**PramatiKart Project**

**Technical Design Document**

**Table of Contents**

[**Details**](#_heading=h.21ngj1xyglrr) **3**

[**HIgh Level Architecture Design**](#_heading=h.7i7b81q88wxz) **3**

[**Components of the application**](#_heading=h.tc33l1yyqk4s) **4**

[**Tech Stack**](#_heading=h.g18zqssf27ey) **4**

[Out of scope](#_heading=h.pqyau46tg0ad) 5

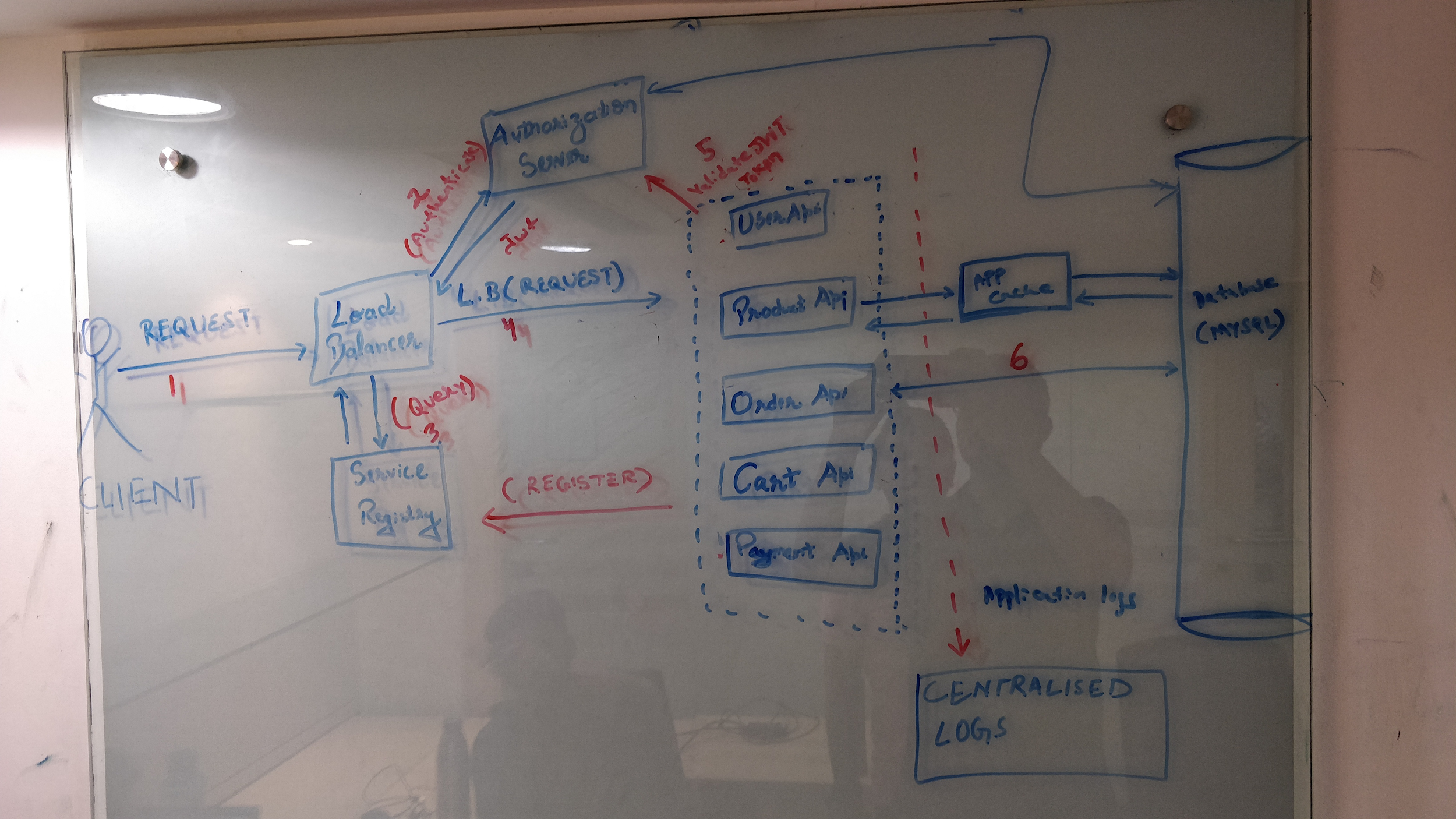
[**Learnings Involved**](#_heading=h.ebqteyv5fzsu) **5**

**Technical Document**

# Details

The introduction of the Technical Requirements Specification (SRS) provides an overview of the entire TDD with purpose, scope, definitions, acronyms, abbreviations, references and overview of the TDD. The aim of this document is to gather and analyze and give an in-depth insight of the complete **PramatiKart** by defining the problem statement in detail. Nevertheless, it also concentrates on the capabilities required by stakeholders and their needs while defining high-level product features. The detailed requirements of the **PramatiKart** are provided in this document.

# HIgh Level Architecture Design



# Flash sale use case:

* User will register for flash sale
* Flash sale will be activated by system admin
* User will be able to participate in flash sale only if user is registered for that
* flash sale will be started and stopped by the admin
* Buyer will check the product, if it's available then buyer can proceed to buy the product
* If the product got sell then application will decrease the quantity of the product.

# Assumptions:

* Flash sale will be only for one product
* Buyer has to register for the sale
* Buyer can buy only one product per sale

# Challenges:

* Application will not allow any oversell or undersell of products
* Application should be able to handle high concurrency requests
* Application should handle high throughput for read and write

# 

# C**omponents of the application**

The components of the application are as follows

**1.Load Balancer**-It acts as a reverse proxy and distributes network or application traffic across a number of servers.Load balancers are used to increase capacity (concurrent users) and reliability of applications.

**2.Authorization Server**- The authorization server validates the credentials and redirects user back to the client with an authorization code. The client talks with the authorization server, confirms its identify and exchanges the authorization code for an access token and optionally a refresh token.

**3.Service Registry**-It stores collective information of services along with related details including the location of service and number of instances.

Registration — whenever a new service or service instance scales in/out, it needs to register/deregister itself with Service Registry.

Discovery — whenever client needs to interact with service, client would lookup for service details or discover service on Service Registry.

**4.Application Cache**-Cache accelerate application response times and help applications scale by placing frequently needed data very close to the application.

**5.Centralized Log Aggregator**-It helps in overall management process in which you consolidate different log formats coming from different sources all into one place to make it easier for you to analyze, search, and report on your data.

# Tech Stack

1. Java 8
2. Spring Boot
3. Spring Cloud
4. Spring Security
5. Kubernetes/Docker
6. MySql
7. Redis Cache
8. Junit and Mockito
9. Swagger
10. Splunk/ELK

## Out of scope

1.Payment API

2.Delivery API

# Learnings Involved

1.Microservices Architecture

2.Spring Security/OAuth 2.0

3. Docker Containers

4.Container Orchestration(Kubernetes)

5. Code Quality via SonarQube

6. Concurrency and locking

7.Service API Documentation

8.Designing API

9.Distributed application programming

10.Centralized Service logging/Tracing

1. **Api Design**
2. **User API**

* This API is responsible to give the information of the user, update the user information. It will also manage the user roles.

1. **Product ApI**

* This API allows client to extract product information(product ID,product descriptions, full product titles, detailed product specs, product and brand images, pricing info, quantity-based discounts, etc.) based on the client's role.

1. **Order API**

* This API allows customer to fetch the order details, to place an order, to cancel the order.

1. **Cart API**

* This API integrate customer features that personalize the customer experience at checkout. These include discounts, applying coupons, adding products to cart, removing product from cart, changing the quantity of the product etc

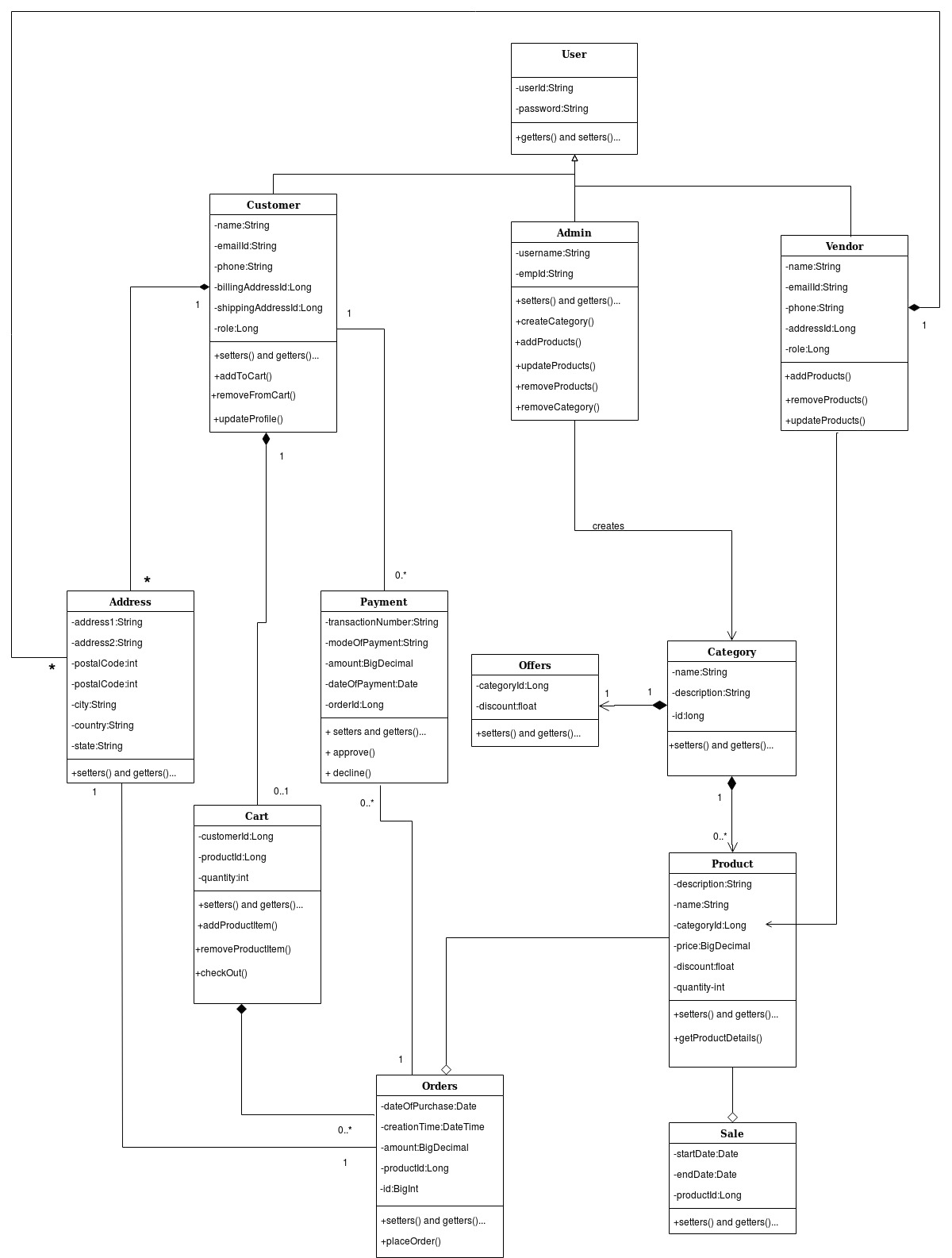
1. **Payment API**

* This API is responsible to do the payment for the order.

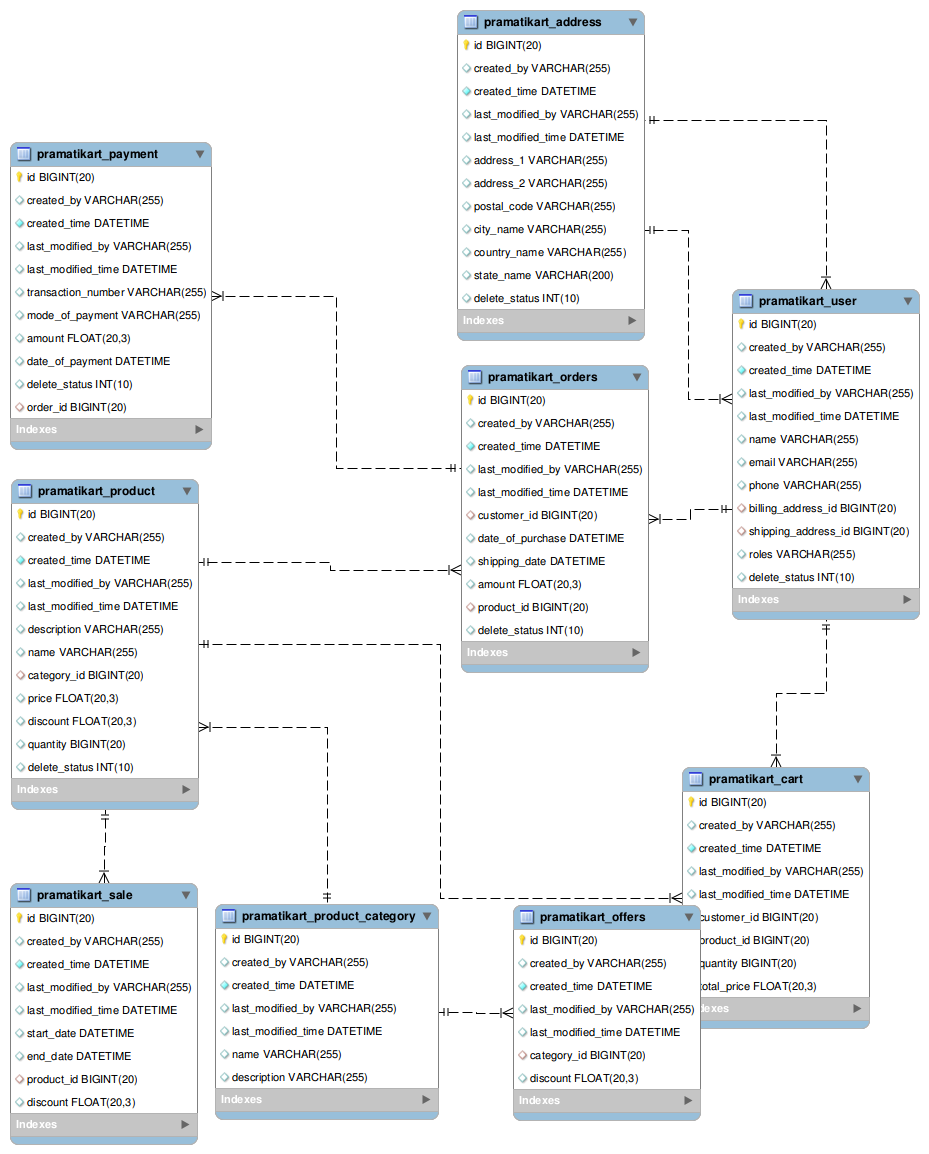
**9 UML Diagrams**

Sequence Diagram

Class Diagram



1. **E-R Diagram**



**6. DB Schema:**

create database pramatikart;

use pramatikart;

CREATE TABLE `pramatikart\_address` (

`id` bigint(20) NOT NULL Primary Key AUTO\_INCREMENT,

`created\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`created\_time` datetime NOT NULL DEFAULT CURRENT\_TIMESTAMP,

`last\_modified\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`last\_modified\_time` datetime DEFAULT CURRENT\_TIMESTAMP,

`address\_1` varchar(255) DEFAULT NULL,

`address\_2` varchar(255) DEFAULT NULL,

`postal\_code` varchar(255) DEFAULT NULL,

`city\_name` varchar(255) DEFAULT NULL,

`country\_name` varchar(255) DEFAULT NULL,

`state\_name` varchar(200) DEFAULT NULL,

`delete\_status` int(10) DEFAULT '0'

) ;

CREATE TABLE `pramatikart\_user` (

`id` bigint(20) NOT NULL Primary Key AUTO\_INCREMENT,

`created\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`created\_time` datetime NOT NULL DEFAULT CURRENT\_TIMESTAMP,

`last\_modified\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`last\_modified\_time` datetime DEFAULT CURRENT\_TIMESTAMP,

`name` varchar(255) DEFAULT NULL,

`email` varchar(255) DEFAULT NULL,

`phone` varchar(255) DEFAULT NULL,

`billing\_address\_id` bigint(20),

`shipping\_address\_id` bigint(20),

`roles` varchar(255) DEFAULT NULL,

`delete\_status` int(10) DEFAULT '0',

FOREIGN KEY (`billing\_address\_id`) REFERENCES pramatikart\_address(id),

FOREIGN KEY (`shipping\_address\_id`) REFERENCES pramatikart\_address(id)

) ;

CREATE TABLE `pramatikart\_product\_category` (

`id` bigint(20) NOT NULL Primary Key AUTO\_INCREMENT,

`created\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`created\_time` datetime NOT NULL DEFAULT CURRENT\_TIMESTAMP,

`last\_modified\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`last\_modified\_time` datetime DEFAULT CURRENT\_TIMESTAMP,

`name` varchar(255) DEFAULT NULL,

`description` varchar(255) DEFAULT NULL

) ;

CREATE TABLE `pramatikart\_product` (

`id` bigint(20) NOT NULL Primary Key AUTO\_INCREMENT,

`created\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`created\_time` datetime NOT NULL DEFAULT CURRENT\_TIMESTAMP,

`last\_modified\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`last\_modified\_time` datetime DEFAULT CURRENT\_TIMESTAMP,

`description` varchar(255) DEFAULT NULL,

`name` varchar(255) DEFAULT NULL,

`category\_id` bigint(20) DEFAULT NULL,

`price` float(20,3) DEFAULT 0,

`discount` float(20,3) DEFAULT 0,

`quantity` bigint(20) DEFAULT 0,

`delete\_status` int(10) DEFAULT '0',

FOREIGN KEY (`category\_id`) REFERENCES pramatikart\_product\_category(id)

) ;

CREATE TABLE `pramatikart\_orders` (

`id` bigint(20) NOT NULL Primary Key AUTO\_INCREMENT,

`created\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`created\_time` datetime NOT NULL DEFAULT CURRENT\_TIMESTAMP,

`last\_modified\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`last\_modified\_time` datetime DEFAULT CURRENT\_TIMESTAMP,

`customer\_id` bigint(20) DEFAULT 0,

`date\_of\_purchase` datetime DEFAULT CURRENT\_TIMESTAMP,

`shipping\_date` datetime DEFAULT CURRENT\_TIMESTAMP,

`amount` float(20,3) DEFAULT 0,

`product\_id` bigint(20) DEFAULT 0,

`delete\_status` int(10) DEFAULT '0',

FOREIGN KEY (`customer\_id`) REFERENCES pramatikart\_user(id),

FOREIGN KEY (`product\_id`) REFERENCES pramatikart\_product(id)

) ;

CREATE TABLE `pramatikart\_payment` (

`id` bigint(20) NOT NULL Primary Key AUTO\_INCREMENT,

`created\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`created\_time` datetime NOT NULL DEFAULT CURRENT\_TIMESTAMP,

`last\_modified\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`last\_modified\_time` datetime DEFAULT CURRENT\_TIMESTAMP,

`transaction\_number` varchar(255) DEFAULT NULL,

`mode\_of\_payment` varchar(255) DEFAULT NULL,

`amount` float(20,3) DEFAULT 0,

`date\_of\_payment` datetime DEFAULT CURRENT\_TIMESTAMP,

`delete\_status` int(10) DEFAULT '0',

`order\_id` bigint(20) DEFAULT 0,

FOREIGN KEY (`order\_id`) REFERENCES pramatikart\_orders(id)

) ;

CREATE TABLE `pramatikart\_cart` (

`id` bigint(20) NOT NULL Primary Key AUTO\_INCREMENT,

`created\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`created\_time` datetime NOT NULL DEFAULT CURRENT\_TIMESTAMP,

`last\_modified\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`last\_modified\_time` datetime DEFAULT CURRENT\_TIMESTAMP,

`customer\_id` bigint(20) DEFAULT 0,

`product\_id` bigint(20) DEFAULT 0,

`quantity` bigint(20) DEFAULT 0,

`total\_price` float(20,3) DEFAULT 0,

FOREIGN KEY (`customer\_id`) REFERENCES pramatikart\_user(id),

FOREIGN KEY (`product\_id`) REFERENCES pramatikart\_product(id)

) ;

CREATE TABLE `pramatikart\_offers` (

`id` bigint(20) NOT NULL Primary Key AUTO\_INCREMENT,

`created\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`created\_time` datetime NOT NULL DEFAULT CURRENT\_TIMESTAMP,

`last\_modified\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`last\_modified\_time` datetime DEFAULT CURRENT\_TIMESTAMP,

`category\_id` bigint(20) DEFAULT NULL,

`discount` float(20,3) DEFAULT 0,

FOREIGN KEY (`category\_id`) REFERENCES pramatikart\_product\_category(id)

) ;

CREATE TABLE `pramatikart\_sale` (

`id` bigint(20) NOT NULL Primary Key AUTO\_INCREMENT,

`created\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`created\_time` datetime NOT NULL DEFAULT CURRENT\_TIMESTAMP,

`last\_modified\_by` varchar(255) DEFAULT 'admin@pramatikart.com',

`last\_modified\_time` datetime DEFAULT CURRENT\_TIMESTAMP,

`start\_date` datetime DEFAULT CURRENT\_TIMESTAMP,

`end\_date` datetime DEFAULT CURRENT\_TIMESTAMP,

`product\_id` bigint(20) DEFAULT 0,

`discount` float(20,3) DEFAULT 0,

FOREIGN KEY (`product\_id`) REFERENCES pramatikart\_product(id)

) ;