|  |  |  |
| --- | --- | --- |
|  | HANOI UNIVERSITYFaculty of Information Technology |  |

61FIT3SADSYSTEM ANALYSIS & DESIGNFINAL REPORT

**Faculty: Information Technology**

**Module Code: 61FIT3SAD**

**Module Name: System Analysis & Design**

**Year: Fall 2020**

**Topic: Food Management System**

**Group:**

**Group members: Nguyễn Thị Ngọc Ánh (1801040015)**

**Nguyễn Thu Trang (1801040226)**

**Nguyễn Thị Thảo (1801040209)**

**Class: 2C18-8C18**

Table of Contents

[ABSTRACT 3](#_Toc60355269)

[GROUP ROLES 4](#_Toc60355270)

[1. Business Modeling (Nguyễn Thị Thảo) 5](#_Toc60355271)

[2. Requirements Definition (Nguyễn Thu Trang) 6](#_Toc60355272)

[3. Prototype (Nguyễn Thị Ngọc Ánh) 7](#_Toc60355273)

[4. Analysis & Design (Nguyễn Thị Ngọc Ánh & Nguyễn Thu Trang) 8](#_Toc60355274)

[5. Conclusion 9](#_Toc60355275)

# ABSTRACT

For over ten years, online ordering has been one of the leading drivers of growth in the restaurant industry.Tech savvy restaurants understand the online food ordering system and its potential to drive more customer engagement and yield reliable growth. Customers, using their computer or smartphone, access a restaurant’s menu online, make their selections, and place an order. By the time a customer arrives at the restaurant, their food is packaged and ready for pickup.In more recent years, the increased use of mobile devices has significantly benefited restaurants as more customers embrace the ability to place orders on the go.The growing popularity of online food ordering is boosted by some very obvious benefits for both restaurants and customers.. *The goal of our project is to follow the Rational Unified Process (RUP) to capture the business context of the system, collect and specify requirement for the system, analysis the requirements to provide an architectural design solution for the Food Management System. The results of our works consist of system documentations for three workflows: business modeling, requirements and analysis & design; Unified Modeling Language (UML) model artifacts (using Visual Paradigm); and a runnable prototype of the system.*

# GROUP ROLES

|  |  |
| --- | --- |
| **Members** | **Roles** |
| Nguyễn Thị Ngọc Ánh (1801040015) | Software Architecture, Database Designer |
| Nguyễn Thu Trang (1801040226) | Requirements Specifier, Designer |
| Nguyễn Thị Thảo (1801040209) | Business Designer, Business Process Analyst |

# Business Modeling (Nguyễn Thị Thảo)

Thảo is the business designer and business process analyst. She is responsible for the  
business architecture. She details the specification of a part of the organization by describing the workflow of one or several business use cases, along with defining the responsibilities, operations, attributes, and relationships of one or several business workers and business entities. She also leads and coordinates business use-case modeling by outlining and delimiting the organization being modeled; for example, establishing what business actors and business use cases exists and how they interact.

# Requirements Definition (Nguyễn Thu Trang)

Trang is the requirements specifier. She detailed the specification of a part of the  
system's functionality by describing the Requirements aspect of one or several use cases and other supporting software requirements. She is also responsible for the use-case package, and maintains the integrity of that package. She detailed the use cases and the supplementary requirements and made them consistent with other requirements discipline artifacts. Besides, she captured requirements on the user interface, including usability requirements.

# Prototype (Nguyễn Thị Ngọc Ánh)

Ánh have implemented a prototype to demonstrate the Food Management System functionality using HTML, CSS & JavaScript (with additional libraries and frameworks such as Bootstrap, …). The prototype is provided a simple Graphical User Interface and have some function of the system specified in requirements definition such as sign-in, display computers, add computers, …The prototype is to demonstrate the system functionality; therefore, there are no server-side code and dedicated database yet. Also, in the prototype, models and controllers are putted into a single file. To ease the demonstration, runtime variables are used to store data objects (products, users, orders, …) instead of a dedicated database; thus, changes will not be saved permanently.

Account for testing prototype:

|  |  |  |
| --- | --- | --- |
| **Username** | **Password** | **Account Type (User)** |
| admin | 123456789 | Admin |
| Raincute31 | ABCDEFGH | Customer |

# Analysis & Design (Nguyễn Thị Ngọc Ánh & Nguyễn Thu Trang)

As a software architecture, Ánh leads and coordinates technical activities and artifacts throughout the project. Her primary responsibility is to establish the overall structure for each architectural view: the decomposition of the view, the grouping of elements. Therefore, she has a breadth view of the system architecture. The table below describes detailed her activities as a software architecture in the Analysis & Design workflow.

|  |  |  |
| --- | --- | --- |
| **Activity** | **Description** | **Output Artifacts** |
| Prioritize Use Cases | Define input to the selection of the set of scenarios and use cases that are to be analyzed in the current iteration.  Define the set of scenarios and use cases that represent some significant, central functionality.  Define the set of scenarios and use cases that have a substantial architectural coverage or that stress or illustrate a specific, delicate point of the architecture. | Software Architecture Document |
| Architectural  Analysis | Define a candidate architecture for the system, based on experience gained from similar systems or in similar problem domains.  Define the architectural patterns, key mechanisms and modeling conventions for the system.  Define the reuse strategy. | Use-Case Realization,  Deployment Model,  Software Architecture  Document, Design Model |
| Identify Design  Mechanisms | Refine the analysis mechanisms into design mechanisms based on the constraints imposed by the implementation environment. | Design Model, Software Architecture Document |
| Incorporate  Existing Design  Elements | Analyze interactions of analysis classes to find design classes.  Refine the architecture, incorporating reuse where possible.  Identify common solutions to commonly encountered design problems.  Include architecturally significant design model elements in the Logical  View section of the Software Architecture Document. | Design Model, Software Architecture Document |
| Describe  Distribution | Describe how the functionality of the system is distributed across physical nodes. | Software Architecture  Document, Deployment Model |
| Identify Design  Elements | Analyze interactions of analysis classes to identify design model elements. | Design Model |

As a designer, Trang is in charge of defines the responsibilities, operations, attributes, and

relationships of several classes, and determines how they will be adjusted to the implementation environment of the Food Management System. Ánh describes software architecture document which provides a comprehensive overview of the architecture of the software system. It serves as a communication medium between the software architect and other project team members regarding architecturally significant decisions which have been made on the project. Then, Trang describes all the fuctions by explaning use case belonged to this function and Ánh draws sequence diagram and class diagram for them. The table below describes detailed his activities as a designer in the Analysis & Design workflow.

|  |  |  |
| --- | --- | --- |
| **Activity** | **Description** | **Output** |
| Use-Case  Analysis | Identify the classes which perform a  use case’s flow of events.  Distribute the use case behavior to those classes, using use-case realizations.  Identify the responsibilities, attributes and associations of the classes.  Note the usage of architectural mechanisms. | Analysis class, Use-Case Realization, Design Model, Analysis Model |
| Use-Case  Design | Refine use-case realizations in terms of interactions.  Refine requirements on the operations of design classes. | Use-Case Realization |
| Class Design | Ensure that the class provides the behavior the use-case realizations require.  Ensure that sufficient information is provided to unambiguously implement  the class.  Handle non-functional requirements related to the class.  Incorporate the design mechanisms used by the class. | Design Class |

As a database designer, Ánh essential obligation is defining the tables, indexes, views, constraints, and other database-specific constructs needed to store, retrieve, and delete persistent objects. The table below describes detailed her activities as a database designer in the Analysis & Design workflow.

|  |  |  |
| --- | --- | --- |
| **Activity** | **Description** | **Output** |
| Database Design | Ensure that persistent data is stored consistently and efficiently.  Define behavior that must be implemented in the database. | Data Model |

# **Conclusion**

Based on knowledge acquired from this course and previous courses, we have designed a management system with requirement match real world business (Food Management System). In our project, we have modeled business context for the system, capture and describe system requirement using both textual and UML notation. Moreover, we have designed the system using Model-View-Controller models as a foundation; therefore, improve robustness and reusability of system components. Also, a runnable prototype is built based on the requirement to demonstrate system functionality. Nevertheless, there is still room for improvement in project. Because our knowledge base limit and inexperience in System Analysis & Design particularly and Information Technology in general, erroneous in our work are inevitable. Therefore, we will continue improve our knowledge to deliver better product in the future.