Assignment 3
Analysis and Design Document

Student:Lazaroiu Andreea Sabina
Group:30432

Table of Contents

1. Requirements Analysis

1.1 Assignment Specification

Design and implement a client-server application for a news agency. The application has 2 types of users: the readers and the writers.

1.2 Functional Requirements

The **readers** can view a list of articles, read an article and do not need to login in order the use the application. The **writers** need to authenticate in order to create, update or delete articles. So the writer accounts are preset by the application developer and cannot be altered.

An article has the following components:

- Title
- Abstract
- Author
- Body

The application must support multiple concurrent users. If a writer posts a new article, the readers must see it in the list of articles in real time, without performing any refresh operation.

1.3 Non-functional Requirements
1. Stability: The objects are stable and do not need changes.
2. Adaptability:

2. Use-Case Model
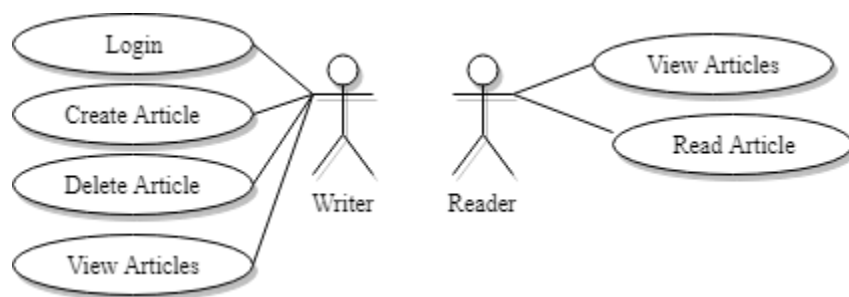
Use-Case description format:
Use case: Create Article
Level: user-goal level
Primary actor: Writer
Main success scenario:

- Login
- Add desired content into the text areas
- Click button to create article

Extensions: Quit creation of article

3. System Architectural Design

3.1 Architectural Pattern Description

The application will be a client-server application.

Client-server architecture is a producer/consumer computing architecture where the server acts as the producer and the client as a consumer. The server houses and provides high-end, computing-intensive services to the client on demand. These services can include application access, storage, file sharing, printer access and/or direct access to the server's raw computing power.

Client-server architecture works when the client computer sends a resource or process request to the server over the network connection, which is then processed and delivered to the client. A server computer can manage several clients simultaneously, whereas one client can be connected to several servers at a time, each providing a different set of services. In its simplest form, the internet is also based on client/server architecture where web servers serve many simultaneous users with website data.

Also an MVC pattern was implemented in the client side for the presentation layer.
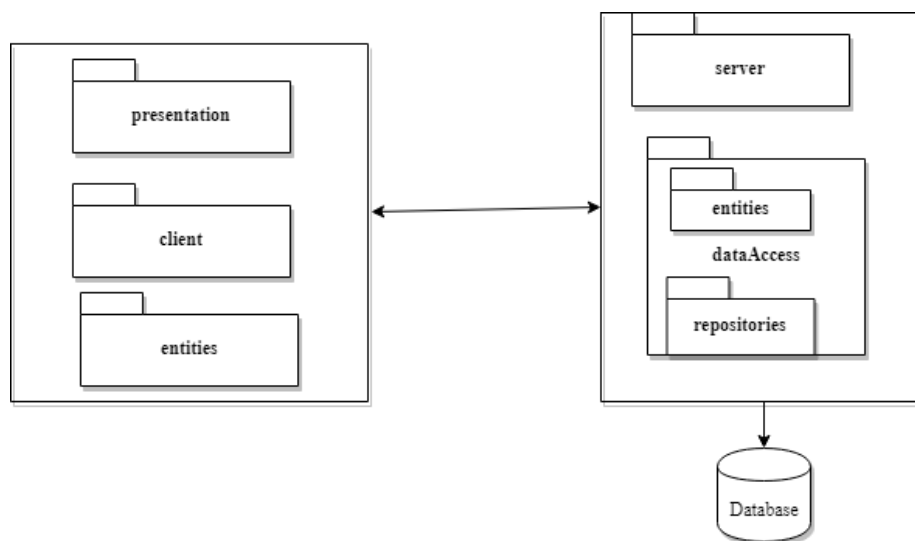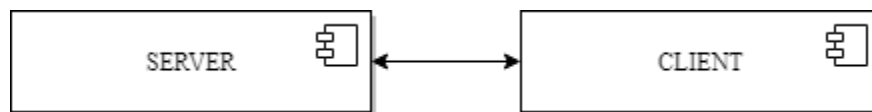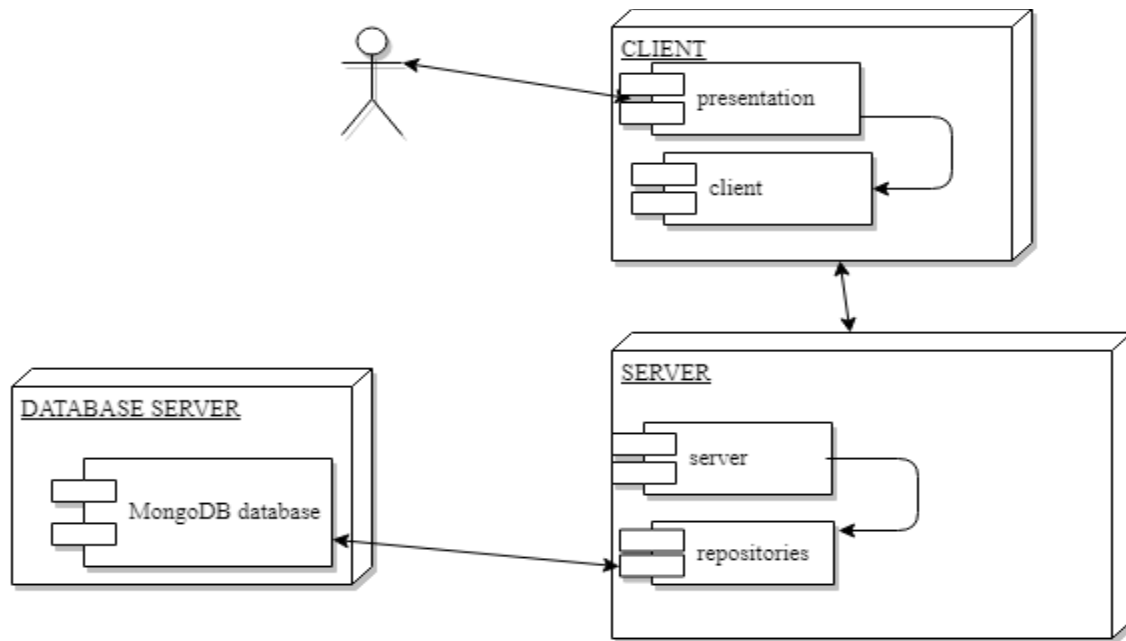

3.2 Diagrams

1. Package Diagram
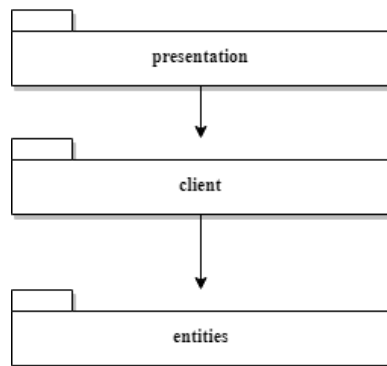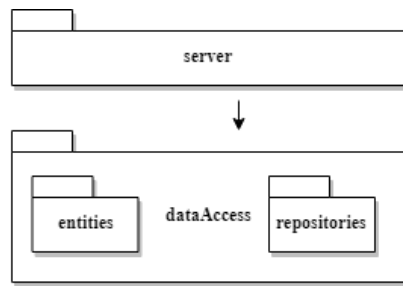
2. Component Diagram

3. Deployment Diagram

4. UML Sequence Diagrams

*[Create a sequence diagram for a relevant scenario.]*
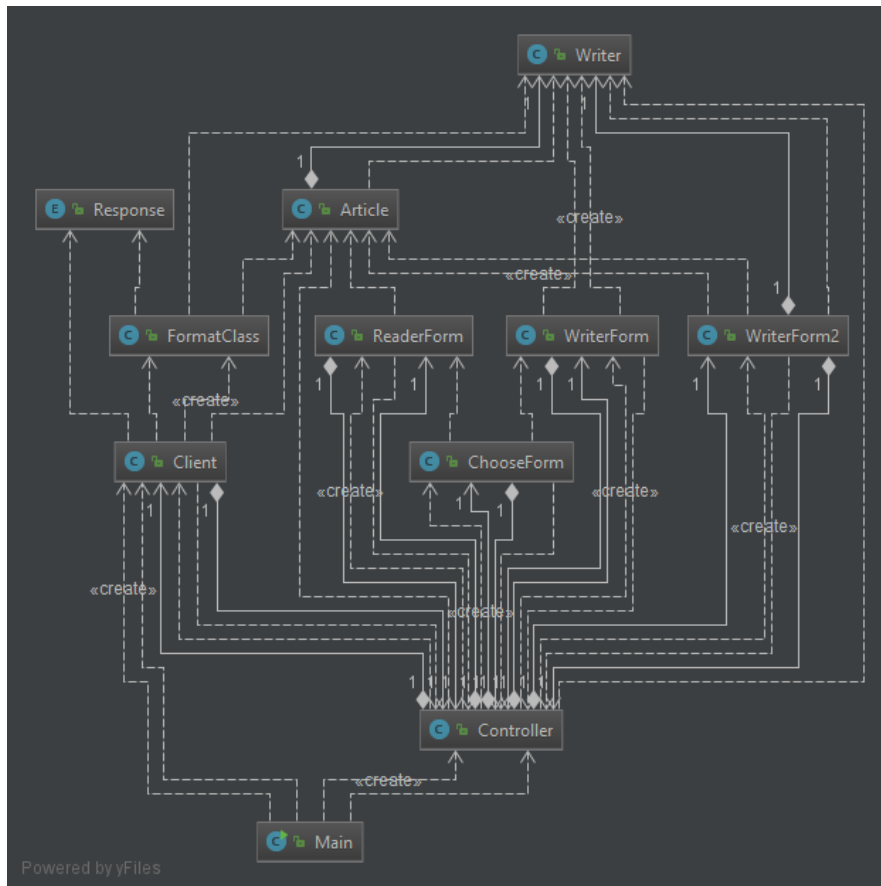
5. Class Design

5.1 Design Patterns Description

The Observer Pattern is used for this application in order to have real updates. The action of adding or deleting articles by writers must be notified to the readers in real time and the readers must see the refreshed list of articles.

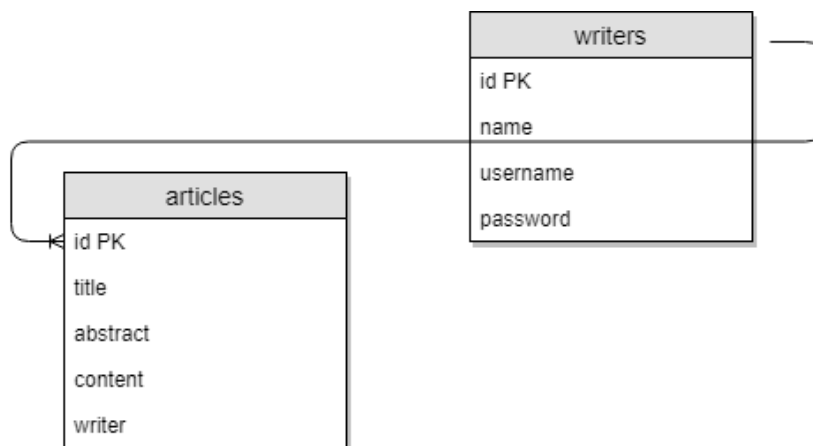## 5.2 UML Class Diagram

Class Diagram Server
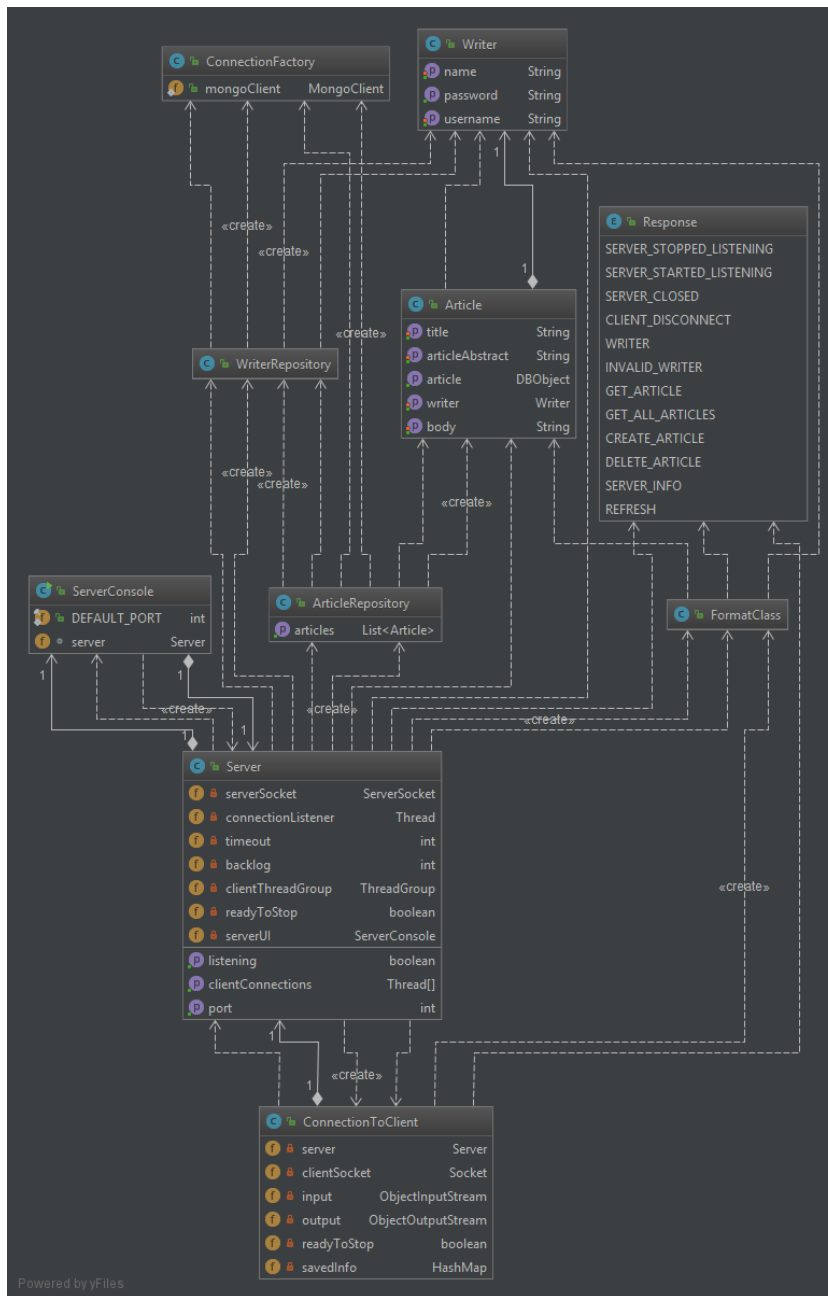
Class Diagram Client

6. Data Model

As Database Management System it has been used MongoDB.

8. Bibliography