|  |  |
| --- | --- |
| logo ngan.png | **MINISTRY OF EDUCATION AND TRAINING** |
|  |  |
|  |  |
|  |  |
|  |  |

**FPT UNIVERSITY**

Capstone Project Document

**Place Suggesting Intelligent Bot**

|  |  |
| --- | --- |
| **Group 3** | |
| **Group members** | Huỳnh Thành Đạt – Team Leader – SE61273  Trương Châu Minh Huy – Team Member – SE61175  Trần Văn Thành – Team Member – SE61373  Đào Công Thuận – Team Member – 60487 |
| **Supervisor** | Kiều Trọng Khánh |
| **Ext. Supervisor** | N/A |
| **Capstone Project code** | PSIB |

- Ho Chi Minh City, 05/2016 -

*This page is intentionally left blank*

Table of Contents

1. Report No. 1 Introduction 6
   1. Project Information 6
   2. Introduction 6
   3. Current Situation 6
   4. Problem Definition 6
   5. Proposed Solution 6
      1. Feature functions 6
      2. Advantages and disadvantages 7
   6. Functional Requirements 7
      1. Web Crawler: 7
      2. Bot Engine 7
      3. Web Training 7
      4. Back-end Training 7
   7. Role and Responsibility 8
2. Report No.2 Software Project Management Plan 9
   1. Problem Definition 9
      1. Name of this Capstone Project 9
      2. Problem Abstract 9
      3. Project Overview 9
   2. Project organization 11
      1. Software Process Model 11
      2. Roles and responsibilities 12
      3. Tools and Techniques 14
   3. Project Management Plan 15
      1. Product Backlog 15
      2. Deliverables 17
      3. All Meeting Minutes 18
   4. Coding Convention 18
3. Report No. 3 Software Requirement Specification 20
   1. User Requirement Specification 20
      1. Guest requirement 20
      2. Member requirement 20
      3. Student requirement 20
      4. Company manager requirement 20
      5. Staff requirement 21
      6. Administrator requirement 21
      7. System requirement 21
   2. System Requirement Specification 21
      1. External Interface Requirement 21
      2. System Overview Use Case 22
      3. List of Use Case 23
   3. Software System Attribute 98
      1. Usability 98
      2. Reliability 98
      3. Availability 98
      4. Security 99
      5. Maintainability 99
      6. Portability 99
      7. Performance 99
   4. Conceptual Class Diagram 99

Figure list

[Figure 1: System Overview Use Case](#Toc410477346) 20

[Figure 2: <Guest>Overview Use Case](#Toc410477347) 21

[Figure 3: <Guest>Sign in](#Toc410477348) 21

[Figure 4: <Guest>Search vacancy](#Toc410477349) 23

[Figure 5: <Guest>Overview Use Case](#Toc410477350) 25

[Figure 6: <Administrator>Manage Event](#Toc410477351) 25

[Figure 7: <Administrator>Create Event](#Toc410477352) 26

[Figure 8: <Administrator>Update Event](#Toc410477353) 27

[Figure 9: <Administrator>Manage Company](#Toc410477354) 29

[Figure 10: <Administrator>Send Email](#Toc410477355) 29

[Figure 11: <Administration>Manage Thread](#Toc410477356) 31

[Figure 12: <Administration> Create Thread](#Toc410477357) 31

[Figure 13: <Administration> Edit Thread](#Toc410477358) 33

[Figure 14: <Administration> Active Thread](#Toc410477359) 35

[Figure 15: <Administration> Deactivate Thread](#Toc410477360) 37

[Figure 16: <Administration>Manage Fee](#Toc410477361) 39

[Figure 17: <Administration> Create Fee](#Toc410477362) 39

[Figure 18: <Administration> Edit Fee](#Toc410477363) 41

[Figure 19: <Administration> Delete Fee](#Toc410477364) 44

[Figure 20: <Staff>Overview Use Case](#Toc410477365) 46

[Figure 21: <Staff>Manage interview](#Toc410477366) 46

[Figure 22: <Staff> View statistic](#Toc410477367) 47

[Figure 23: <Staff> Approve](#Toc410477368) 48

[Figure 24: <Staff> Add interview record](#Toc410477369) 50

[Figure 25: <Staff>Create interview](#Toc410477370) 52

[Figure 26: <Staff> Update interview](#Toc410477371) 54

[Figure 27: <Manager>Overview Use Case](#Toc410477372) 56

[Figure 28: <Company manager>Manage vacancy](#Toc410477373) 56

[Figure 29: <Company manager>Create vacancy](#Toc410477374) 57

[Figure 30: <Company manager> Edit vacancy](#Toc410477375) 59

[Figure 31: <Company manager> Enable/Disable vacancy](#Toc410477376) 61

[Figure 32: <Company manager>Manage staff](#Toc410477377) 63

[Figure 33: <Company manager>Add staff](#Toc410477378) 63

[Figure 34: <Company manager> Update staff info](#Toc410477379) 66

[Figure 35: <Company manager>Active/Deactive staff](#Toc410477380) 68

[Figure 36: <Company manager> Manage interview](#Toc410477381) 70

[Figure 37: <Company manager>View statistic](#Toc410477382) 72

[Figure 38: <Company Manager>Manage Thread](#Toc410477383) 73

[Figure 39: <Company Manager> Create Thread](#Toc410477384) 74

[Figure 40: <Company Manager> Edit Thread](#Toc410477385) 76

[Figure 41: <Company Manager> Delete Thread](#Toc410477386) 78

[Figure 42: <Company Manager> Register Event](#Toc410477387) 80

[Figure 43: <Company Manager>Manage QR Code](#Toc410477388) 82

[Figure 44: <Administrator>Create QR Code](#Toc410477389) 82

[Figure 45: <Staff>Overview Use Case](#Toc410477390) 84

[Figure 46: <Student>Create Resume](#Toc410477391) 84

[Figure 47: <Student>Create Resume](#Toc410477392) 85

[Figure 48: <Student>Update Resume](#Toc410477393) 86

[Figure 49: <Student>Delete Resume](#Toc410477394) 88

[Figure 50: <Student>Search vacancy](#Toc410477395) 89

[Figure 51: <Student> Attempt Event](#Toc410477396) 90

[Figure 52: <Staff>Overview Use Case](#Toc410477397) 92

[Figure 53: Conceptual Class Diagram](#Toc410477398) 94

**Definitions, Acronyms, and Abbreviations**

|  |  |
| --- | --- |
| **Name** | **Definition** |
| **PSIB** | **Place Suggesting Intelligent Bot** |
|  |  |

# Report No. 1 Introduction

## Project Information

* Project name:  **Place Suggesting Intelligent Bot**
* Project Code: **PSIB**
* Product Type: **Web Application & Engine**
* Start Date: **09-May-2016**
* End Date: **19-August-2016**

## Introduction

In this document, we introduce a solution for users who want to find a coffee shop or small shop that its foods are delicious and affordable in a quickly way when they are locating somewhere. Facebook and Facebook Messenger are more and more popular nowadays and appear on every platform. E-commerce pages on Facebook was common with every people. But those E-commerce pages on Facebook do not have function auto answer and the owners have to answer their customer by themself. It takes a lot of time and sometimes they can missing the inboxes. Based on the demand of the users, we proposed a solution for them in Vietnam.

We build a system, which help the owner E-commerce pages on Facebook to solve their current problems. In the process of analysis, we believe that Message Platform of Facebook and Api.ai are popular and suitable to build the bot can answer message automatically and intelligently. By using Facebook Messenger, user can ask the page directly and the bot will reply right after that. Beside of that we also provide a function to train for the bot, make the bot more intelligent.

This document also describes our working process in 4 months, includes our perspective in the system, component designs and detailed core workflows. We hope that our solution will make a big change in Vietnam with the most convenient with the lastest tech trend that will go nowhere.

## Current Situation

Users are locating in somewhere, and they want to find some place to meet their friend at coffee shop or they just want to eat something, but they do not know where to go.

Some popular behaviors of users are using the smartphone with 3G to search on google or “around me”, foody, diadiemanuong applications; or asking residents; or asking friends, … However, those tasks are time-consuming and not very effective; and sometimes you are shy, scary, or not confident to communicate, or your language is a barrier to yourself. However, those applications are not supported their API.

## Problem Definition

Below are advantages / disadvantages of the current situation:

* **Advantages:**
* Natural feeling when communicate between human and human.
* **Disadvantages:**
* Spend too much cost: the product or service provider must have staff to contact with customer when customer need information. Staff may always available to solve all customer demand. It cost time and human resource.
* Information is overview: due to the huge number of customer. The staff just can provide the information in a brief way.
* Lack/wrong information: when user use search engine to find the place they want. The result may not give a right information they need. There are to much result but less information.
* User must do too much steps: use search engine, ask for friend, contact to provider, view the product, view review. Earch step must be use different tool.
* Bad user experience: user must be use many tool to meet a request. The change between many tool make they tired and give up if they cannot have a information they want.

## Proposed Solution

Our proposed solution is to build a system name “PSIB” that is chat bot engine integrating with Messenger Paltform, which can suggest for customer depend on their demand. We also design a Web application to help user training bot to make it more intelligent.

PSIB system include a Web application and an engine processing bot.

### Feature functions

* Web application (for admin):
* Manage data: set schedule for update and crawler data into database. Beside that, admin can have an action on crawler web to choose which information need to save in database.
* Manage bot: Export json file for api.ai and update core bot in api.ai
* Web application (for user):
* Training bot: user can trainning bot by example and syntax to max it more intelligent.
* Processing engine bot:
* Module process with api.ai: processing with entiy and intent in api.ai, process keyword and collect the response when user ask.
* Module processing with Facebook Messenger Platform: process in tructrured message for send and response, processing with location and connect with api.ai.

### Advantages and disadvantages

#### Advantages

* + - * Can integrate with existed applition, Facebook Messenger, do not need to build the new one.
      * Multiple platforms – build one, run any where.
      * Can train the bot more and more intelligent: the system can collect common requests from clients. It helps user can train the system how to react with it.
      * Reduce time, cost and human resources: users do need to reply their clients request, the bot can do it automatically.

#### Disadvantages

* + - * System requires user must have the Internet to communicate with it.
      * Require initial data before the system can operate.
      * User need to train the system manually, it can not learn by it-self yet.

## Functional Requirements

Function requirement of the system are listed as below:

* **Admin component**
* Manage data: Schedule and select data to crawler.
* Manage bot: manage interact with api and has tool to export json.
* **User component**
* Training bot: training bot by example. Update bot intent and entity using api.ai sdk.
* **Bot component**
* Api.ai: connect with api and processing response.
* Facebook Messenger platform: processing message with reply and response. Display structured message and processing location of user.

## Role and Responsibility

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Full Name** | **Role** | **Position** | **Contact** |
| 1 | Kiểu Trọng Khánh | Project Manager | Supervisor | KhanhKT@fpt.edu.vn |
| 2 | Huỳnh Thành Đạt | Developer | Leader | DatHTSE61273@fpt.edu.vn |
| 3 | Trương Châu Minh Huy | Developer | Member | HuyTCMSE61175@fpt.edu.vn |
| 4 | Trần Văn Thành | Developer | Member | ThanhTVSE61373@fpt.edu.vn |
| 5 | Đào Công Thuận | Developer | Member | ThuanDC60487@fpt.edu.vn |

# Report No.2 Software Project Management Plan

## Problem Definition

### Name of this Capstone Project

Official name: Place Suggesting Intelligent Bot

Vietnamese name: Chat bot gợi ý địa điểm thông minh

Abbreviation: PSIB

### Problem Abstract

As current in Vietnam, E – Commerce are growing up and up every day. Many E – Commerce pages on Facebook appear. Using page on Facebook are convinient with the owner because they do not need to spend money to build another website with the same demand, but it always takes too much time to answer the inboxes of customers. So we use the Facebook Message Platform and Api.ai building a bot to resolve this problem. The bot can answer automatically and give information to customers by Facebook Messenger. Customers is not asked to install any application else.

We also provide a website for the owners of bot to train their bot. The bot can study the symnonym to become more intelligent and answer many type question of customer. For example, if customer ask: “hôm nay tôi thèm món Nhật” or “Chúng tôi muốn thưởng thức đồ ăn Nhật”, the bot will understand and give customer suggestion about Japanese food.

In addition, we also provide an website crawler to get data from other website to make our data become more powerful.

### Project Overview

#### **Current Situation**

Below are the problems encountered in this project:

**Absence of team members:** team members casn get sick or unexpected problems.

**Language understand:** do not support well Vietnamese and need to train a lot.

**Data Dependant:** do not own data and depend on data from other website, depend on Api.ai framework

**Lack of syntax:** cannot cover all syntax of language.

**Around me function:** cannot get exactly location of customers.

#### **The Proposed System**

According to the framework researches, we found out that the Facebook Message Platform is very capable of resolve the problem for E – Commerce page on Facebook.

  The basic idea is to use Api.ai to build an engine for the bot and use Facebook Message Platform to transfer message from bot to customer.

We also build a highly available web server to maintain the main system to work 24/7 and make sure that if customer need ask for the information from the bot, there will be always available.

Our system includes two subsystems:

* Engine for bot process natural language from customer and use Facebook Message Platform to transfer message.
* Web crawler to get data from other website and standardize data to add to database, support the owner training their bot.
  + - * 1. **Website Website**

Main web application is common communication portal for administrators to craw data from other website and the place for the owner training their bot.

For administrators:

* Get all data from the website
* Get data following the post on the website.

For the bot owner:

* Training bot with synonyms.

##### **Engine for Bot**

The engine process natural language from customer using Api.ai framework. Convert the sentence into word entities, understand it as an request of customer and give back the customer a suitable response.

The engine can study the synonym of existed entities when the ownsers train for bot.

Processing with messenger platform, engine can send and receive message with structure and handle many type of message from text message, attactments,…

#### **Boundaries of the System**

The system contains the engine and web application for administrators, below is boundaries of system:

* The engine combines Facebook Message Platform and Api.ai to process natural language and transfer response message.
* Bot will answer the customer automatically with the natural language syntax **(**Either English or Vietnamese**)** of the users that wants to ask information.
* The engine can allow to train the bot to make it more intelligent.
* A Web Crawler is used to extract data on foody.vn and diadiemanuong.com
  + - 1. **Future plan**

Current engine only supports the owner to training their bot by using the synonyms. Training work are still manually. So we design the engine easily to scale bigger:

* **Training with syntax:** The owner can train new syntax and control their bot completely.
* **Learning automatically:** because of learning synonyms are still manually. The owner have to spend their time for training and it is not usable. The engine will be build for bot can learning automanically if there is new words from customer.

#### **Development Environment**

##### **Hardware requirements**

* For website:

|  |  |  |
| --- | --- | --- |
|  | **Minimum** | **Recommended** |
| Operating system | Windows XP SP3 | Windows 8 Professional |
| Internet Connection | Cable, Wireless | Cable, Wireless |
| Computer Processor | Intel Core 2 Duo | Intel Core i5 |
| Computer Memory | 2GB | 4GB |

##### **Software requirements**

* Operating system: Windows, Mac OS, Linux.
* Development language: Java (JDK7).
* DBMS: MySQL.
* IDE: WebStorm, Spring Tool Suite.
* Source Code Control: Git.
* Communication: Facebook, Slack, Skype.

## Project organization

### Software Process Model

The model use for developing this project is Scrum model. We choose this model because of following reason:

* Scrum define a flexible, holistic product development strategy where a development team works as a unit to reach a common goal.
* For our solution, it is very suitable for us to self-organize and face-to-face communication between team member.
* Easy to change requirement.

OVERVIEW DIAGRAM

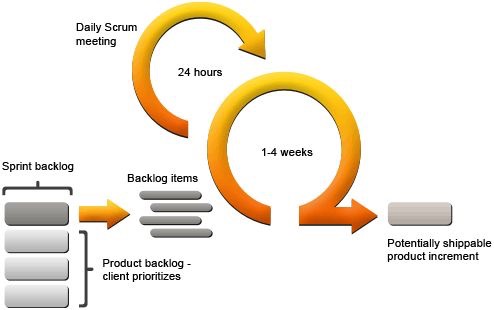


Figure 3-1 Scrum model

(Image source: http://www.certiconglobal.com/)

### Roles and responsibilities

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Full name** | **Team Role** | **Scrum Team Role** | **Responsibilities** |
| 1 | Kiều Trọng Khánh | Supervisor | Product Owner | * Specify user requirements * Control the development process * Support in technique and business logic. |
| 2. | Huỳnh Thành Đạt | Team Leader  Developer | Scrum Master | * Manage process. * Planning Scrum implementations. * Create system architecture design. * Support technique. * Create product Increment. * Create test plan. * Configure developing environment. * Code. * Test. * Write document and reports. |
| 3. | Trương Châu Minh Huy | Developer | Scrum Team Member | * Commit individual product on time * Support each other to complete team work * Create product Increment. * Code. * Test. * Write document and reports. |
| 4 | Trần Văn Thành | Developer | Scrum Team Member | * Commit individual product on time * Support each other to complete team work * Code * Test |
| 5 | Đào Công Thuận | Developer | Scrum Team Member | * Commit individual product on time * Support each other to complete team work * Code * Test |

### Tools and Techniques

|  |  |  |
| --- | --- | --- |
| Technique | Front end | * JSP * HTML * JavaScript * CSS |
| Back end | * Java * Spring MVC * Selenium * JavaScript ECMAScript 5.1 * Apiai v2.0.3 * Express v4.13.4 |
| Web servers | | * TomCat portable v7.0 * NodeJS v4.4.4 |
| Database Management System | | * MySQL |
| Developing tool | | * JetBrains WebStorm 2016.1.1 * Spring Tool Suite |
| Source Control | | * Git (server github.com) |

## Project Management Plan

### Product Backlog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Story  ID** | **Features** | **Task  ID** | **Task description** | **Sprint** |
| 1 | Create Product Backlog | 1.1 | Create Product backlog | 1 |
| 2 | Create Introduction document | 2.1 | Create Introduction document | 1 |
| 2.2 | Review Introduction document | 1 |
| 3 | Learning facebook api and api.ai | 3.1 | Learning facebook api and api.ai | 1 |
| 4 | Create mockup UI | 3.1 | Log in UI | 1 |
| 3.2 | Daskboard UI | 1 |
| 3.3 | Manage/ Config schedule crawler UI | 1 |
| 3.5 | Training bot UI | 1 |
| 4 | Create Software Project Management Plan | 4.1 | Problem definition | 2 |
| 4.2 | Project organization | 2 |
| 4.3 | Project management plan | 2 |
| 4.4 | Coding convention | 2 |
| 5 | Create Software Requirements Specifications | 5.1 | User Requirement Specification | 3 |
| 5.2 | External Interface Requirement | 3 |
| 5.3 | Use case diagram | 2 |
| 5.4 | Software System Attributes | 3 |
| 5.5 | Database diagram | 2 |
| 6 | Create Software Design Description | 6.1 | Design Overview | 4 |
| 6.2 | System Architectural Design | 4 |
| 6.3 | Component Diagram | 4 |
| 6.4 | Detailed Description of Components | 4 |
| 6.5 | Sequence Diagram | 3 |
| 6.6 | User Interface Design | 2 |
| 6.7 | Database Design | 4 |

### Deliverables

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Deliverable** | **Delivery date** | **Delivery location** | **Note** |
| 1 | Introduction Document | 16/5/2016 | FU – CMS | Report No.1 |
| 2 | Software Project Management Plan | 23/5/2016 | FU – CMS | Report No.2 |

\* For each sprint, deliverables is *potentially shippable product*, which can be a part of documents or runnable software application which is implementation of some program features.

### All Meeting Minutes

All Meeting Minutes are put in folder

## Coding Convention

Java: Using to develop crawler web application and bot training web application

Summary:

* Naming Convention:
* For variable’s name, use camel case. E.g.: minValue, maxValue…
* For function name, use camel case. E.g.: searchProduct…
* For class name, use Pascal case. Ex: Product, Order…
* Layout Convention:
* Write only one statement/declaration per line.
* Indent continuation one tab stop (four spaces).
* Add at least one blank line between method definitions and property definitions.
* Use parentheses to make clauses in an expression apparent.
* Commenting Convention:
* Place the comment on a separate line, not at the end of a line of code.
* Begin comment text with an uppercase letter.
* End comment text with a period.
* Insert one space between the comment delimiter (//) and the comment text.
* Do not create formatted blocks of asterisks around comments.

NodeJS: Using to develop service API

Summary:

* Naming Convention:
* For variable’s name, use camel case. E.g.: minValue, maxValue…
* For properties name, user camel case E.g.: minValue, maxValue…
* For function name, use camel case. E.g.: searchProduct…
* For class name, use upper camel case. Ex: BankAccount, OrderDetails…
* For constance, use upper case: Ex: KEY\_MAP, PAGE\_ACCESS\_KEY....
* Layout Convention:
* Write only one statement/declaration per line.
* Indent continuation one tab stop (four spaces).
* Add at least one blank line between method definitions and property definitions.
* Use parentheses to make clauses in an expression apparent.
* Limited 80 charaters per line.
* No trailling whitespace.
* Use single quotes. E.g.: var foo=’bar’,
* Commenting Convention:
* Place the comment on a separate line, not at the end of a line of code.
* Begin comment text with an uppercase letter.
* End comment text with a period.
* Insert one space between the comment delimiter (//) and the comment text.
* Do not create formatted blocks of asterisks around comments.