*App Name* Design Document

(image of logo)

Designed by:

Lucian Irsigler

Banzile Nhlebela

Laaiqah Bayat

Prashan Rajaratnam

Version:1.0

Date Created: 20 May 2023

Contents

[1. Introduction 3](#_Toc136116822)

[1.1 App Purpose 3](#_Toc136116823)

[2. User Interface Design and Data Security 4](#_Toc136116824)

[2.1 Login Screen 4](#_Toc136116825)

[2.2 Buyer Screens 5](#_Toc136116826)

[2.2.1 Review Product 5](#_Toc136116827)

[2.3 Seller Screens 6](#_Toc136116828)

[2.3.1 Adding Product 6](#_Toc136116829)

[3. Business Rules 7](#_Toc136116830)

[4. Entity Relational Diagrams (ERD) 8](#_Toc136116831)

[4.1 ERD 8](#_Toc136116832)

[4.2 Implementation of ERD and Addressing Issues 9](#_Toc136116833)

[4.3 Normalization 10](#_Toc136116834)

# 1. Introduction

## 1.1 App Purpose

“AppName” is a user-friendly mobile marketplace designed for buyers and sellers to interact in a seamless manner. Our app will act as an intermediary between the two users by use of a multitude of different features. When logging in and creating an account, the user will get the choice of whether they want to be a buyer or a seller. The following options will be available to the customer based on if they choose to be a buyer or seller.

As a buyer, you have access to the following features:

1. Create an account
2. Search for items
3. View items details in a sorted list
4. Review an item that you have purchased

As a seller, you have access to the following features:

1. Create an account
2. Post items and all their essential details
3. View how your customer’s have reviewed your products.

“AppName” offers a reliable and efficient mobile marketplace experience, promoting ease of access, security and seamless transactions between buyers and sellers.

# 2. User Interface Design and Data Security

This section will detail all the screens in which data validation and data verification is used throughout the app on different screens in order for all the necessary information to be as concise and accurate as possible.

## 2.1 Login Screen

(pic of log in)

|  |  |  |
| --- | --- | --- |
|  | Data Validation/ Verification/ Interactability | Function and Description |
| First name | Yes | Uses NAME\_REGEX to ensure that the user’s input starts with a capital letter followed by more letters, spaces, hyphens, or apostrophes. No numbers can be input here. |
| Last name | Yes | Uses NAME\_REGEX to ensure that the user’s input starts with a capital letter followed by more letters, spaces, hyphens, or apostrophes. No numbers can be input here. |
| Phone Number | Yes | Uses TextInputLayout to ensure that the user’s input is only numbers. We have made it such that the user must be South African and therefore their number starts with +27 followed by nine other numbers. |
| Date of Birth | Yes | Uses TextInputLayout to ensure that the user’s input is only numbers and dashes. We have also provided the format that the date should be in above where the user inputs their data. |
| Address | Yes | Uses TextInputLayout to ensure that the users can input anything required for an address, that being letters, numbers, and all types of punctuation. |
| Username | Yes | Uses TextInputLayout to ensure that the users can input anything required for an address, that being letters, numbers, and all types of punctuation. |
| Password | Yes | Uses PASSWORD\_REGEX to ensure that the user’s input has at least one uppercase, lower case, digit, and special character. It must be minimum 8 characters. |
| Email | Yes | Uses EMAIL\_REGEX to ensure that the user’s input starts with one or more letters followed by an optional hyphen or period. This is followed by “@” then one or more letters followed by an optional hyphen or period which represents the user’s email account domain. |
| Type of User | Interactable | Drop-down box for the user to choose if they are going to be acting as a customer or a seller. |

## 2.2 Buyer Screens

After logging in to the app, a buyer will have the chance to input data when searching for a specific product or reviewing a product. The search is based purely on the LIKE SQL command and therefore requires correct spelling from the user. Our app also uses a rating system that will be presented once the buy button has been tapped. The purchase and review screen can be seen below.

### 2.2.1 Review Product

(review/ purchasing screen)

|  |  |  |
| --- | --- | --- |
|  | Data Validation | Function and Description |
| Star Rating | NA | This will only appear once the potential buyer has pressed the buy button so we can ensure that only people who have bought this product can review it, allowing for the rating to be accurate. |
| Description | NA | No form of data validation or verification is needed here. |

## 2.3 Seller Screens

After logging in to the app, a seller will have the chance to input data when adding their product to the store. A picture of this screen and how data validation is used can be seen below.

### 2.3.1 Adding Product

(add product screen)

|  |  |  |
| --- | --- | --- |
|  | Data Validation/ Verification? | Function and Description |
| Product Name | Yes | Uses TextInputLayout to ensure that the user’s input can be letters, numbers, and any form of punctuation. We will also have a character limit to ensure that the layout of this page is not made untidy. |
| Description | N/A | No form of data validation or verification is needed here. |
| Price | Yes | Uses TextInputLayout to ensure that the user’s input is only numbers. We have made it such that the user enters both Rands and Cents as our app is South African based and accuracy is important. |
| Category |  | Seller can input the type of category or choose from a pre-made one. |

# Business Rules

Our app will run based on the following business rules

User Account:

* A seller can sell more than one product.
* A customer can write more than once review.
* Each seller that has had a review written on their product, will have an aberage review score.
* Each user must have a unique account tied to a valid email address and username.
* Users must provide accurate and complete information when registering.
* On the administration end, we will assign the user an ID.
* Each user must have the following required attributes: user ID username, password, Salt, first name, last name, email, phone, and address.

Products List:

* Sellers can list products that they have the rights to sell.
* A product may have many reviews.
* Every product belongs to a category.
* Each product must have the following required attributes: a product ID (primary key), a user ID, name, description, price, category (foreign key to Categories table) and a dated added.

Reviews:

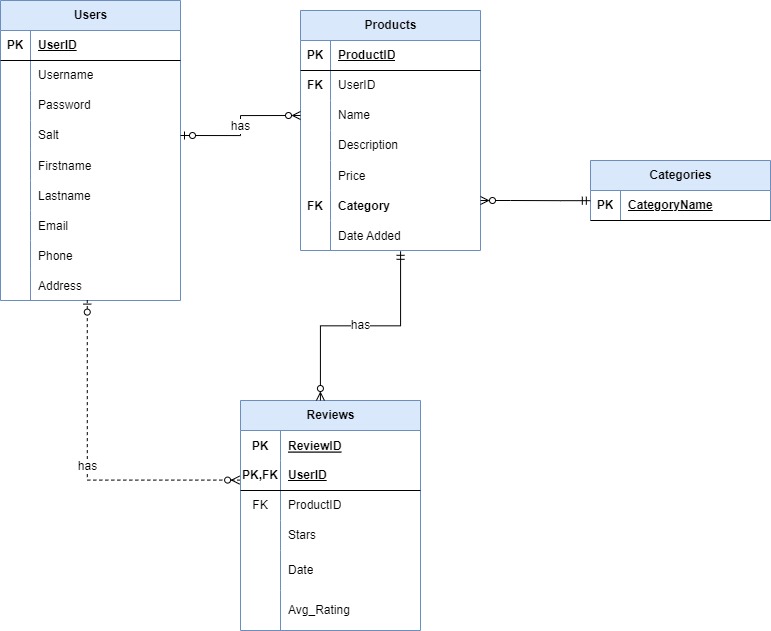
* Each review belongs to one product.
* A product may have a review with the option of multiple reviews from different buyers.
* Each review must have the following required attributes: a review ID (primary key), user ID (primary key as well as foreign key to Users table), product ID, the rating, and a date.

Categories:

* Each product belongs to a category.
* Each category may have many products.
* The categories table has 1 required attribute which will act as the primary key and that is the category name.

# Entity Relational Diagrams (ERD)

## ERD



## Implementation of ERD and Addressing Issues

PHP: For the PHP file structure, each table has a PHP file that contains it’s corresponding SQL query, i.e., insert, search and update. An example of this in practice is the Products table has a products.php that has the selectFromProducts, insertIntoproducts and so on.

Multivalued Attributes: Other than the user’s address in the Users table, there are no multivalued attributes. For the address attribute, we kept the address as one field instead of splitting it into street, suburb, and city because our mobile market app does not use any form of location-based services.

Entity Relations: We have developed our ERD in such a way that we have avoided the use of M: N to relationships such that fewer data anomalies occur.

Null variables: All of this has already been discussed in the data input screens. To recap, all necessary fields will always have some value in it because we used TextInputLayout to ensure that the data is correct and not Null.

## 4.3 Normalization

Legend for Normalization:

Underlined means primary key.

Red means foreign key.

**2NF**:

Users (UserID, Username, Password, Salt, Firstname, Lastname, Email, Phone, Address)

Products (ProductID, UserID, Name, Description, Price, Category, Date Added)

Foreign key to link the products a customer buys or sells.

Category (CategoryName)

Reviews (ReviewID, UserID, ProductID, Stars, Date, Avg\_Rating)

Foreign key to link Reviews to the product being reviewed.

**3NF**:

Users (UserID, Username, Password, Firstname, Lastname, Email, Phone, Address)

Salt (Password, Salt)

Products (ProductID, UserID, Name, Description, Price, Category, Date Added)

Foreign key to link the products a customer buys or sells.

Category (CategoryName)

Reviews (ReviewID, UserID, ProductID, Stars, Date)

Foreign key to link Reviews to the product being reviewed.

AverageStars (ReviewID, Avg\_Rating)

Foreign key to link transitive relationship between Reviews and AverageStars table.

THINGS TO DO

Screen descriptions and interactable can be used to show data validation and verifications

Check relation between seller and reviews. Because a seller has one avg review but a customer can write many reviews

Required methods:

Search by name

Sort by new items

Click on item and display name, a description, a price and a date posted

Rate items and show avg

Categories display