

Mr. Smith's Grocery Store App

Software Requirement Specification

Developers: Jeffrey Farnworth and Josh Wilding

Version 1.0 approved

9/21/16

1. Introduction

1.1 Purpose: Create an application for Mr. Smith's grocery store

We will create an executable that will run on a Windows computer that will fulfill these basic functions:

- User Login/Management
- Goods Database Management
- Customer Checkout and Receipt Printing

1.1. Intended Audience and Reading Suggestions

This document is prepared for Mr. Smith and his employees. It is designed to be read straight through.

1.2. Product Scope

This software is designed to aid Mr. Smith in the running of his grocery store. It is meant to aid in employee management, product management, and in the checkout process and printing of receipts.

2. Overall Description

2.1. Product Perspective

This product is a new, self-contained system that has the potential for additional features in future versions.

2.2. Product Functions

User Login

The program will require a login for all users. These logins will be classified as either Admin or Employee. Users will automatically log out when idle.

Products Database Management

The Products Database will work out of the box. However, users can import existing Excel spreadsheets (.xls) to the database.

Each good has the following information:

- name
- barcode number
- number of available units
- price per unit
- provider name

- provider contact information

Admins will be able to add new goods and edit/delete existing goods.

Customer Management

The system will track customers of the store.

Will these be customers that sign up?

Will customers that don't volunteer information still be tracked by payment method?

What information will be tracked?

Admins/Employees will be able to edit the Customer Database.

Customer Checkout and Receipt Printing

Employees/Admins will be able to use Customer Checkout

Each transaction will be assigned the employee that initiated it.

Each good will be "scanned" (barcode number entered) or searched for and added.

Quantity automatically be 1, but can be altered.

Scale can be used when goods require.

Inventory will auto-update.

Receipt will contain the following information:

- Company logo
- Company info
- Receipt number
- Employee ID of employee who initiated checkout
- List of goods, prices, and quantities
- Subtotal
- Sales tax
- Total
- Payment method

2.3. User Classes and Characteristics

Admin

Admins will have all privileges within the system. They will have access to:

- User Management for creating and managing Employees/Admins
- Goods Database Management for managing/adding goods
- Customer Checkout for checkout out customers

Employee

Employees will have access to:

- Customer Checkout for checking out customers

2.4. Operating Environment

The application will be designed to run on a Windows computer

2.5.Design and Implementation Constraints

Each function will need to take less than a second and security will require a user login and secure database.

2.6.User Documentation

A User Manual will accompany the first iteration deliverable and will detail how to accomplish each function.

3.External Interface Requirements

3.1.User Interfaces

The program will use a standard GUI for navigation. The GUI will feature tabs for selecting functionality and popup windows where necessary.

3.2.Hardware Interfaces

The application will not interface directly with the barcode scanner nor the scale, but users will be able to easily enter this information manually. The application will be able to print receipts directly on a printer.

3.3.Software Interfaces

This application will use PyQt5 for the User Interface, ReportLab for receipt generation and printing. The database will be implemented using SQLite.

4.System Features

4.1.System Features

Product Features

Manage Employees

Create a New Employee

- 1) Select **Create a New Employee** button
- 2) New Employee ID will automatically be created
- 3) Enter Employee name (e.g. John Smith)
- 4) Enter Employee password (e.g. 1234)
- 5) Select whether Employee is an Admin or Cashier
- 6) Select **Update Employee** button

Update an Employee

- 1) Select Employee from list
- 2) Edit Employee name, password, or role
- 3) Select **Update Employee** button

Delete an Employee

- 1) Select Employee from list
- 2) Select **Delete Employee** button

Manage Products

Add a New Product

- 1) Select **Add a New Product** button
- 2) Enter the following information:
 - 1) Product Name
 - 2) Barcode
 - 3) Available Units
 - 4) Price
 - 5) Customer Price
 - 6) Provider ID
- 3) Select **Update Product** button

Import a Spreadsheet

- 1) Select the **Import Spreadsheet** button
- 2) Browse to the desired file and select **Open**
- 3) Select the **Update Product** button

Barcode Search

Update a Product

- 1) Enter Product barcode into Barcode Search field
- 2) Select **Search** button
- 3) Update any necessary Product Information fields
- 4) Select **Update Product** button

Delete a Product

- 1) Enter Product barcode into Barcode Search field
- 2) Select **Search** button
- 3) Select **Delete Product** button

Begin Checkout Barcode Search

Add Item

- 1) Enter Product barcode into Barcode Search field
- 2) Select **Search** button
- 3) Enter quantity OR enter weight
- 4) Select **Add Item** button

Remove Item

- 1) Select an item in the Receipt List
- 2) Select the **Remove Item** button

Get Payment

- 1) When item scanning is complete select the **Get Payment** button

Payment Dialog

Print Receipt

- 1) Select Cash/Credit/Check for transaction payment
- 2) Select **Print Receipt** button

5.Other Nonfunctional Requirements

5.1.Performance Requirements

Each function will need to be run in less than a second.

5.2.Safety Requirements

The program must calculate costs correctly and security will prevent misuse by users and store patrons.

5.3.Security Requirements

User login will prevent unauthorized users from mismanaging the user or product databases. This will also prevent store patrons from accessing the system.