|  |  |
| --- | --- |
|  | **MINISTRY OF EDUCATION AND TRAINING** |

**FPT UNIVERSITY**

|  |  |
| --- | --- |
| Capstone Project Document | |
| **Face Detection For Delivery Payment** | |
| **Group 5 - IS** | |
| **Group members** | Nguyễn Văn Hiếu – SE61961 (Leader)  Đinh Phú Thắng – SE62528  Trần Trọng Nghĩa – SE62278  Nguyễn Hữu Lâm– SE61234 |
| **Supervisor** | Nguyễn Huy Hùng |
| **Ext. Supervisor** | N/A |
| **Project Code** | FDDP |

– **Ho Chi Minh City, 20 January 2019** –

Nội dung

[**A.** **Introduction** 2](#_Toc6507282)

[**B.** **Software Project Management** 2](#_Toc6507283)

[**1.** **Problem Definition** 2](#_Toc6507284)

[**1.1** **Name of this Capstone Project** 2](#_Toc6507285)

[**1.2** **Problem Abstract** 2](#_Toc6507286)

[Face detection for delivery payment is a system that aims at specifying customer face then use it for paying their order. 2](#_Toc6507287)

[Detecting users’ face requires knowledge about AI Machine Learning. For detecting, our system needs customers to provide their face which will use for training AI. Finally, when paying for the order, the user will be taken their face again by the shipper for AI verify with their face which provided. 2](#_Toc6507288)

[**1.3** **Project Overview** 2](#_Toc6507289)

[**1.3.1** **Current Situation** 2](#_Toc6507290)

[- Strengths: 2](#_Toc6507291)

[**+** **Knowledge on software development:** The whole team has experience in software development. All members are familiar with the same server-side programming language**.** 2](#_Toc6507292)

[- Weakness: 2](#_Toc6507293)

[**1.3.2** **The Proposed System** 3](#_Toc6507294)

[**1.3.2.1** **Website** 3](#_Toc6507295)

[**1.3.2.2** **Mobile Application** 4](#_Toc6507296)

[**1.3.3** **Boundaries of the System** 4](#_Toc6507297)

[The system can: 4](#_Toc6507298)

[**1.3.4** **Future Plans** 5](#_Toc6507299)

[- Check out payment for shipper. 5](#_Toc6507300)

[- Get approved from Paypal to call API accept order. 5](#_Toc6507301)

[**1.3.5** **Development Environment** 5](#_Toc6507302)

[**1.3.5.1** **Hardware requirements** 5](#_Toc6507303)

[**For Server** 5](#_Toc6507304)

[**For Computer** 5](#_Toc6507305)

[**For Tablet/Mobile** 5](#_Toc6507306)

[**1.3.5.2** **Software requirements** 5](#_Toc6507307)

[**For Web** 5](#_Toc6507308)

[**For Mobile Application** 6](#_Toc6507309)

[**2.** **Project Organization** 6](#_Toc6507310)

[**2.1 Software Process Model** 6](#_Toc6507311)

[**2.2 Roles and Responsibilities** 7](#_Toc6507312)

[**2.3 Tools and Techniques** 8](#_Toc6507313)

[**3.** **Project Management Plan** 8](#_Toc6507314)

[**3.1. Sprint Backlog** 8](#_Toc6507315)

[**3.2. Product Backlog** 8](#_Toc6507316)

[**3.3. All Meeting Minutes** 8](#_Toc6507317)

[Meeting Minutes can be found here 8](#_Toc6507318)

[**4.** **Coding Convention** 8](#_Toc6507319)

# **Introduction**

1. **Software Project Management**
2. **Problem Definition**
   1. **Name of this Capstone Project**

* Official name: Face Detection For Delivery Payment.
* Vietnamese name: Hệ Thống Hỗ Trợ Thanh Toán Thông Qua Nhận Diện Khuôn Mặt Khi Giao Hàng.
* Abbreviation: FDDP.
  1. **Problem Abstract**

Face detection for delivery payment is a system that aims at specifying customer face then use it for paying their order.

Detecting users’ face requires knowledge about AI Machine Learning. For detecting, our system needs customers to provide their face which will use for training AI. Finally, when paying for the order, the user will be taken their face again by the shipper for AI verify with their face which provided.

* 1. **Project Overview**
     1. **Current Situation**
* Strengths:

**+** **Knowledge on software development:** The whole team has experience in software development. All members are familiar with the same server-side programming language**.**

* Weakness:

**+** **Lack of Deep Learning knowledge**: The application using human face detection technology requires Deep Leaning. However, none of the team members has learned Deep Learning or any related techniques. Thus, the research is extremely important and necessary.

**+ Lack of Ionic Framework knowledge:** The application using 2 apps for customer and shipper on mobile device. We are using Ionic framework for apply on multiple platform (Android and IOS). However, none of us has learned and worked with this framework.

**+ Lack of Business knowledge:** The application applied for shipping business. However, none of team members has worked in this business. So we have to research about this business and try our best to done it well. However, maybe some case of business we couldn’t handle.

**+ Lack of support from a payment company:** Our application has to build process of payment to improve it with face recognize. So that, we have to have the permission to access customer’s payment information and the API for approve the payment. However, we have no support for that methods except open source from Paypal. We got many troubles with this problem which we’ll discuss in the another report in this document.

- Opportunities:

* **Implementing Deep Learning not necessary:** Just understanding the mechanism is enough. In addition, there are a lot of libraries to support third-party to work with Deep Learning easily.

* + 1. **The Proposed System**

We create 2 applications in client side for shipper and customer, 1 web application for admin to manage orders, accounts, stores and products to apply a small business. So that we can apply function for payment over detecting human face.

For detecting human face, as mentioned above, we decide to use TensorFlow library as third-party which supported in Deep Learning. This library responses to face points by coordinates. Those points will be assigned to multiple image frames posted by customer as when they assign to system.

After that, the system will compare coordinates on images to find the matched and give result in checkout order, we decide to use Inception Resnet V1 to resolve it. This is the best feasibility solution that our team can perform.

We decide to develop the system with functions:

* + - 1. **Website**
* **For admin (manager)**
* Admin can manage account:
* Admin (Manager)
* View list of admin.
* Search admin with all field.
* View data of admin.
* User (Customer)
* View list of users.
* Search user with all field.
* View data of user.
* Shipper (Staff)
* View list of shippers.
* Search shipper with all field.
* View data of shipper.
* Add new shipper.
* Deactivate shipper.
* Admin can manage order:
* View order list by day.
* Search order with all field.
* Edit order (show who made the actions).
* Admin can manage store:
* View list store partners in the system.
* Search store with all field.
* View data of store.
* Add new data of store.
* Deactivate store.
* Admin can manage product:
* View list of products of store.
* Search product with all field.
* View data of product.
* Add new product depend on store.
* Admin can manage feedback:
* Can view feedback from users.
* Can make feedback for shipper.
  + - 1. **Mobile Application**
* **For driver (staff)**
* Shipper can take orders of customer.
* Shipper can edit their password.
* Shipper can view their profit.
* Shipper can view their routing on google map.
* **For users (customer):**
* User can register new account with their mobile phone.
* User can log in or log out system of mobile phone.
* User can choose the food or item what they like to add to their cart.
* User can edit some of their account’s information.
* User can feedback shipper.
* User can make paying online with face detection.
* User can see list of order they already made.
  + 1. **Boundaries of the System**

The system can:

* Allow Manager to manage user accounts .
* Allow Manager to manage stores and products of store.
* Allow Manager to approve face for payment in case shipper can’t use this function.
* Allow Manager to see system report.
* Allow Shipper to receive orders from customer.
* Allow Shipper to see their profile and profit.
* Allow Shipper to view history.
* Allow Shipper verify customer order by face recognize.
* Allow Customer to create new accounts.
* Allow Customer to create order.
* Allow Customer to view history orders.
* Allow Customer to edit their profile
* Allow Customer to edit their receive member face.

The system is not a complete business, it’s just a model to apply payment over face recognize so it can be miss some field in business.

* + 1. **Future Plans**
* Check out payment for shipper.
* Get approved from Paypal to call API accept order.
  + 1. **Development Environment**
       1. **Hardware requirements**

**For Server**

|  |  |  |
| --- | --- | --- |
| **Windows** | **Minimum Requirement** | **Recommended** |
| **Internet Connection** | Cable, Wi-Fi (6 Mbps) | Cable, Wi-Fi (16 Mbps) |
| **Operating System** | Windows 7 or above | Windows 7 or above |
| **Computer Processor** | Core i7 (5th) | Core i7 (7th) |
| **Computer Memory** | 8GB RAM | 16GB RAM |

**For Computer**

|  |  |  |
| --- | --- | --- |
| **Windows** | **Minimum Requirement** | **Recommended** |
| **Internet Connection** | Cable, Wi-Fi (6 Mbps) | Cable, Wi-Fi (16 Mbps) |
| **Operating System** | Windows 7 or above | Windows 7 or above |
| **Computer Processor** | Core i3 (5th). | Core i5 (5th) |
| **Computer Memory** | 4GB RAM | 8GB RAM |

**For Tablet/Mobile**

|  |  |  |
| --- | --- | --- |
| **Windows** | **Minimum Requirement** | **Recommended** |
| **Internet Connection** | Cable, Wi-Fi (6 Mbps) | Cable, Wi-Fi (16 Mbps) |
| **Operating System** | Android 6 | Android 8 |
| **Computer Processor** | Snapdragon 710. | Snapdragon 840. |
| **Computer Memory** | 4GB RAM | 8GB RAM |

* + - 1. **Software requirements**

**For Web**

|  |  |  |
| --- | --- | --- |
| **Software** | **Name / Version** | **Description** |
| **Operating system** | MacOS, windows 7 or above, linux, ….. | Support for mutiple Systems. |
| **Environment** | Java 8 platform | Install java platform jdk8 |
| **Modeling tool** |  |  |
| **IDE** | IntelliJ Idea Ultimate | Jetbrain IDE |
| **DBMS** | Mysql Database | Mysql database |
| **Source control** | Github | Github.com |
| **Web browser** | Chrome, Firefox, Safari, Edge | Support for multiple platform. |

**For Mobile Application**

|  |  |  |
| --- | --- | --- |
| **Software** | **Name / Version** | **Description** |
| **Operating system** | Android OS 6.0 or above, IOS 10 or above. | Support for both Android and IOS. |
| **Environment** | Android or IOS environment. | Mobile device using Android or IOS system. |
| **Modeling tool** | Android, IOS or Chrome browser. | Android or IOS device . |
| **IDE** | Visual Code | Visual Code IDE. |
| **DBMS** | Local Storage | Android local storage. |
| **Source control** | Github | Github.com |

1. **Project Organization**

## **2.1 Software Process Model**

This project is developed using Scrum model – part of an agile framework for Software development project. Our team choose Scrum model because of the following reasons:

* Pose Detection is a very new issue with us. We need to focus on researching about deep learning and experiment the most suitable deep learning library with our abilities.
* After detecting human pose, we need to keep researching about algorithm which help us compare the matching percentage between two poses.

For those reasons we choose Scrum model for our process.

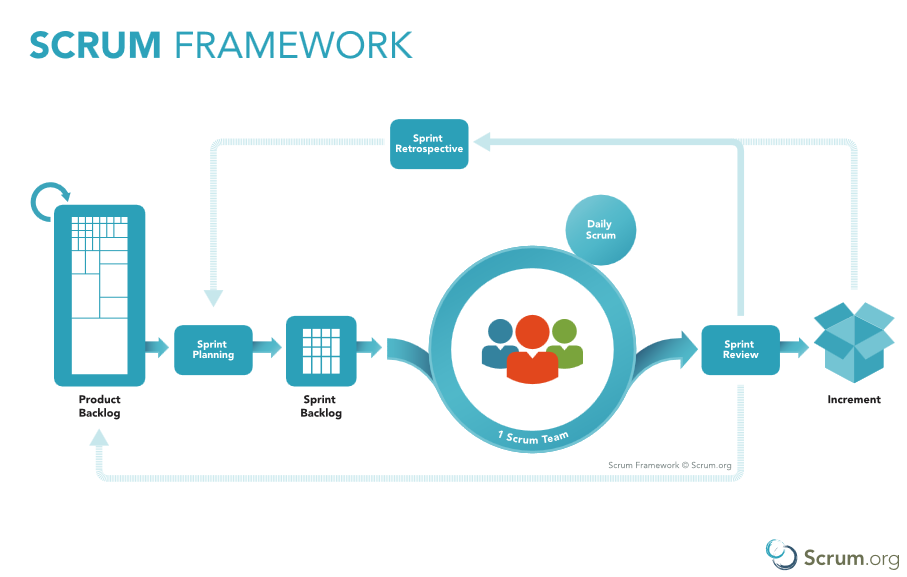


Figure 1 Scrum Framework

[*https://www.scrum.org/resources/what-is-scrum*](https://www.scrum.org/resources/what-is-scrum)

## **2.2 Roles and Responsibilities**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Full name** | **Role in Group** | **Responsibilities** |
| **1** | Nguyễn Huy Hùng | Project manager | • Specify user requirement  • Control the development process  • Give out technique and business analysis support |
| **2** | Nguyễn Văn Hiếu | Team Leader, BA, DEV, Tester | • Managing process  • Designing database  • Clarifying requirements  • Prepare documents  • GUI Design  • Create test plan  • Coding  • Testing |
| **3** | Đinh Phú Thắng | Developer/Tester | • Coding  • Testing |
| **4** | Trần Trọng Nghĩa | Developer/Tester | • Coding  • Testing |
| **5** | Nguyễn Hữu Lâm | Developer/Tester | • Coding  • Testing |

## **2.3 Tools and Techniques**

|  |
| --- |
| **Techniques** |
| Developing tool | Front-end | HTML 5, CSS 3, JavaScript, jQuery 3.1, Bootstrap 4.1 |
| Back-end | Spring Boot MVC Framwork, JPA, Tensorflow, DLib, OpenCV. |
| Web Servers | Apache Tomcat | |
| Mobile | Ionic Framework | |
| Database Management System | Microsoft MySQL server. | |

1. **Project Management Plan**

### **3.1. Sprint Backlog**

Sprint Backlog can be found [here](https://drive.google.com/open?id=1InYqyDqNsp_rIDQsVx8tM0BOcU0HbvMi)

### **3.2. Product Backlog**

Product Backlog can be found [here](https://drive.google.com/open?id=1Kk6lrjphJiMvoETUEk27QsY8z6ZabXZR)

### **3.3. All Meeting Minutes**

Meeting Minutes can be found [here](https://drive.google.com/open?id=1b3XmAZa-3p9I9eWR6zpcKpfYfvhmAft3)

1. **Coding Convention**

* **Naming Convention**:
* Variable names should be short yet meaningful. The choice of a variable name should be designed to indicate to the casual observer the intent of its use.
* Methods should be verbs, in mixed case with the first letter lowercase, with the first letter of each internal word capitalized.
* **Indentation**:
* One declaration per line is recommended since it encourages commenting. In absolutely no case should variables and functions be declared on same line.
* Do not put different types on the same line.
* **Declarations Convention:**
* One declaration per line is recommended since it encourages commenting.
* Using Java Code Convention from:

<http://www.oracle.com/technetwork/java/codeconvtoc-136057.html>