

Software Architecture Document

Improvement for Lunch Order Module

Prepared by: **Thu Nguyen**
Jirakit Paitoonnaramit

September, 2017

Revision History

Date	Version	Description
<5/10/2017>	<1.0>	First Draft of the Software Architecture Doc.

Table of Contents

1. Introduction	4
1.1 Purpose	4
1.2 Scope	4
1.3 Overview	4
2. Architectural Representation	4
3. Architectural Goals and Constraints	5
4. Use-Case View	5
4.1 Use-Case Diagram	6
4.1.1 Create new order	6
4.1.2 Feeling lucky	6
4.1.3 Favorite order	6
4.1.4 Choose order	6
4.1.5 Send email	6
4.1.6 View history	7
4.1.7 Control Account	7
4.1.8 Manage product	7
4.1.9 Manage categories	7
4.2 Activity Diagram	7
4.2.1 Favorite Order Activity Diagram	7
4.2.2 Sending Email Activity Diagram	7
4.3.1 Module Diagram	8
4.3.2 Class Diagram	9

Software Architecture Document

1. Introduction

This introduction provides an overview of the entire *Software Architecture Document* for the Conservation Planning Support System. It includes the purpose, scope, definitions, acronyms, abbreviations, references, and overview of the system.

1.1 Purpose

This document provides a comprehensive architectural overview of the system, using a number of different architectural views to depict different aspects of the system. It is intended to capture and convey the significant architectural decisions which have been made on the system.

1.2 Scope

This Software Architecture Document provides an architectural overview of the Lunch module. The Lunch module is being developed by SSE student to add some function.

1.3 Overview

This document consists of 4 sections, which are described below:

- Section 1 is simply an introduction to the software architecture
- Section 2 addresses the goals and constraints of the system's architecture
- Section 3 describes the architectural representation of the system.
- Section 4 describes the process by use-case diagram, activity diagram and architecture overview.

2. Architectural Representation

This document presents the architecture as a series of views; use case view, activity view, architectural view.

3. Architectural Goals and Constraints

There are some key requirements and system constraints that have a significant bearing on the architecture. They are:

- 1 The software is based on Odoo, therefore the software has to follow the MVC design pattern as other modules of Odoo.
- 2 The existing lunch module in Odoo must be interfaced with to support user to choose favorite order and support vendor by sending email to vendor.

4. Use-Case View

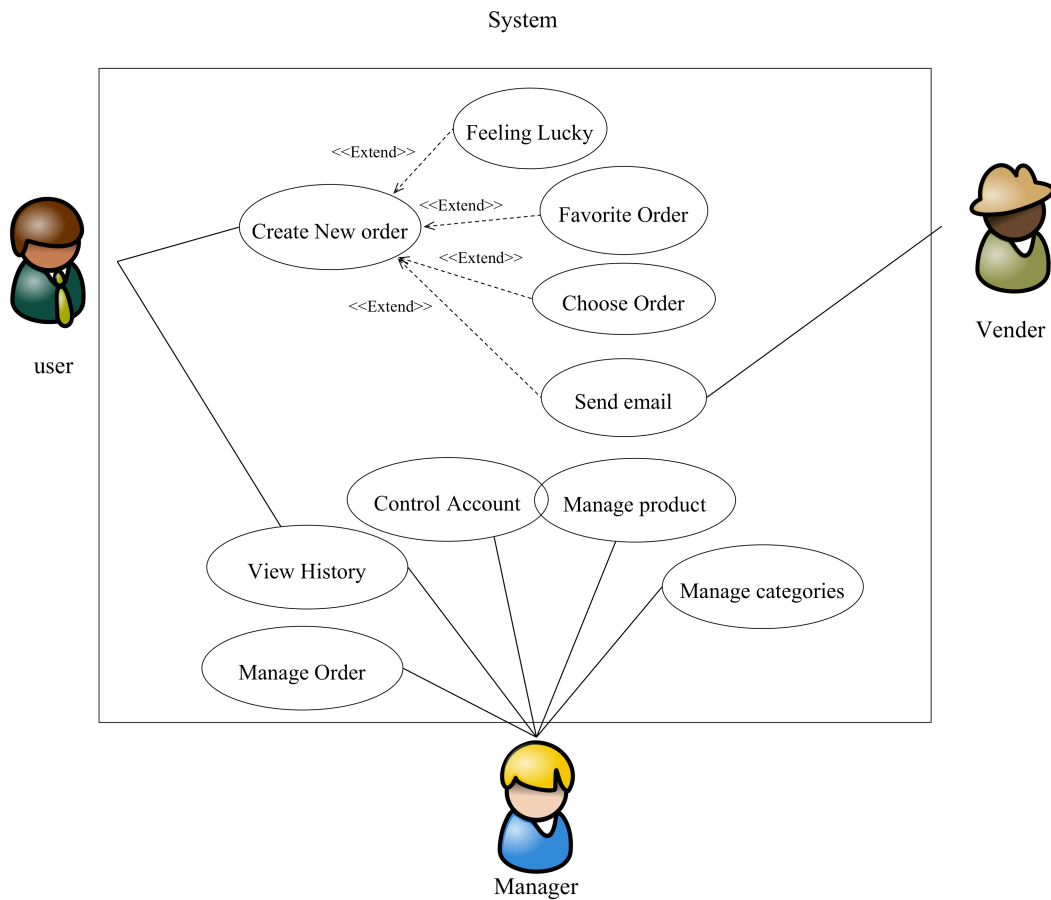
A description of the use-case view of the software architecture. The Use Case View is important input to the selection of the set of scenarios and/or use cases that are the focus of an iteration. It describes the set of scenarios and/or use cases that represent some significant, central functionality. It also describes the set of scenarios and/or use cases that have a substantial architectural coverage (that exercise many architectural elements) or that stress or illustrate a specific, delicate point of the architecture.

The Lunch module use cases are:

- Create new order
- Feeling lucky
- Favorite order
- Choose order
- Send email
- View history
- Control Account
- Manage product
- Manage categories

These use cases are initiated by the user and vendor.

4.1 Use-Case Diagram



Actor : user

4.1.1 Create new order

Brief Description : This use case describes how the user create a new lunch order by several method.

4.1.2 Feeling lucky

Brief Description : This use case describes how the user create a new lunch order by random from vendors and budget limit.

4.1.3 Favorite order

Brief Description : This use case describes how the user create a new lunch order by choose from user's favorite order(sort by most order menu).

4.1.4 Choose order

Brief Description : This use case describes how the user create a new lunch order by manually add menu mane in order line.

Actor : vendor

4.1.5 Send email

Brief Description : This use case describes how the system send all of order sort by vender via email to a vender.

Actor : manager

4.1.6 View history

Brief Description : This use case describes how the manager view employee's order and manage the employee payment.

4.1.7 Control Account

Brief Description : This use case describes how the manager add, edit and remove vendor account.

4.1.8 Manage product

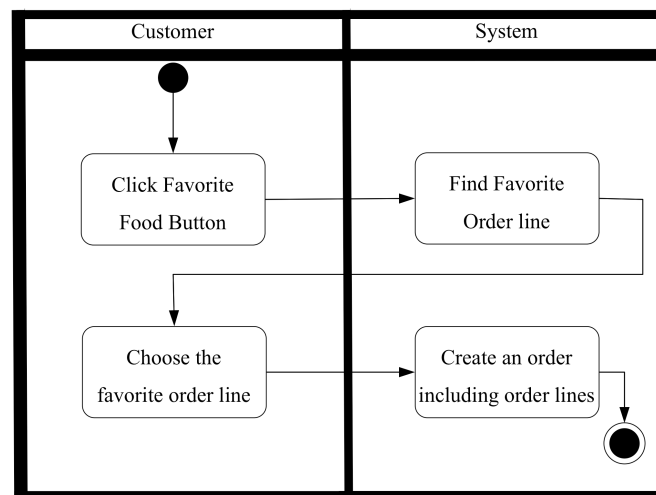
Brief Description : This use case describes how the manager add, edit and remove product.

4.1.9 Manage categories

Brief Description : This use case describes how the manager add, edit and remove order categories.

4.2 Activity Diagram

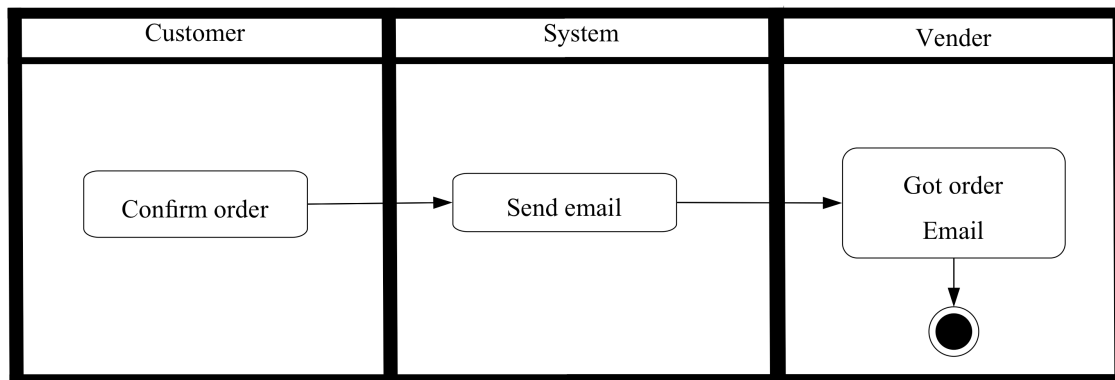
4.2.1 Favorite Order Activity Diagram



Favorite Order Activity Diagram showing the process of how to order by favorite order method. First, user have to click on Favorite menu button. Then server will find a favorite of user by sorting the most order menu and send data to wizard. Finally, user choosing the menu and order has created.

4.2.2 Sending Email Activity Diagram

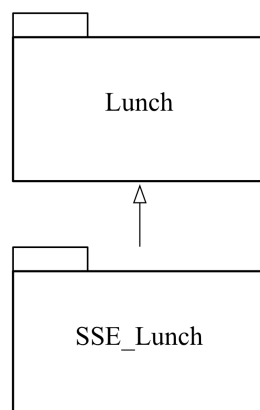
Sending Email Activity Diagram showing the process of emailing the orders to vendor. When at a setting time to send the email. System will get all of data of the menu then sorting by vendor and send it to the vender.



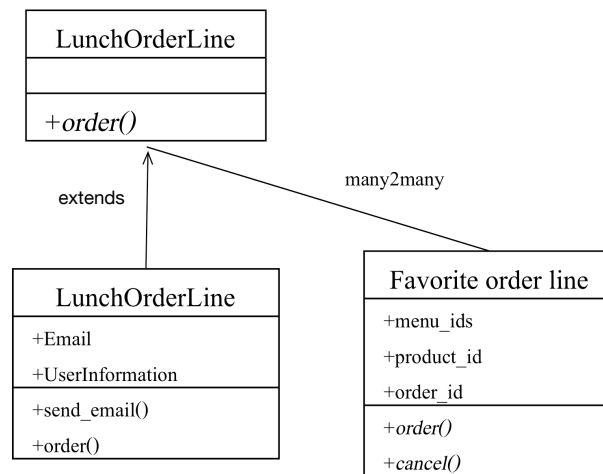
4.3 Architecture Overview

4.3.1 Module Diagram

In this design. SSE_lunch has been create and inherit to Lunch module class as shown on picture below



4.3.2 Class Diagram



4.3.2.1 LunchOrder Line: The original class to order the lunch.

4.3.2.2 LunchOrderLineSendEmail : The class for send email to the vender.

4.3.2.3 Favorite order line : The class for calculate favorite order.