

System Design Document

<Joy’s Toys>

<Seth White>

CSC-289 Programming Capstone Project

Guilford Technical Community College

Professor Scott Ilkenhons, MS

|  |  |
| --- | --- |
| Document ID | SYSTEM DESIGN-v0.1 |
| Version Number | 0.1 |
| Issue Date | Friday, November 4, 2022 |
| Classification | Private |

**Table of Contents**

[List of Tables 3](#_Toc117936549)

[List of Figures 4](#_Toc117936550)

[1 Introduction 5](#_Toc117936551)

[1.1 Purpose of Document 5](#_Toc117936552)

[1.2 Acronyms and Abbreviations 5](#_Toc117936553)

[2 SWOT Analysis 5](#_Toc117936554)

[2.1 5](#_Toc117936555)

[3 Database Design 5](#_Toc117936556)

[4 Application Detailed Design 5](#_Toc117936557)

[5 External Interface Design 5](#_Toc117936558)

[5.1 Interface Architecture 5](#_Toc117936559)

[5.2 Interface Detailed Design 5](#_Toc117936560)

[6 Network and Physical Architecture Design Overview 6](#_Toc117936561)

[6.1 Background Information 6](#_Toc117936562)

[6.2 System Evolution Description 6](#_Toc117936563)

[7 Training Plan 6](#_Toc117936564)

# List of Tables

2.1 [SWOT Analysis](#_SWOT_Analysis)

# List of Figures

3 [Database Design](#_Database_Design)

3.2 Database Function

4 [Application Design](#_Application/Website_Detailed_Design)

Figure 4.1

Figure 4.2

Figure 4.3

Figure 4.4

6.1 [System Evolution Description](#_System_Evolution_Description)

# Introduction

This is the Software Design Document for the development of the Web Application for Joy’s Toys. Refer to this document for the list of concerns that will be/have been addressed.

## Purpose of Document

This SDD will identify opportunities that our application will address for the business *Joy’s Toys.* Throughout the development process this document will serve as a source of guidance. This document divides itself into a SWOT Analysis, Database Design, Application/Website Detailed Design, External Interface Design, an overview of Network and Physical Architecture Design, and a Training Plan.

The SWOT Analysis identifies the Strengths, Weaknesses, Opportunities, and Threats of our application.

The Database Design provides an abstract overview of our database.

The Application/Website Detailed Design will contain an abstract overview/layout of the design of the application.

The External Interface Design section will describe the architecture and design of the External Interface.

The overview of the Network and Physical Architecture Design will address the hardware requirements and network design and set up for this application.

Finally, the Training Plan will include a structured plan to train employees on how to use this system. Training will be broken down into sessions that employees will attend.

The system name is Joy’s Toys. The app will be identifiable through key features of Inventory Tracking, Customer Account creation and management as well as access for customers to product availability, Organization for Employees, eCommerce, and a system to access and modify financial data for specified employees.

## Acronyms and Abbreviations

|  |  |
| --- | --- |
| GUI | Graphical User Interface |
| SDD | System Design Document |
| ERD | Entity Relationship Diagram |
| APP  CRUD | Web/Mobile Application  Create, Read, Update, Delete |

# SWOT Analysis

|  |  |
| --- | --- |
| Strengths   * Good reputation among local community * Familiarity with customer base | Weaknesses   * Organization/Structure * Tracking * Accessibility for customers and employees |
| Opportunities   * Inventory tracking * eCommerce(selling, buying, etc) * Daily Balancing(reconciling credits and debits) * Payroll * Accounts Receivable * Accounts Payable * Order forms * Customer receipt generator * Customer Request form * New Customers * Memoization for employees * Customer Accounts (online accounts, buying history, etc) | Threats   * Other online competitors * Supplier shortages/lack of storage space * Losing employees to other businesses |

# Database Design

\*Entity Relationship Diagram\*

Diagram

Description automatically generated

\*this design is outdated and has been changed in miro

This ERD details the tables and illustrates the relationships between these tables. This will be the overall organization of Joy’s Toys database thus far.

The tables involved are:

Customer

Payment Information

Customer\_Order

Customer\_Transaction

Customer\_Order\_Product\_Association

Products

Categories

Product\_Category\_Association

Inventory\_Order

Inventory\_Transaction

Delivery

Supplier

Employee

Job\_Role

A Customer is a defined object for a customer that will be able to access the app, which will in turn use information that the Customer will provide to the database for purchases and profile purposes.

A customer’s payment information will be stored in the Payment Information table, which will be related to Customer through a foreign key in the Payment Information table.

Customers can have multiple Orders, but only one Customer per Order. Orders may also contain many Products, and many Products can be in many Orders. This will be accomplished through an Associative Entity (or Junction table), called Customer\_Order\_Product\_Association.

Many Products can be in Many Categories, and one Category can contain Many Products. This will be accomplished through an Associative Entity called Product\_Category\_Association.

Once a Customer\_Order is complete, a Customer\_Transaction is generated for the customer. Only one Customer\_Transaction is generated per order, which will contain the Order ID number along with Payment Information.

Inventory is tracked through the Products and Inventory\_Order table. Once a Product decrements to a certain amount in Quantity the designated Employee can make an order for more inventory.

The Inventory\_Order table contains a foreign key to the Products table. It will also contain an ID, an order date, the purchasing price per unit, and a quantity ordered.

The Inventory\_Order table is linked by foreign key to the Delivery table. The Delivery table contains an ID, a Supplier ID, the expected delivery date, and the aforementioned Inventory order ID.

The Supplier table will be linked by foreign key to the Delivery table. The Supplier table will contain a name, address, phone number, and email of the Supplier.

An Employee will contain standard information about an employee, such as name, address, phone number, email, bank account number, etc…

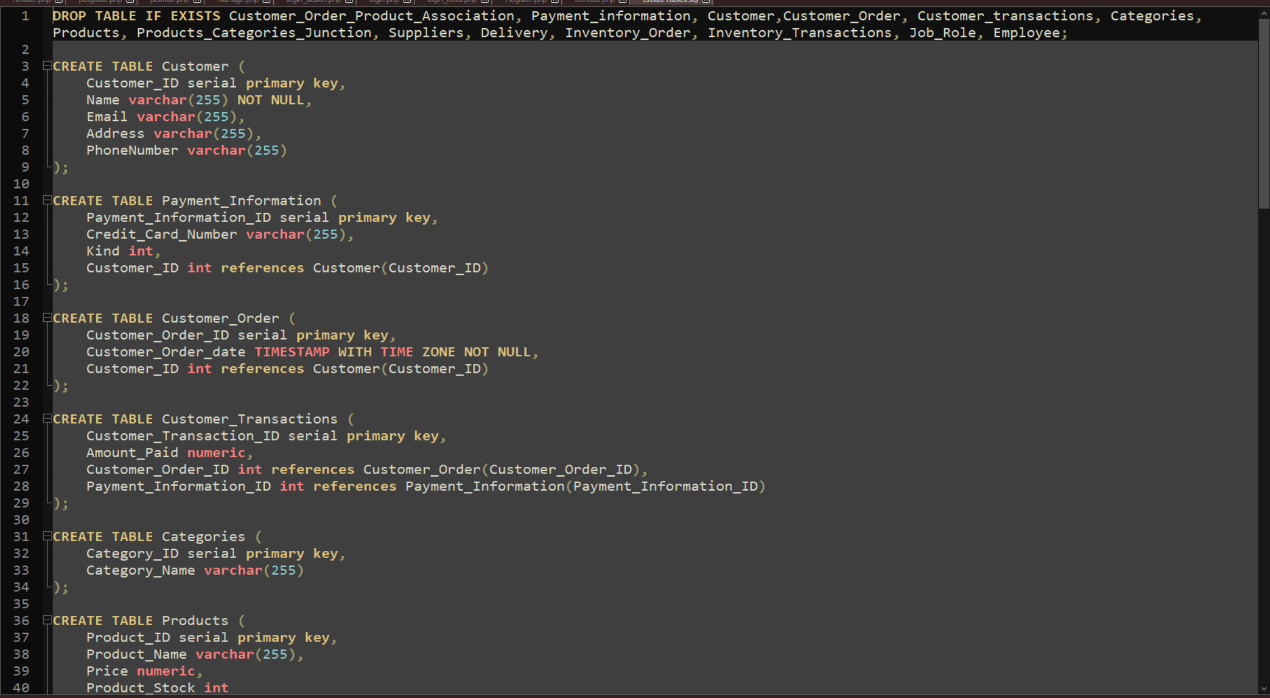
The number of hours worked by a particular Employee will be part of the Employee object as well. One Employee may only have one Job Role, however a specific Job Role can be assigned to many different Employees.

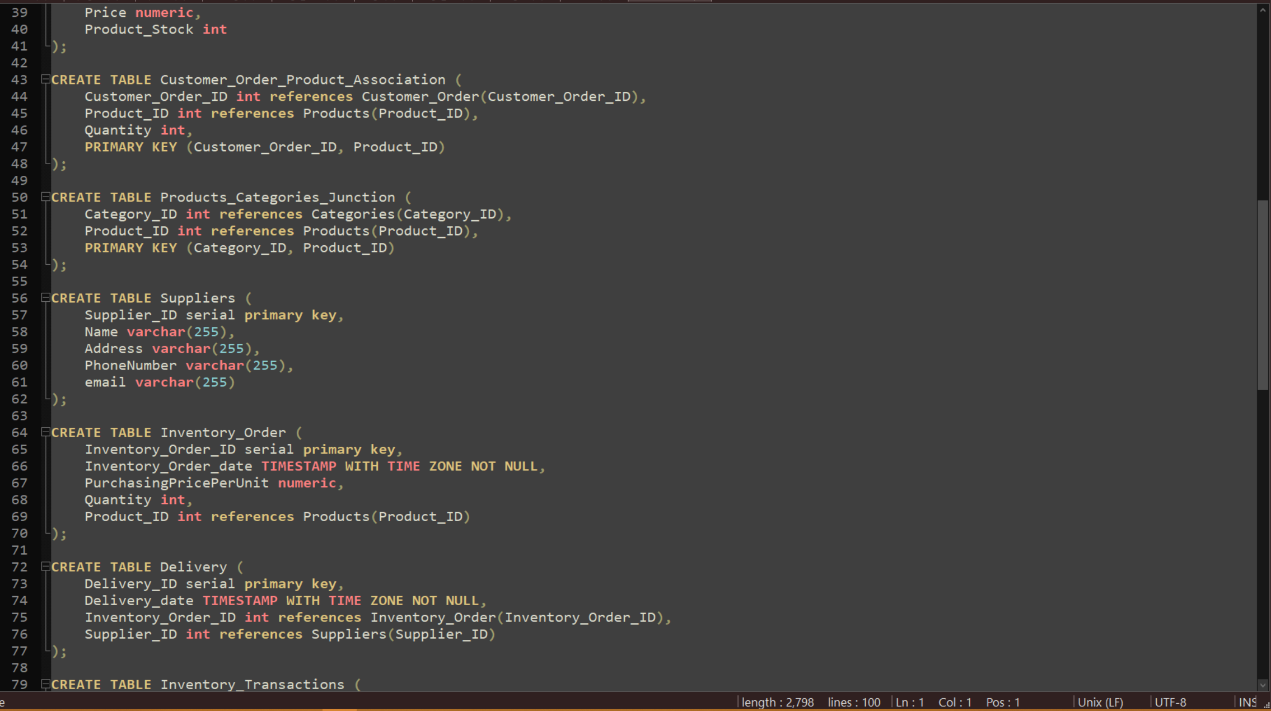
A specific Employee is granted access to complete an Inventory\_ Order. These are transactions that order more inventory of one specific item from a specific Supplier. Once an Inventory\_Order is complete an Inventory\_Transaction is generated that corresponds to that particular Inventory\_Order. Only one transaction per one order. Along with the Inventory\_Transaction, a Delivery\_date will be assigned from the particular Supplier and stored in the Delivery object. Only one supplier can supply one Delivery\_date per order.

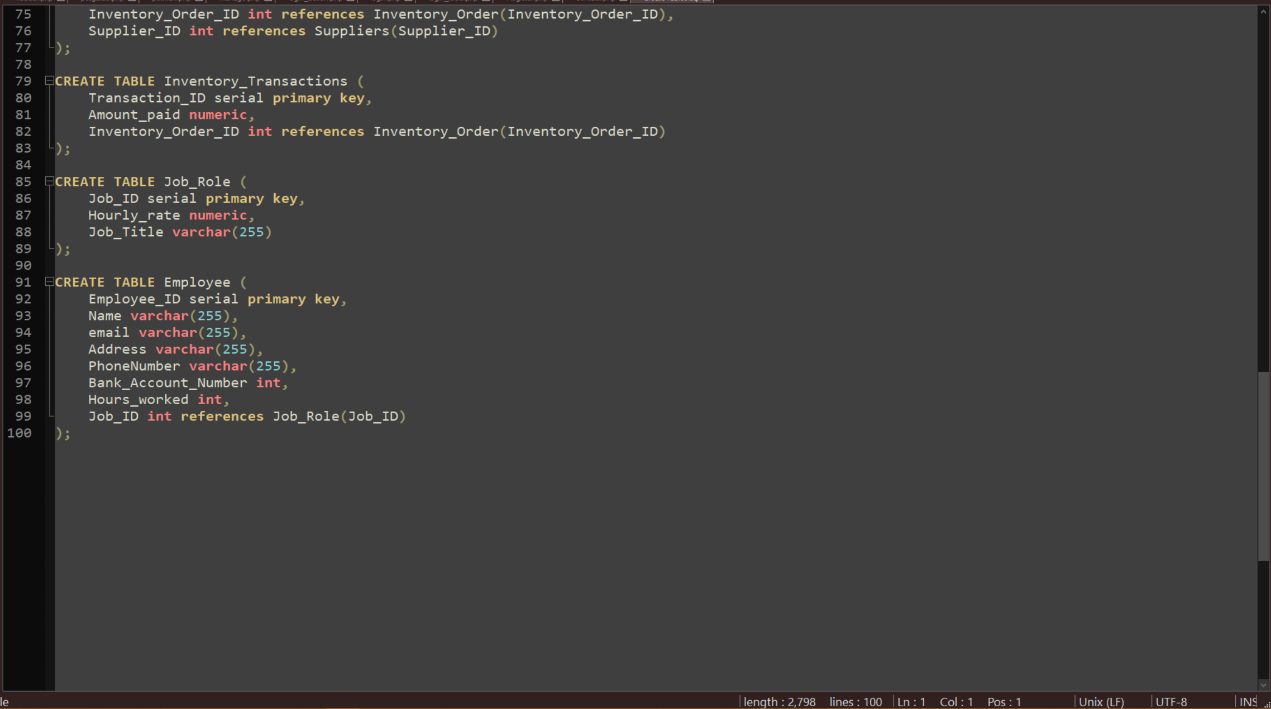
3.2 Database Function:

Below are images of the function and design of the database.

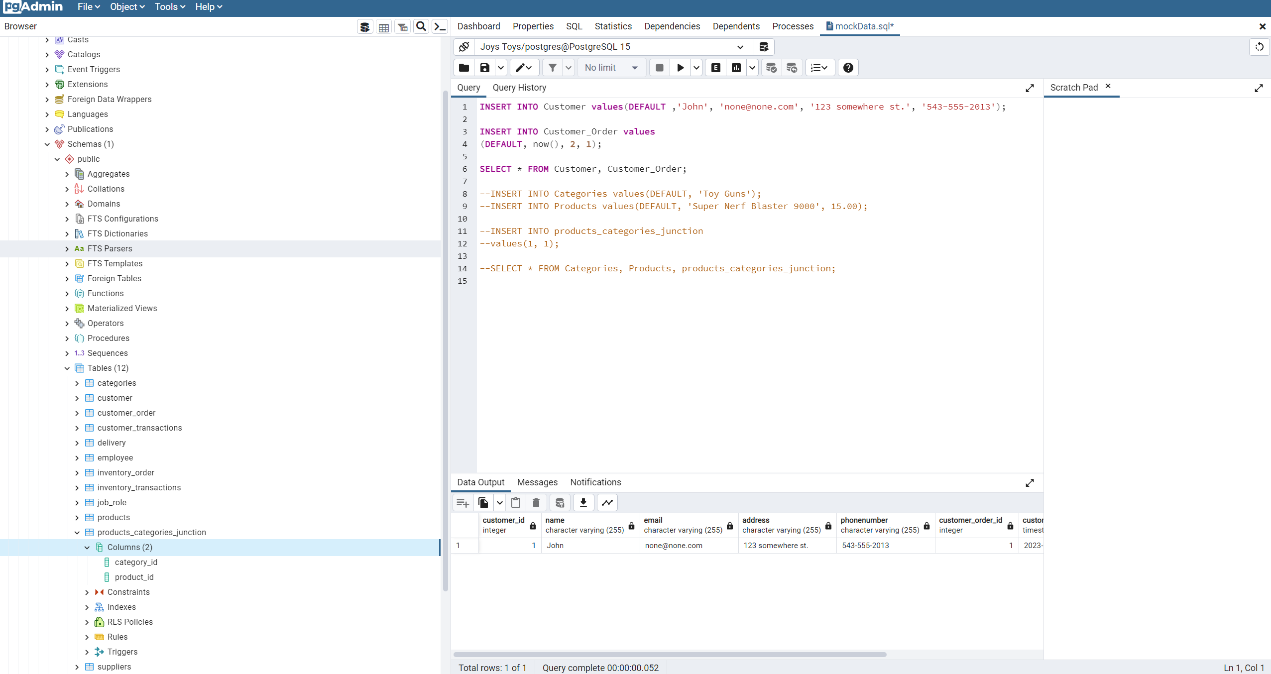
Table Creation:



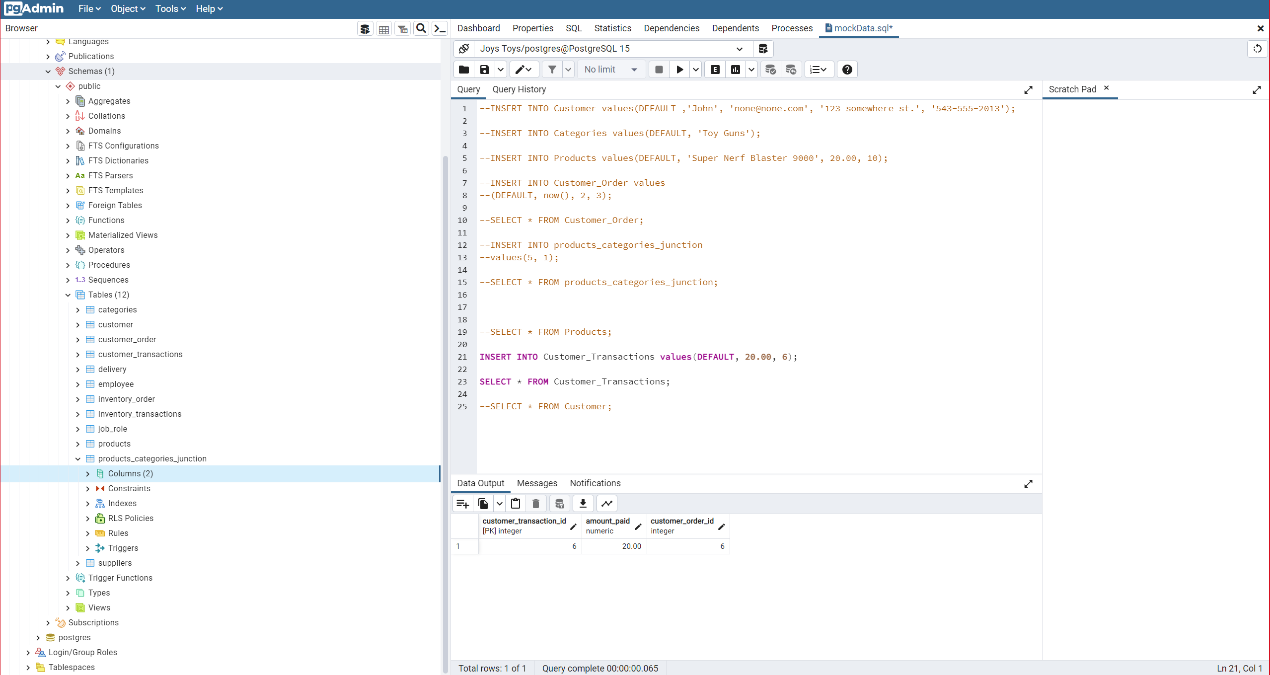


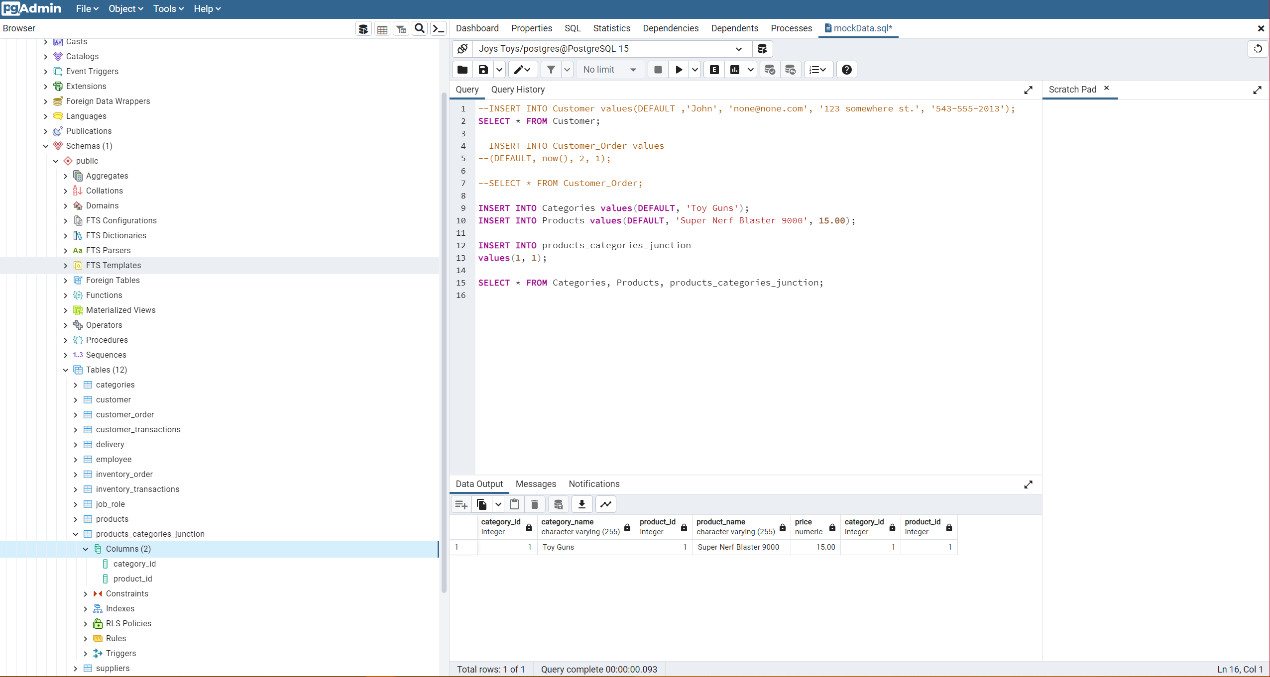


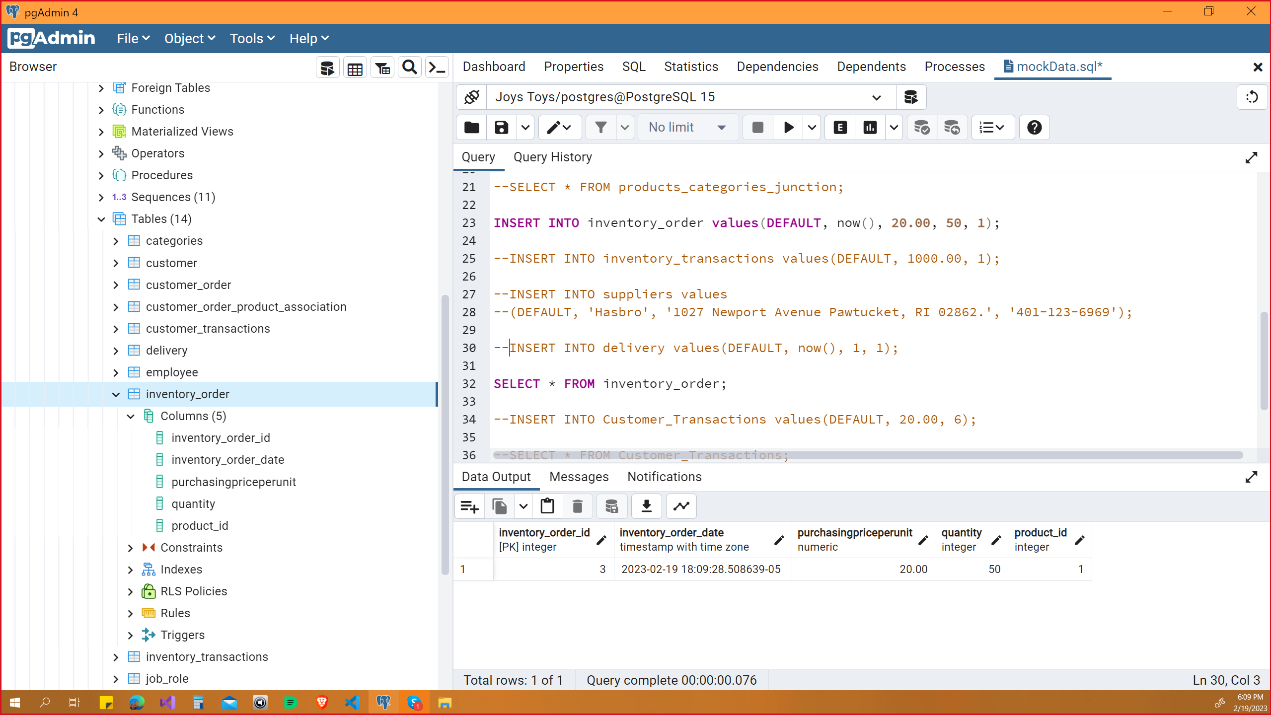
Customer and Customer\_Order:



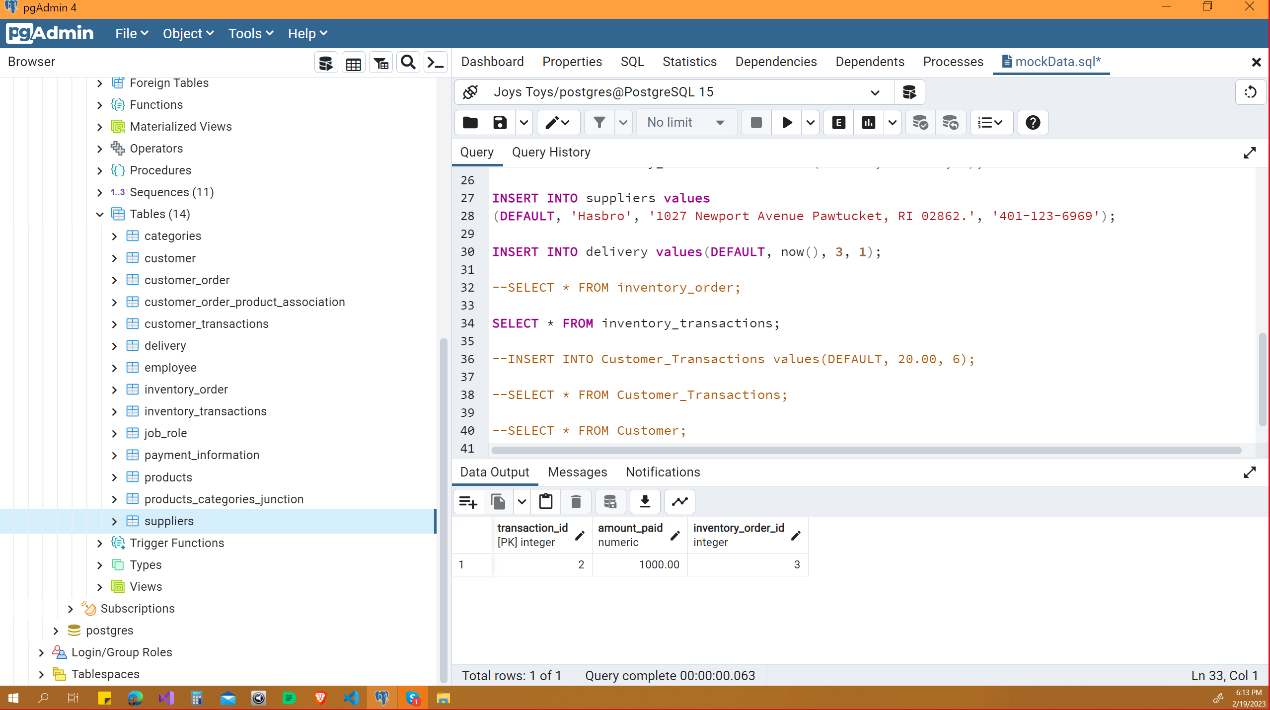
Customer\_Transaction:

Product\_Category\_Association:

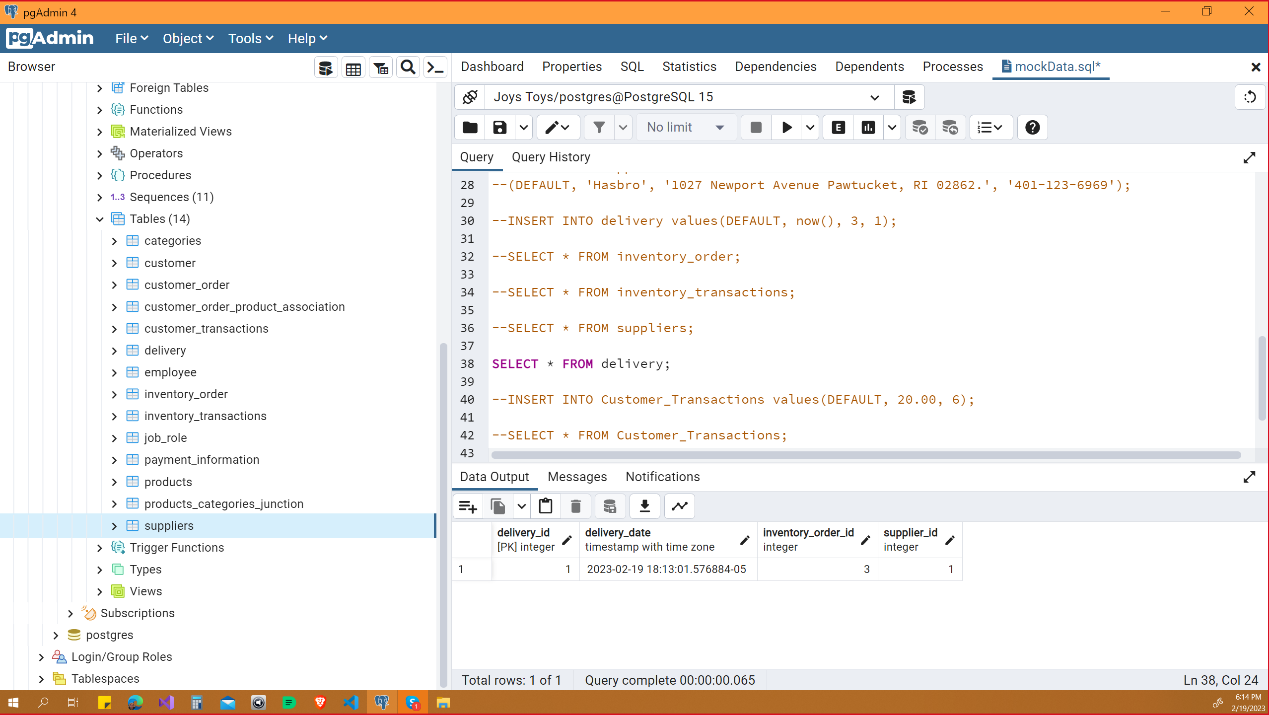


Inventory\_Order:

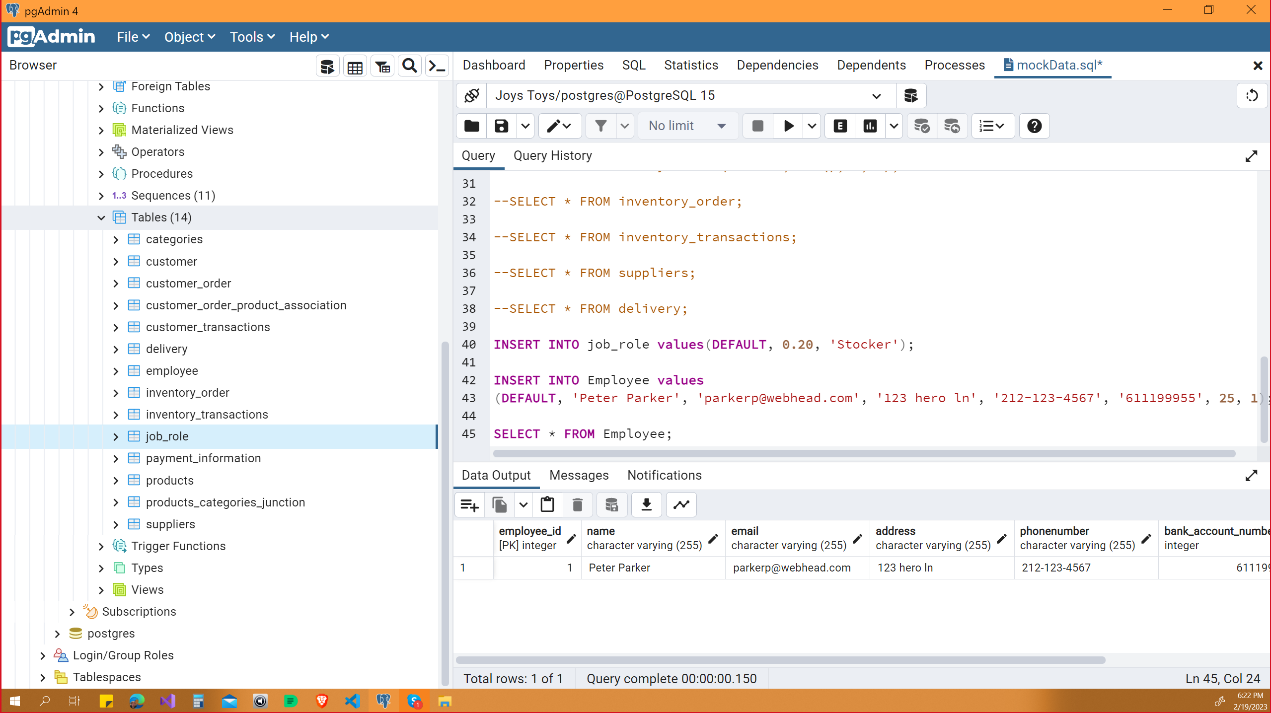
Inventory\_Transaction:



Delivery:



Employee and Job\_Role:



# Application/Website Detailed Design

The application will follow the Client-Server Architecture. The Server will manage the database and expose the web APIs for which the Clients will then talk to. The Client being the web application.

The client of the application is implemented with the React framework. The client provides the GUI that the end user will view and interact with. More importantly, the client speaks to the server, which is connected to the database, in order to populate the webpage with the data from the database in the structure that the client provides.

The Server is implemented in Java SpringBoot. Its implementation models the structure of the PostgreSQL database with “entities”, “repositories”, and “controllers”. Entities model the tables, and its records and fields. The repositories, using the Jpa repository, implicitly constructs the SQL statements necessary to perform CRUD operations. Finally, controllers are responsible for presenting the data to the end user. In our case, the client will interact with the exposed APIs by the controllers from the server.

Currently, the application is incomplete. The items are being summoned from the database and presented on the webpage. However, the shop functionality, user creation/login process, and employee portal have yet be implemented.

The overall schema of the application is modeled by Figure 4.5. This is the UML diagram of the Joys Toys Application, where user interaction and intended functionality are modeled.

The shop will be visible to the public, where potential customers may browse the categories and the products contained within. Customers who wish to make purchases will need to create an account, which will contain purchase history and payment information. Customers may edit their accounts in order to provide the experience they want.

Employees will also have separate access to the app, with necessary privileges to access information and to perform specified tasks.

Figure 4.1 shows the server implementation In SpringBoot.

Figure 4.2 shows the data in JSON format that is summoned from SpringBoot.

Figure 4.3 shows the client implementation in React.

Figure 4.4 show the webpage that is generated with React connected to SpringBoot.

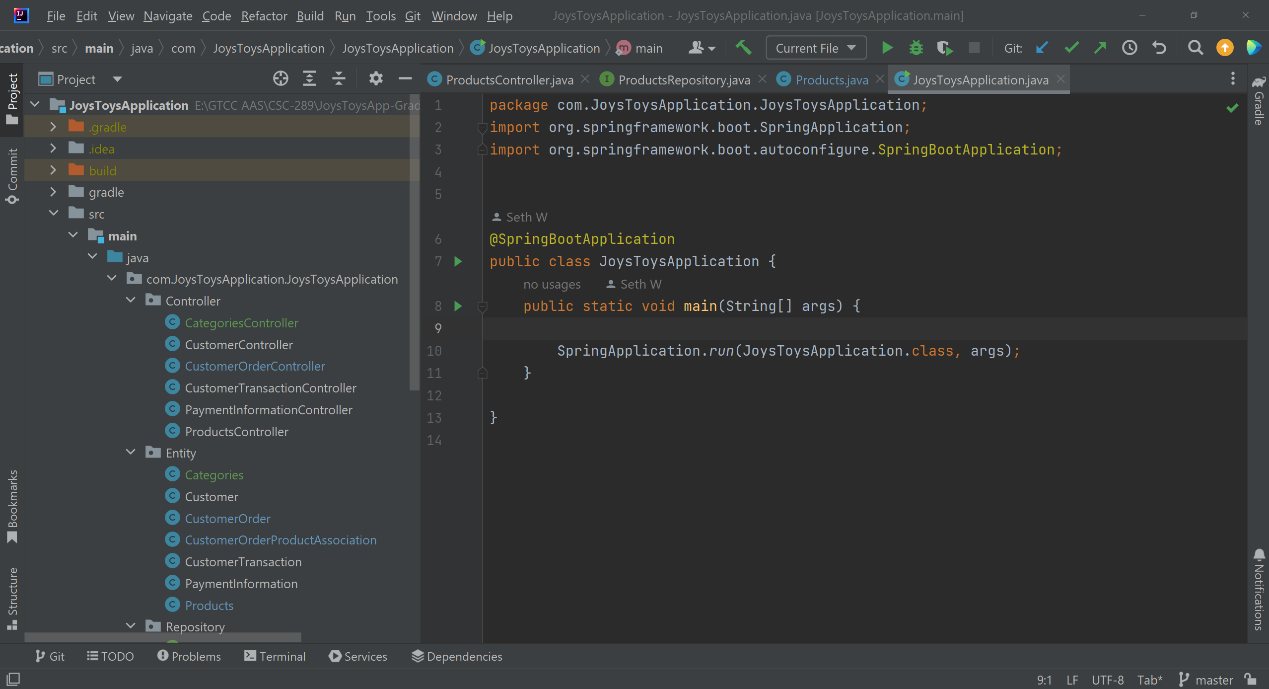


Figure 4.1

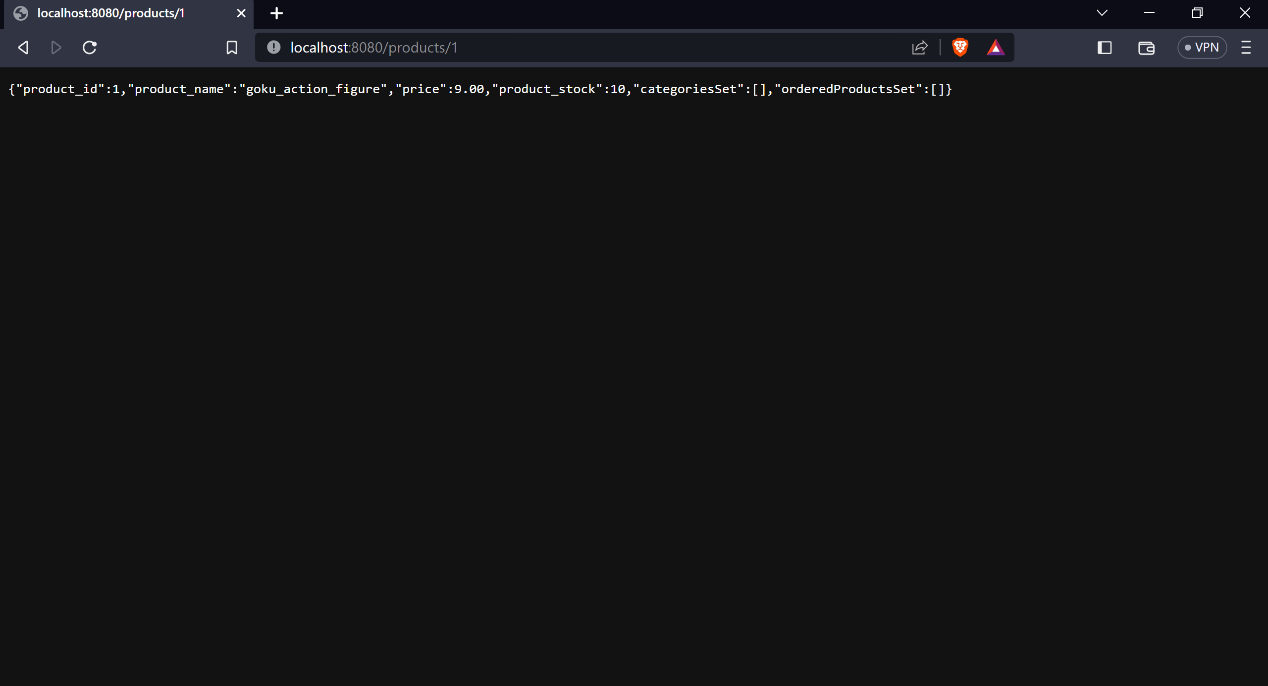
Figure 4.2



Figure 4.3

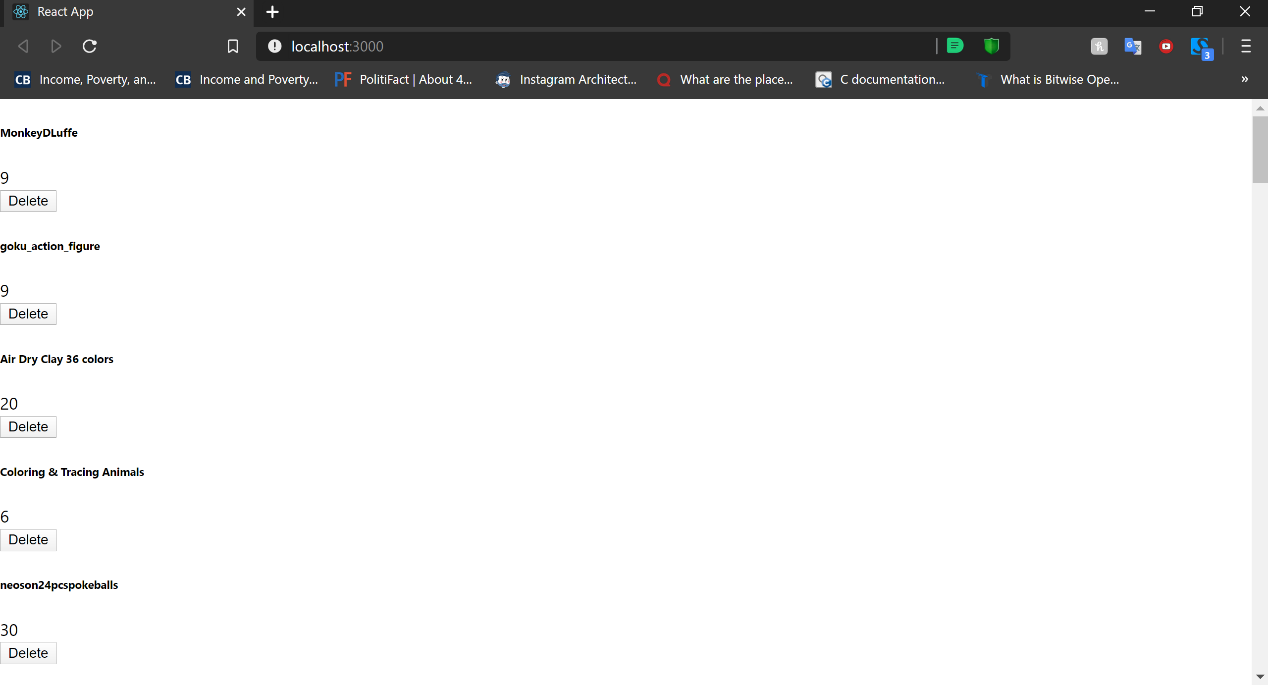


Figure 4.4

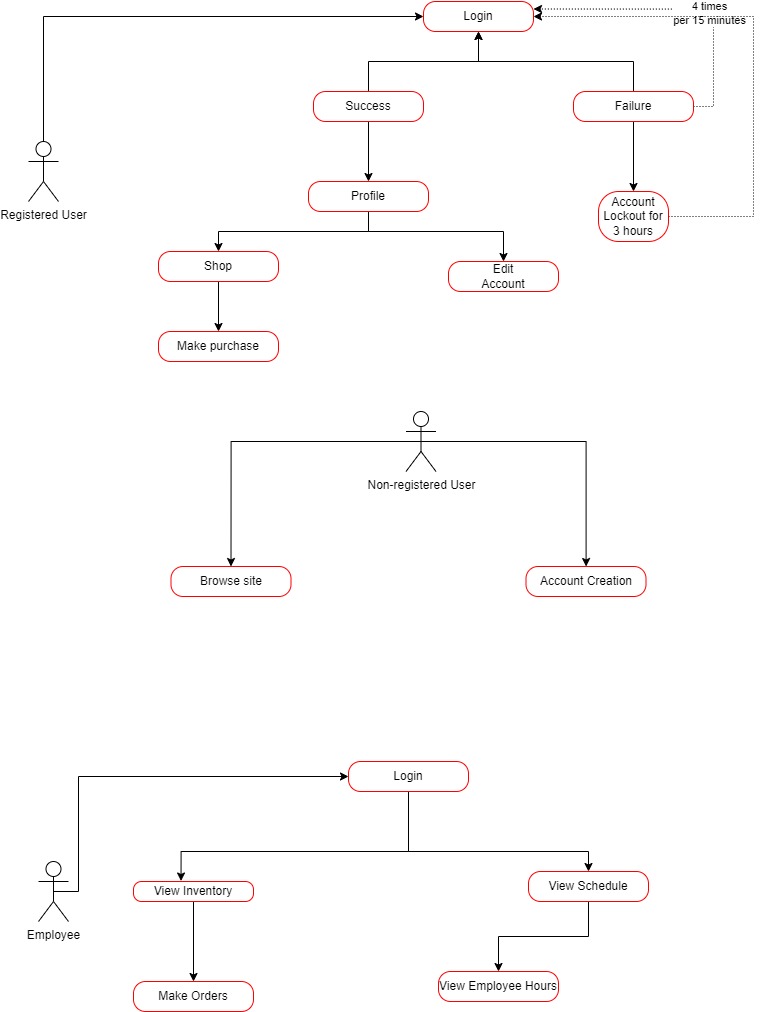


Figure 4.5

# External Interface Design

## Interface Architecture

<Describe the interface architecture>

## Interface Detailed Design

<Describe the interface detailed design>

# Network and Physical Architecture Design Overview

## Background Information

Currently, Joy’s Toys do not have any computers in the building, as well as no app that customers/employees can access and make purchases, changes, or track history with. We are attempting to build an app that will do the aforementioned for Joy’s Toys. The app will require certain hardware to be introduced to the business.

In order to create the app for Joy’s Toys, the hardware/software will include a Server, a network firewall, a modem and router, 4-5 tablets, and a thin-client desktop for data entry. The mobile app will be accessed through a mobile phone.

## System Evolution Description

\*Network Diagram\*

A picture containing text, sky

Description automatically generated

Currently, Joy’s Toys Network Architecture is nonexistent. A Server will be added to provide services to the app, while the modem and router shall provide access to the internet. The network firewall will be placed in between the internet and the Server for extra protection. Clients will then be able to access the Server via wireless and wired connection.

A backup server will be provided, in the case of disasters occurring.

## Disaster Recovery Plan

A list of hardware that will be required for Joys Toys web infrastructure will include:

* A stable internet connection
* A server for the web application
* A backup server for recovery
* 1 router
* A firewall
* Thin clients, including two desktops, and pad mobile devices

The likelihood of a natural disaster occurring is relatively low. Even still, we will provide a level of redundancy for the case that a natural disaster does occur.

The web server will be rented from a server site, which means that it will be located at an offsite facility. So in the case of a natural disaster occurring, the data for Joys Toys Application will be secure.

The backup server will also be kept at an offsite location. This will provide a level of redundancy for data recovery, in the case that something disastrous happens at the server site location.

Due to construction of the React framework, many dependencies are included in the construction of the Joys Toys application. Fortunately, React is a well-maintained framework making it very easy to manage the security to these dependencies. Most updates cover many of the security vulnerabilities. An upgrade schedule will be created and adhered to. This will fall under the category of “regularly scheduled maintenance” which shall occur according to the upgrade schedule.

Spring Boot utilizes hibernate, which uses prepared SQL statements. As well as providing a level of abstraction, this also provides a level of security against SQL injection.

Payment information will be stored on the server and backup server which will then be protected by the Oauth2 authentication process.

Occasionally, downtime may occur. Whether the cause is regularly scheduled maintenance, network issues, or an issue with the server itself. As far as server issues are concerned, the backup server provides a level of redundancy. Thus, if an issue with the server does occur, then the backup server will take over the main role of responsibility in providing functionality of the web application.

As for other issues, an investigation shall be conducted once an issue is detected. Any issue with the network, or otherwise will be fixed as promptly as possible.

# Training Plan

Joy Johnson: Joy will have full access to the application. Therefore, she will require the most training out of all the employees, save for Bea.

The training will include

* Computer Training
* Mobile phone training
* Application training
  + How to use both the web application and the mobile application
  + Admin role creation and assignment
  + Security Training

Bea Bernard: Bea is the accountant, so she will have full access to the information the database holds. This will include special authorization within the app itself, where she will be able to access financial tables and records, stock and inventory, as well as employee data. Bea’s training will include:

* Application Training
  + How to use both the web application and the mobile application
  + Admin role creation and assignment
  + Security Training
* Security Training

Peter Parker: Peter will be responsible for stock/inventory. He will be able to access and order inventory through the app. Peter’s training will include:

* Application training
  + How to use the web and mobile application
  + How to access inventory information and make orders
* Customer Service
  + Customer training and support

Gwen Stacy: Gwen will be the Customer Sales Representative, so she will be the first point of contact between customers and Joy’s Toys. Therefore, she will need:

* Application Training
  + Sales
  + Stock
  + Receipt
* Customer Service support
  + Helping customers use the app