**ASSIGNMENT 1 FRONT SHEET**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing** | | |
| **Unit number and title** | Unit 30: Application Development | | |
| **Submission date** | 06/03/2024 | **Date Received 1st submission** | 06/03/2024 |
| **Re-submission Date** | 13/03/2024 | **Date Received 2nd submission** | 13/03/2024 |
| **Student Name** | Tran Duc Luong | **Student ID** | BH00108 |
| **Class** | IT0502 | **Assessor name** | Nguyen Thanh Trieu |
| **Student declaration**  I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice. | | | |
|  |  | **Student’s signature** | Luong |

**Grading grid**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| P1 | P2 | P3 | M1 | M2 | D1 |
|  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **❒ Summative Feedback: ❒ Resubmission Feedback:** | | |
| **Grade:** | **Assessor Signature:** | **Date:** |
| **Lecturer Signature:** | | |

Table of Contents

[INTRODUCTION 4](#_Toc161212590)

[CONTENTS 5](#_Toc161212591)

[(P3) Research the use of software development tools and techniques and identify any that have been selected for the development of this application. 5](#_Toc161212592)

[1. UML 5](#_Toc161212593)

[1.1 UML definition 5](#_Toc161212594)

[1.2 Some popular UML Diagrams 6](#_Toc161212595)

[1.2.1 Class diagram 6](#_Toc161212596)

[1.2.2 Use case diagram 7](#_Toc161212606)

[1.2.3 Activity diagram 8](#_Toc161212607)

[1.2.3 Component diagram 9](#_Toc161212609)

[1.3 Use the UML tool 10](#_Toc161212610)

[1.3.1 Diagrams.net 10](#_Toc161212611)

[1.3.2 LucidChart 11](#_Toc161212612)

[2. Development Tools and Techniques 13](#_Toc161212613)

[2.1. C# 13](#_Toc161212614)

[2.2. ASP.NET Core MVC 14](#_Toc161212615)

[2.3. Visual Studio 2022 15](#_Toc161212616)

[2.4. SQL Server 16](#_Toc161212617)

[CONCLUSION 18](#_Toc161212618)

[REFERENCES 19](#_Toc161212619)

[Bibliography 19](#_Toc161212620)

Table of Contents

[Figure 1 : Class Diagram 5](#_Toc160543374)

[Figure 2 : Use case Diagram 6](#_Toc160543375)

[Figure 3 : Activity Diagram 8](#_Toc160543376)

[Figure 4 : Component Diagram 9](#_Toc160543377)

[Figure 5 : Diagram.Net 11](#_Toc160543378)

[Figure 6 : LucIdChart 13](#_Toc160543379)

[Figure 7 : Gleek.io 15](#_Toc160543380)

[Figure 8 : C# 18](#_Toc160543381)

[Figure 9 : ASP.NET Core MVC 19](#_Toc160543382)

[Figure 10 : Visual Studio 2022 21](#_Toc160543383)

[Figure 11 : SQL Server 23](#_Toc160543384)

# INTRODUCTION

Along with the rapid development of technology. Web applications are increasingly popular because of the benefits they bring such as: High interactivity, fast data storage and retrieval. Integrate multiple tasks to support features, management, and monitoring. Increase user experience, increase brand effects and save maintenance costs. FPT Company is no exception to that trend. They wish to develop a web-based system to manage "Training" activities for the company's internal training program to ensure a continuous learning environment throughout FPT Corporation. The system is used by the human resources department. In asm I will present the methodological tools I use to develop the system.

# 

# CONTENTS

# (P3) Research the use of software development tools and techniques and identify any that have been selected for the development of this application.

## UML

### 1.1 UML definition

**Unified Modeling Language (UML)** is a general-purpose modeling language. The main aim of UML is to define a standard way to **visualize** the way a system has been designed. It is quite similar to blueprints used in other fields of engineering. UML is not a programming language, it is rather a visual language.

* We use UML diagrams to portray the behavior and structure of a system.
* UML helps software engineers, businessmen, and system architects with modeling, design, and analysis.
* The Object Management Group (OMG) adopted Unified Modelling Language as a standard in 1997. It’s been managed by OMG ever since.
* The International Organization for Standardization (ISO) published UML as an approved standard in 2005. UML has been revised over the years and is reviewed periodically.

UML encompasses a diverse range of diagrams, each serving specific purposes in the software development lifecycle. These diagrams can be broadly categorized into three main groups:

* **Structure diagrams**: These diagrams depict the static structure of a system, including its components, relationships, and configurations. Examples include class diagrams, object diagrams, component diagrams, and deployment diagrams.
* **Behavior diagrams**: Behavior diagrams capture the dynamic aspects of a system by illustrating how it responds to internal and external stimuli over time. Examples include activity diagrams, state machine diagrams, and use case diagrams.
* **Interaction diagrams**: Interaction diagrams focus on the exchange of messages and interactions among system components, emphasizing the flow of control and data during runtime. Examples include sequence diagrams, communication diagrams, and timing diagrams.

The development of UML was driven by the need to standardize the disparate notations and methodologies used in software design. By providing a common language and notation, UML facilitates collaboration among stakeholders, enhances communication, and improves the overall quality of software systems (Geeksforgeeks, 2024)

### 1.2 Some popular UML Diagrams

#### 1.2.1 Class diagram

Class diagrams are the main building block of any object-oriented solution. It shows the classes in a system, attributes, and operations of each class and the relationship between each class.

In most modeling tools, a class has three parts. Name at the top, attributes in the middle and operations or methods at the bottom. In a large system with many related classes, classes are grouped together to create class diagrams. Different relationships between classes are shown by different types of arrows.

Below is an image of a class diagram.

A diagram of a software application

Description automatically generated

Figure 1 : Class Diagram

The class diagram is a central modeling technique that runs through nearly all object-oriented methods. This diagram describes the types of objects in the system and various kinds of static relationships which exist between them.

Relationships

There are three principal kinds of relationships which are important:

Association - represent relationships between instances of types (a person works for a company, a company has a number of offices.

Inheritance - the most obvious addition to ER diagrams for use in OO. It has an immediate correspondence to inheritance in OO design.

Aggregation - Aggregation, a form of object composition in object-oriented design. (Geeksforgeeks, 2024)

#### 1.2.2 Use case diagram

A use-case model describes a system's functional requirements in terms of use cases. It is a model of the system's intended functionality (use cases) and its environment (actors). Use cases enable you to relate what you need from a system to how the system delivers on those needs.

Think of a use-case model as a menu, much like the menu you'd find in a restaurant. By looking at the menu, you know what's available to you, the individual dishes as well as their prices. You also know what kind of cuisine the restaurant serves: Italian, Mexican, Chinese, and so on. By looking at the menu, you get an overall impression of the dining experience that awaits you in that restaurant. The menu, in effect, "models" the restaurant's behavior.

Because it is a very powerful planning instrument, the use-case model is generally used in all phases of the development cycle by all team members. (Handaz, 2024)

Use Case Diagram Example:

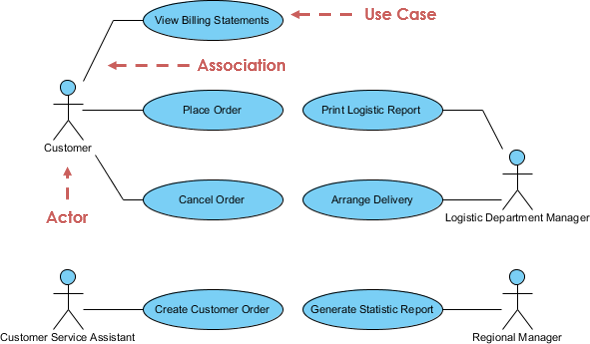


Figure 2: use diagram

#### 1.2.3 Activity diagram

Activity diagrams represent workflows in a graphical way. They can be used to describe the business workflow or the operational workflow of any component in a system. Sometimes activity diagrams are used as an alternative to State machine diagrams. Check out this wiki article to learn about symbols and usage of activity diagrams. You can also refer this easy guide to activity diagrams. (Geeksforgeeks, 2024)

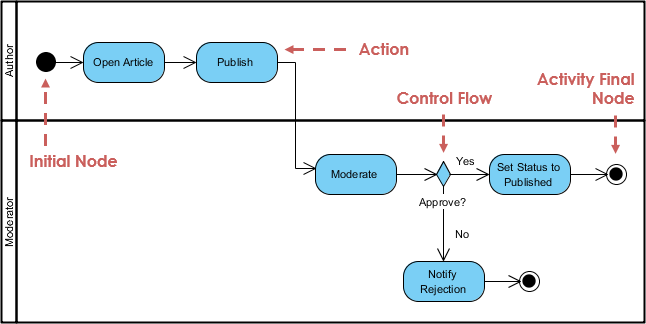


Figure 3 : Activity Diagram

#### 1.2.3 Component diagram

A diagram of a component

Description automatically generated

Figure 4 : Component Diagram

A Component Diagram in the Unified Modeling Language (UML) is a type of diagram used to model the physical aspects of object-oriented systems. It provides a visual representation of the components and their relationships within a system, helping to visualize, specify, and document component-based systems. (Athuraliya, 2023)

* Key components of a Component Diagram include:

**Components**: These are modular parts of a system that encapsulate functionality and data. Components represent the building blocks of the system and can be replaced or reused within its environment. Each component is responsible for one clear aim within the entire system.

**Interfaces**: Interfaces define the points of interaction between components. There are two types of interfaces: provided interfaces, represented with a complete circle at their end, indicating that the component provides this interface, and required interfaces, represented with only a half circle at their end, indicating that the component requires this interface to interact with other components.

A Component Diagram breaks down the actual system under development into various high levels of functionality. It helps to illustrate how different components interact with each other to achieve specific objectives. Components encapsulate functionality and data, promoting modularity, reusability, and maintainability within the system. (Athuraliya, 2023)

### 1.3 Use the UML tool

#### 1.3.1 Diagrams.net

**A screenshot of a computer

Description automatically generated**

Figure 5 : Diagram.Net

Diagrams.net, formerly known as draw.io, is a versatile cross-platform graph drawing software developed using HTML5 and JavaScript. It provides a user-friendly interface for creating various types of diagrams, including:.

Here are some key features of diagrams.net:

Security-First Approach: It prioritizes security and privacy. You can store your data wherever you prefer, and the platform does not access your data directly.

Integration with Google Drive: diagrams.net seamlessly integrates with Google Drive, allowing you to create and edit diagrams directly within your Google Drive environment.

Powerful Collaboration Tools: Collaborate with others using shared cursors in real-time. It offers everything you’d expect from a professional diagramming tool.

Free and Accessible: No login or registration is required. You can use it online or download the desktop app for offline use. (PerHart, 2020)

#### LucidChart

**A screenshot of a computer

Description automatically generated**

Figure 6 : LucIdChart

Lucidchart, an intelligent diagramming application, empowers teams to collaborate, visualize complex ideas, and make informed decisions. Here’s what you need to know about Lucidchart:

Versatile Diagram Creation: Lucidchart allows you to create various types of diagrams, including:

* Flowcharts
* Wireframes
* UML diagrams
* Organizational charts
* Network diagrams

Security and Privacy: Lucidchart prioritizes security and privacy. You can store your data wherever you prefer, and the platform does not directly access your data.

Google Drive Integration: Seamlessly integrate Lucidchart with Google Drive for easy diagram creation and editing within your Google environment.

Collaboration and Communication:

Common Visual Language: Lucidchart promotes collaboration by providing a common visual language. When everyone collaborates in the same space, alignment becomes easier.

Real-Time Collaboration: Collaborate with shared cursors in real time, ensuring everyone stays on the same page.

Bring Ideas to Life: Lucidchart helps you bring plans to life. Whether you’re designing software architecture or planning workflows, Lucidchart keeps you focused and moving forward with purpose1.

Intelligent Features:

ChatGPT Plugin: Leverage intelligent features like the ChatGPT plugin and AI Prompt Flow to build diagrams automatically.

Enterprise Ready: Lucidchart offers secure, scalable solutions for organizations of all sizes.

Integration with Industry-Leading Apps: Communicate visually and keep teams aligned by connecting Lucidchart with apps like Google Workspace, Microsoft, Atlassian, and Slack. (Adrew, 2022)

## 2. Development Tools and Techniques

### 2.1. C#

A purple hexagon with a white circle and a white circle with a white symbol on it

Description automatically generated

Figure 7 : C#

C#, pronounced “C-Sharp”, is a programming language created by Microsoft. C# combines the computational power of C++ and the ease of Visual Basic, Microsoft's event-based programming language and environment. C# is derived from C++, has similar features to Java, and is used in many areas of software development.

Here are some important points about C#:

Versatile language: C# allows you to create many types of charts, including:

* Flow chart
* Wireframe
* UML diagram
* Organizational chart
* Network diagram

Security and privacy: C# prioritizes security and privacy. You can store data anywhere you want, and the platform doesn't access your data directly.

Integration with Google Drive: C# integrates easily with Google Drive, allowing you to create and edit charts directly in your Google environment.

Collaboration and communication tools:

Common visual language: C# promotes collaboration by providing a common visual language. When everyone is working together in the same space, consensus becomes easier.

Real-time Collaboration: Collaborate with shared cursors in real-time, ensuring everyone is on the same page.

Create applications with C#: C# is the most popular language for development on .NET. With .NET, you can create applications that run on any platform. Leverage your favorite skills, code, and libraries across all platforms in a familiar environment. This helps you build applications faster and save costs.

Learn C# with developers: Explore the concepts and syntax of the C# programming language through live videos in the “C# for Beginners” video series. Once you've mastered the basics, you can explore other lessons on the .NET videos page, where you can learn how to build any type of application with C# (Anon., 2019).

### 2.2. ASP.NET Core MVC

A logo of a company

Description automatically generated

Figure 8 : ASP.NET Core MVC

ASP.NET Core MVC is a lightweight, open source and easy to test framework that integrates seamlessly with ASP.NET Core. It provides a template-based way to build dynamic websites, allowing clean separation between related sections.

Here are some important points about ASP.NET Core MVC:

MVC model: The Model-View-Controller (MVC) architectural model separates the application into three main groups of components: Models, Views and Controllers.

Optimized for ASP.NET Core: ASP.NET Core MVC is a framework optimized for use with ASP.NET Core. It helps you build dynamic websites efficiently and easily test.

Routing: ASP.NET Core MVC supports flexible routing, which helps you define how URLs are mapped to handlers.

Model binding: Allows you to bind data from HTTP requests to handler method parameters.

Model data checking: ASP.NET Core MVC provides a mechanism to check Model data to ensure the validity of the data before processing.

Dependency injection: Supports easy dependency injection, helping you manage dependencies in the application.

Filters: Allows you to perform actions before or after processing a request.

Areas: Helps you organize source code into separate, easy-to-manage areas. (Anon., n.d.)

### 2.3. Visual Studio 2022

A close up of a logo

Description automatically generated

Figure 9 : Visual Studio 2022

Visual Studio 2022 is a 64-bit integrated development environment (IDE) that helps you code, debug, test, and deploy software applications for Windows, Mac, Linux, iOS, and Android. Here are some important points about Visual Studio 2022:

GitHub Copilot: Visual Studio 2022 integrates with GitHub Copilot, an intelligent AI-powered coding tool. You'll get multiline hints from the source code and comments in your code, helping you complete tasks faster and easier.

Cross-platform development: Visual Studio 2022 allows you to develop cross-platform mobile and desktop applications. Use .NET MAUI to build cross-platform apps, and Blazor to create responsive Web interfaces using C#.

Deploy to Azure: Visual Studio 2022 supports deploying applications to the Azure platform easily.

AI integration: Artificial intelligence-based code completion features help you code quickly and accurately.

Real-time collaboration: Visual Studio 2022 lets you work together in real-time shared coding sessions.

Integrate with GitHub: Clone repos, manage work items and make commits easily.

Faster coding speeds: Visual Studio 2022 provides a fast experience and good performance, even with large and complex projects. (Anon., n.d.)

### 2.4. SQL Server

A close-up of a logo

Description automatically generated

Figure 10 : SQL Server

SQL Server, developed by Microsoft, is a relational database management system (RDBMS). It provides a robust platform for storing, managing, and retrieving data. Here are some key points about SQL Server:

Versions and History:

SQL Server has evolved over the years, with various versions released:

* SQL Server 2000: Released in 2000.
* SQL Server 2005: Released in 2005.
* SQL Server 2008: Released in 2008.
* SQL Server 2012: Released in 2012.
* SQL Server 2014: Released in 2014.
* SQL Server 2016: Released in 2016.
* SQL Server 2017: Released in 2017.
* SQL Server 2019: Released in 2019.
* SQL Server 2022: Released in 2022.

Each version brings enhancements, performance improvements, and new features.

Editions:

SQL Server comes in different editions, catering to various needs:

Enterprise Edition: Comprehensive features for large-scale applications.

Standard Edition: Suitable for mid-sized applications.

Web Edition: Designed for web hosting scenarios.

Developer Edition: For development and testing purposes.

Express Edition: Free and lightweight, ideal for small applications.

T-SQL (Transact-SQL):

T-SQL is the proprietary language used by SQL Server for querying and managing data.

It extends SQL with additional features and capabilities.

Tools and Management:

SQL Server Management Studio (SSMS): A powerful tool for managing databases, writing queries, and administering SQL Server.

Azure Data Studio (ADS): Formerly known as SQL Operations Studio (SOS), it’s a cross-platform tool for managing SQL Server and other databases.

Integration and Security:

SQL Server integrates with other Microsoft technologies, such as Azure services.

It provides robust security features, including authentication, encryption, and access control.

Service Packs and Updates:

Regular updates, service packs, and cumulative updates enhance stability and security.

Check your SQL Server version using the SERVERPROPERTY() function.

In summary, SQL Server is a versatile RDBMS used for building and managing data-driven applications. (Anon., n.d.)

# CONCLUSION

Before any software could be programmed decisions are needed to be made on which programming language should be used. The software been a desktop application, visual studio 2022 was used as the IDE and to ensure a standardized object oriented program in its entire ramification, C sharp programming language was used, after Microsoft.Net was used to connect to the database which was created using Microsoft SQL Server. The implementation was carried out using C# .Net Framework with windows applications which serves as the Graphical User Interface (GUI). The application was built on windows operating system, with visual studio a good graphical user interface was design, then using the C sharp programming as code behind to perform all the functionalities.

For this project we decided to use Visual Studio for web application development. Visual studio is an effective support software to support website programming work. It is also the system that brings together everything related to application development, including the code editor, designer, and debugger. That is, you can write code, fix bugs, edit application design easily with just 1 Visual Studio software.

The language we choose for software development is C#.

The framework we use in this project is ASP.NET MVC (.NET Framework). Because ASP.Net is a powerful technology used to build rich web applications. With it's MVC implementation it just became more systematic, enterprise and rugged. Most importantly, it is very suitable for FPT Company to run enterprise applications related to scalability aspect. Because MVC is a cleaner, systematic, and more advanced way of implementing structured growth of software.

# REFERENCES

# Bibliography

Adrew, 2022. *Lucidchart.* [Online]   
Available at: Lucidchart.com  
[Accessed 01 03 2024].

Anon., 2019. *dotnet.microsoft.* [Online]   
Available at: dotnet.microsoft.com  
[Accessed 01 03 2024].

Anon., n.d. *Hacker Noon.* [Online]   
Available at: https://hackernoon.com/7-of-the-top-best-languages-for-web-applicationdevelopment-8q1531zd  
[Accessed 06 03 2024].

Anon., n.d. *towardsdatascience.* [Online]   
Available at: https://towardsdatascience.com/top-10-databases-to-use-in-2021-d7e6a85402ba  
[Accessed 06 03 2024].

Athuraliya, A., 2023. *The Easy Guide to Component Diagrams.* [Online]   
Available at: https://creately.com/blog/software-teams/component-diagram-tutorial/  
[Accessed 05 03 2024].

CMARIX, n.d. [Online]   
Available at: https://www.cmarix.com/blog/6-benefits-of-asp-net-mvc/  
[Accessed 06 03 2024].

Geeksforgeeks, 2024. *Activity Diagrams | Unified Modeling Language (UML).* [Online]   
Available at: https://www.geeksforgeeks.org/unified-modeling-language-uml-activity-diagrams/  
[Accessed 05 03 2024].

Geeksforgeeks, 2024. *Class Diagram | Unified Modeling Language (UML).* [Online]   
Available at: https://www.geeksforgeeks.org/unified-modeling-language-uml-class-diagrams/  
[Accessed 05 03 2024].

Handaz, 2024. *What is Use Case Diagram?.* [Online]   
Available at: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/  
[Accessed 06 03 2024].

Luidad, 2024. *What is Unified Modeling Language (UML)?.* [Online]   
Available at: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/  
[Accessed 05 03 2024].

PerHart, A., 2020. *draw.* [Online]   
Available at: draw.io  
[Accessed 01 03 2024].