Food Management System

Version <1.0>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <24/12/2020> | <1.1> | Final Draft | Nguyen Thi Ngoc Anh |

Table of Contents

1. Introduction 2

1.1 Purpose 2

1.2 Scope 2

1.3 Definitions, Acronyms, and Abbreviations 2

1.4 References 2

1.5 Overview 2

2. Flow of Events—Design 2

3. Derived Requirements 2

# Introduction

## Purpose

This document describes how the Edit Order Status Use-Case is realized within the design model, in terms of collaborating objects.

## Scope

This document applies to the Food Management System which will be developed by TAT Restaurant.

## Definitions, Acronyms, and Abbreviations

None.

## References

None.

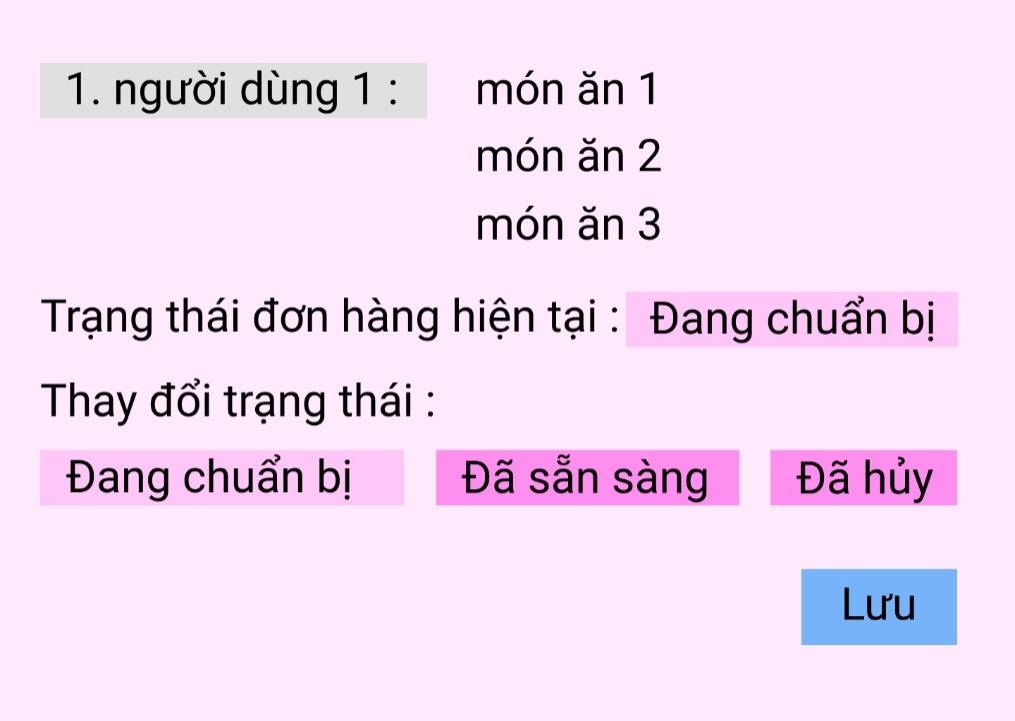
## Overview

In the following section, Use-Case Realization Specification of the Edit Order Status Use-Case of the Food Management System is provided in detail. The first section is a textual description of the Use-Case specification. The following section contains diagrams (sequence and collaboration diagrams) describing how the use case is realized in terms of collaborating objects. The third section includes class diagrams with relationships that participate in the realization of the use case. The last section is an analysis of all requirements, such as non-functional requirements, on the use-case realization that are not considered in the design model, but that need to be taken care of during implementation.

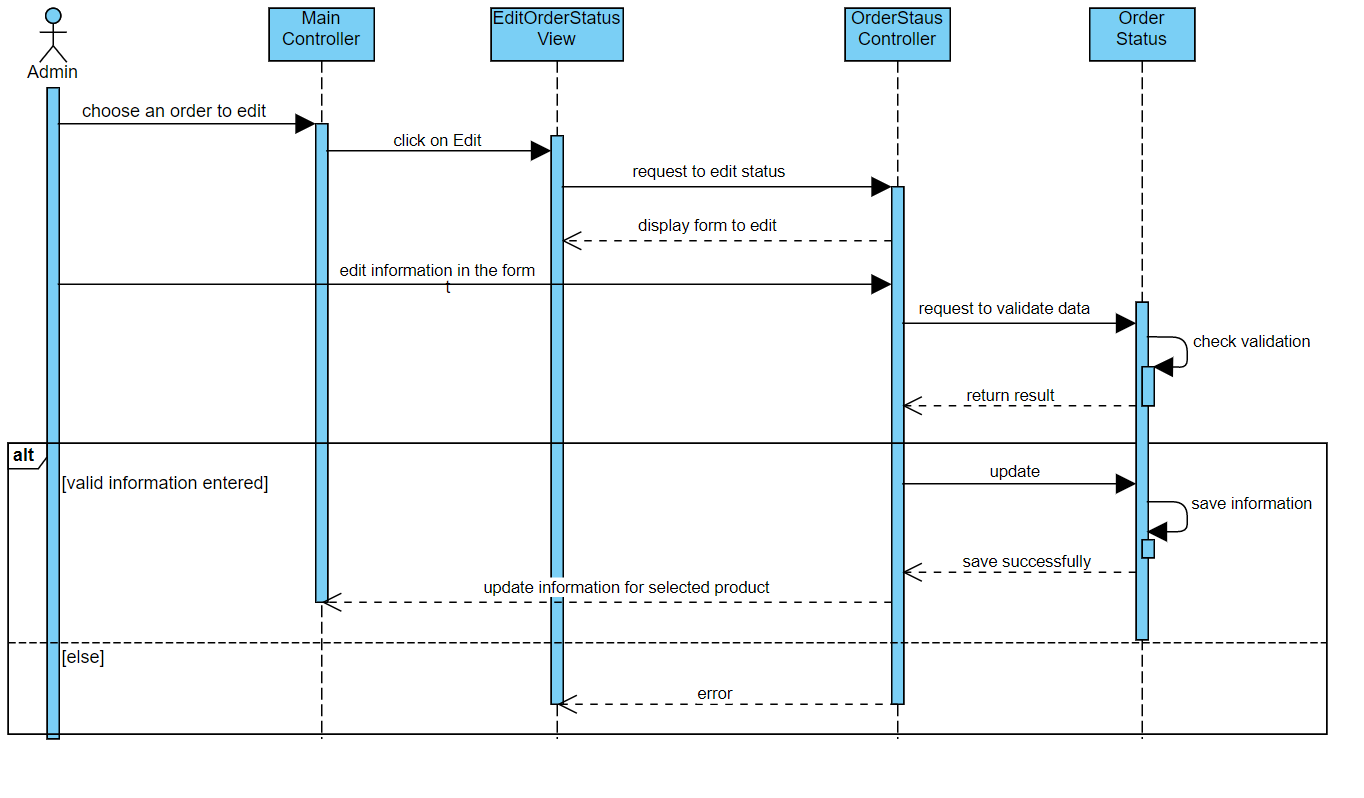
# Flow of Events—Design

**2.1. Use Case Specification (script)**

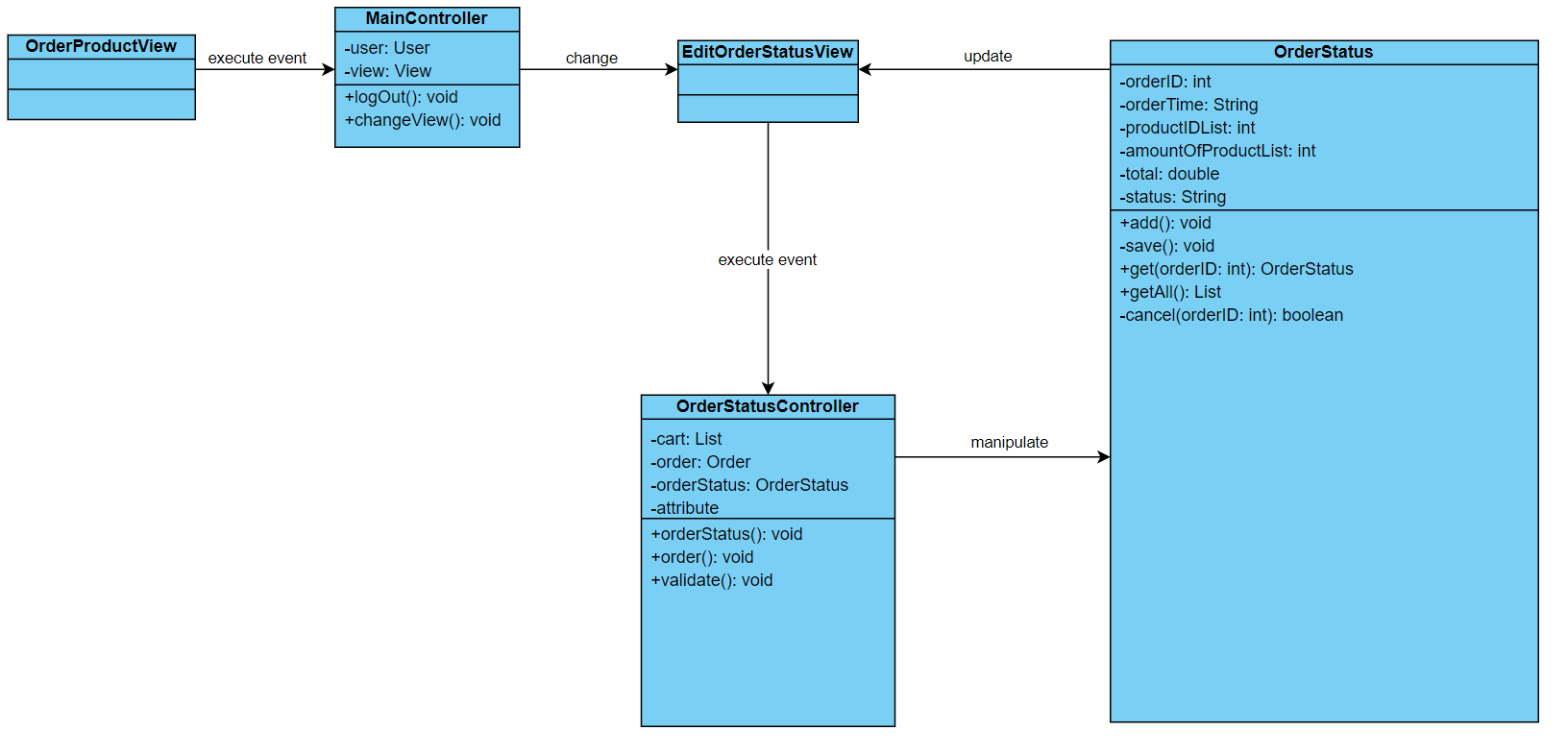
|  |  |
| --- | --- |
| **Use Case** | Edit Order Status |
| **Description** | Admin changes the status of orders that have been ordered |
| **Actor(s)** | Admin |
| **Flows of Events** |  |
| **Basic Flow** | |
| 1. The user goes to edit order status page.  2. The user updates status for an order.  3. The user submits information.  4. The system verifies information, and save order status with specified information to database | |
| **Alternative Flows** | |
| **Title** | **Description** |
| Invalid Information Entered | 1. User enter invalid information.  2. System displays information with appropriate message to correct invalid information. |
| **Pre-conditions** | |
| The user is logged in. | |
| **Post-conditions** | |
| **Title** | **Description** |
| Success | Success message is prompted and the order with updated status is saved to database |
| Failure | The user is unable to update status of order in database for one or more reasons and an error is prompt. |
| **Extension Points** | |
| Cancel Order. | |

**

**2.2. Sequence Diagram:**



**2.3. Class Diagram:**



# Derived Requirements

None.