

ASSIGNMENT A3

1. Objective

The objective of this assignment is to allow students to become familiar with the client-server architectural style and the Observer design pattern.

2. Application Description

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 to 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.

An article has the following components:

- Title
- Abstract
- Author
- Body
- List of related articles

When reading an article the user should be able to see the title and the abstract of the related articles. By clicking on the title of the related article, he will be taken to a page that displays the full article.

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.

Bonus points: If you can include pictures in the articles you get 1 bonus point.

3. Application Constraints and Technical 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.

4. Deliverables

In the same Github repository as Assignment 1 and Assignment 2, add a new folder with the following files:

- Analysis and design document.
- Implementation source files.
- Readme file that describes the installation process of the application and how to use it:
 - how to install your application on a clean computer
 - how to access your application and with what users
 - images with all use cases and their scenarios implemented

5. Grading

Grade	Functionality
5	Analysis and design document Client-Server Architecture Json Serialization for data sent between client and server
6	View List of Articles Writer Login Write Simple Article
7	Open Article Articles Update in real-time (Observer Design Pattern)
8	Article with references to other articles
9	Quality of Implementation and Documentation (Validated Input Data + Code + UI)
10	Persist Articles Persist Writer Accounts CRUD on Writer Accounts

6. References

<http://download.oracle.com/javase/tutorial/networking/sockets/index.html>

https://docs.oracle.com/javafx/2/get_started/jfxpub-get_started.htm

https://www.tutorialspoint.com/jackson/jackson_first_application.htm

https://en.wikipedia.org/wiki/Composite_pattern

https://en.wikipedia.org/wiki/Observer_pattern

[https://msdn.microsoft.com/en-us/library/54xbah2z\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/54xbah2z(v=vs.110).aspx)

[https://msdn.microsoft.com/en-us/library/e80y5yhx\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/e80y5yhx(v=vs.110).aspx)