Politechnika Warszawska

WYDZIAŁ MATEMATYKI
I NAUK INFORMACYJNYCH

# Praca dyplomowa licencjacka

na kierunku Matematyka

NAJPIERW WYGENERUJ STRONĘ TYTUŁOWĄ

KTÓRA ZNAJDUJE SIĘ

Numer albumu 000000

promotor

W KATALOGU title_page

konsultacje

I OTRZYMANEGO PDF-A NAZWIJ titlepage.pdf I WSTAW DO KATALOGU Z SZABLONEM!

WARSZAWA 2018

..............................................                ..............................................

supervisor's signature                author's signature

## Abstract

A System for Resources Management in a Small Chemical Laboratory

Firstly we will write an abstract of our work in English.

**Keywords:** keyword1, keyword2, ...

**Streszczenie**

System zarządzania zasobami małego laboratorium chemicznego

**Słowa kluczowe:** slowo1, slowo2, ...

Warsaw, ..................

Declaration

I hereby declare that the thesis entitled „A System for Resources Management in a Small Chemical Laboratory", submitted for the Engineer degree, supervised by dr inż. Agnieszka Jastrzębska, is entirely my original work apart from the recognized reference.

..............................................

# Contents

# Introduction

For our engeeniring project, we have designed and implemented 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 further describiton of the project is contained in the following chapters.

# 1. Work Division Plan

The project was designed so as it could be completed by the group of three people. The tasks were divided equally. That being said, all three participants of the project contributed towards the building of the system, its design and the frame application. What is more, great emphasis has been put on collaboration and team work, resulting in members of the group often performing tasks outside their divised part, which contributed toward the project's final success.

## 1.1. Work Division

Table 1.1: Work Division

| Name | Responsibility |
|---|---|
| Klaudia Jarosz | Implementation of the frame application |
| | Creating a database containg data about users, resources, suppliers and user's activity |
| | Design and implementation of a user-friendly interface |
| | Testing |
| Maciej Głowala | Implementation of the frame application |
| | Handling users with different roles |
| | Saving and restoring system state |
| | Virtual server setup |
| | Testing |
| Aleksandra Bułka | Implementation of the frame application |
| | Implementation of reports and notifications |
| | Forecasting module |
| | Testing |

# 2. Business Analysis

TODO

## 2.1. Business Analysis

## 2.2. Other Known Solutions

# 3. Requirement Specification

## 3.1. Functional Requirements

The functional reuirements of the applications are different for different users of the application. The tables contained in the following chapters provide descriptions of use cases for different groups of application users. The three groups of users of the application are:

1. Administrator (manager of the whole system and its users)
2. Manager (a person with rights for laboratory resources management)
3. Registered user

### 3.1.1. Administrator

Table 3.1: Functional Requirements for Administrator

| Actor | Name | Description |
|---|---|---|
| Administrator | Login | Log in to the system |
| | Password Management | Recover his password |
| | | Change his password |
| | 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 View | Display nicely current availability of resources and their categorization - multiple viewing perspectives, sorting and filtering are available |

| | | |
|---|---|---|
| | 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 |
| | | Delete a resource from database |
| | 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 |
| | Product View | Display nicely products produced by laboratory and their categorization |
| | Product Management | Define a new type of product, describe it with description card, add multimedia content to this product (for example a photograph) |
| | | Modify a product, change description card, multimedia content associated with this product (for example a photograph) |
| | | Delete a product from database |
| | Product Group Management | Create a group of laboratory product, define which products will belong to this group |
| | | Modify a group of laboratory products, redefine which products will belong to this group |
| | | Delete a group of laboratory products |
| | Report Definition | Define what the report will contain, for example a plot of production 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 |
| | | Generate the desired report |
| | | Define the recipients and send the report |
| | Prediction Report Definition | Define what the report will contain, this can be either for example production in time or demand for some resource in time |

| | | |
|---|---|---|
| | | Define for which resources, products or groups of resources or products the prediction should be made |
| | | Generate the desired prediction report |
| | | Define the recipients and send the prediction report |
| | Notification Definiton | Define whether notifications (an alert about low level of some chemical reagent) will it be sent |
| | | Define when the notifications will it be sent (set the value which is critical for each reasource) |
| | Order Management | Define how many and which resources are to be ordered |
| | | Send an order |
| | | Turn on automatic ordering of resources or turn it off |
| | System State Management | Save the current state of the system |
| | | Shedule an automatic back-up of a system state |
| | | Restore system state based on an archived backup |

### 3.1.2. Manager

Table 3.2: Functional Requirements 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 |
| | 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 |

| | | |
|---|---|---|
| | | Delete a resource from database |
| | 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 |
| | Product View | Display nicely products produced by laboratory and their categorization |
| | Product Management | Define a new type of product, describe it with description card, add multimedia content to this product (for example a photograph) |
| | | Modify a product, change description card, multimedia content associated with this product (for example a photograph) |
| | | Delete a product from database |
| | Product Group Management | Create a group of laboratory product, define which products will belong to this group |
| | | Modify a group of laboratory products, redefine which products will belong to this group |
| | | Delete a group of laboratory products |
| | Report Definition | Define what the report will contain, for example a plot of production 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 |
| | | Generate the desired report |
| | | Define the recipients and send the report |
| | Prediction Report Definition | Define what the report will contain, this can be either for example production in time or demand for some resource in time |
| | | Define for which resources, products or groups of resources or products the prediction should be made |
| | | Generate the desired prediction report |
| | | Define the recipients and send the prediction report |
| | Notification Definiton | Define whether notifications (an alert about low level of some chemical reagent) will it be sent |

| | | |
|---|---|---|
| | | Define when the notifications will it be sent (set the value which is critical for each reasource) |
| | Order Management | Define how many and which resources are to be ordered |
| | | Send an order |
| | | Turn on automatic ordering of resources or turn it off |

### 3.1.3. User

Table 3.3: Functional Requirements 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 |
| | 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 |
| | | Delete a resource from database |
| | Product View | Display nicely products produced by laboratory and their categorization |
| | Product Management | Define a new type of product, describe it with description card, add multimedia content to this product (for example a photograph) |
| | | Modify a product, change description card, multimedia content associated with this product (for example a photograph) |
| | | Delete a product from database |

| | | |
|---|---|---|
| | Report Definition | Define what the report will contain, for example a plot of production 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 |
| | | Generate the desired report |
| | | Define the recipients and send the report |
| | Prediction Report Definition | Define what the report will contain, this can be either for example production in time or demand for some resource in time |
| | | Define for which resources, products or groups of resources or products the prediction should be made |
| | | Generate the desired prediction report |
| | | Define the recipients and send the prediction report |
| | Order Management | Define how many and which resources are to be ordered |
| | | Send an order |

## 3.2. Non-functional Requirements

## 3.3. User Stories

## 3.4. Use Cases

Table 3.4: Non-functional requirements

| Area | Number | 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 and 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. |

# 4. Development Model and Methodology

## 4.1. Methodology

For our project, the development methodology of choice was the Waterfall Development Model. This methodology was highly recommended to us.

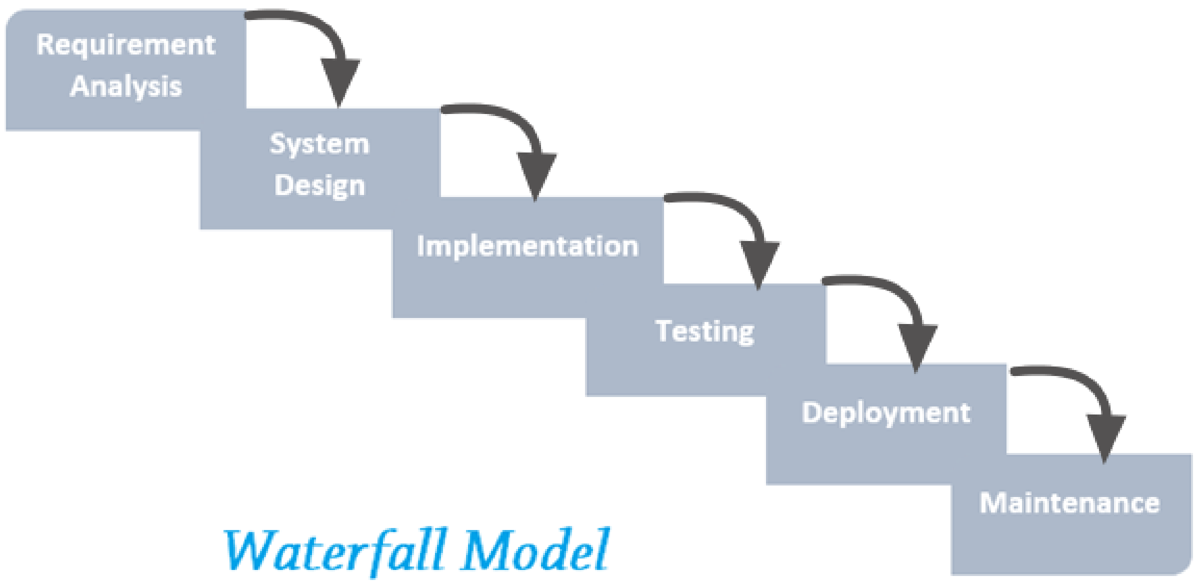

Figure 4.1: Waterfall Model

In Waterfall Model phases are executed sequentially, in linear way. We have a steady set of requirements, non changeable in time. The system is developed progressively and the user is involved only in the early phases Advantages of this model include it being easy to manage. Disadvantages are that it contains a strict sequence of activities, it has high cost of errors in beginning stages and high importance and cost of documentation and also a contact with the customer is weak. Therefore it can be used when it is possible to precisely define the requirements

### 4.1.1. Argumentation

The reasons we have chosen this particular development model are presented below:

- The sequential order of phases matched perfectly the organized schedule of the development of our Engineering Thesis
- Fixed set of requirements, as the initial requirements waere submitted by team to the Faculty and could not be changed later
- Easy management is facilitating our work as we work as a team and have no manager
- It was highly recommended by the coordinators of the Group Project

## 4.2. Development Model

# 5. System Architecture

## 5.1. Software Architectural Pattern

To facilitate the design of classes, we have decided to follow a software architectural pattern. A pattern of choice here was Model–view–controller (MVC) Pattern. This pattern is used to separate a given application into three interconnected parts. This is done to separate internal representations of information from the ways information is presented to, and accepted from, the user.

- Model - Model represents an object carrying data. It can also have logic to update controller if its data changes.
- View - View represents the visualization of the data that model contains.
- Controller - Controller acts on both model and view. It controls the data flow into model object and updates the view whenever data changes. It keeps view and model separate.

As mentioned before, it has helped us develop the division into classes, visible on the class diagram below. LIME – is the program's main class. Then, the Model, View and Controller Classes were designed for each UI element. The database objects are handled by Servlet (classes taking an article from http POST and passing it to JDBC), DAO (classes responsible for the communication with database) and Query (classes to parse SQL) classes.

## 5.2. Backend Architectural Pattern

L.I.M.E. project base on many architectural patterns. Precisely chosen patterns make overall architecture strong and well built.

### 5.2.1. Data Access Object Pattern

Data access object (DAO) is an object that provides an abstract interface to some type of database or other persistence mechanism. By mapping application calls to the persistence

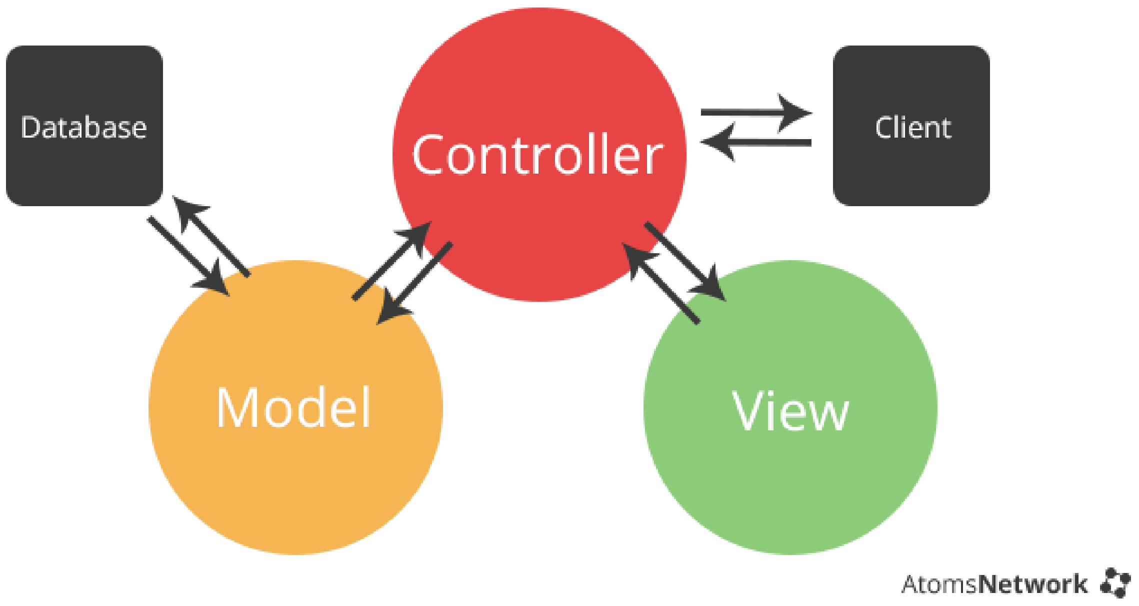

Figure 5.1: MVC Pattern

layer, the DAO provides some specific data operations without exposing details of the database. This isolation supports the Single responsibility principle. It separates what data access the application needs, in terms of domain-specific objects and data types (the public interface of the DAO), from how these needs can be satisfied with a specific DBMS, database schema, etc. (the implementation of the DAO). The most important fact and advantage is the relatively simple and rigorous separation between two important parts of an application that can but should not know anything of each other, and which can be expected to evolve frequently and independently. Changing business logic can rely on the same DAO interface, while changes to persistence logic do not affect DAO clients as long as the interface remains correctly implemented. All details of storage are hidden from the rest of the application (see information hiding). Thus, possible changes to the persistence mechanism can be implemented by just modifying one DAO implementation while the rest of the application isn't affected. L.I.M.E. implements one template parent interface with methods which are used by all DAO classes. IBasicCrudRepository extends hibernate CrudRepository and inherits from it all CRUD database operations. Singular DAO classes implement service-specific methods which are defined in DAO interfaces.

# 6. Technical Analysis

More detailed - all the classes and functions

## 6.1. General Program Workflow

## 6.2. Program Interface

## 6.3. System Classes

## 6.4. Database Design

## 6.5. Prediction Model

# 7. Experimental Evaluation

This is important - these are various tests, trying to cover all the possible thing user can do in the application

## 7.1. Evaluation Plan

# 8. Acceptance Tests

This is for other tests

## 8.1. Test Plan

## Conclusions

1 Page of Our Conclusions of Writing This Project

# Bibliography

[1] A. Author, *Title of a book*, Publisher, year, page–page.

[2] J. Bobkowski, S. Dobkowski, Title of an article, *Magazine X, No. 7*, year, PAGE–PAGE.

[3] C. Brink, Power structures, *Algebra Universalis 30(2)*, 1993, 177–216.

[4] F. Burris, H. P. Sankappanavar, *A Course of Universal Algebra*, Springer-Verlag, New York, 1981.

## Glossary

| | |
|---|---|
| **LIME** | Laboratory Internal Management Entity - the name of the application discussed in this document. |

# List of Figures

If you don't need it, delete it.

## List of tables

If you don't need it, delete it.

## List of appendices

## A. User Manual

There we will have manual

## B. Testing Scenarios

Testing Scenarios