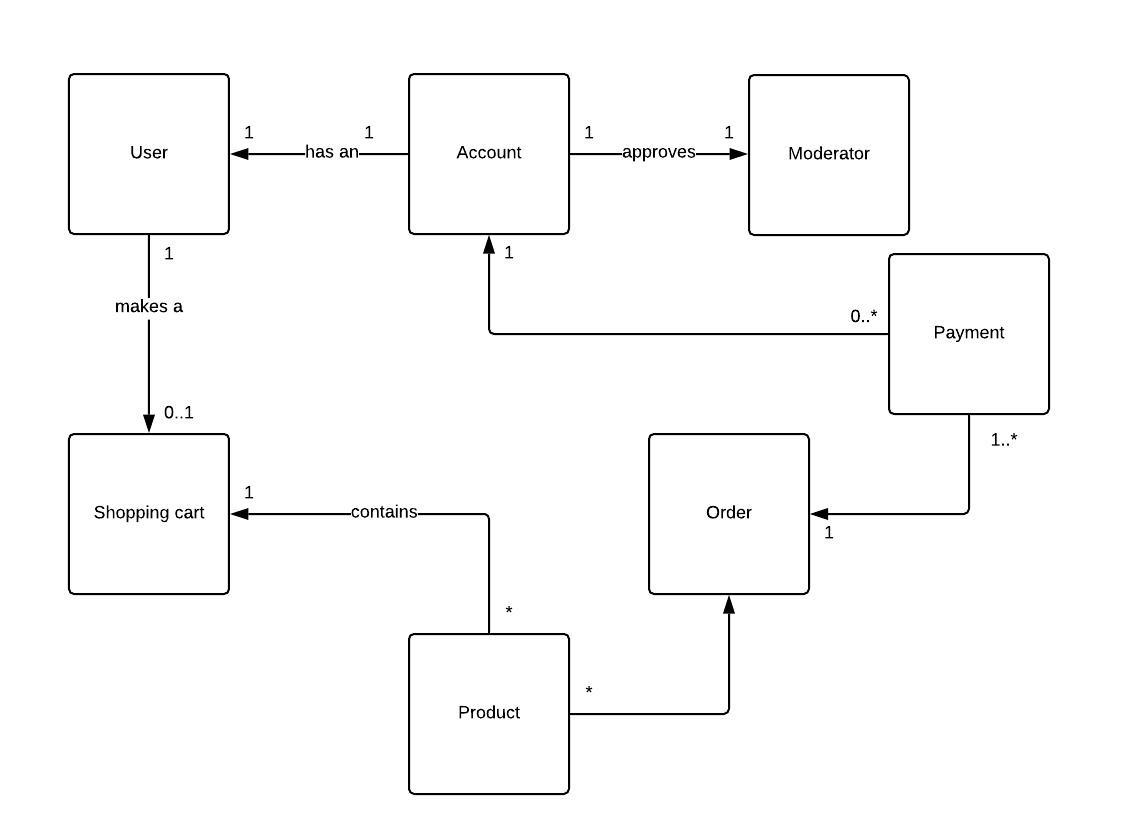
IT Store

Analysis and Design Document

# Elaboration – Iteration 1.1

# Domain Model



# Architectural Design

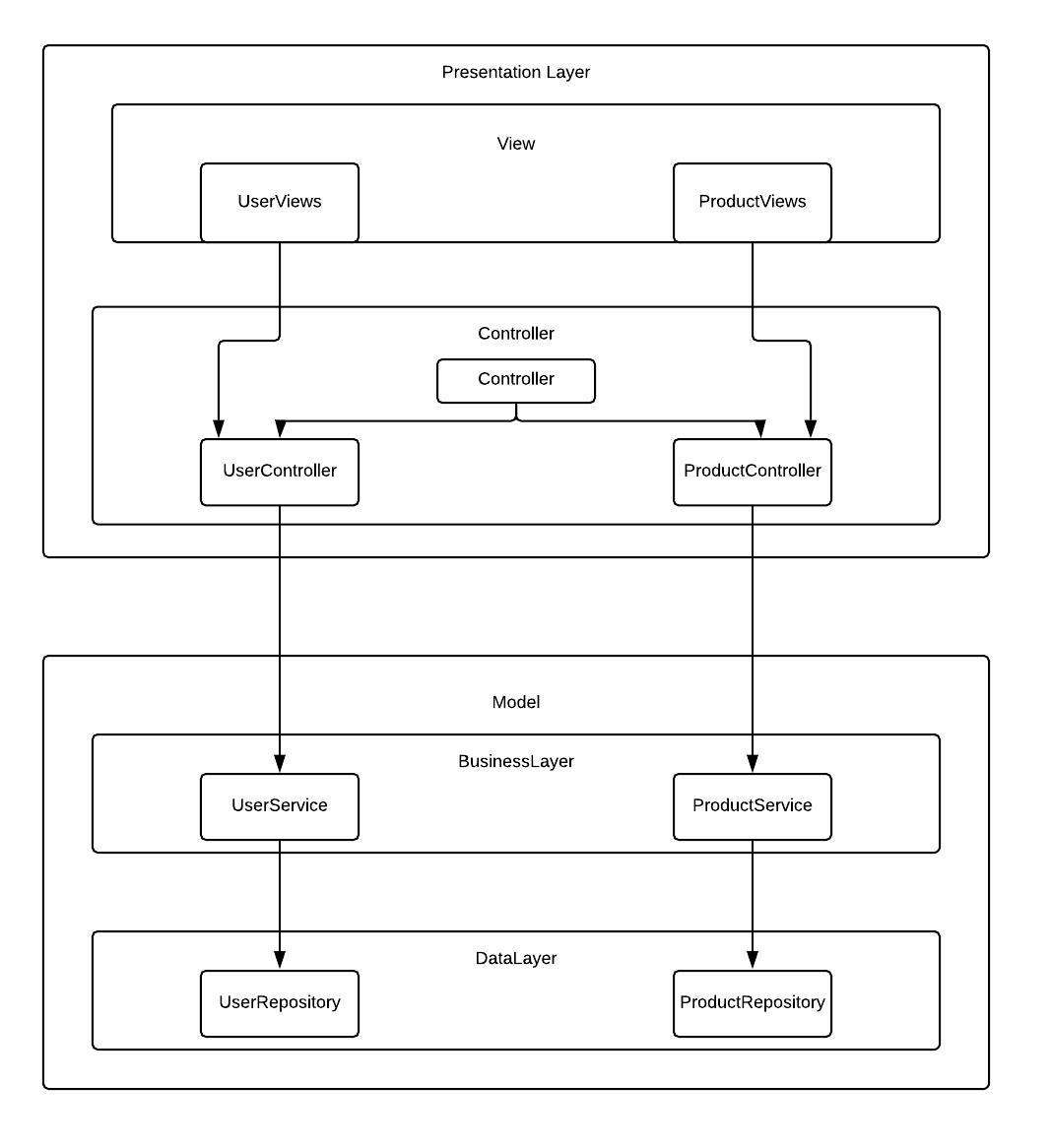
## Conceptual Architecture

The two architectural patterns to be used for this application are Model-View-Controller and Layered architecture which is applied on the model part of the MVC.

MVC is an architectural pattern which is separates the data and the data management part of an application from the view part, which has the main advantage of making the application easier to port on different systems and increases reusability of the implemented parts (view and model).

The Layered architectural pattern is a pattern which groups components into multiple logical layers that can be decoupled. It will be used on the model component of the MVC application, since the model needs to be simplified.

Through these architectural patterns, a high level of cohesion and a low level of coupling is achieved, making the project easier to understand and to maintain.



## Package Design

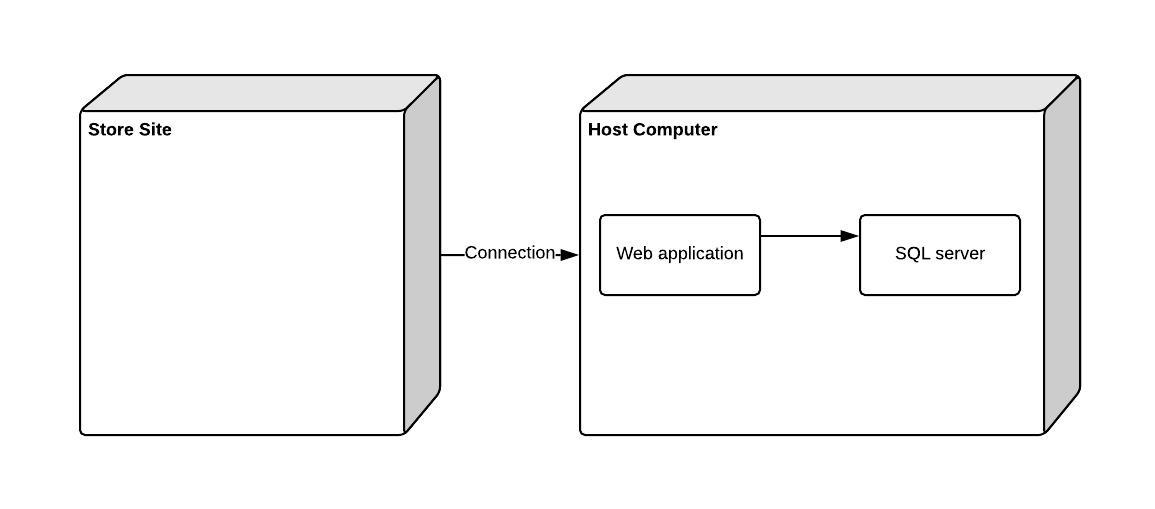
## 

## Component and Deployment Diagrams

Component Diagram:

# 

Deployment diagram:

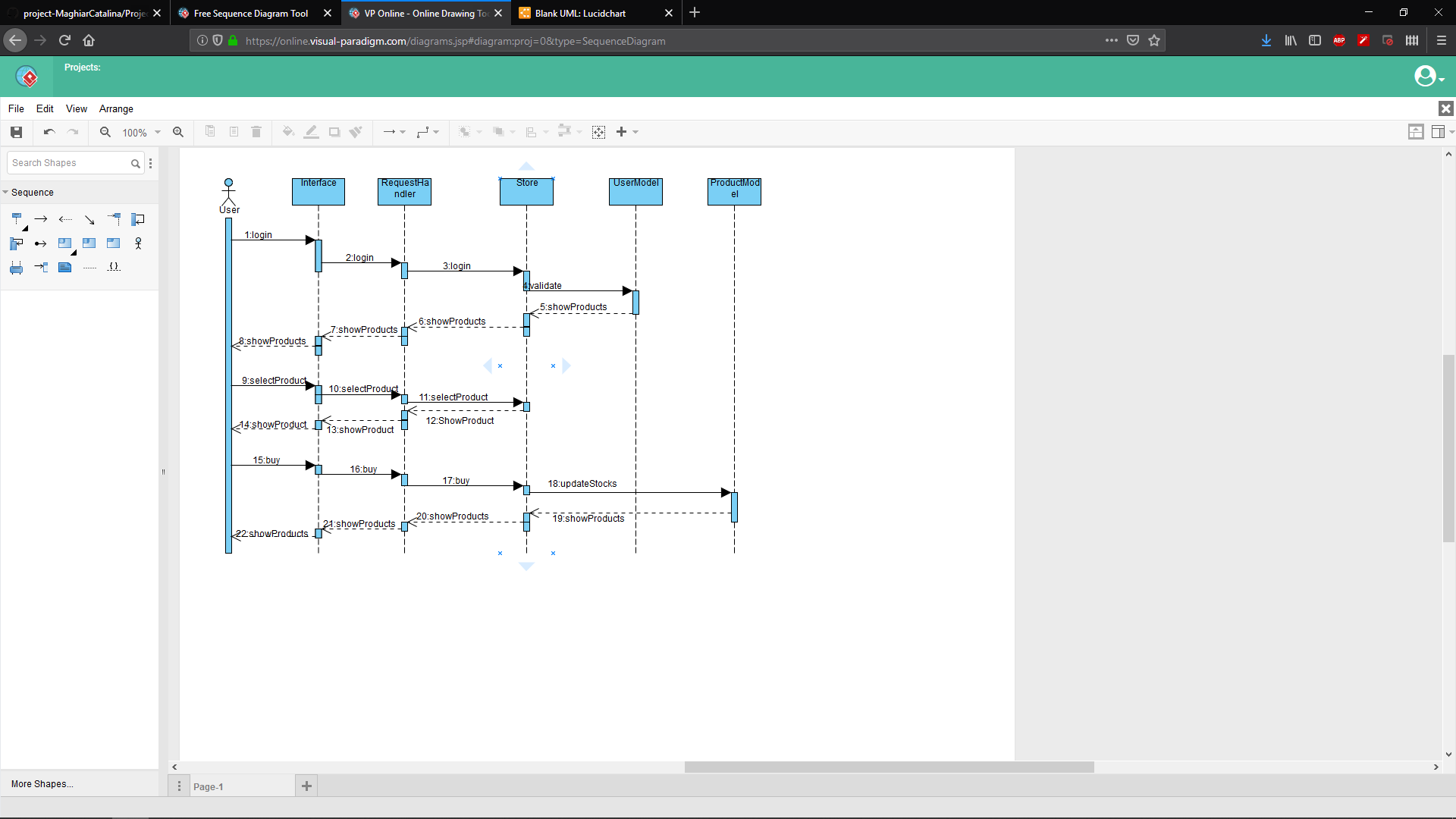


# Elaboration – Iteration 1.2

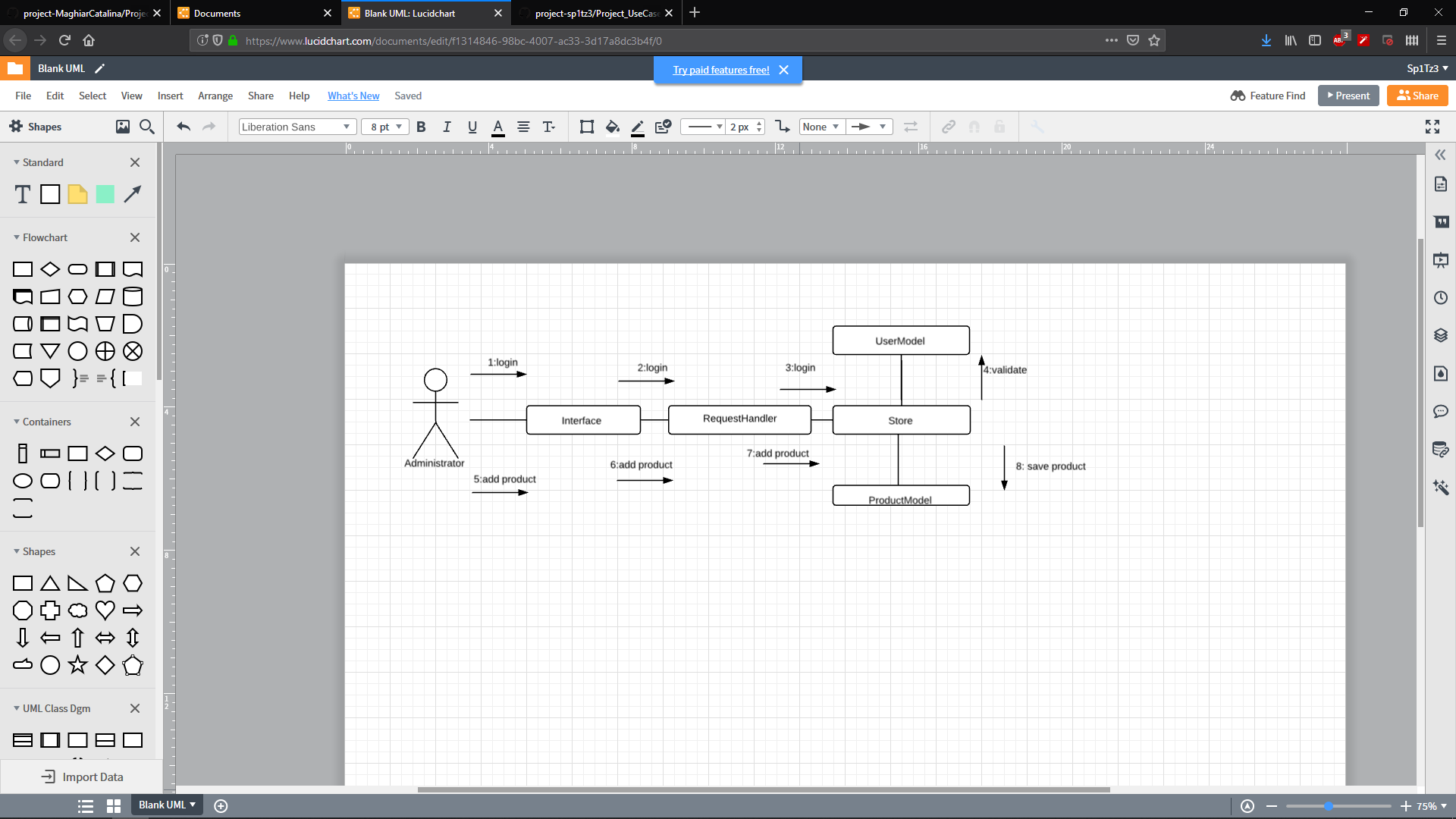
# Design Model

## Dynamic Behavior

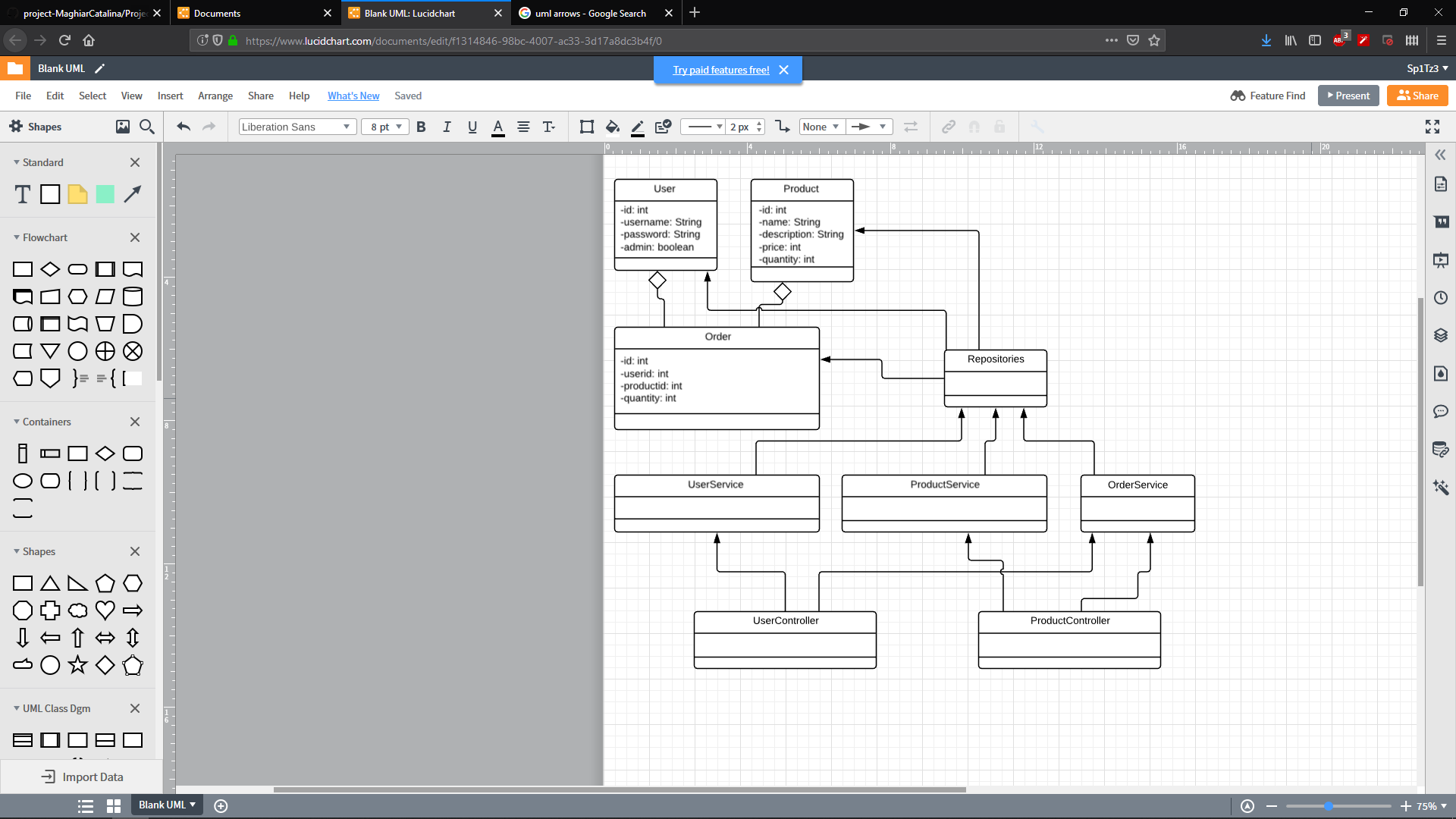
Sequence diagram: user logs in and buy something



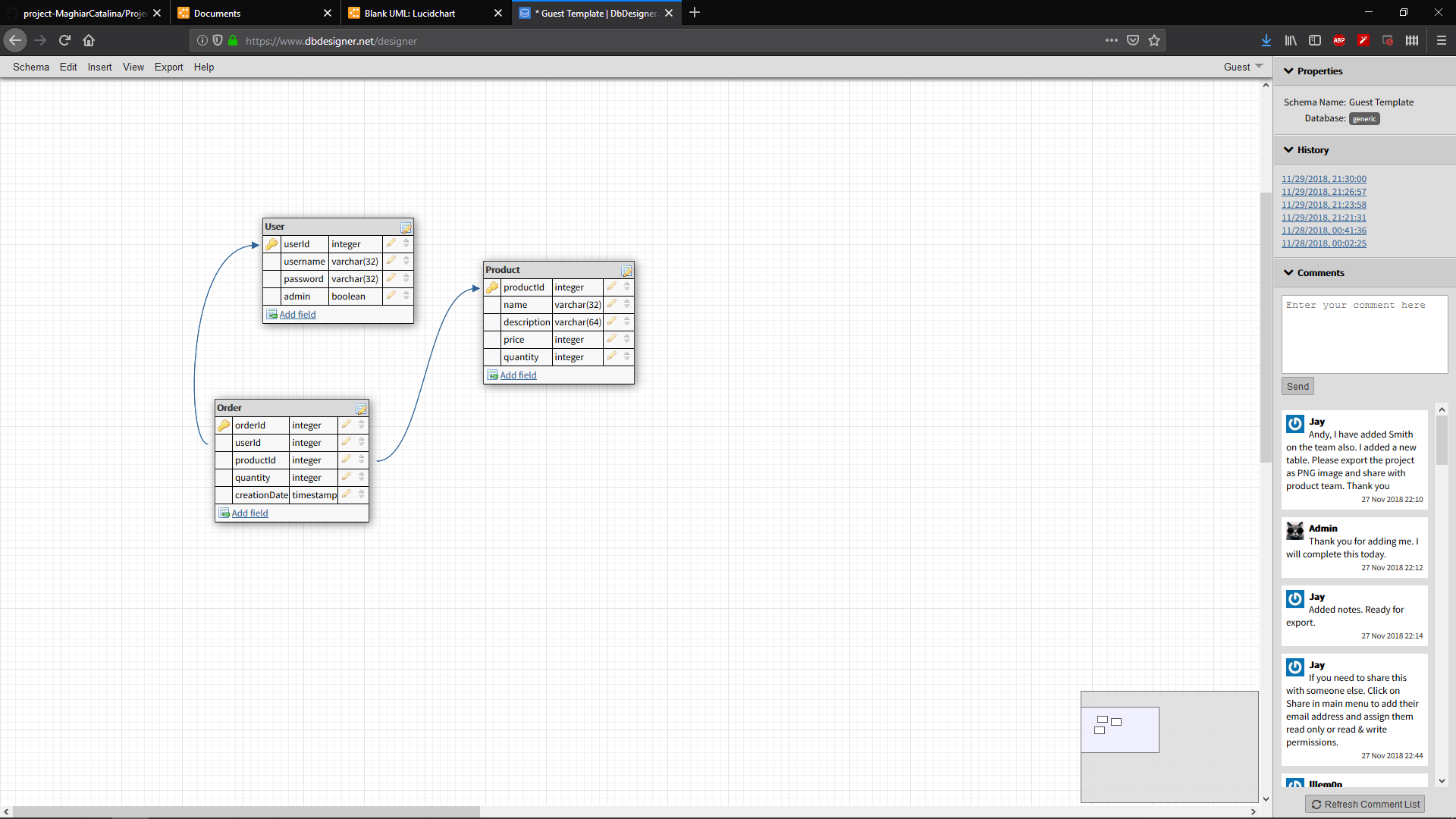
Communication diagram: administrator adds a new product



## Class Design



# Data Model



# Test Strategy

Tests that will be used are:

Unit tests to check that the Service layer works as intended. Examples of such tests are:

-editing a product,

-adding/removing a product as administrator.

Integration tests to check if the system works as a whole.

# Elaboration – Iteration 2

# Architectural Design Refinement

*[Refine the architectural design: conceptual architecture, package design (consider package design principles), component and deployment diagrams. Motivate the changes that have been made.]*

# Design Model Refinement

## *[Refine the UML class diagram by applying class design principles and GRASP; motivate your choices. Deliver the updated class diagrams.]*

# Construction and Transition

# System Testing

*[Describe how you applied integration testing and present the associated test case scenarios.]*

# Future improvements

*[Present future improvements for the system]*