Project Planning

Group Project

Kevin Wu 2242923

Haris Hussain 2234354

Tony Do 2239932

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Kevin Wu | Haris Hussain | Tony Do | Milestones |
| Week 1 | Make ERD  Insert data | Normalize data | Create Tables | Database set up  Fully inserted data |
| Week 2 | -uninstall subprogram  -Check for stock available or not subprogram  -subprogram for the total stocks across different warehouses  -2 java classes (objects) | -save order subprogram  -Avg review function  -SQL Exceptions  -2 java classes (objects) | -modify stocks subprogram  -check for flagged customers reviews  -Audit logs  -2 java classes (objects) | Finished SQL and started Java classes |
| Week 3 | -procedures for the 2 classes (add object and such) while implementing JDBC  -Set up connection methods (close and get)  -Store procedures in their respective packages | -procedures for the 2 classes (add object and such) while implementing JDBC  -Test java methods  -Input for SQL validation | - procedures for the 2 classes (add object and such) while implementing JDBC  -Input validation for Java | Finished Java application |

# Understanding the Project & Planning

1. The project is to implement a database design and implement JDBC with a Java application that interacts with the database. It will track a customer’s orders, inventory and their information, including customer reviews. We must analyze a dataset given by our teachers and normalize it. The problem we must solve is to keep track of its products, its inventory, and the customer’s orders through different stores. We’re given a super store in the format of a spreadsheet, and we must normalize it, create the tables and data.
2. Products are articles that are manufactured for sale. Each product has a unique name, it’s impossible to have 2 products with the same name. Each product has a unique name, a category, a review, a review score (1 to 5) and a price.
3. An order is made through a store, by a customer only and only if a product has more than 1 quantity and only if there is available stock in the warehouse. Each order keeps track of the customer detail, address, detail of the product, order date, quantity, and their review score and review description.
4. A store is a retail establishment selling products to the public. For which we’re making inside the database to store the data.
5. It will run using PL/SQL and runnable on PDBORA19C.
   1. Java.sql.\*; the sql library
   2. It can be managed on different files. But in SQL Oracle, it will be managed in packages. Code should also be managed in procedures. Functions should compute derived data and perform validation.
   3. Put the dependency in the pom.xl, copy the url to connect to the database (url)

Create maven project.

* 1. Input the username and password.

1. Use gitlab, having a private project shared between teammates which will use different merges and merge requests within our own respective branches so that each teammate can simply look at the code and double check if everything is all good before merging to Main.
2. We have until November 19th to finish the project.
3. Every lab
4. November 19th

* A readme.md that includes to link to your gitlab repo,
* Ensure all steps to setting up the database are well documented.
* Ensure all steps to compile and run the Java application are well documented.
* Ensure names and student ids of group members are listed –
* Design.pdf containing ERD diagrams describing your final design.
* Source code for the PL/SQL packages setup.sql which setups up the database, creates the appropriate packages etc.
* remove.sql which removes any tables, types, and packages created by setup.sql
* Source code for the Java application Must include any necessary configuration files such as a POM.xml to run the code.

# Examining the Requirements

## Database

* 1. The database needs to capture information related to the products, orders, customers, reviews, inventory, store. Each of the entities should have tables to store their own data.
  2. Since we have quite a few tables, each table will be somehow related products will be related to orders, orders are related to customers, customers are related to reviews, products are related to inventory and inventory is related to store

1. Currently unsure but we will have Procedures for placing orders, updating inventory, remove or add products, functions to find specific products or anything to sort.
2. Procedures in the PL/SQL packages will be responsible for performing any action that revolves around updating, removing, inserting data in the database. Like Mentioned above, update inventory, update price and such.

1. Functions will be used to compute derived data and returning specific values we are looking for. For example, order total, average review, search for specific values.

## Application

1. As of now we will use the command line interface but if time permits, we will learn and implement A GUI
   1. Users will enter information in the command Line in java using JDBC
   2. Information will be displayed to the user through text in The CLI of whatever IDE is being used.
2. A user could place orders for products, view product details and their availabilities, making and seeing reviews, checking inventory levels, checking customer information (modify if he has admin rights)
3. The application will use JDBC to establish a connection to the database. It will send sql queries and commands to retrieve, update, manage data. Call made procedures and functions from packages.
4. Objects will represent the entities we have such as products, orders, customers and etc. So, the amount will be based on the number of Tables we will have.

## Starting Point

1. The first step will be normalizing the provided dataset and design a schema. Making the tables structures, their relationship, PKS and FKS and constraints such as Not null.
2. After designing the schema, next step will be to create and set up necessary tables and relationships in the database. Additionally, we will start setting up the Java project structure.
3. To efficiently manage the project each person will do a little bit of everything since all of us need to be involved in each part of the work. We will divide everything based on comfortability and availabilities for each.

# Code Design and Setup

1. -Fall2023Database2FinalProject (folder)

-sql (folder)

-setup.sql

-remove.sql

-documents (folder)

-ProjectPlanningExercise.doc

-ProjectPlanning.pdf

-Design.pdf

-code (folder)

-src (folder)

-main (folder)

-java (folder)

-(package) (folder)

-(group) (folder)

-App.java

-Services.java (all the methods)

-product.java

-orders.java

-customers.java

-reviews.java

-inventory.java

-store.java

-test (folder like main)

-target (folder)

-classes (folder like main)

-test-classes (folder like main)

-pom.xml

1. The scripts needed will be: setup.sql, remove.sql and Pom.xml for dependencies.
2. <https://gitlab.com/wuKevin/fall2023database2finalproject>
3. Check gitlab already done.