Assignment 3
Analysis and Design Document

Student: Danila Lucia-Diana
Group: 30433

Table of Contents

1. Requirements Analysis

1.1 Assignment Specification

Use Java/C# API to design and implement a client-server application for a news agency. The application has three types of users: the readers, the writers and an administrator. 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. The **admin** is the only one who can create writer accounts, but cannot create new admin accounts. So the admin accounts are preset by the application developer and cannot be altered.

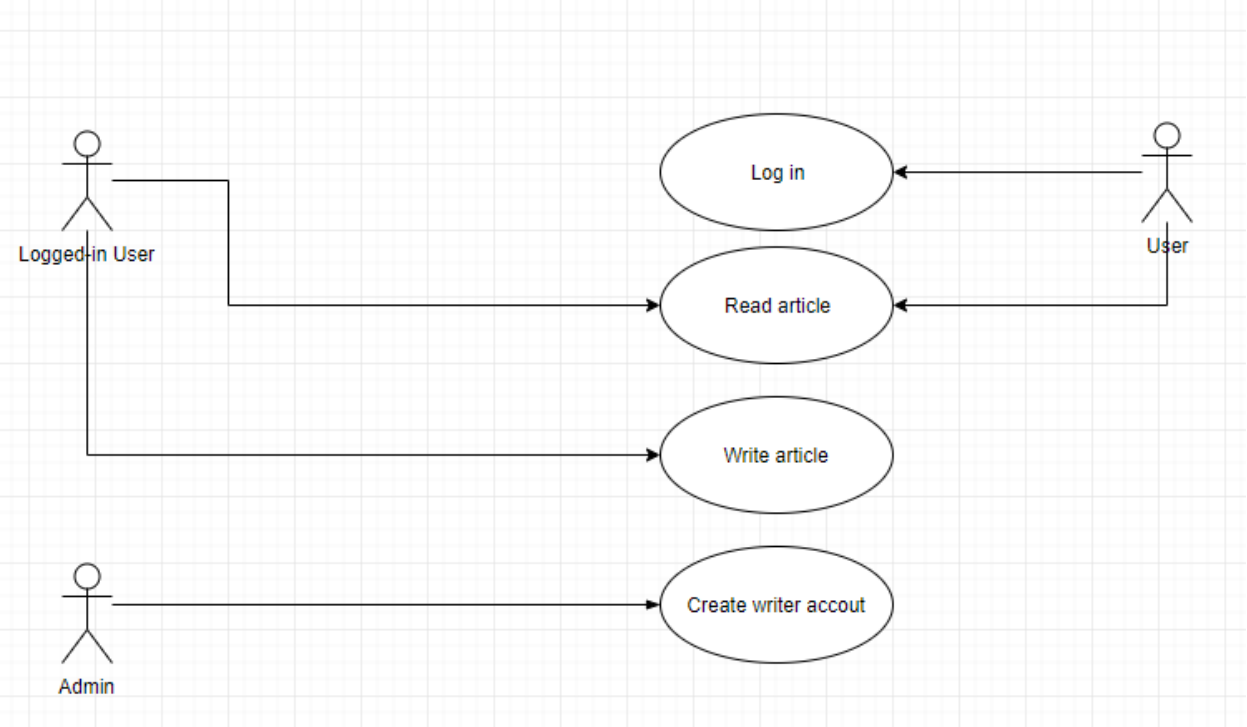
1.2 Functional Requirements

- an article has the following components: title, abstract, body and author, list of related articles
- writer must login to write articles
- any user can read articles
- only the admin can create writer accounts

1.3 Non-functional Requirements

- The application must be client-server.
- Use the Observer design pattern for updating the list of articles in real time
- For sending data from the client to the server use JSON serialization.
- When writing an article, show a list that supports multi-select for choosing the related articles.

2. Use-Case Model



Use case: read article
Level: user-goal level
Primary actor: user (not logged in)
Main success scenario: *article is read*
Extensions: *the application runs succesfully*

3. System Architectural Design

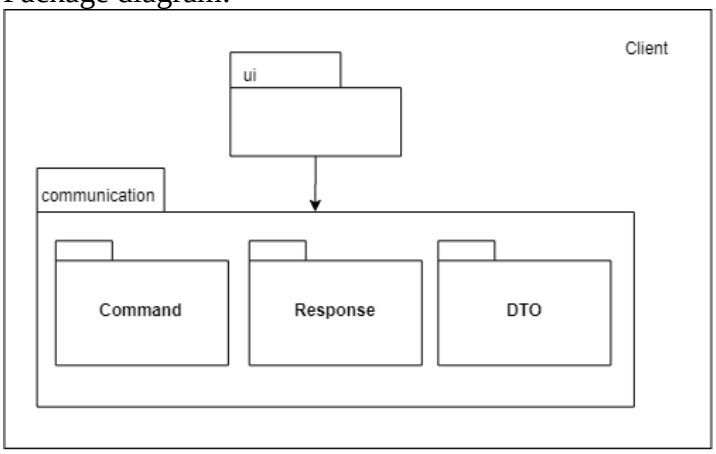
3.1 Architectural Pattern Description

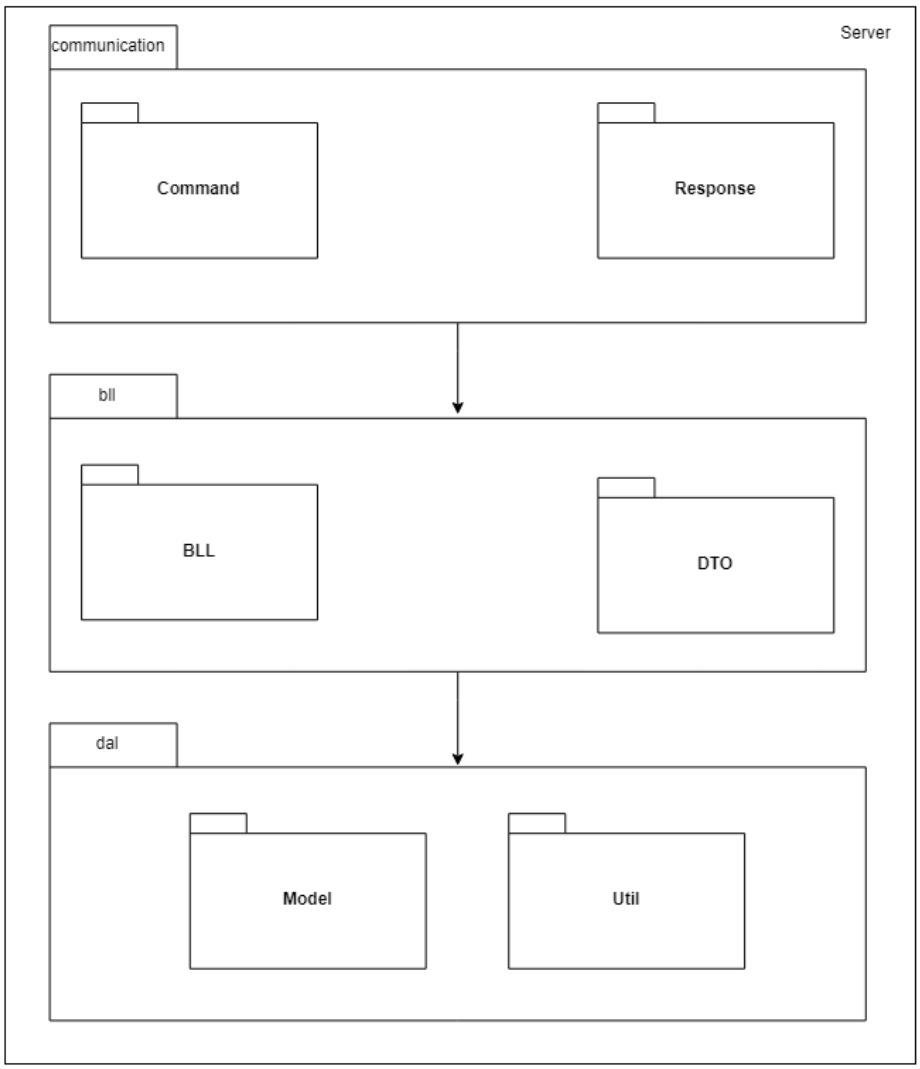
Client/server architecture is a computing model in which the server hosts, delivers and manages most of the resources and services to be consumed by the client. This type of architecture has one or more client computers connected to a central server over a network or internet connection. This system shares computing resources.

Client/server architecture is also known as a networking computing model or client/server network because all the requests and services are delivered over a network.

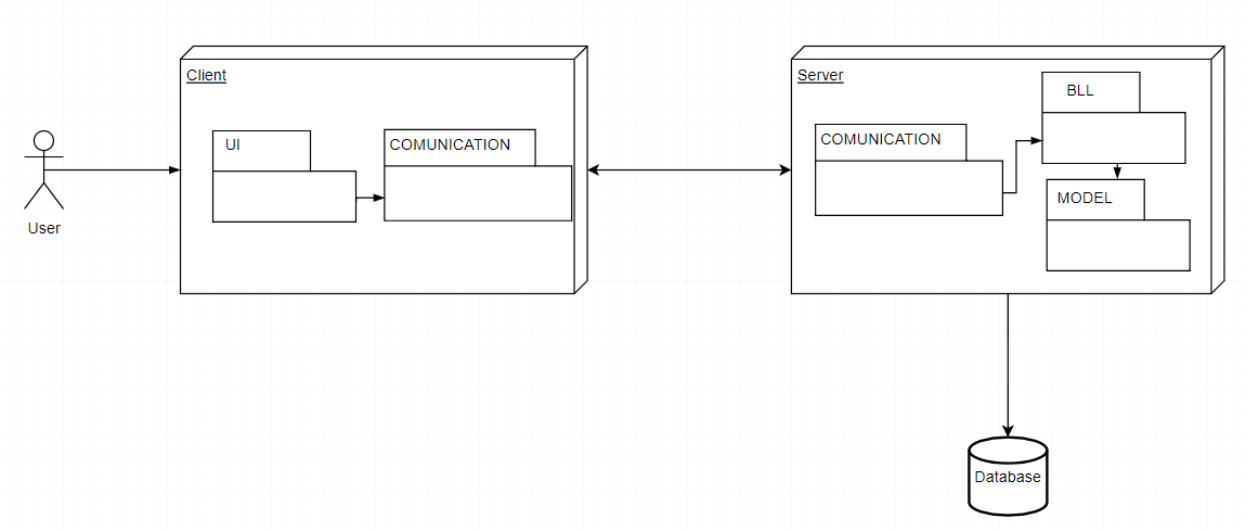
3.2 Diagrams

Package diagram:

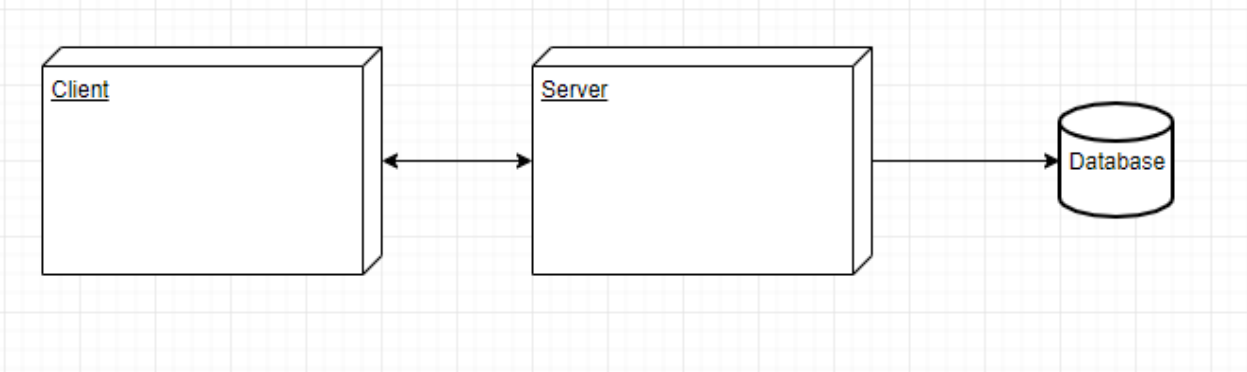




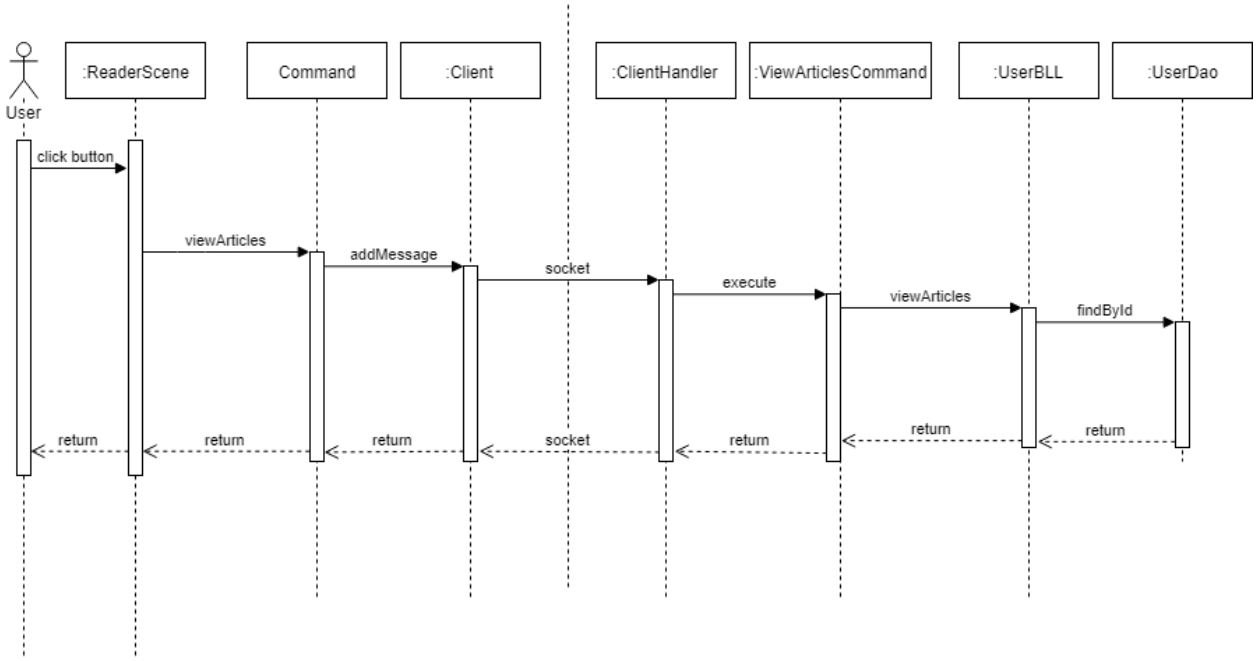
Component diagram:



Deployment diagram:



4. UML Sequence Diagrams



5. Class Design

5.1 Design Patterns Description

Design Patterns Description

The used design Patterns are Client-Server, DAO, and Table Module.

Dao: access to data varies depending on the source of the data. Access to persistent storage, such as to a database, varies greatly depending on the type of storage (relational databases, object-oriented databases, flat files, and so forth) and the vendor implementation.

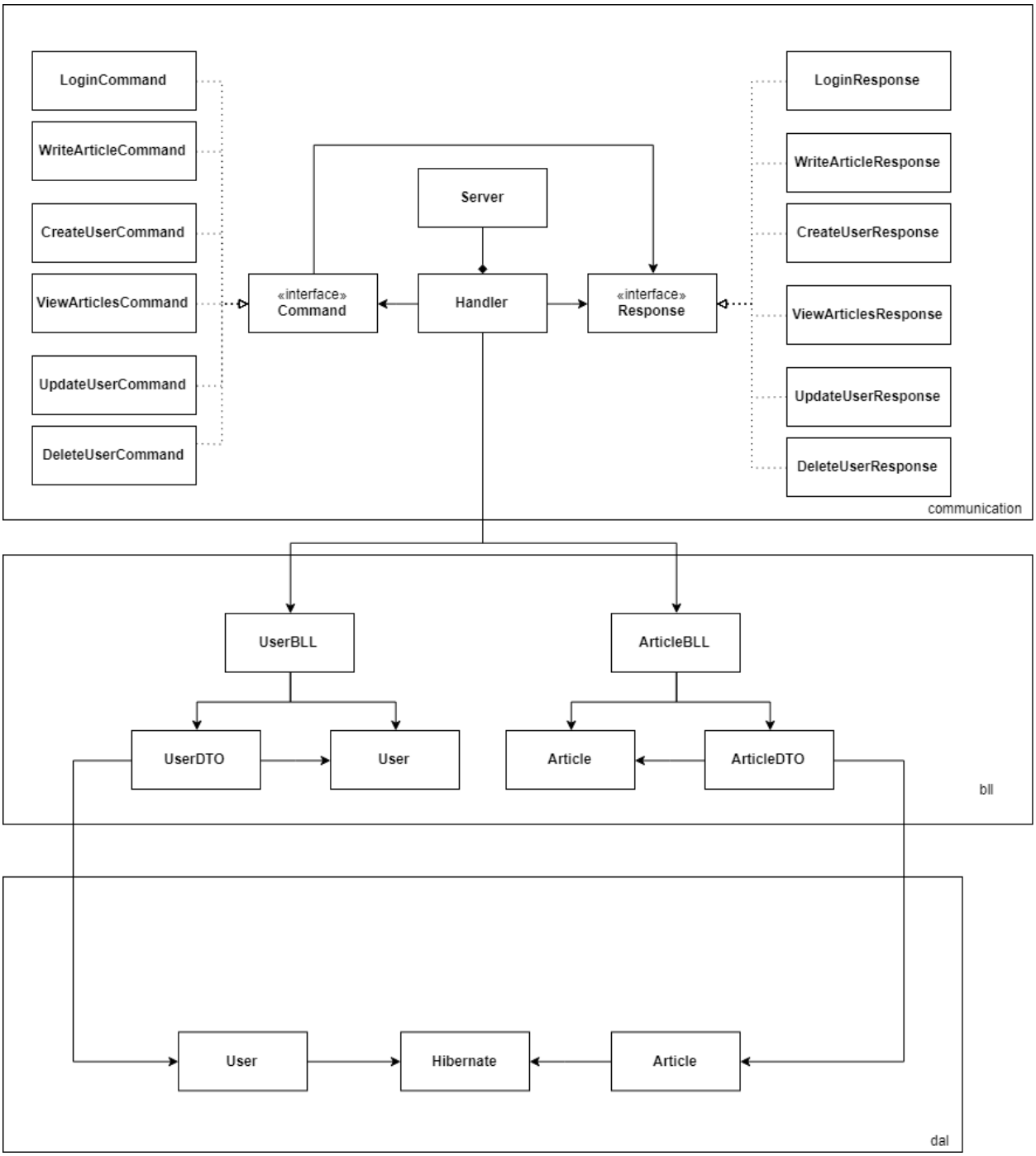
Table Module: A single instance that handles the business logic for all rows in a database table or view.

Client/server architecture is a computing model in which the server hosts, delivers and manages most of the resources and services to be consumed by the client. This type of architecture has one or more client computers connected to a central server over a network or internet connection. This system shares computing resources.

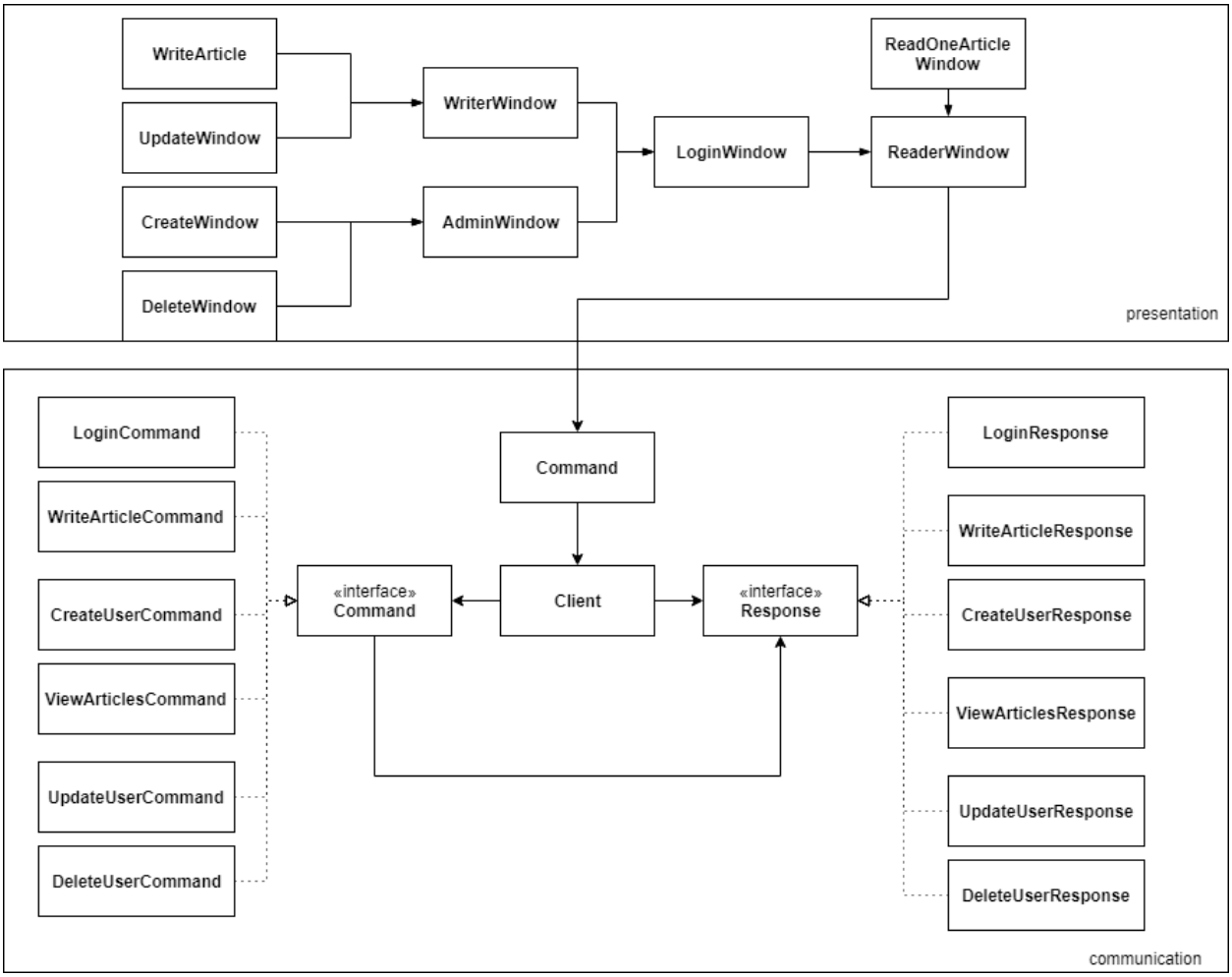
Client/server architecture is also known as a networking computing model or client/server network because all the requests and services are delivered over a network.

5.2 UML Class Diagram

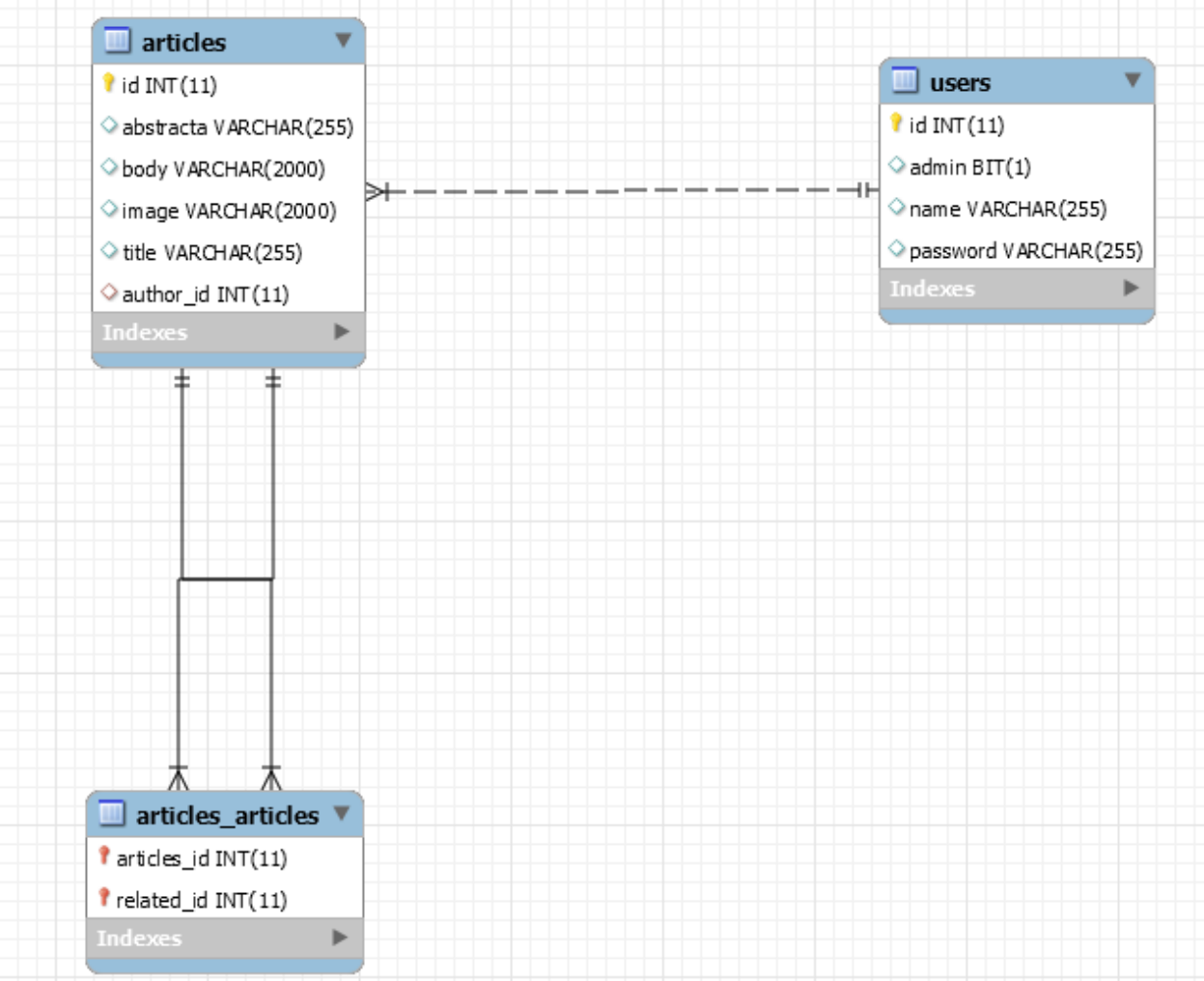
Server:



Client:

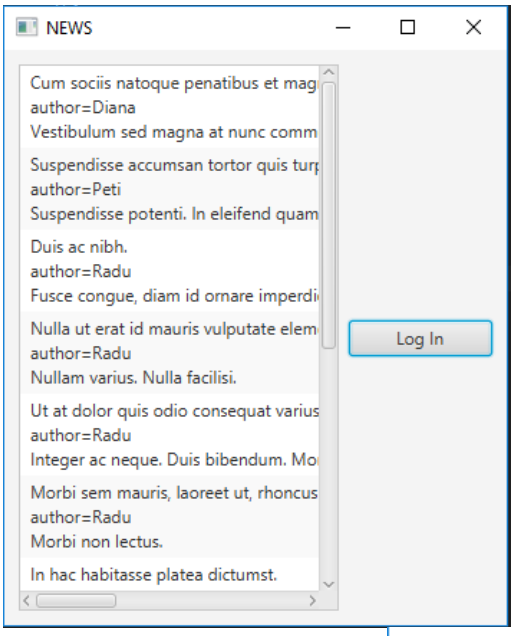

WriteArticle
UpdateWindow
CreateWindow
DeleteWindow
WriterWindow
AdminWindow
LoginWindow
ReadOneArticle Window
ReaderWindow
presentation
LoginCommand
WriteArticleCommand
CreateUserCommand
ViewArticlesCommand
UpdateUserCommand
DeleteUserCommand
Command
«interface» Command
Client
«interface» Response
LoginResponse
WriteArticleResponse
CreateUserResponse
ViewArticlesResponse
UpdateUserResponse
DeleteUserResponse
communication

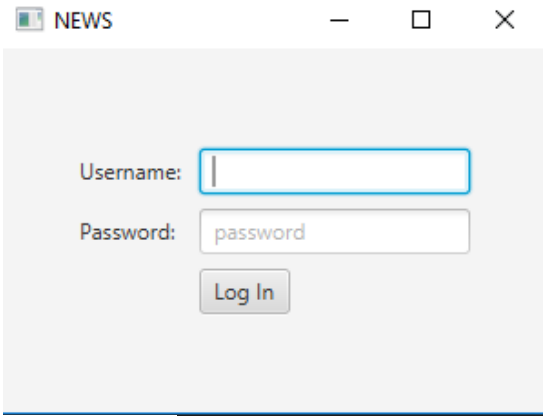
6. Data Model

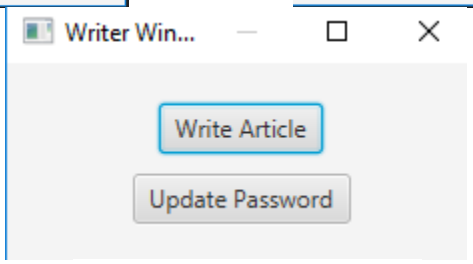


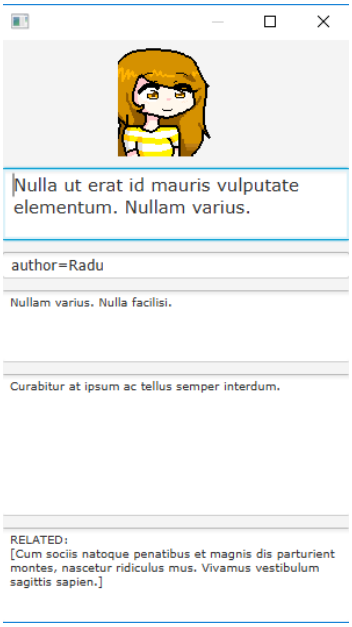
7. System Testing

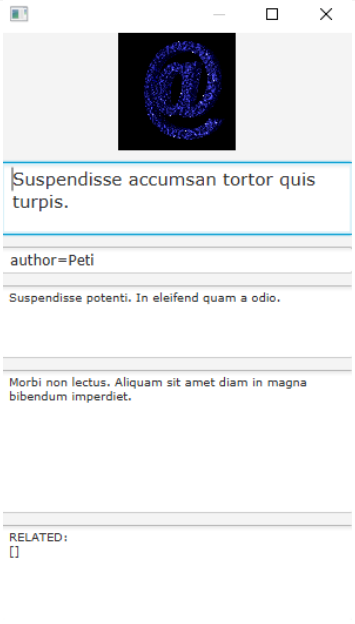
The system was tested and the errors were solved with system.out.prinln().
Some print screens from using the application are:

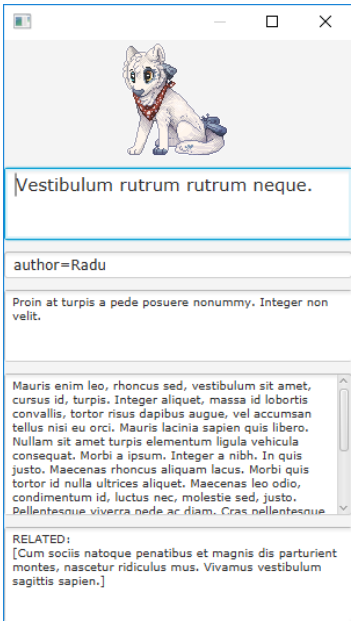

NEWS
Cum sociis natoque penatibus et mag
author=Diana
Vestibulum sed magna at nunc comm
Suspendisse accumsan tortor quis turp
author=Peti
Suspendisse potenti. In eleifend quam
Duis ac nibh.
author=Radu
Fusce congue, diam id ornare imperdi
Nulla ut erat id mauris vulputate elem
author=Radu
Nullam varius. Nulla facilisi.
Ut at dolor quis odio consequat varius
author=Radu
Integer ac neque. Duis bibendum. Mo
Morbi sem mauris, laoreet ut, rhoncus
author=Radu
Morbi non lectus.
In hac habitasse platea dictumst.
Log In

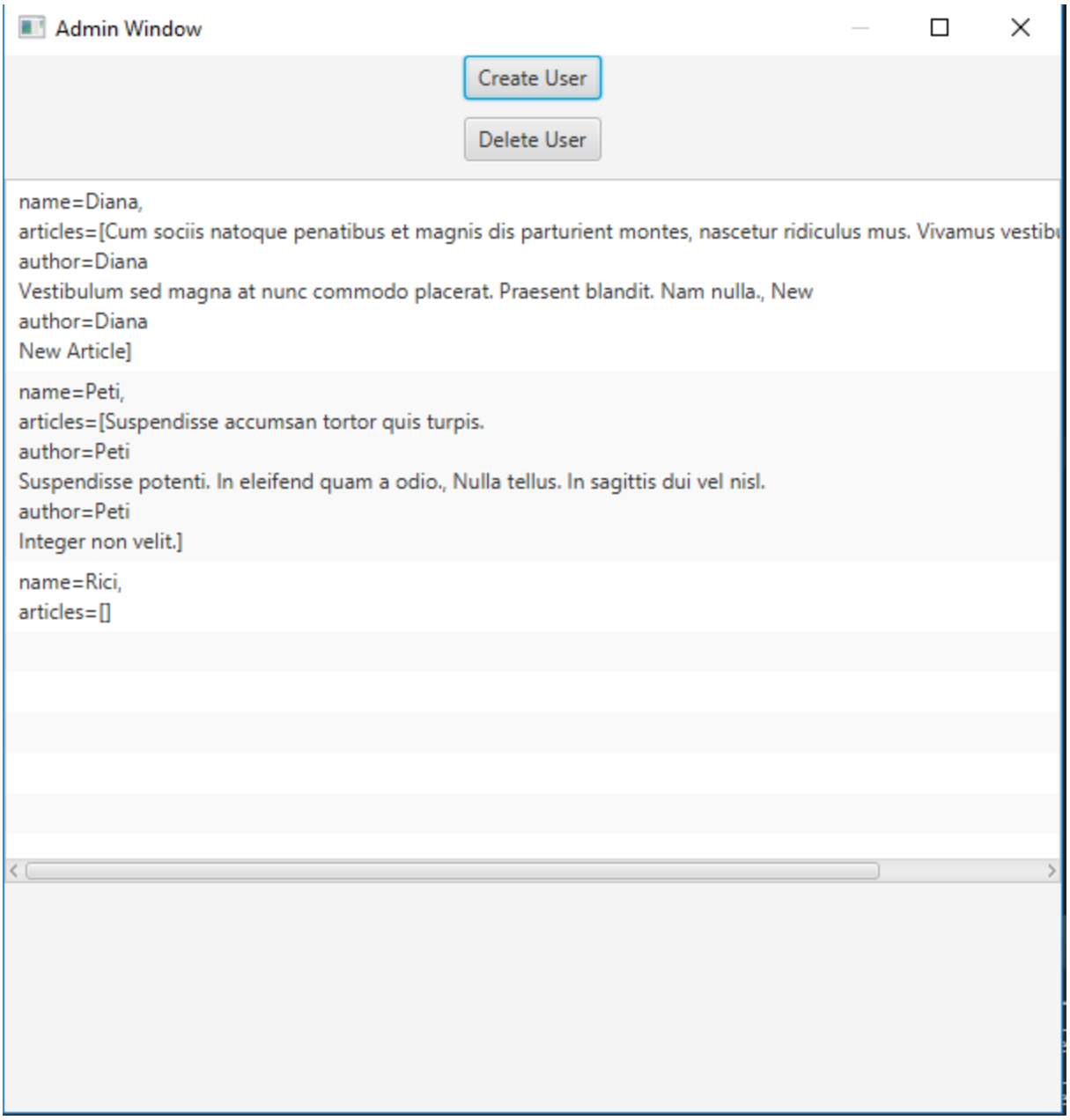

NEWS
Username:
Password:
password
Log In


Writer Win...
Write Article
Update Password


Nulla ut erat id mauris vulputate elementum. Nullam varius.
author=Radu
Nullam varius. Nulla facilisi.
Curabitur at ipsum ac tellus semper interdum.
RELATED:
[Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Vivamus vestibulum sagittis sapien.]


Suspendisse accumsan tortor quis turpis.
author=Peti
Suspendisse potenti. In eleifend quam a odio.
Morbi non lectus. Aliquam sit amet diam in magna bibendum imperdiet.
RELATED:
[]


Vestibulum rutrum rutrum neque.
author=Radu
Proin at turpis a pede posuere nonummy. Integer non velit.
Mauris enim leo, rhoncus sed, vestibulum sit amet, cursus id, turpis. Integer aliquet, massa id lobortis convallis, tortor risus dapibus augue, vel accumsan tellus nisi eu orci. Mauris lacinia sapien quis libero. Nullam sit amet turpis elementum ligula vehicula consequat. Morbi a ipsum. Integer a nibh. In quis justo. Maecenas rhoncus aliquam lacus. Morbi quis tortor id nulla ultrices aliquet. Maecenas leo odio, condimentum id, luctus nec, molestie sed, justo.
RELATED:
[Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Vivamus vestibulum sagittis sapien.]



8. Bibliography

- https://www.techopedia.com/definition/438/clientserver-architecture
- https://www.google.ro/search?rlz=1C1OKWM_enRO775RO775&tbm=isch&q=pictures+100x100+pixels&chips=q:pictures+100x100+pixels,online_chips:100x100+gif&sa=X&ved=0ahUKEwjUtMKW74DbAhUL16QKHVRVDSQQ4lYIJygB&biw=1920&bih=900&dpr=1#imgrc=r0MtwRMuExiDtM:
- https://stackoverflow.com/questions/37111582/simplest-way-to-add-an-image-in-javafx
- https://www.sqlservercentral.com/Forums/Topic625247-145-1.aspx
- http://cs.lmu.edu/~ray/notes/javanetexamples/
- https://docs.oracle.com/javase/tutorial/networking/sockets/clientServer.html