Faculty of Mathematics and Information Science Warsaw University of Technology



A System for Resources Management

in a Small Chemical Laboratory

Klaudia Jarosz

Maciej Głowala

Aleksandra Bułka

Version no. 1.0

22.10.2017.

Table of Contents

Table of Contents 2

1 History of Document Changes 2

2 Specification 2

2.1 Executive summary 3

2.2 Functional requirements 3

2.3 Non-functional requirements 5

2.4 Project schedule 6

# History of Document Changes

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Author** | **Description** | **Version** |
| 22.10.2017 | Aleksandra Bułka | Initial version. | 1.0.0 |
| 24.10.2017 | Aleksandra Bułka | Added functional requirements.. | 1.0.1 |
| 24.10.2017 | Maciej Głowala | Added non-functional requirements and user interface design. | 1.0.2 |
| 27.10.2017 | Aleksandra Bułka | Added Schedule | 1.0.3 |

# Specification

## Executive summary

The students will design and implement a system supporting management of supplies in a small chemical laboratory (chemical reagents, instruments, etc.). The system keeps track of the state of resources in the laboratory, and stores the data in a database. The system has graphical user-friendly interface which facilitates displaying and modifying the gathered data. Multiple functionalities in the system allow the user to:

* Classify the resource into groups
* Assign descriptions with multimedia content to resources
* Define and generate reports and notifications displaying current state of resources as well as plot of activity in time, plot of demand for some resource in time, bar chart, pie chart, table with adjustable columns/rows, an alert about low level of some chemical reagent, etc.
* For each resource, store suppliers’ contact data and order new resources directly from the application
* Predict future demand for resources based on available historical data

The initial set of requirements for the project is described in the following chapters..

## Functional requirements

Three tables below contain a description of use cases for different actors.

The three actors = three possible users of our system will be:

1. Administrator (master user, manages the system as a whole, including its users)
2. Manager (a person with rights for laboratory resources management)
3. Registered user

Table 1. Description of use cases for Administrator

|  |  |  |
| --- | --- | --- |
| Actor | Name | Description |
| Administrator | Login | Log in to the system |
| Password Management | Recover his password |
| Change his password |
| Resource View | Display nicely current availability of resources and their categorization - multiple viewing perspectives, sorting and filtering are available |
| Report Definition | Define the method of sending a report and its recipients |
| Define what the report will contain , for example a plot of activity in time, plot of demand for some resource in time |
| Define how the data will be presented, for example bar chart, pie chart, table with adjustable columns/rows |
| Report Generation | Select and generate a Report |
| Send a Report |
| Notification Definiton | Define type of notification and when will it be sent, for example an alert about low level of some chemical reagent |
| Prediction View | Select data resources or groups of resources to be variables and generate a prediction of future demand for resources based on available historical data (multivariate time series prediction) |
| Plan Management | Create a plan |
| Modify a plan |
| Remove a plan |
| User Account Management | Create an account in the system, assign the account to a role (user, manager) |
| Modify an account in the system – change either personal data or assignment to a role (user, manager) |
| Remove an account from the system |
| Resource Management | Define a new type of resource, describe it with description card, add multimedia content to this resource (for example a photograph) and assign the resource with a supplier |
| Modify a resource, change description card, multimedia content associated with this resource (for example a photograph) and its assignment to a supplier |
| Resource Group Management | Create a group of laboratory resources, define which resources will belong to this group |
| Modify a group of laboratory resources, redefine which resources will belong to this group |
| Delete a group of laboratory resources |
| System State Management | Save the current state of the system |
| Restore system state based on an archived backup |

Table 2. Description of use cases for Manager

|  |  |  |
| --- | --- | --- |
| Actor | Name | Description |
| Manager | Login | Log in to the system |
| Password Management | Recover his password |
| Change his password |
| Resource View | Display nicely current availability of resources and their categorization - multiple viewing perspectives, sorting and filtering are available |
| Report Definition | Define the method of sending a report and its recipients |
| Define what the report will contain, for example a plot of activity in time, plot of demand for some resource in time |
| Define how the data will be presented, for example bar chart, pie chart, table with adjustable columns/rows |
| Report Generation | Select and generate a Report |
| Send a Report |
| Notification Definiton | Define type of notification and when will it be sent, for example an alert about low level of some chemical reagent |
| Prediction View | Select data resources or groups of resources to be variables and generate a prediction of future demand for resources based on available historical data (multivariate time series prediction) |
| Order Management | Choose a method of sending orders for new resources – either manual or automatic |
| If manual option is chosen – order new resources |
| Modify suppliers data |
| Resource Management | Add a new resource, describe it with description card, add multimedia content to this resource (for example a photograph) and assign the resource with a supplier |
| Add instances of a resource |
| Modify a resource, change description card, multimedia content associated with this resource (for example a photograph) and its assignment to a supplier |
| For instances of resource present in database – modify quantity |
| Delete resources from database |

Table 3. Description of use cases for User

|  |  |  |
| --- | --- | --- |
| Actor | Name | Description |
| User | Login | Log in to the system |
| Password Management | Recover his password |
| Change his password |
| Resource View | Display nicely current availability of resources and their categorization - multiple viewing perspectives, sorting and filtering are available |
| Report Management | Redefine what the report will contain, for example a plot of activity in time, plot of demand for some resource in time |
| Redefine how the data will be presented, for example bar chart, pie chart, table with adjustable columns/rows |
| Report Generation | Select and generate a Report |
| Send a Report |
| Prediction View | Select data resources or groups of resources to be variables and generate a prediction of future demand for resources based on available historical data (multivariate time series prediction) |
| Resource Management | Add instances of a resource |
| For instances of resource present in database – modify quantity |
| Delete instances of resources from database |

## User Interface Design

### General Description

1. Common:

* Login page
* Password recovery page
* Welcome page after logging in
* Change password subpage
* General elements of page:

1. Page-height menu on the left side of screen
2. Footer

1. Administrator:

* Subpage for add/remove/edit plans
* Subpage for managing users
* Subpage for reports and predictions

1. Manager:

* Subpage for reports and predictions
* Subpage for managing orders
* Subpage for managing data

1. User:

* Subpage for choosing plan to do and input data.

### Detailed description:

On each page there will be strip on the top of the page with the logo and name of the application.

1. Login Page:
2. Input boxes for username and password
3. Login button
4. Forgotten password button
5. Password recovery page:
6. Email input box
7. Ok button
8. Welcome page after logging:
9. Logo of the application
10. Information about application
11. Change password subpage:
12. Input box for present password
13. Input box for new password
14. Input box for password confirmation
15. Ok button
16. Subpage for add/remove/edit plans:
17. General table of the plans with the column for add/remove/edit buttons
18. Pop-up window for adding, editing and remove confirmation
19. Subpage for managing users:
20. General table of the users with the column for add/remove/edit buttons
21. Pop-up window for adding, editing and remove confirmation
22. Subpage for reports and predictions:
23. Predictions section
24. Set of radio buttons for report type selection
25. Input for date range
26. Prepare and download button
27. Subpage for managing orders:
28. General table of orders
29. Add new order button
30. Pop-up window for adding orders
31. Subpage for managing data:
32. General table of available materials
33. Subpage for choosing plan to do and input data:
34. Selection list of a plan
35. Accept button
36. Pop-up for data input

## Non-functional requirements

|  |  |  |
| --- | --- | --- |
| **Area** | **Nr** | **Details** |
| *Usability* | 1 | Application must be responsive. It must be working on PC, tablets and phones with resolution at least 720p. |
| *Reliability* | 2 | Application must be of type *High Availability. It should be available 24h/7d between* 08:00 and 23:00. There could be service breaks during the week between 24:00 and0 8:00. |
| 3 | Application must have quick restart in case of app machine failures. |
| *Recovery* | 4 | Application must have daily database recovery performed between 24:00 and 08:00 |
| *Performance* | 5 | Application should respond no longer than 3 seconds while strain being on level 100 queries per minute. |
| *Supportability* | 6 | Documentation should contain instruction for recovery data from database backup. |
| 7 | Application should keep backward compatibility between the released versions. |
| *Security* | 8 | Application must have user levels security. It shall not pass a user who has inappropriate privileges. |

## Project schedule

### Project Schedule

|  |  |
| --- | --- |
| 1. **Part 1** | **01.10.2017.-26.10.2017.** |
| 1. Functional Requirements (Use Cases) | 01.10.2017.-20.10.2017. |
| 1. Non-functional requirements | 01.10.2017.-20.10.2017. |
| 1. User Interface Design | 08.10.2017.-23.10.2017. |
| 1. Glossary | 14.10.2017.-26.10.2017. |
| 1. Project schedule | 20.10.2017.-26.10.2017. |
| 1. **Part 2** | **27.10.2017. – 09.11.2017.** |
| 1. Technical Project | 24.10.2017. – 09.11.2017. |
| 1. Choice of Development Model | 27.10.2017. – 09.11.2017. |
| 1. **Part 3** | **10.11.2017. – 23.11.2017.** |
| 1. Developing Modules | 08.11.2017. – 23.11.2017. |
| 1. Unit Test Projects | 16.11.2017. – 23.11.2017. |
| 1. **Part 4** | **24.11.2017. – 07.12.2017.** |
| 1. Integrating Modules | 22.11.2017. – 07.12.2017. |
| 1. Acceptance Test Project | 30.11.2017. – 07.12.2017. |
| 1. **Part 5** | **08.12.2017. – 05.01.2017.** |
| 1. Preparing final version, fixing bugs | 07.12.2017. – 02.01.2017. |
| 1. Documenting Code | 07.12.2017. – 02.01.2017. |
| 1. User Manual | 20.12.2017. – 02.01.2017. |
| 1. Reports from Tests | 20.12.2017. – 02.01.2017. |
| 1. List of Changes | 30.12.2017. – 05.01.2017. |

### Gannt Chart

