<Assignment 3>

Analysis and Design Document

Student:Bianca-Andreea Tusa

**Group:302310**

Table of Content

1. Requirements Analysis 3

1.1 Assignment Specification 3

1.2 Functional Requirements 3

1.3 Non-functional Requirements 3

2. Use-Case Model 3

3. System Architectural Design 3

4. UML Sequence Diagrams 3

5. Class Design 3

6. Data Model 3

7. System Testing 3

8. Bibliography 3

1. Requirements Analysis Chira2018

# 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.

# 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 tocreate, 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. The admin also can perform CRUD on categories.

.

# Non-functional Requirements

Modified data in a database should be updated for all users accessing in real time.

2. Use-Case Model

[Create the use-case diagrams and provide one use-case description (according to the format below).

Use-Case description format:

Use case: creating a new category for articles

Level: <one of: user-goal level

Primary actor: admin user

Main success scenario: access the data-base, introducing all the required parameters for creating a new category(id, name).

Extensions: : if the information is incorrect, an error will appear.

]

3. System Architectural Design

**3.1 Architectural Pattern Description**

**Client-Server:**

The client–server model is a distributed application structure that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients. Often clients and servers communicate over a computer network on separate hardware, but both client and server may reside in the same system. A server host runs one or more server programs which share their resources with clients.

**Layered Architecture:**

The most common architecture pattern is the layered architecture pattern, otherwise known as the n-tier architecture pattern. This pattern is the de facto standard for most Java EE applications and therefore is widely known by most architects, designers, and developers.

**3.2 Diagrams**

*[Create the system’s conceptual architecture; use architectural patterns and describe how they are applied. Create package, component and deployment diagrams]*

4. UML Sequence Diagrams

Scenario: View request (User Interface) -> Processing request (Manager User)-> List All Articles -> (Data Base) -> View All (User Interface).

5. Class Design

**5.1 Design Patterns Description**

*[Describe briefly the used design patterns.]*

**5.2 UML Class Diagram**

*[Create the UML Class Diagram and highlight and motivate how the design patterns are used.]*

6. Data Model

Users: Admin, Regular User, Writer.. Admin and Writer -> CRUDs and other actions. Regula User can View.

7. Bibliography

<https://stackoverflow.com/>

https://www.tutorialspoint.com/index.htm