

Software Architecture Document

-Yo!Kart

Table of Contents

1. Introduction	3
1.1. Purpose	3
1.2. Scope	3
2. Architectural Representation	3
3. Architectural Goals	3
4. Architectural Constraints	3
5. Use-Case View	4
5.1. Architecturally Significant Use Cases	5
6. Logical View: Architecture Overview	7
7. Process View	8
7.1. Processes	8
8. Deployment View	9
8.1 Desktop PC / Web Browser	9
8.2 Public Frontend	9
8.3 Login Protected Dashboard	9
8.4 Deployment Server	10
8.5 Size and Performance	10

1. Introduction

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 Yo!kart multi vendor ecommerce system designed by the FATbit technologies to support the concept of online buying and selling. The e commerce system in a multi vendor system which means multiple users can buy or sell items through the system.

2. Architectural Representation

This document presents the architecture as a series of views;

- Use case view
- Logical view
- Process view
- Deployment view.

There is no separate implementation view described in this document.

3. Architectural Goals

There are some key goals that have a significant bearing on the architecture, which are as follows:

- All Admin, buyers, and seller functionality must be available for access by the respective users.
- The Yo!Kart System must ensure complete protection of data from unauthorized access.
- All remote accesses are subject to user identification and password control.
- The Yo!Kart System will be implemented as a client-server system.

4. Architectural Constraints

There are some key system constraints that have a significant bearing on the architecture, which are as follows:

- All performance and load balancing requirements will be taken care of and load balancing must be done in order to handle high volume traffic on the system.
- The system is constrained to depend on the third party APIs like – payment gateways, SMS, Shipping APIs etc. Any response break from third parties will ultimately affect the functionality of the integrated API.
- The External mailing server (if integrated) will be responsible for the mail notifications sent to different users. The External mailing server (if integrated) will be responsible for the mail notifications to the system users.

5. 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 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 Seller use cases are as follows:

- Login/Register
- Buy Subscriptions (If enabled by Admin)
- List & Pick Products from Catalogue
- Add Custom Catalogue
- View Orders
- Manage Order Status
- Manage Shipping
- Manage Cancellation and Return Requests
- View Wallet, Submit Withdrawal Requests
- View & Download Reports

The Buyer use cases are as follows:

- Login/Register
- Search Products
- Add Products to Cart
- Order Placement
- Apply coupons
- View Wallet, Recharge & Submit Withdrawal Requests
- Payment Gateway payment processing
- Raise Return and cancellations Requests

These use cases are initiated by the buyer, seller or the admin. In addition, interaction with external actors; payment gateway and shipping gateways will occur.

5.1. Architecturally Significant Use Cases

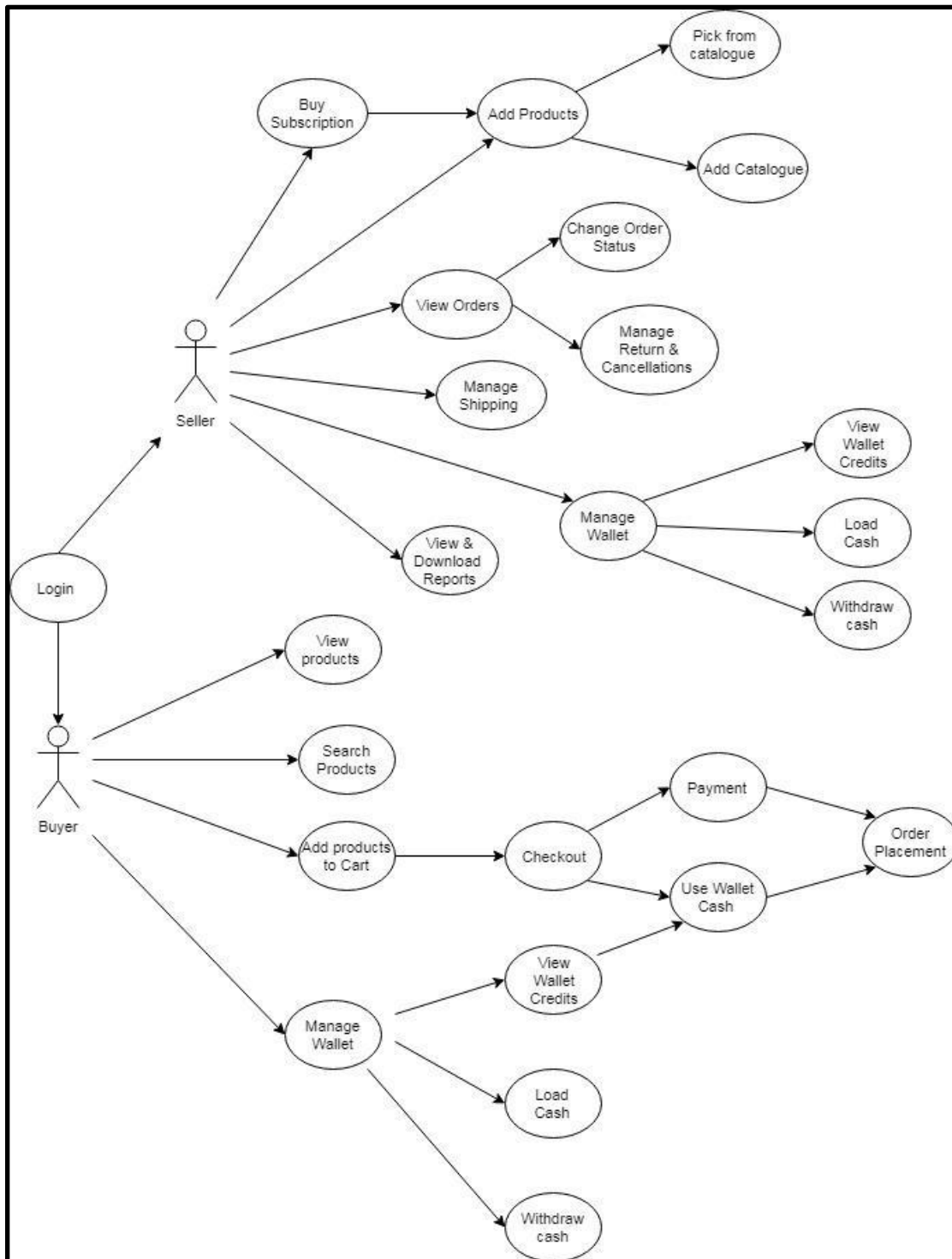


Diagram Name: Architecturally Significant Use-Cases

5.1.1 Buy Subscription

This use case will allow the owner to buy subscription plans to list their products on website. The subscription plan activation by seller is only required if the site admin has made it mandatory for the sellers for listing their product. The actor of this use case is Seller.

5.1.2 Add Products

The user can pick products from the catalogue defined by the site admin or add their own catalogue. The product moderation may or may not be required, as per the admin settings. The actor in this case is Seller.

5.1.3 Orders Management

The user can view the list of orders and manage the orders. The user will also be able to view the return and refund orders and take necessary actions for them. The actor in this case is seller.

The user will be able to place orders, view the list of orders and place return or refund requests for specific orders. The actor in this case is Buyer.

5.1.4 Manage Shipping

The user can will be responsible for managing the shipping of the orders unless otherwise defined by the site admin for managing shipping. The actor in this case is seller.

5.1.5 Manage Wallet

The user can view the wallet balance, wallet transactions and load cash and place a withdrawal request for funds transfer to the stated bank account. The actor in this case is both Seller & Buyer.

5.1.6 View & Download Reports

The user can view the reports and download reports for products inventory, products performance and products inventory status. The actor in this case is Seller.

5.1.7 View & Search Products

The user can view and search products according to their wish. The actor in this case is Guest, Seller or Buyer.

5.1.8 Add Products to Cart

The user can browse products, add products from the products listings or details page into the cart and update the quantity count as per their requirement.

The actor in this case is Guest, Seller or Buyer.

5.1.9 Checkout & Order Placement

To accomplish the checkout process a user will be redirected to the payment gateway options upon click of the Checkout button on the cart page. The user will be redirected the payment gateway page and make payment with an authentic credit card / debit card. However user can also make payment through direct bank transfer ort virtual wallet which is loaded with cash credits. The actor in this case is Buyer.

6. Logical View: Architecture Overview

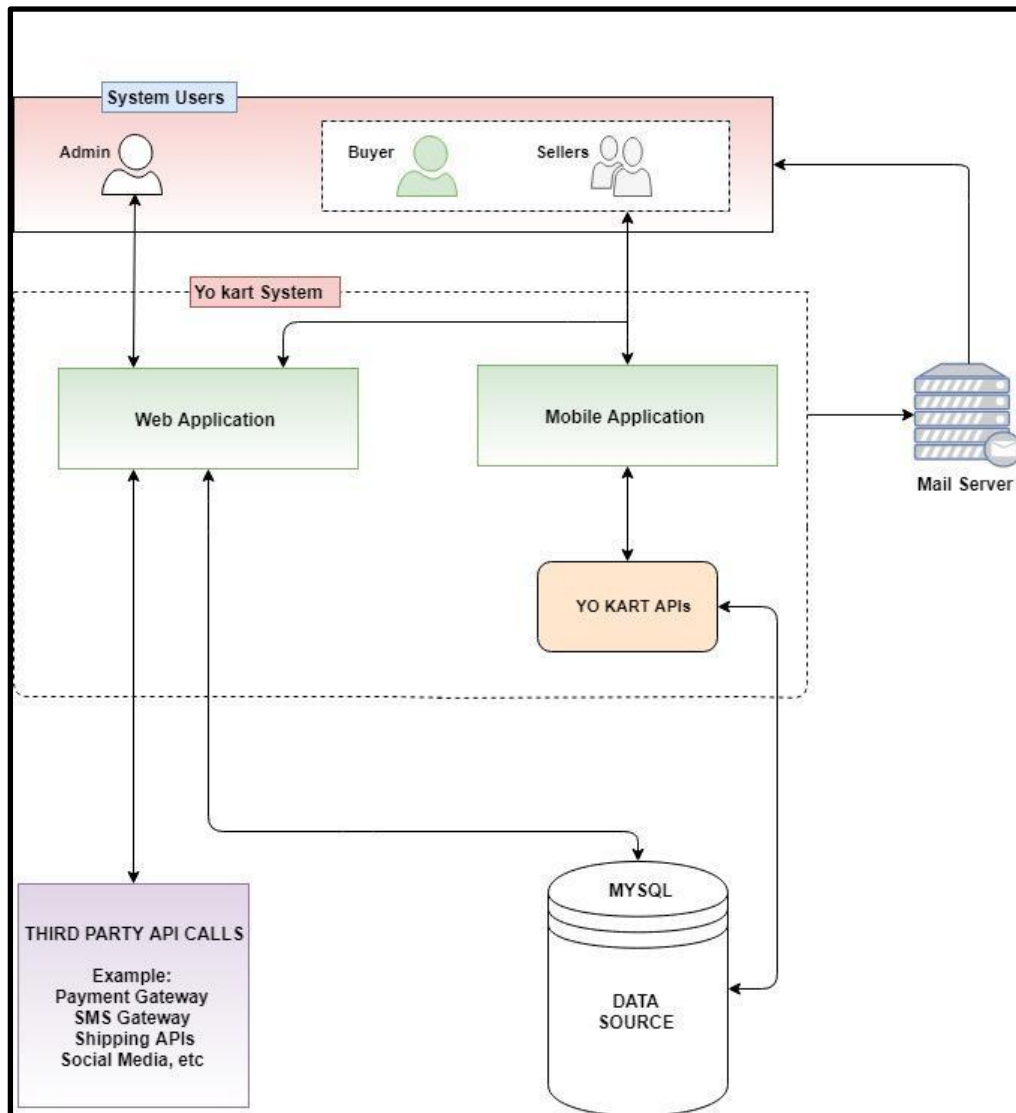


Diagram Name: Architecture Overview

6.1.1 Application layer

This application layer has all the boundary classes that represent the application screens that the user sees. This layer depends upon the Process Objects layer; that straddles the separation of the client from mid-tier.

6.1.2 Business Services layer

The Business Services process layer has all the controller classes that represent the use case managers that drive the application behavior. This layer represents the client-to-mid-tier border. The Business Services layer depends upon the Process Objects layer; that straddles the separation of the client from mid-tier.

6.1.3 Middleware layer

The Middleware layer supports access to Relational DBMS-MYSQL.

6.1.4 Base Reuse

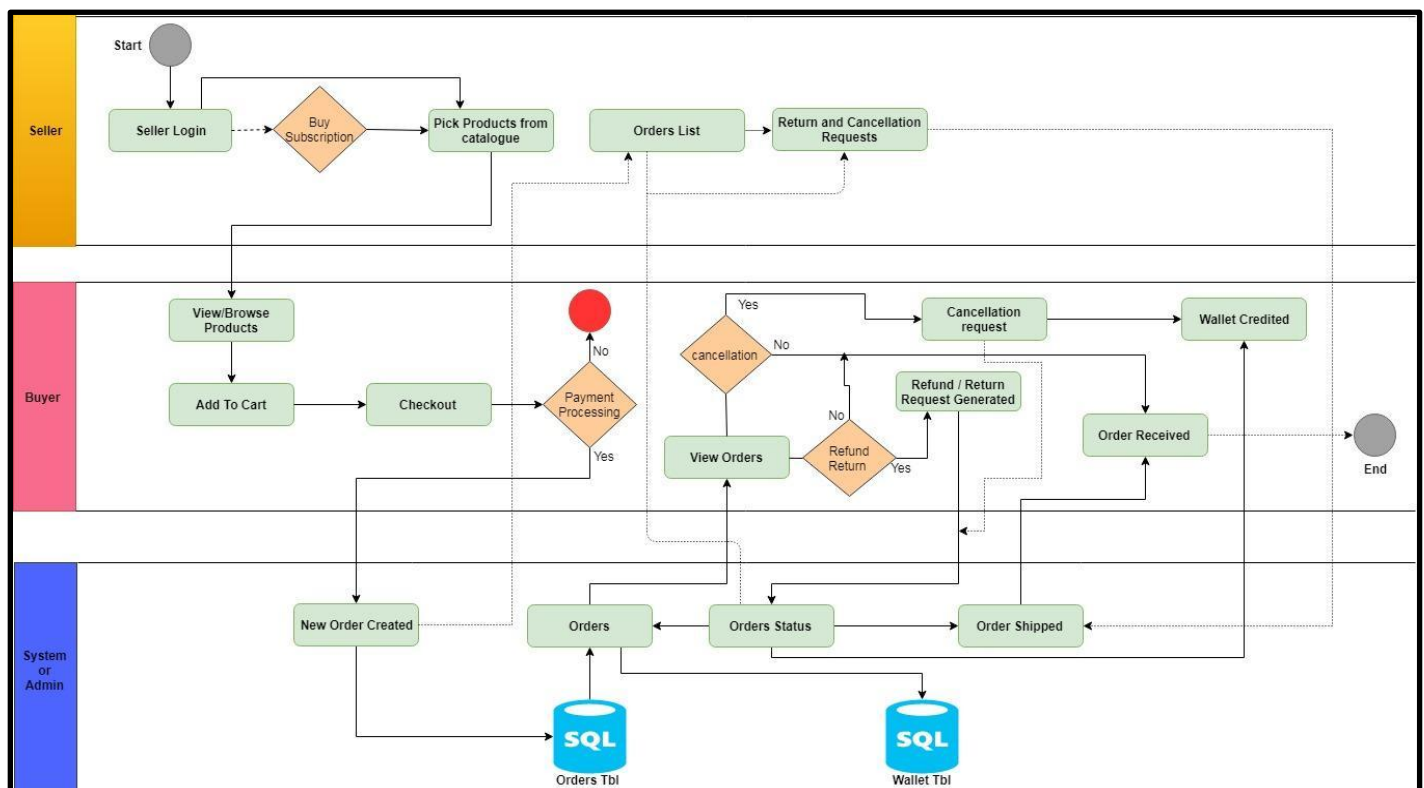
The Base Reuse package includes classes to support list functions and patterns.

7. Process View

A description of the process view of the architecture. Describes the tasks (processes and threads) involved in the system's execution, their interactions and configurations. Also describes the allocation of objects and classes to tasks.

7.1. Processes

The Core buying process and order refund/ return and cancellation process for the Yo kart is explained below in the



process view diagram

Diagram Name: Process View

8. Deployment View

A description of the deployment view of the architecture describes the various physical nodes for the most typical platform configurations. Also describes the allocation of tasks (from the Process View) to the physical nodes. This section is organized by physical network configuration; each such configuration is illustrated by a deployment diagram, followed by a mapping of processes to each processor.

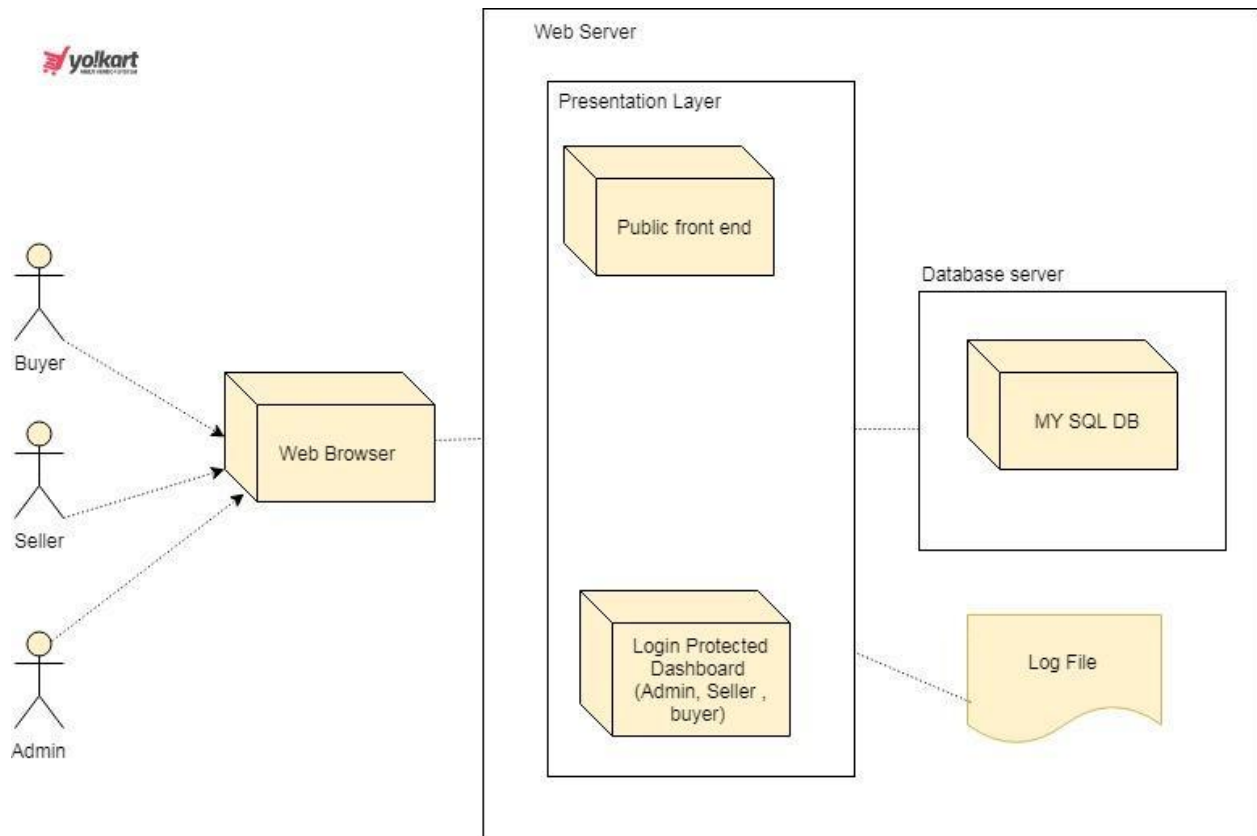


Diagram Name: Deployment View

8.1 Web Browser

Buyer can register via Desktop PCs browsers that are connected directly to the Web Servers. The Seller also uses these PCs browser to register and maintain products on website.

8.2 Public Frontend

Public frontend is the main client side part which lies under the presentation layer. All the users have access to public frontend through the web browsers.

8.3 Login Protected Dashboard

The login protected dashboard that allows users to manage their respective information in the system. Access to dashboard is available through web browser.

8.4 Deployment Server

8.4.1 Supported Operating Systems

Linux x86, x86-64

8.4.2 Supported Web Servers

Apache 2.2.x – Apache 2.4.x

8.4.3 PHP Compatibility

5.6 and above versions

8.4.4 Required Extensions

- GD with Free Font support
- Zlib with zip support
- DOM
- Mbstring should be enabled
- Iconv function should be enabled
- Fileinfo function should be enabled
- Ioncube Loader
 - ✓ Safe mode off
 - ✓ Memory limit 32M or more (Some pages with advanced feature may use upto 128M)

8.4.5 My SQL

5.6 Or newer

8.4.6 Server Hosting setup

Ability to run scheduled jobs (crontab) with PHP

8.4.7 Compatible Web Hosting

- Shared Servers
- Virtual Private Server (VPS)
- Dedicated Servers
- Cloud Server

8.5 Size and Performance

The chosen software architecture supports the key sizing and timing requirements, as stipulated in the Supplementary Specification [8.4]:

- The system browsing bandwidth is directly proportional to the server bandwidth.
- The system shall provide access to the product catalog database with not more than 10 seconds latency.
- The system must be able to complete 80% of all transactions within 2 minutes.

The selected architecture supports the sizing and timing requirements through the implementation of client-server architecture.