Online shopping system (Shoppe)

Software Architecture Document

Version <1.0>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 5/12/2021 | 1.0 | Final version | Nguyễn Xuân Bách |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

1.1 Purpose 4

1.2 Scope 4

1.3 Definitions, Acronyms, and Abbreviations 4

1.4 References 4

1.5 Overview 4

2. Architectural Representation 4

3. Architectural Goals and Constraints 4

4. Use-Case View 5

4.1 Use-Case Realizations 6

5. Logical View 10

5.1 Overview 10

5.2 Architecturally Significant Design Packages 11

6. Size and Performance 14

7. Quality 14

Software Architecture Document

# Introduction

## 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.

## Scope

This Software Architecture Document provides an architectural overview of the Online Shopping System. The Online Shopping System is being developed by Shopee.

## Definitions, Acronyms, and Abbreviations

\_ User – a person who use the system, can be seller or buyer

\_ Buyer Account – The default type of account when a person registers an account on Shopee. Buyer Account can only purchase products available on Shopee.

\_ Seller Account – User with the seller account has the buyer account features plus the basic vendor permission such as: create product, edit product, manage process order, advertise product.

## References

None

## Overview

In the following section, architectural design of the Computer Shop Management System is provided in detail. First, the primary software architecture of the system will be defined. Then, there are further discussion about the goals and constraints that will be imposed upon the quality of the final product. In the precedence sections, the key views of the system are demonstrated to depict different aspects of the system. Lastly, criteria concerning with size, performance and quality of the system will be proposed.

# Architectural Representation

This document presents the architecture as a series of views; use case view, logical view, process view and deployment view. There is no separate implementation view described in this document.

# Architectural Goals and Constraints

There are some key requirements and system constraints that have a significant bearing on the architecture.  
They are:  
• The Online Shopping System must be designed to fulfills all system requirements  
specified in requirements definition.  
• The Online Shopping System design must be structured to be robust, easy to change if and when functional requirements change.  
• The Online Shopping System must be designed to allow the re-use of business logic  
across applications; therefore, the design separate the three components: model, view and  
controller.  
• The separation of the three components: model, view and controller are also necessary to provide a convenient cooperation between different development teams.  
• The Online Shopping System will run on a dedicated platform with access to a database.  
• The Online Shopping System website provides most of the content display. An interface to this system must be capable of handling large traffic volumes

# Use-Case View

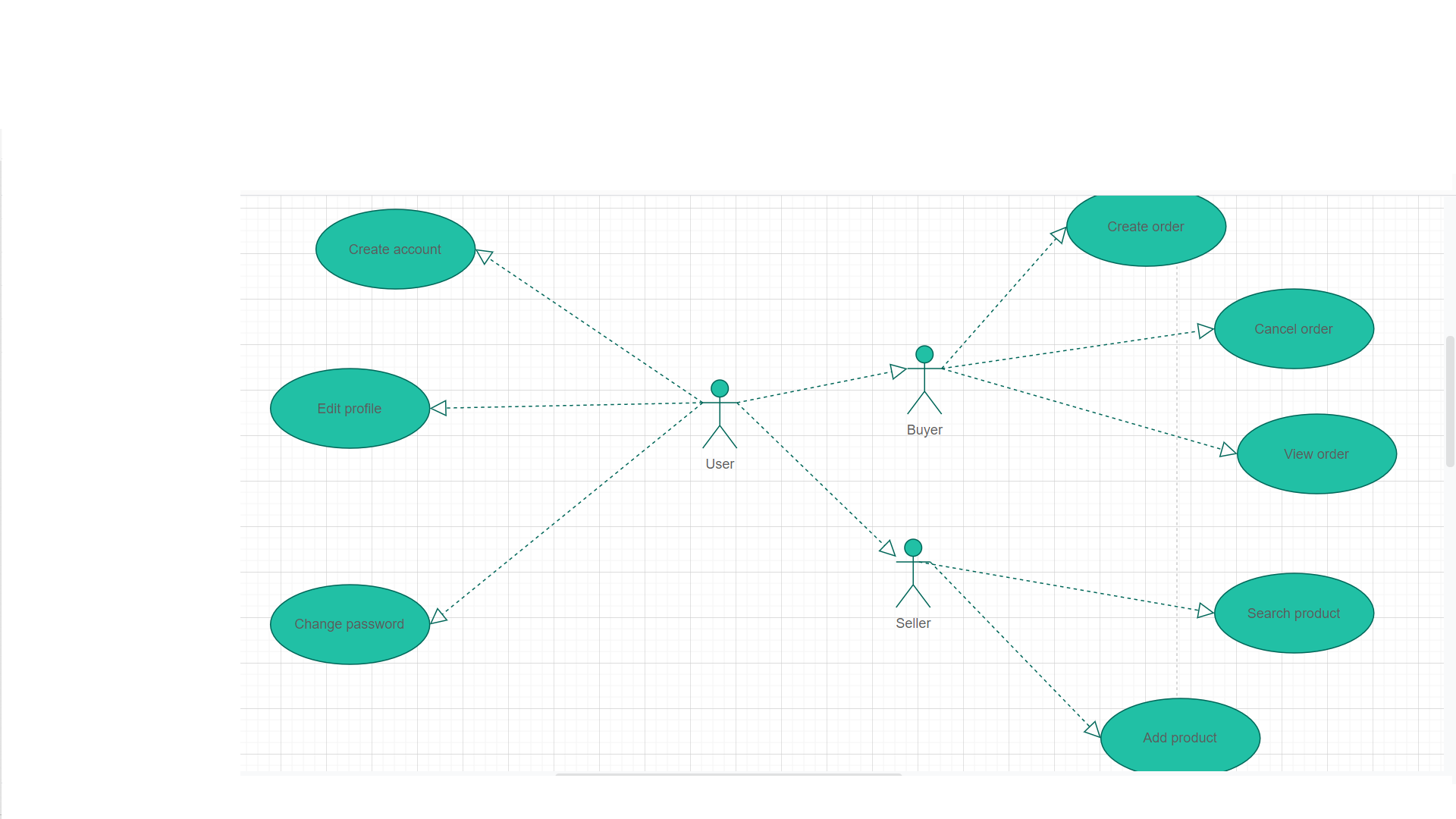
A description of the Use-Case View of the system architecture. The Use Case View is important input to the selection of the set of scenarios and/or use cases that are the focus of an iteration. It describes the set of scenarios and/or use cases that represent some significant, central functionality. 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 significant use cases in this system are listed below:  
♣ Creating Account

♣ Editing Profile  
♣ Changing Password

♣ View Order  
♣ Create Order  
♣ Cancel Order

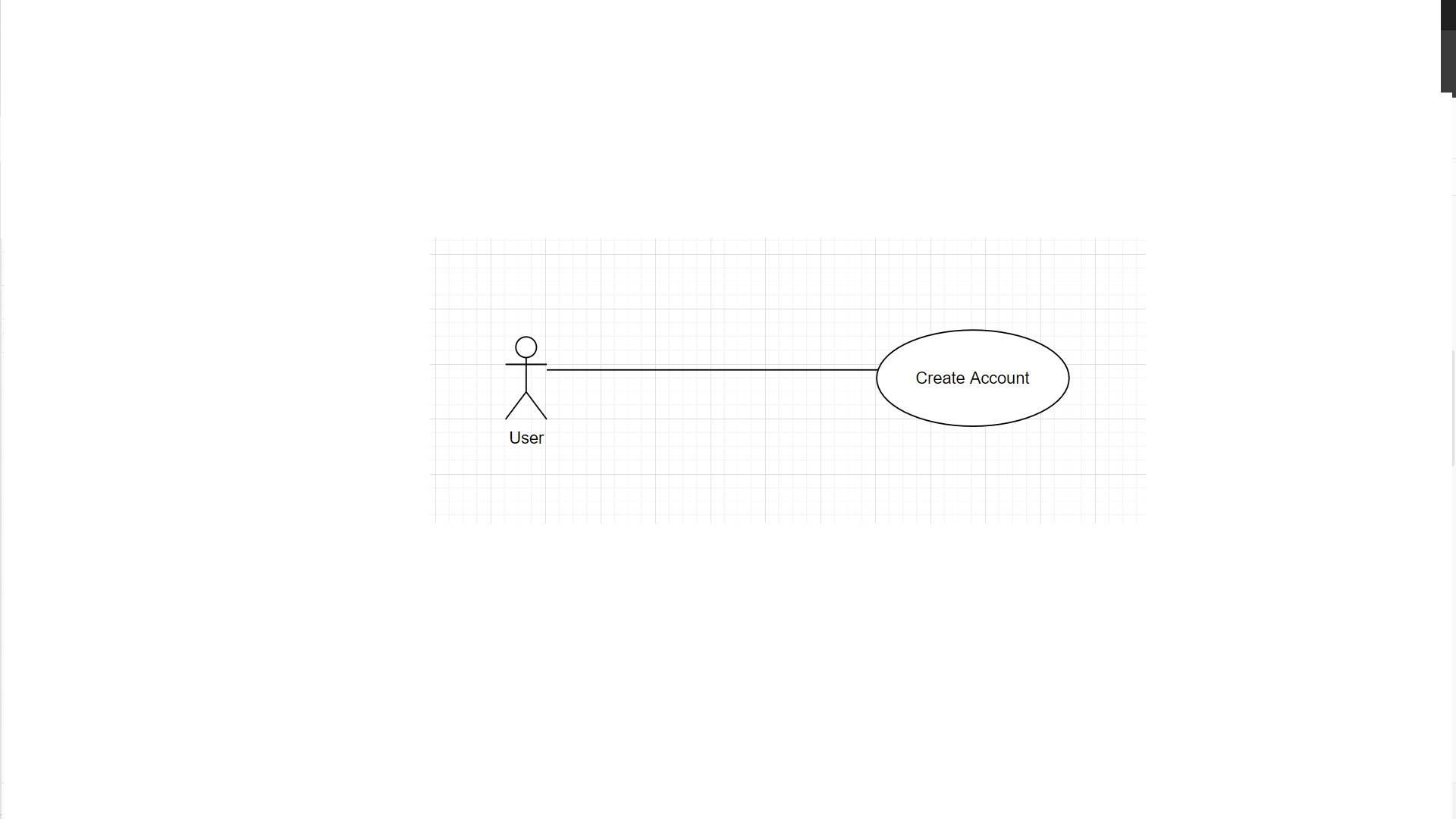
♣ Search product

♣ Add product



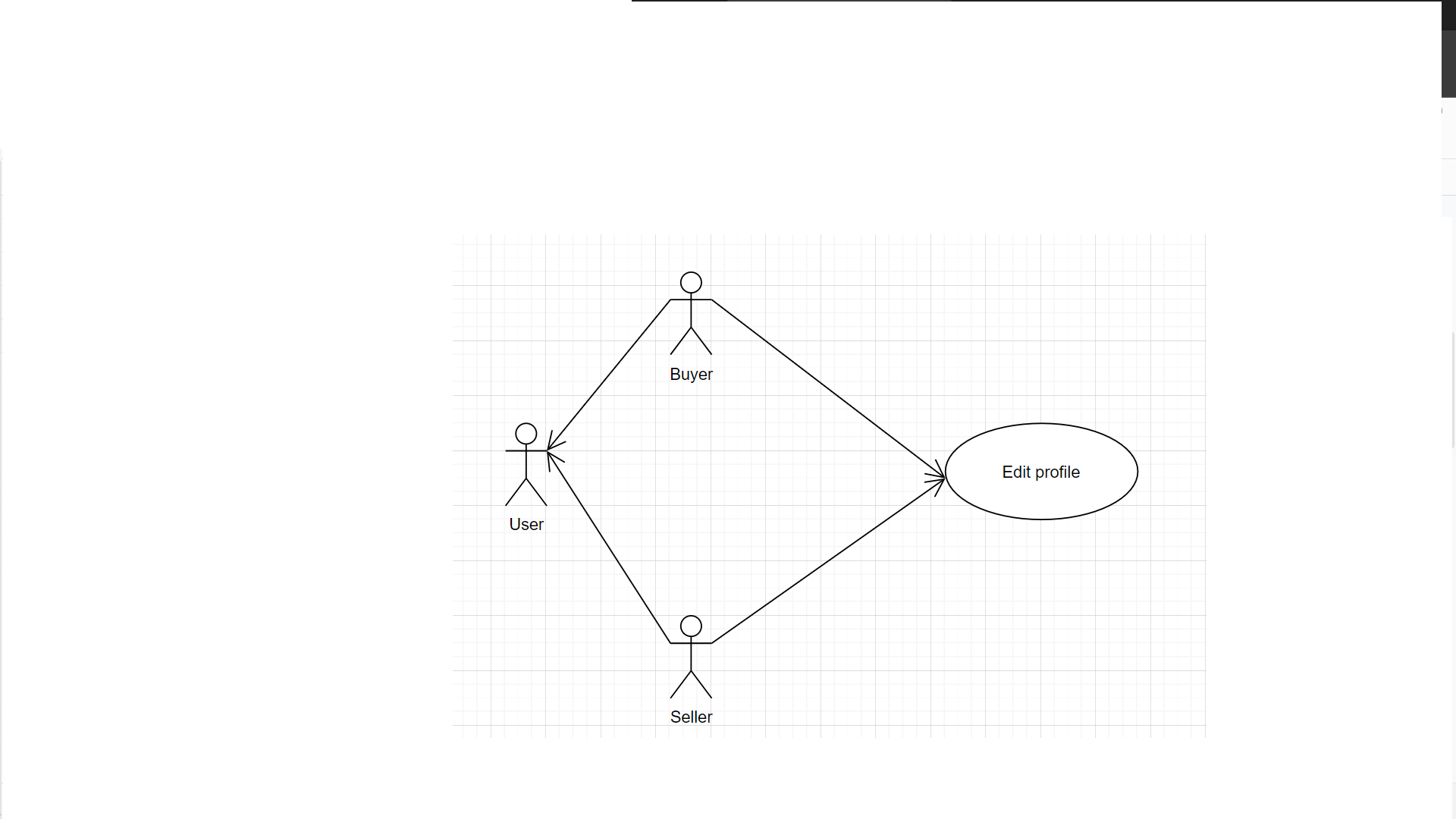
## Use-Case Realizations

**Create Account:**

****

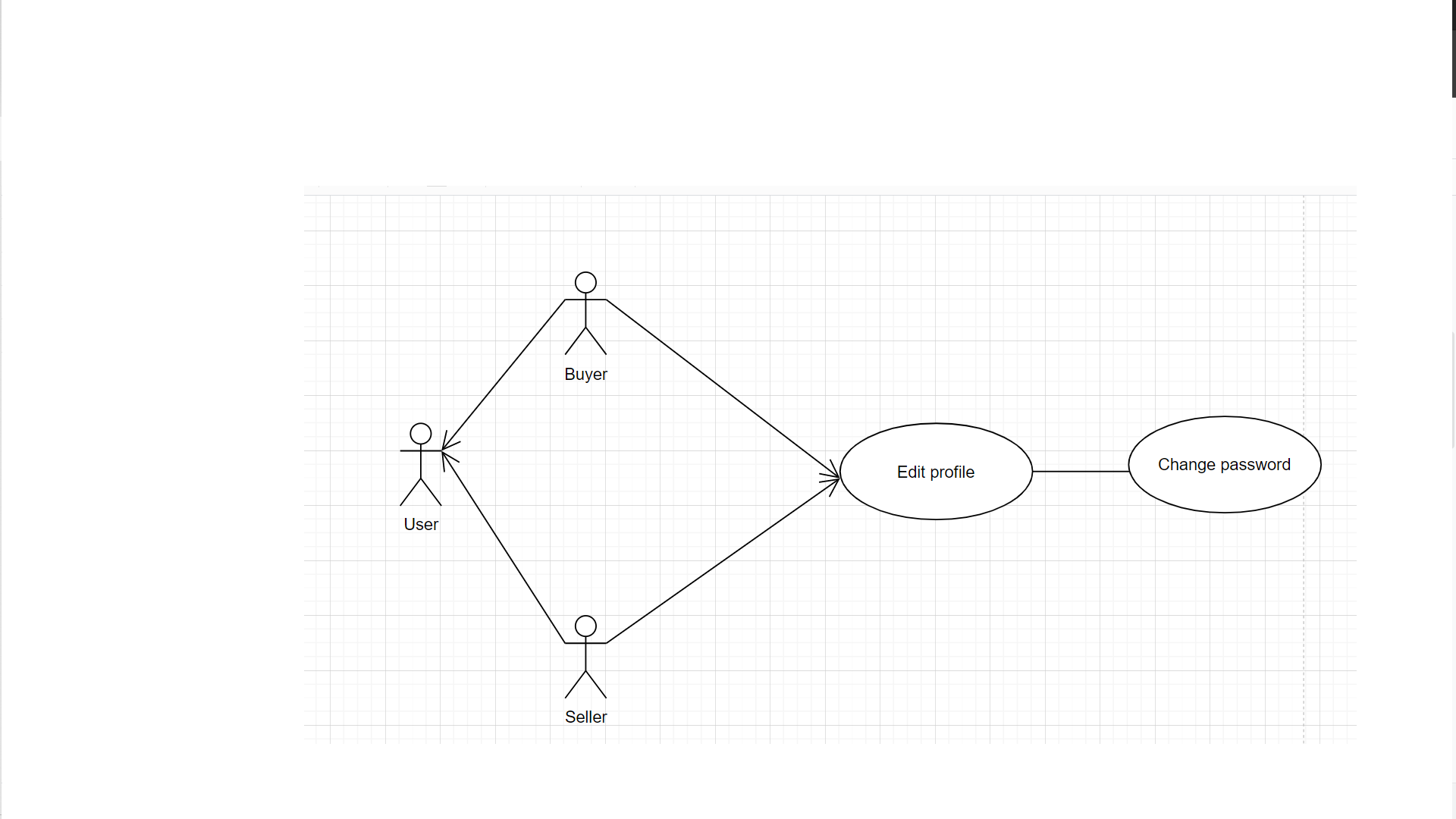
• **Brief Description:** An user creating account.  
• **Specification:** See Use-Case-Realization Specification: Manage Account

**Edit Profile**

****

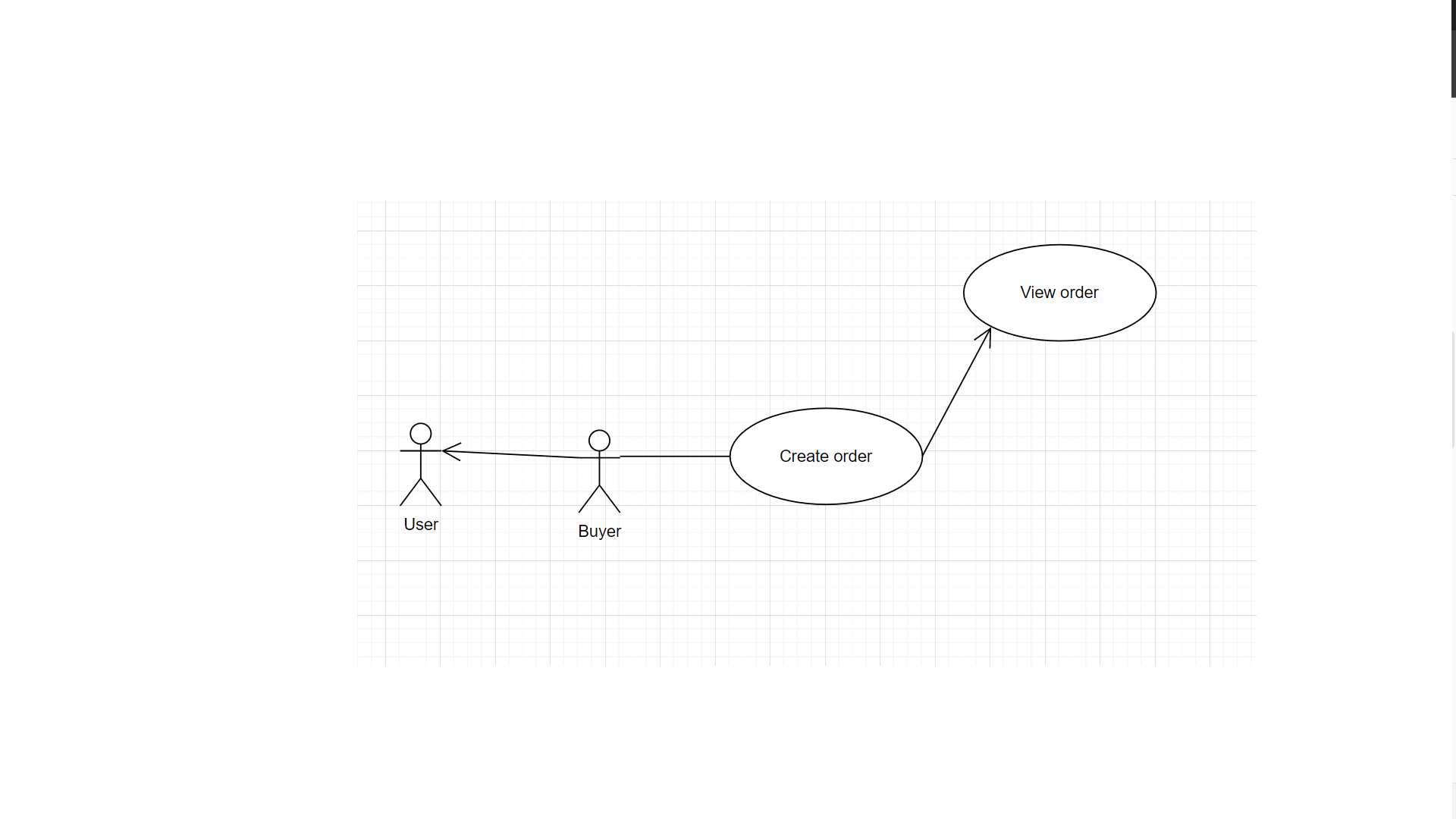
• **Brief Description:** An user editing account.  
• **Specification:** See Use-Case-Realization Specification: Manage Account

**Change Password**

****

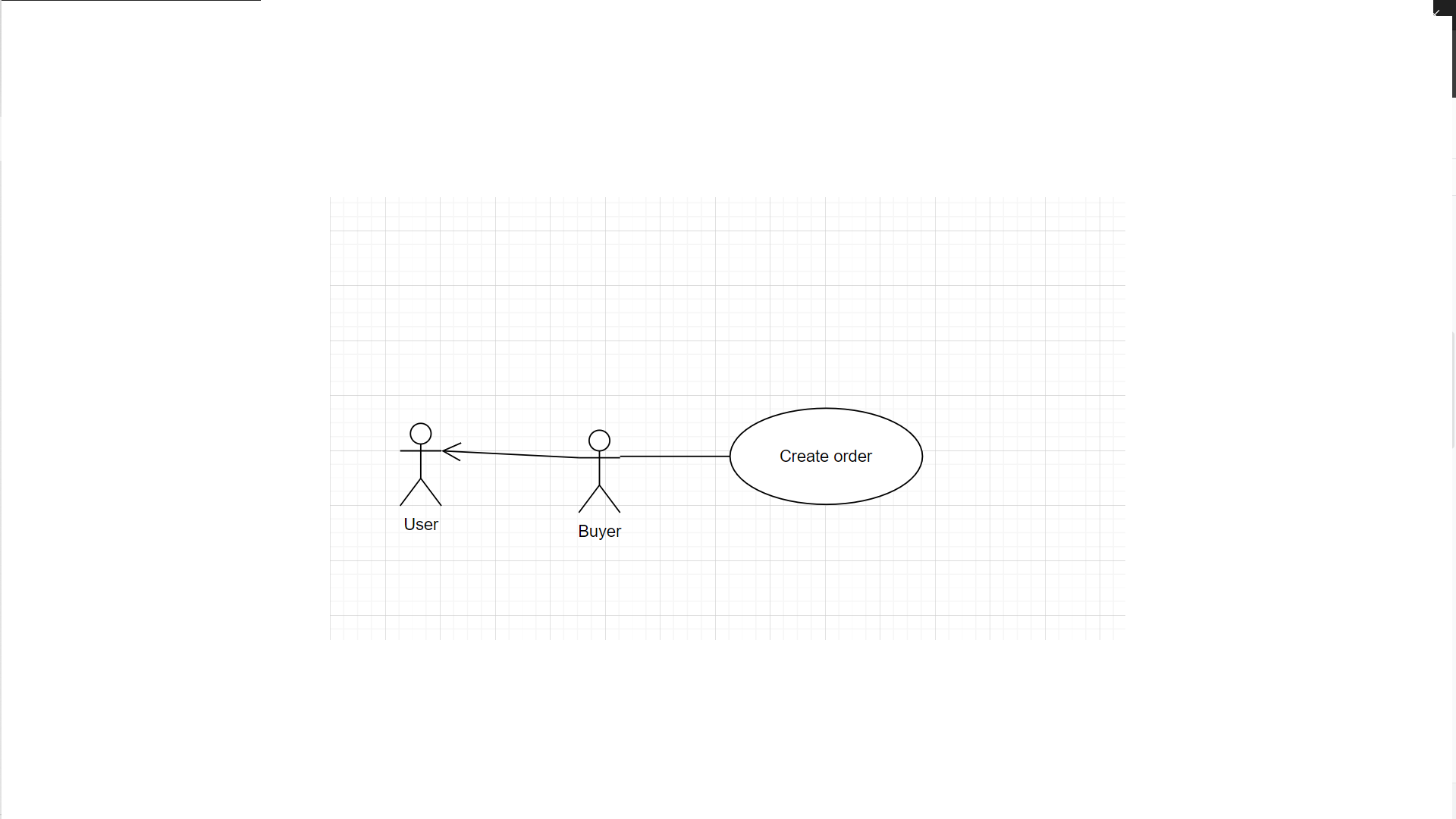
• **Brief Description:** An user changing account.  
• **Specification:** See Use-Case-Realization Specification: Manage Account

**View order**

****

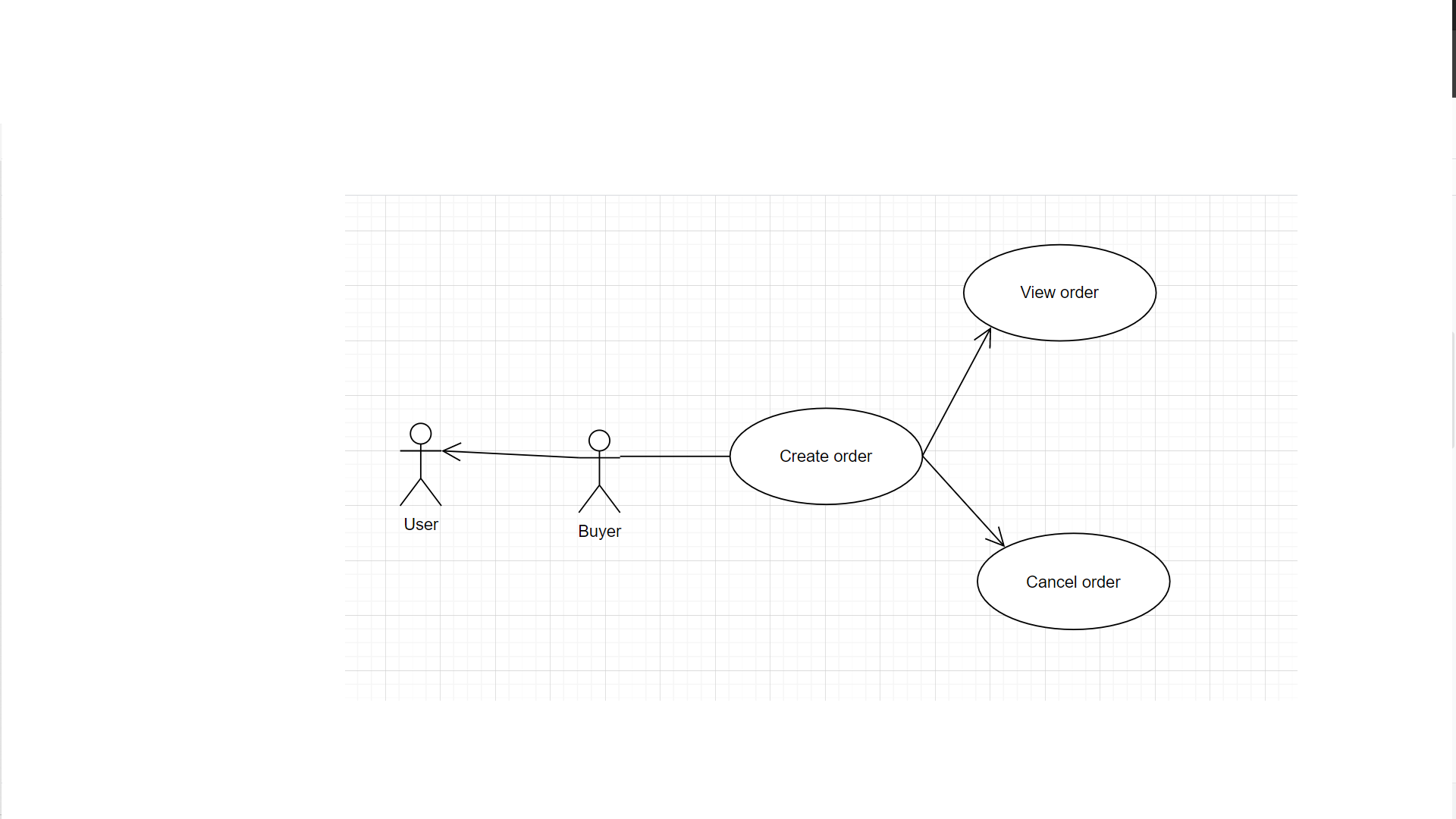
• **Brief Description:** An buyer viewing order.  
• **Specification:** See Use-Case-Realization Specification: Buyer Manage Order

**Create order**

****

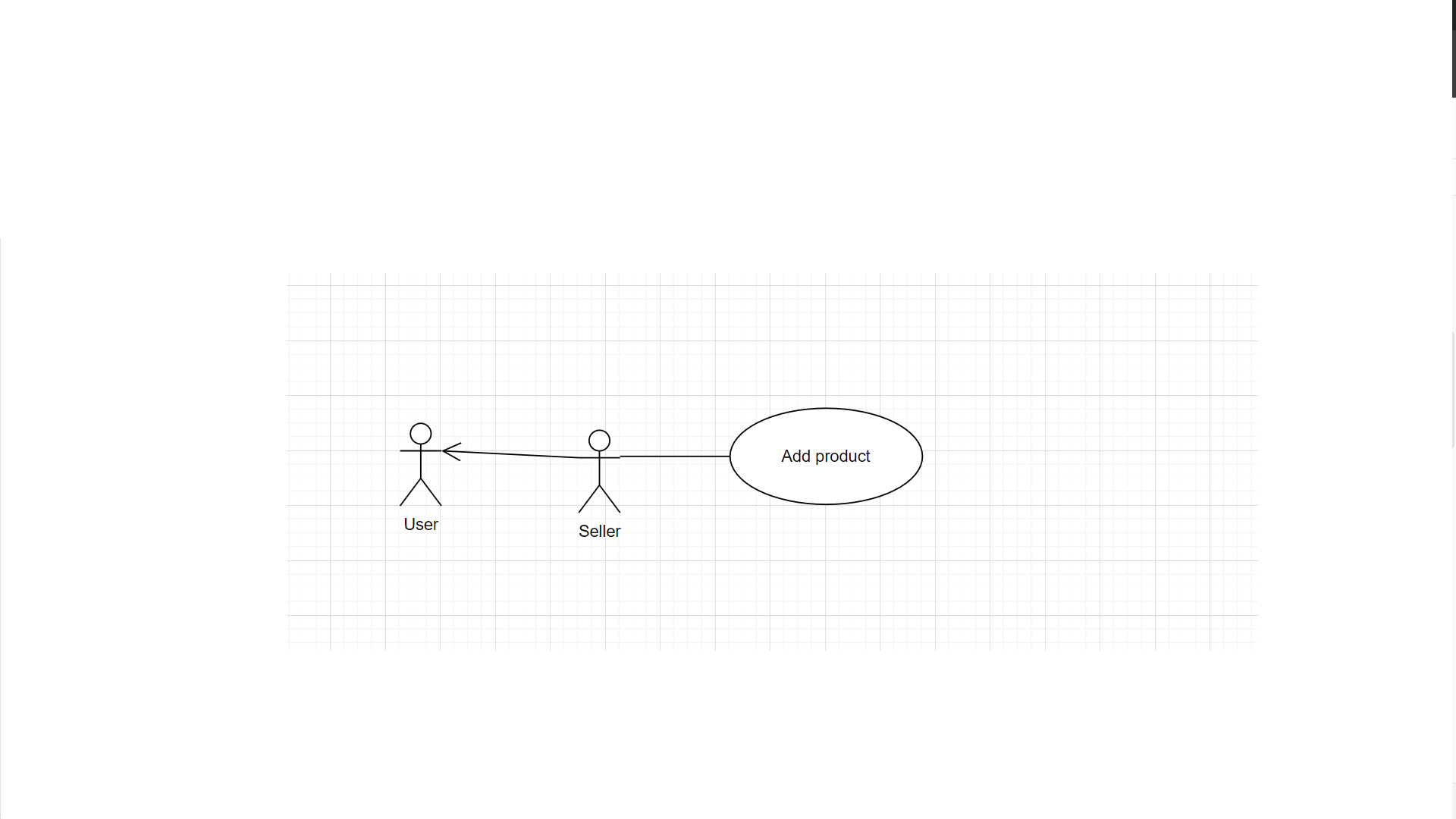
• **Brief Description:** An buyer creating order.  
• **Specification:** See Use-Case-Realization Specification: Buyer Manage Order

**Cancel order**

****

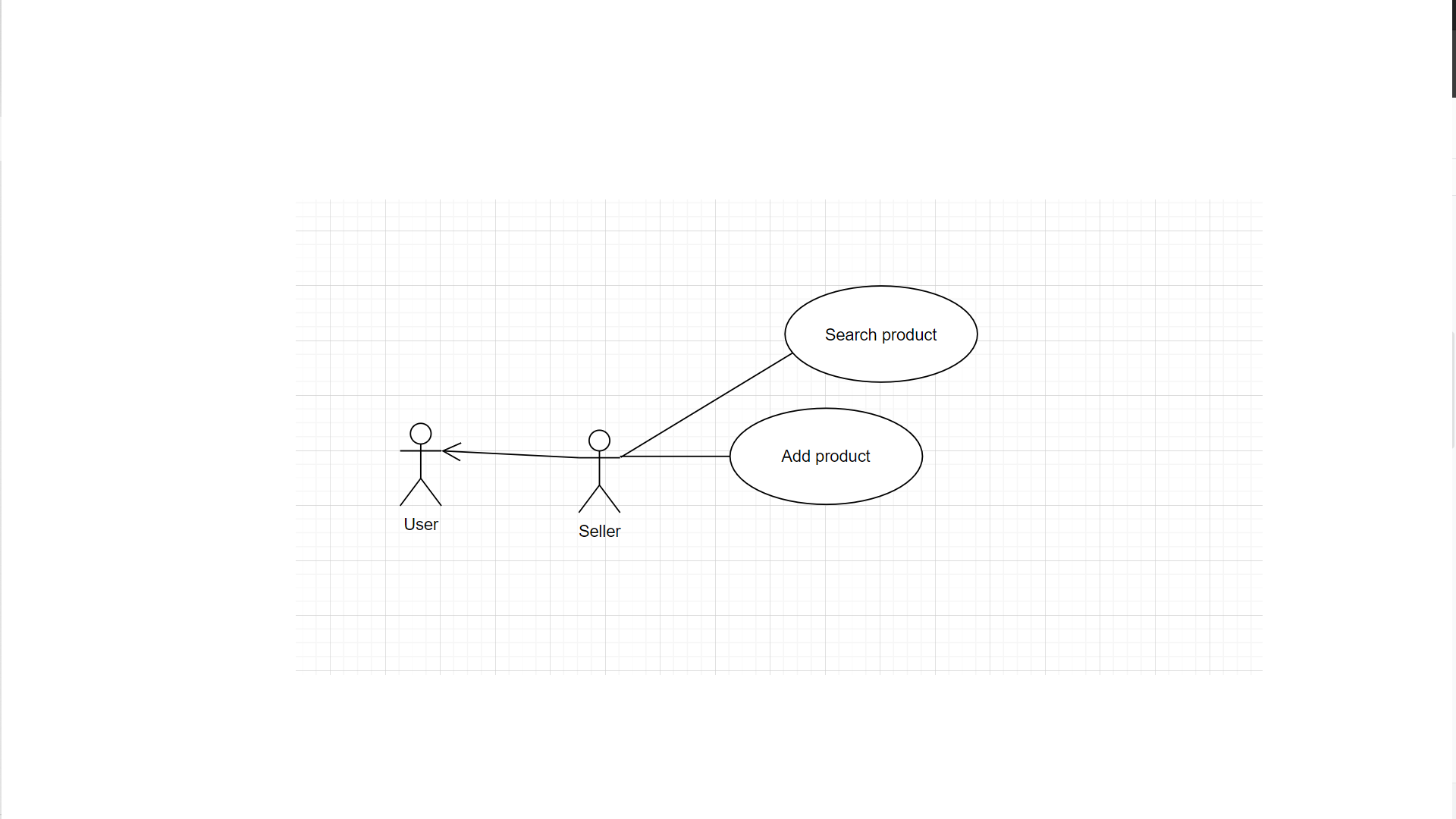
• **Brief Description:** An buyer cancelling order.  
• **Specification:** See Use-Case-Realization Specification: Buyer Manage Order

**Add product**

****

• **Brief Description:** A seller adding product.  
• **Specification:** See Use-Case-Realization Specification: Product management

**Search product**

****

• **Brief Description:** A seller searching product.  
• **Specification:** See Use-Case-Realization Specification: Product management

# Logical View

## Overview

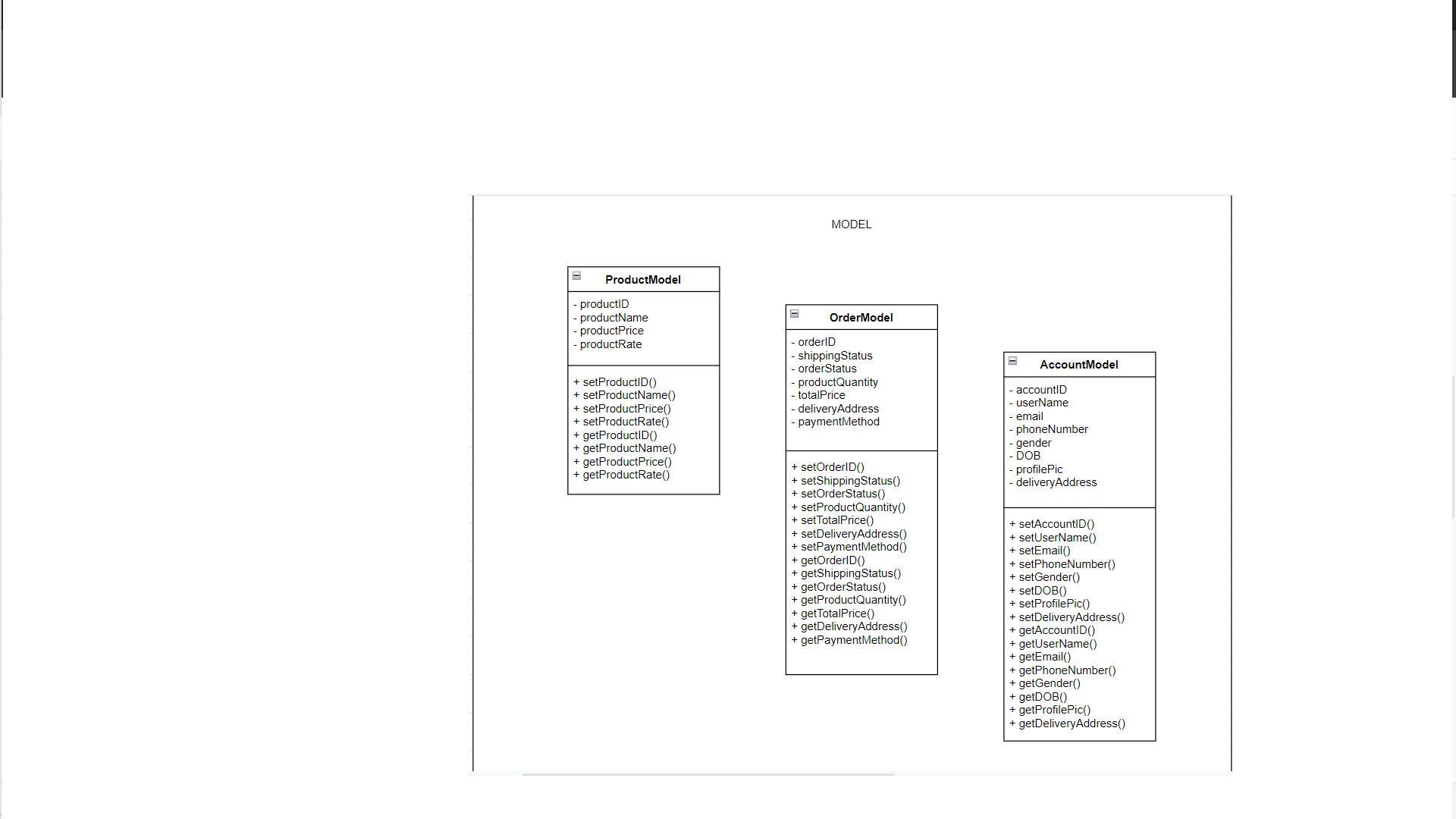
A description of the logical view of the architecture. Describes the overall decomposition of the design  
model in terms of package hierarchy and layers.  
The logical view of the Computer Shop Management System is comprised of 3 significant packages:

• **model:** contains classes that directly manages the data, logic and rules of the Online Shopping System and displayed in the view.  
• **view:** contains classes that generates output representation of information to the user based on changes in the model.  
• **controller:** contains classes that can send commands to the model to update the model’s state (e.g.,add a new product); it can also send commands to its associated view to change the view’s presentation of the model.

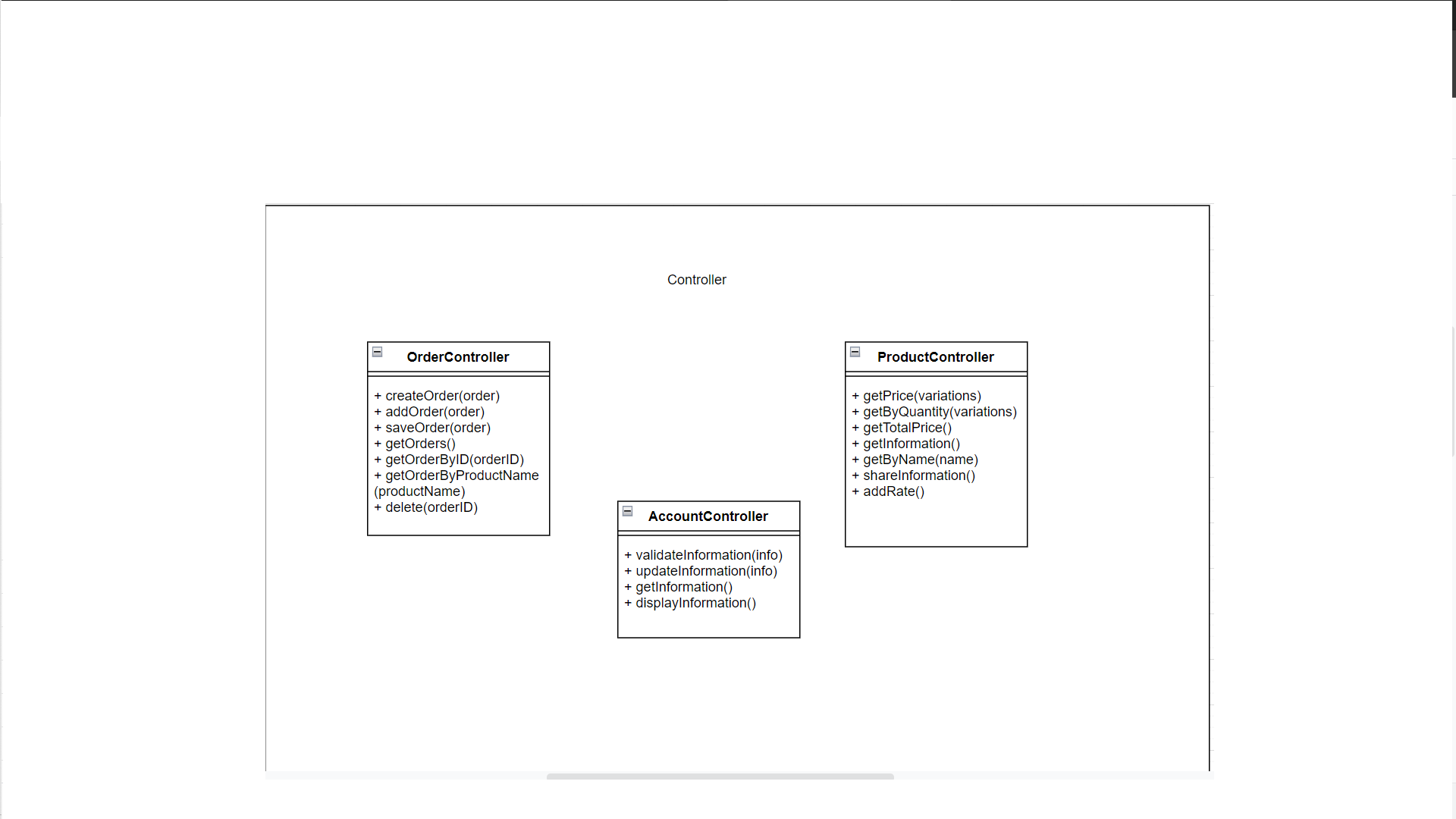


## Architecturally Significant Design Packages

**Package model:**

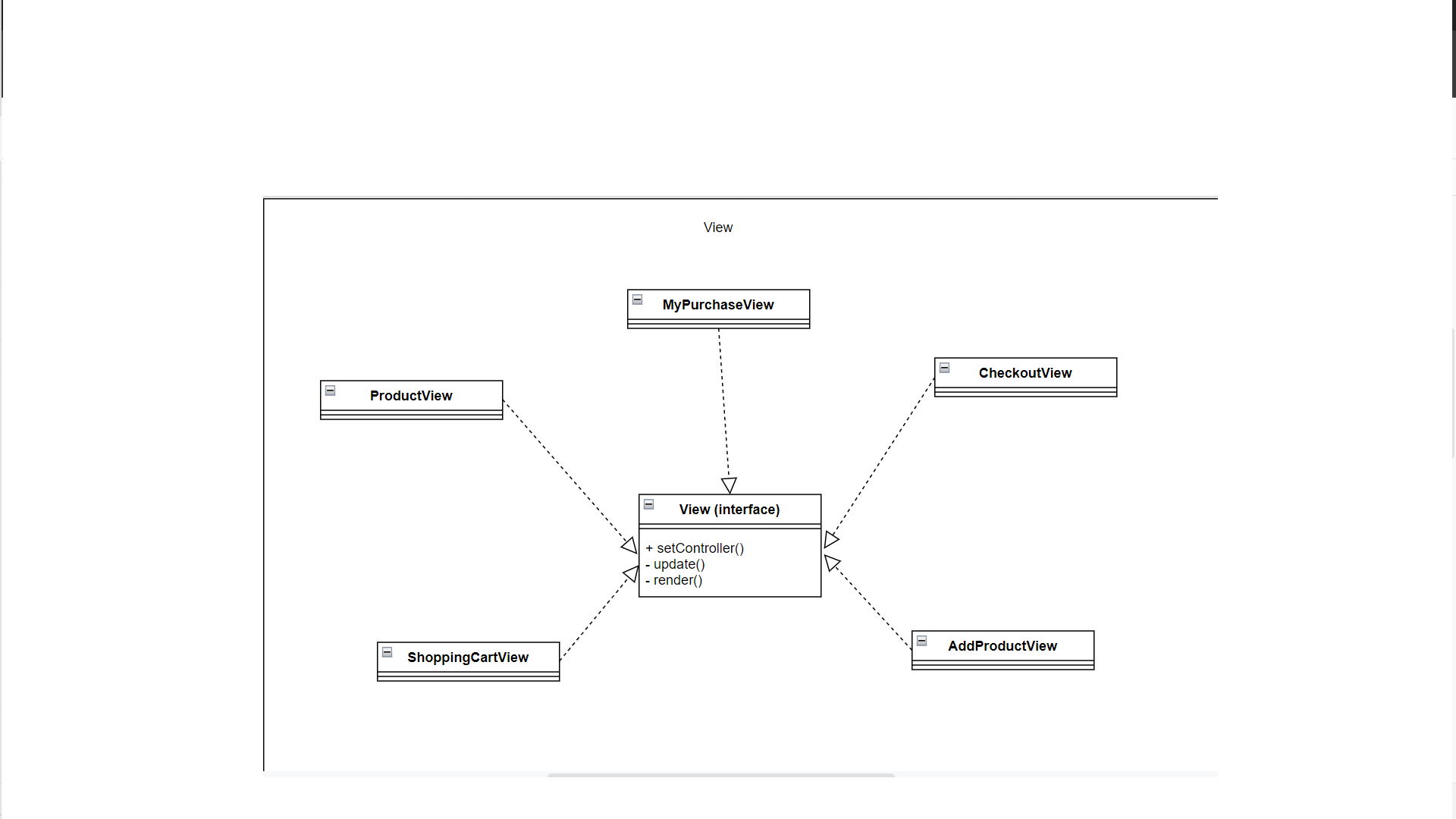
****

|  |  |
| --- | --- |
| **Name** | model |
| **Brief Description** | Contains classes that directly manages the data, logic and rules of the Online Shopping System and displayed in the view. |
|  |  |
| **Classes** | ProductModel, OrderModel, AccountModel |

**Package controller:**

|  |  |
| --- | --- |
| **Name** | controller |
| **Brief Description** | Contains classes that directly manages the data, logic and rules of the Online Shopping System and displayed in the view. |
|  |  |
| **Interfaces** | Controller. |
| **Classes** | AccountController, ProductController, OrderController, |
| AuthenticationController. |  |

**Package View:**

****

|  |  |
| --- | --- |
| **Name** | view |
| **Brief Description** | Contains classes that generates output representation of information to the user based on changes in the model. |
|  |  |
| **Interfaces** | View. |
| **Classes** | ProductView, MyPurchaseView, CheckoutView, ShoppingCartView, AddProductView |
|  |  |

# Size and Performance

The major dimensioning characteristics of the software that impact the architecture and performance constraints:  
The system shall support up to 5000 concurrent users against the primary database at any given time, and up to 3000 concurrent users against the local servers at any one time.  
The system must perform all functions with minimal time delays.  
The system must also accurately save all information transactions.

# Quality

The system architecture supports the quality requirements:  
• In order to maintain the highest degree of system integrity, the system is capable of  
ensuring that all information transitions are saved.  
• Databases will be backed up on a daily basis in concern with safety implications.  
• The system website is capable of display correctly on different devices web browser of  
any screen size (i.e. responsive design).  
• All system website functions are available through popular web browsers; for instance,  
Google Chrome, Mozilla Firefox, Opera, Safari, Microsoft Edge, Internet Explorer.