system as admin</li><li>• Must visit manage orders page</li><li>• Must be a new order</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• He can cancel the new order by clicking in the cancel option</li><li>• The product that has been ordered isn't available for the moment in the warehouse so the admin should email the client.</li></ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• Admin hasn't been logged in as admin</li><li>• The product is available so the order is approved by the admin</li></ul>
Non functional requirements	<ul style="list-style-type: none"><li>• The order details are shown only to admin and the client. They are invisible to other users.</li></ul>
Postconditions	<ul style="list-style-type: none"><li>• Admin is redirected to the manage orders page</li><li>• Order is canceled.</li></ul>

UC_1.9	Requesting supplies
Summary	Admin and Economist can request supplies when stock is low
Dependency	
Actors	<i>Admin, Economist</i>

### ***ElectroSpaceX Requirements Specification***

Preconditions	<ul style="list-style-type: none"><li>• Admin must be logged in the system as admin</li><li>• Economist must be logged in</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• At the moment that one product is low on stock then a warning is generated at the admin or economist screen.</li><li>• Selects the option request supply</li></ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• Admin hasn't been logged in as admin</li><li>• Economist hasn't been logged in</li><li>• There isn't a product with low stock</li></ul>
Non functional requirements	
Postconditions	<ul style="list-style-type: none"><li>• An automatic email is sent to the supplier</li><li>• <i>Admin or economist is redirected to the dashboard</i></li></ul>

UC_2.0	Creation of daily and monthly report
Summary	Admin and Economist can request supplies when stock is low
Dependency	
Actors	<i>Economist</i>
Preconditions	<ul style="list-style-type: none"><li>• Economist must be logged in</li></ul>
Description of the Main Sequence	<ul style="list-style-type: none"><li>• The daily and monthly report for all order proceedings (in total of products used and income) is produced.</li><li>• This report is sent to the admin for a verification</li></ul>
Description of the Alternative Sequence	<ul style="list-style-type: none"><li>• Admin doesn't approve the report</li></ul>
Non functional requirements	
Postconditions	<ul style="list-style-type: none"><li>• Admin approves the report and the report is ready to be printed</li></ul>

## 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. Definitions, Acronyms, and Abbreviations**

Define all terms, acronyms, and abbreviations used in this document.

### **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 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 Assign BU Conceptual Design	Accepted
		BUA-PF-01 Derive Bargaining Unit-Process Flow Diagram	Accepted
		BUA-PF-01 Derive Bargaining Unit-Process Flow Diagram	Accepted
BR_LR_09 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 Assign BU Conceptual Design	Accepted
		BUA-PF-02 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

### ElectroSpaceX Requirements Specification

BR_LR_01	1	BUA	BUA-PF-01	Derive Bargaining Unit-Process Flow Diagram	Accepted
BizReqID	Pri	Major Area	DevTstItems DelivID	Deliv Name	Status
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
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	

### ElectroSpaceX Requirements Specification

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.

### ***ElectroSpaceX Requirements 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.