**HANOI UNIVERSITY**

**Faculty of Information Technology**

**61FIT3SAD**

**SYSTEM ANALYSIS & DESIGN**

**ACTIVITY REPORT**

| **Faculty:** | **Information Technology** |
| --- | --- |
| **Module Code:** | **61FIT3SAD** |
| **Module Name:** | **System Analysis & Design** |
| **Year:** | **Fall 2021** |
| **Topic:** | **Online Learning Platform** |
| **Group:** | **9** |
| **Group members** | **Nguyễn Tuấn Nghĩa (1901040146)** |
|  | **Nguyễn Như Hồng Phúc (1901040159)** |
|  | **Lê Hồng Hải (1901040069)** |
|  | **Bùi Hoàng Quân (1901040166)** |
| **Class:** | **8C19 + 7C19** |

**TABLE OF CONTENTS**

ABSTRACT 3

GROUP ROLES 4

1. Business Modeling (**Nguyễn Tuấn Nghĩa**) 5

2. Requirements Definition **(Nguyễn Như Hồng Phúc**) 6

3. Prototype **(Lê Hồng Hải**) 7

4. Analysis & Design **(Bùi Hoàng Quân**) 8

5. Conclusion 11

**ABSTRACT**

*Human learning is improving as a result of the rapid advancement of computer science and technology. Because of their applicability and convenience, online courses are progressively displacing traditional courses. In this article, we propose an online course management system - Google Classroom is a promising system with extensive online course administration capabilities, including course management, user management, report production, and viewing, among other things. To determine the business context, capture and clearly characterize system requirements, build architectural design, and data design for the Online Course Management System, our project uses the Rational Unified Process (RUP). The project's output will contain documentation from three processes: business modeling, requirements definition, and analysis and design, as well as UML model artifacts for the system in Visual Paradigm and Draw.io, and a set prototype image of the functional system.*

|  | **GROUP ROLES** | |
| --- | --- | --- |
|  |  |  |

| **Members** | **Roles** |
| --- | --- |
| Nguyễn Tuấn Nghĩa (1901040146) | Business Designer |
| Nguyễn Như Hồng Phúc (1901040159) | Requirements Specifier |
| Lê Hồng Hải (1901040069) | Prototype |
| Bùi Hoàng Quân (1901040166) | Software Architecture, Designer, Database Designer |

**1. Business Modeling (Nguyễn Tuấn Nghĩa)**

Nghia is the company's business designer and process analyst. He is in charge of the company's business architecture. He describes the workflow of one or more business use cases, as well as the duties, activities, qualities, and relationships of one or more business employees and business entities, in order to specify a component of the organization. He also leads and manages business use-case modeling, which entails sketching and delimiting the organization being modeled, such as determining what business actors and business use cases exist and how they interact.

1. **Requirements Definition (Nguyễn Như Hồng Phúc)**

The Requirements Specifier (Phuc) is a program that allows you to specify requirements. He details certain needs of numerous use-cases and other system requirements, giving us a thorough perspective of the definition of a portion of the system's capabilities. He also examines and summarizes the use-case and extra needs requirements elements. She is also responsible for the use-case package and maintenance. He's also someone who documents all of the system's needs.

**3. Prototype (Lê Hồng Hải)**

Hai is in charge of putting the prototype together to show how the Thinkific Online Course Management System works. Prototypes are based on genuine photos from the system's operation. provides a functioning Graphical User Interface for system tasks including registration, login, course management, student management, and so on. The prototype's objective is to illustrate the system's functioning.

Link to online system Google Classroom: [Classes (google.com)](https://classroom.google.com/u/0/h)

Accounts for testing the prototype:

| Email | Password | Account Type |
| --- | --- | --- |
| Lehonghai123@gmail.com | haile2021@ | User |

1. **Analysis & Design (Bùi Hoàng Quân)**

Quan worked as a software architect, studying systems and their components to identify their objectives. His responsibilities include problem-solving approaches that improve the system and ensure that all of its components work together to meet its objectives. He also investigated the process of developing a new business system or updating an existing one by specifying its components or modules in order to fit specific requirements. He goes through the prior system in detail and evaluates how computers may be used most effectively to increase productivity.

| **Activity** | **Description** | **Output Artifacts** |
| --- | --- | --- |
| Prioritize Use Case | To determine technical content and the order of following iterations, choose a defined number of scenarios and use cases to investigate and produce.  Based on staff availability, the various development teams completed and polished the project.  Define client expectations for deliverables, tool and commercial-off-the-shelf (COTS) product availability, and other project needs. | Software Architecture Document |
| Architectural Analysis | Define the system's prospective architecture as well as the architectural approaches that will be used.  To restrict and concentrate the design, gather knowledge from similar systems or issue areas, so that time and effort are not squandered on architectural rediscovery.  Define the architecture; architectural analysis is especially crucial when creating new and unusual systems. | Use-Case Realization, Deployment Model, Software Architecture Document, Design Model |
| Identify Design Mechanisms | Determine who each analysis mechanism's clients are.  Determine the characteristics of each analytical technique.  Clients are divided into groups based on how they utilize characteristic profiles. | Software Architecture Document, Deployment Model |
| Incorporate Existing Design Elements | Find interfaces, design classes, and design subsystems by analyzing the interactions of analysis classes.  Refine the architecture whenever practical, embracing reuse.  Identify typical solutions to design issues that arise frequently.  Include architecturally relevant design model elements in the Software Architecture Document's Logical View section. | Design Model, Software Architecture Document |
| Describe Distribution | Analyze the interactions of analysis classes to find interfaces, design classes, and design subsystems.  When possible, refine the architecture while accepting reuse.  Identify conventional solutions to regularly occurring design difficulties.  Include architecturally important design model elements in the Logical View section of the Software Architecture Document. | Software Architecture Document, Deployment Model |
| Identify Design Elements | Represent a set of responsibilities with a lot of subtlety.  Represent a set of coarse-grained responsibilities, which may be made up of other subsystems but are ultimately a set of classes.  The control threads of the system should be reflected.  Express abstract assertions about the duties of a class or a subsystem. | Design Model |

Quan's job as a designer include defining the responsibilities, actions, attributes, and connections of a variety of classes, as well as determining how they will be customized for the Online Course Management System's implementation environment. The table below shows his actions as a designer in the Analysis & Design workflow.

| **Activity** | **Description** | **Output Artifacts** |
| --- | --- | --- |
| Use-Case Analysis | Creating a system from the users perspective Communicating system activity in language that the user understands Defining all behaviors that are observable from the outside | Analysis class, Use-Case Realization, Design Model, Analysis Model |
| Use-Case Design | Figure out who will be using the website.  Select a user from the list.  Define the user's goals for visiting the site. A use case is created for every activity a user performs on the site. | Use-Case Realization |
| Class Design | Ascertain that the class has the desired behavior for the use-case realizations.  Ascertain that sufficient information is provided in order to implement the class without ambiguity.  Maintaining the class's non-functional requirements requires including the class's design mechanisms. | Design Class |

As a database designer, Quan's major role is to establish the tables, indexes, views, constraints, and other database-specific structures needed to store, retrieve, and delete persistent objects. Her actions as a database designer are detailed in the Analysis & Design workflow in the table below.

| **Activity** | **Description** | **Output Artifacts** |
| --- | --- | --- |
| Database Design | Design databases are easy to maintain, improve data consistency, and save money on disk storage. The database designer decides how data items are related to one another and what data must be saved. | Data Model |

**5. Conclusion**

We assessed a management system using the information we gained from this course (69FIT3SAD) (Online Learning Platform). We modeled business context for the system in our project, collecting and defining system requirements in both textual and UML notation. Based on the MVC paradigm, we offer an analysis and design system. To show system requirements, we try to utilize various diagrams such as use-case diagrams, class diagrams, package diagrams, and sequence diagrams. We also go through the system's flow in great detail and make it simple to grasp. However, the project still has some unforeseen twists and turns. Because of our poor knowledge and lack of expertise in System Analysis and Design. As a result, we will continue to expand our expertise in order to provide the greatest product possible in the future.