**ASSIGNMENT FRONT SHEET**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing** | | | | |
| **Unit number and title** | **﻿WEBG301 – Project Web** | | | | |
| **Submission date** | 15/7/2022 | **Date Received 1st submission** | |  |
| **Re-submission Date** |  | **Date Received 2nd submission** | |  |
| **Group** | **Student ID & Name** | **Final Score** | | **Student’s signature** |
| 1. GCC200058-Dang Nguyen Dang Khoi |  | | Dang Khoi |
| 2. GCC200203-Truong Van Tuan Kiet |  | | Tuan Kiet |
| 3. GCC200064-Mai Hoang Minh Thai |  | | Minh Thai |
| **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. | | | | | |
| **Class** | GCC0903 | | **Assessor name** | Tran Thi Kim Khanh | |

### OBSERVATION RECORD

|  |  |  |  |
| --- | --- | --- | --- |
| **Student name:** | **Dang Nguyen Dang Khoi** | | |
| **Description of activity undertaken** | | | |
| With group, I have the role of leader of the group. I have assigned tasks to each team member. In addition, I always help members when they have difficulty or assist them in fixing bugs when they code. With my own task, with source code, I have created function register, login, logout, add product to cart, update, delete product in cart, order, authorization for Admin / User, admin confirm order, change password, account manager, admin account registration, check order (user can't order if he has no product in cart), update information for user account. With document, I wrote chapter 4 of document of team. | | | |
| **Assessment criteria** | | | |
|  | | | |
| **How the activity meets the requirements of the assessment criteria** | | | |
|  | | | |
| **Student signature:** | **Dang Khoi** | **Date:** | **15/7/2022** |
| **Assessor name:** | **Tran Thi Kim Khanh** | | |
| **Assessor signature:** |  | **Date:** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Student name:** | **Truong Van Tuan Kiet** | | |
| **Description of activity undertaken** | | | |
| In the group, I have the role of a member of the group. I take quests from the leader and execute them. In addition, I always help members when they have difficulty or assist them in fixing bugs when they coding. I have a background in design so I was tasked with drawing logos and icons for the website. About the Back-end I was assigned functions like the Search function, Compare product function, Index product, Index cart user, Index product detail, and check Username duplicate. Regarding the front-end, I was assigned functions such as changing the website color header, adding a logo, creating a website footer, adding about us, and icons of social media. With the document, I wrote chapter 3 of the document team. | | | |
| **Assessment criteria** | | | |
|  | | | |
| **How the activity meets the requirements of the assessment criteria** | | | |
|  | | | |
| **Student signature:** | **Tuan Kiet** | **Date:** | **15/7/2022** |
| **Assessor name:** | **Tran Thi Kim Khanh** | | |
| **Assessor signature:** |  | **Date:** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Student name:** | **Mai Hoang Minh Thai** | | |
| **Description of activity undertaken** | | | |
| In the group, I have the role of a member of the group. I take quests from the leader and execute them. In addition, I always help members when they have difficulty or assist them in fixing bugs when they coding.  - With the source code, I created and edited the homepage interface, added the latest product slideshow to the homepage, added a navbar to the homepage and made the website responsive, changed the login interface, created the add function, update function of Product, create interface, add function and update function of Brand, create Order\_history.  - With the document, I wrote chapters 1 and 2 of the team document. | | | |
| **Assessment criteria** | | | |
|  | | | |
| **How the activity meets the requirements of the assessment criteria** | | | |
|  | | | |
| **Student signature:** | **Minh Thai** | **Date:** | **15/7/2022** |
| **Assessor name:** | **Tran Thi Kim Khanh** | | |
| **Assessor signature:** |  | **Date:** |  |

|  |  |  |
| --- | --- | --- |
| **❒ Summative Feedback: ❒ Resubmission Feedback:** | | |
| **Grade:** | **Assessor Signature:** | **Date:** |
| **Internal Verifier’s Comments:** | | |
| **IV Signature:** | | |

**Contents**

[Chapter 1 – Users’ requirements 7](#_Toc108268892)

[1.1 Project Specification 7](#_Toc108268893)

[1.2 Users 7](#_Toc108268894)

[1.3 Input data is taken from the Users 7](#_Toc108268895)

[1.4 Output data is displayed to the user 7](#_Toc108268896)

[1.5 The processes of the system 7](#_Toc108268897)

[1.6 Scope of the work 8](#_Toc108268898)

[1.7 Customer Acceptance Criteria 9](#_Toc108268899)

[Chapter 2 – System Design 10](#_Toc108268900)

[2.1 About Symfony 10](#_Toc108268901)

[2.2. Git/GitHub 11](#_Toc108268906)

[2.3 Use Case Diagram 13](#_Toc108268907)

[2.4 Entity Relationship Diagram 14](#_Toc108268908)

[2.5 Sitemap 14](#_Toc108268909)

[2.6 System Architecture Diagram 16](#_Toc108268910)

[Chapter 3 – Implementation 17](#_Toc108268911)

[3.1 Sample Source Code 17](#_Toc108268912)

[3.2 Images of final Application 24](#_Toc108268913)

[3.3 GitHub Repository evidences 26](#_Toc108268914)

[Chapter 4 – Conclusion 27](#_Toc108268915)

[4.1 What went well 27](#_Toc108268916)

[4.2 What did no go well 27](#_Toc108268917)

[4.3 Lessons learned and further improvements 27](#_Toc108268918)

[Reference 29](#_Toc108268919)

**Assignment**

# **Chapter 1 – Users’ requirements**

## **1.1 Project Specification**

We are working as a Website Developers for an IT consultancy company. The company was approached by a new branch of KTK Shop in Can Tho City, a smartphone store. They asked us to design an online smartphone shopping website for them. The website will be developed for the Windows Platform using HTML5, CSS, JavaScript and PHP. The site will work fine on all the leading browsers including Chrome, IE, Firefox, etc.

## **1.2 Users**

- Company: Staffs of KTK Shop is Admin

- Customer: Customers who have age from 18 to 30 years old that as student or people who want

## **1.3 Input data is taken from the Users**

- Information about smartphone: name, price, quantity, description of product

- Information of users: Username, password, birthdate, fullname, email, address, tel.

- Name of product which user need to find.

- Information of order: name, price, quantity, total.

- Detail of 2 ProductID which user need to compare.

- Information of brand.

## **1.4 Output data is displayed to the user**

- List all smartphones and detail information of smartphone.

- Result of searching.

- Detail of 2 products to compare.

- Notification such as user enter error information,...

- Order history of user.

- List all brand.

## **1.5 The processes of the system**

- Search product items by name

- Add order

- Add/ update product

- Order history

- Add admin account

- Update user account, change password

- Compare 2 product

- Add/ update brand

- Add/update/delete product in cart

- Processing customer’s registration.

- Processing customer’s login.

- Authorization for Admin / User

- Admin confirms order

## **1.6 Scope of the work**

- Design a website for online shopping with multi pages, KTK Shop

- Front-end:

* For customer: customer can register an account and login with the registered username to buy a smartphone. Besides, customer can update account, change password, watch order history, search product by name, compare two product, watch detail information and add product into cart. In addition, in page cart, customer can update, delete and order their order.
* For admin: admin can add, update product and brand. Besides, admin can management account of customer and add admin account, watch order detail . In addition, admin can confirms customer's order.

- Back-end:

* Add, update product: add, update information of smartphone in database
* Add, update brand: add, update information of brand in database
* Add account for admin
* Compare 2 product, if user check on 3 product to compare the website will be display a error message
* Search product: searching a smartphone by name as a user’s input
* Register: if there is no duplicate username account and valid data, a new account is added to the database
* Register: if password least 8 characters long or long 25 characters, the website will be display a error message.
* Login: if username and password had in database, user will be login successfully
* Update account for user and change password
* Order: add, update, delete product in cart in database
* Payment: add information order with name of customer in database
* Confirms: Confirms by change the status of an order in database.

## **1.7 Customer Acceptance Criteria**

|  |  |
| --- | --- |
| **No** | **Customer’s Acceptance Criteria** |
| **1** | The Home Page should be created by making use of a suitable logo. |
| **2** | The site must contain the links to navigate through various smartphones such as (Apple, Samsung, etc). |
| **3** | Administrators can management account of customer but don't update or delete |
| **4** | Pricing of smartphones should be added along with the images |
| **5** | Customer can register member of store and then can login the website |
| **6** | Administrators can perform the following functions: Add, update products |
| **7** | Administrators can perform the following function: confirms customer's order. |
| **8** | Users can perform the following functions: Add, edit and delete their order |
| **9** | Users can search any product when they enter name of product and press search button |
| **10** | The site cart must display all information such as price, quantity and total of order |
| **11** | The site detail product must display all information of product such as image, specification |
| **12** | Customer can read all information of KTKShop at footer of website. |

# **Chapter 2 – System Design**

## **2.1 About Symfony**

According to (Golofit, 2021):

* What is symfony?

Symfony is a relatively new PHP framework and reusable PHP components provider. It exists as a company and a community of over 600,000 developers-users from multiple countries. And that's where the usefulness of Symfony comes from. Due to its flexibility and active community, it has always been meeting the latest trends. Being available under an open-source license, Symfony can be adjusted to the **specific needs of every single developer by adding single-handedly any required module of their own**. And that's precisely how it works. The size and diversity of the Symfony community create equally diverse possibilities for every user.

* The Model-View-Controller Architecture.

To use PHP Framework Symfony in a more efficient way, it can be structured with MVC pattern. This stands for Model View Controller, and it is an architecture pattern, which helps us design a website or web app in a much more structured, layered, and logical way. It does so by dividing your project into three pieces:

* Model, everything related to databases and maintaining the data itself.
* View stands for components responsible for presenting data to a user.
* Controller, which is an intermediate between the other two. It connects them by sending data from Model to View, or, in other words, processes data, updating it when needed, and explains how it should be presented to the user.
* Why should use Symfony?

### Flexibility of framework

Framework Symfony doesn't tie a user to all its functionalities. Instead, it allows us to use only the very needed ones. You can choose only the tools you need without having to browse every single time through the rest of them. The adaptability of **Framework Symfony helps users speed up the pace of work** without making it unnecessarily complex or heavy.

### Fast performance

Symfony is considered to be one of the best, if not the best, since it doesn’t require a lot of resources to run efficiently. You can also **speed up the performance, optimizing your servers and apps**, with a help from Symfony support. It might be worth mentioning that it also consumes less memory while still allowing the development of apps at high speed.

### User-friendly

PHP Framework Symfony is accessible to a newcomer and an expert at the same time. This was achieved with professional support from the company and a huge community, working together and developing the possibilities of Symfony for every user. That's why it's considered to be a friendly environment even for beginners.

### Expandability of Framework

Using the Symfony framework, you can work on a system of bundles, each one with its specific functionality. You can **reuse every bundle you added and share every single one with the Symfony community.** This system allows you to change even the very core of your framework behavior without reconfiguring everything from the start.

## **2.2. Git/GitHub**

According to (Github, 2022):

* What is Git/Github?

Git is distributed version control software. Version control is a way to save changes over time without overwriting previous versions. Being distributed means that every developer working with a Git repository has a copy of that entire repository - every commit, every branch, every file.

There are a few things should know before getting started with Git:

* Branches are lightweight and cheap, so it's OK to have many of them.
* Git stores changes in SHA hashes, which work by compressing text files. That makes Git a very good version control system (VCS) for software programming, but not so good for binary files like images or videos.
* Git repositories can be connected, so we can work on one locally on our own machine, and connect it to a shared repository. This way, we can [push](https://github.com/git-guides/git-push) and [pull](https://github.com/git-guides/git-pull) changes to a repository and easily collaborate with others.
* Why use Git/Github?

With Git, we can make a "commit", or a save point. You can also go back to previous commits. This takes the pressure off of us while you're working. Commit often and commit early, and we will never have that gut-sinking feeling of overwriting or losing changes.

* There are some major advantages:
* Speed

Like we mentioned above, Git uses SHA compression, which makes it very fast.

* Merge conflicts

Git can handle merge conflicts, which means that it's OK for multiple people to work on the same file at the same time. This opens up the world of development in a way that isn't possible with centralized version control. We have access to the entire project, and if we are working on a branch, we can do whatever we need to and know that our changes are safe.

* Cheap branches

Speaking of branches, Git offers a lot of flexibility and opportunity for collaboration with branches. By using branches, developers can make changes in a safe sandbox.

Instead of only committing code that is 100% sure to succeed, developers can commit code that might still need help. Then, they can push that code to the remote and get fast feedback from integrated tests or peer reviews.

Without sharing the code through branches, this would never be possible.

* Ease of rollback

If we make a mistake, it's OK! Commits are immutable, meaning us can't be changed. This means that if we do make a mistake, even on an important branch like main, it's OK. We can easily revert that change, or roll back the branch pointer to the commit where everything was fine.

The benefits of this can't be overstated. Not only does it create a safer environment for the project and code, but it fosters a development environment where developers can be braver, trusting that Git has their back.

## **2.3 Use Case Diagram**

**Diagram

Description automatically generated**

**Picture 1.** Use case diagram

## **2.4 Entity Relationship Diagram**

**Graphical user interface, application

Description automatically generated**

**Picture 2.** Entity Relationship Diagram

## **2.5 Sitemap**

- Don’t login

Diagram

Description automatically generated

**Picture 3.** Sitemap when user don’t login

- Customer login

Diagram

Description automatically generated

**Picture 4.** Sitemap when customer login

- Admin login

Diagram, Teams

Description automatically generated

**Picture 5.** Sitemap when admin login

## **2.6 System Architecture Diagram**

Graphical user interface

Description automatically generated

**Picture 6.** System Architecture Diagram

### **Chapter 3 – Implementation**

## **3.1 Sample Source Code**

**Text

Description automatically generated**

* The controller gets the result from the ShowCart and sumCart function and then returns it to cart/index.html.tiwg.

**Text

Description automatically generated**

* Show cart repo ShowCart function on CartRepository gets the userID and cart ID from the controller. This function will select all information from the user cart and return it to the controller.

**Text

Description automatically generated**

* The SumCart function on the CartRepository gets the userID and cart ID from the controller. This function will calculate the total amount of the product from the user's cart and send it back to the controller.

**Text

Description automatically generated**

**Text

Description automatically generated**

Text

Description automatically generated

* The template receives the customer's cart information from the controller. We use the for loop to process and display shopping cart information

**Text

Description automatically generated**

* CheckCartDetail on CartRepository gets the product ID and cart ID from the controller. Use the COUNT() function to count the product quantity in the cart. Then return the quantity of the product. This is a controller to add products to the user cart.

**Text

Description automatically generated**

**Text

Description automatically generated**

* The controller gets the product quantity of the product then the controller will check if the product has been added by the user. If there are already products in the user's cart, the system will update 1 more product quantity. If the product has not been added, the system will add that product to the user's cart with the quantity of 1.

**Text

Description automatically generated**

* This controller will look for the ProductID contained in the CartDetail of the currently logged-in user's cart. Then remove that product from CartDetail.

Text

Description automatically generated

Text

Description automatically generated

* This controller will get the product quantity that user want to update and find the ProductID contained in the CartDetail of the currently logged-in user's cart. And then Updates the product quantity in the CartDetail.

## **3.2 Images of final Application**

**Graphical user interface, application

Description automatically generated**

* This is the user's cart when the user has not added a product.

**Graphical user interface, application

Description automatically generated**

**-** This is the cart when the user has added the product.

**Graphical user interface, application

Description automatically generated**

**-** This is the shopping cart when the user updates the product quantity, product quantity and total price are all changed.

**Graphical user interface, application

Description automatically generated**

**-** This is the user's cart before delete product.

**Graphical user interface, application, website

Description automatically generated**

**-** This is the user's cart after delete product.

## **3.3 GitHub Repository evidences**

Our repository: <https://github.com/KTK-Shop/Shop-KTK.git>

### **Chapter 4 – Conclusion**

## **4.1 What went well**

• My website has full functions such as login, register, order,... It fully meets the requirements of users.

• I have encrypted the password of the user account with symfony's special encryption function, so the user information is pretty well secured.

• User-friendly interface.

## **4.2 What did no go well**

With function of our website, there are not many options to log in with other accounts such as Google, Facebook,..., Some features like feedback and display of purchases from customers are missing, There are no security functions in the database yet, .... With during the working process, we still make mistakes when defining the relationships between entities, so it takes us a lot of time to correct them. In addition, we sometimes forget and confuse the commands of the symfony framework, which costs us time to find and adjust. Finally, we still haven't drawn the use case well, the direction of include and extend is still flawed.

## **4.3 Lessons learned and further improvements**

- Lessons learned: We realize we need to improve more on our knowledge. First of all, we need to learn more about entity relationships and how to draw use case diagrams to master the basics and not spend a lot of time the same mistakes. In addition, we need to do more projects related to the symfony framework to improve our ability to code with this framework. Moreover, through this project, we realized that we still have a lot of knowledge to learn and basic knowledge to master, so we need to search for documents in Google and YouTube to learn more. knowledge to do better on other projects in the future.

- Further improvements:

* We will update more login options with other accounts such as Google, Facebook,...When a user enters their account information, our system will automatically get the user's information from that account and create it for them a account in our system.
* We will create an additional column in the product table in the database to store the total number of purchases of that product, then we will redesign the interface for the product section to add more information about the purchase of the product for the customer. products so that customers can trust and buy products from the store.
* We will create a feedback column in the order detail table and add a feedback frame for customers when they receive the goods to receive feedback from which to further improve KTK Shop's services.
* We will learn more and add API to the website so that users don't have to reload the page when choosing any function. Thereby increasing the friendliness and performance of the website

# **Reference**

Github, 2022. *github.* [Online]   
Available at: https://github.com/git-guides

Golofit, P., 2021. *accesto.* [Online]   
Available at: https://accesto.com/blog/what-is-php-framework-symfony-explained-for-executives/