Point of Sales Transaction System for Domino’s Pizza

Software Design Description

Version 1.0

October 7, 2014

**Number One Best Professional Development Practices**

**Authors: Richard Tyaba, Abu Kebbie-Anthony, Chris McEligot, Nicolas Mouriski**

Table of Contents

1 Introduction 3

1.1 Purpose 3

1.2 Scope 3

1.3 Definitions, Acronyms and Abbreviations 3

1.4 References 3

1.5 System Overview 3

2 Data Design 4

2.1 Internal Data Structures 4

2.2 Data Flow Diagrams 4

2.3 Data Dictionary 5

3 Architecture Design 5

3.1 Program Structure 6

4 Interface Design 7

4.1 Internal System Interfaces 7

4.2 External System Interfaces 7

4.3 User Interfaces 7

4.3.1 Login Screen 8

4.3.2 POS Screen 8

4.3.3 Pizza Creation Dialog 9

4.4 Add Soda Dialog 9

4.4.1 Price & Tax Rate Configuration Screen 10

4.4.2 Employee Account Administration Screen 10

4.4.3 Add/Edit Employee Dialog 11

5 Procedural Design 12

6 Miscellaneous 12

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author(s)** |
| 10/07/2014 | 1.0 | Initial version of the Software Design Description | Kebbie-Anthony, McEligot, Mouriski, Tyaba |

# Introduction

## Purpose

The Software Design Document (SDD) is a comprehensive software design model consisting of four distinct but interrelated activities: data design, architectural design, interface design, and procedural design. The document is used as a tool to communicate preliminary design concepts to customers, users, and among the software engineering team members.

The Data Design describes structures that reside within the software. Attributes and relationships between data objects dictate the choice of data structures.

The Architecture Design uses information flow characteristics, and maps them into the program structure. Transformation mapping method is applied to exhibit distinct boundaries between incoming and outgoing data. The Data Flow diagrams allocate control input, processing, and output along three separate modules.

The Interface Design describes internal and external program interfaces as well as the design of human interface. Internal and external interface designs are based on the information obtained from the analysis model.

The Procedural Design describes structured programming concepts using graphical, tabular, and textual notations. These design mediums enable the designer to represent procedural detail that facilitates translation to code. This blueprint for implementation forms the basis for all subsequent software engineering work.

The SDD contains the following sections:

* Data Design
* Architecture Design
* Interface Design
* Procedural Design

## Scope

The SDD contains the design description and models of four distinct but related aspects: data design, architecture design, interface design, and procedural design. All of the requirements that dictate the design of each of these aspects are outlined in the SRS. The SDD must contain all of the necessary information about the design of the software and must not have any hole of leave any unanswered questions.

## Definitions, Acronyms and Abbreviations

* #1BPDP – #1 Best Professional Development Practices
* GUI – Graphic user interface
* POS – Point of sales
* SDD – Software Design Description
* SRS – Software Requirements Specification

## References

Below is the list of all the documents referenced elsewhere in the SDD:

* SRS; Version 1.0; September 23, 2014; #1BPDP

## System Overview

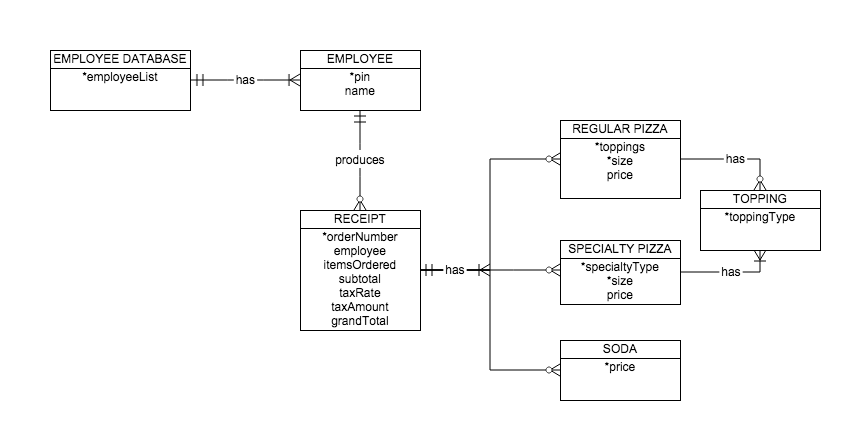
The objective of this project is to develop a Point of Sales (POS) Transaction System for Domino’s Pizza. The system will have the ability to log on with different user accounts, contain a GUI for placing orders consisting of common items for sale by Domino’s Pizza, and display an onscreen receipt for a completed order.

# Data Design

This section describes how the date of the program is going to be stored. Specific data and external file structures are indicated, as well as the relationships between structures. Actions that can be performed on these structures are also described.

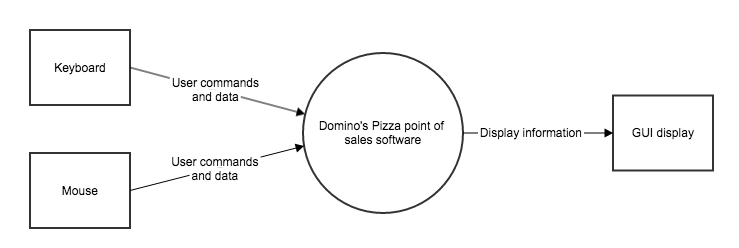
## Internal Data Structures

Entity Relationship Diagram

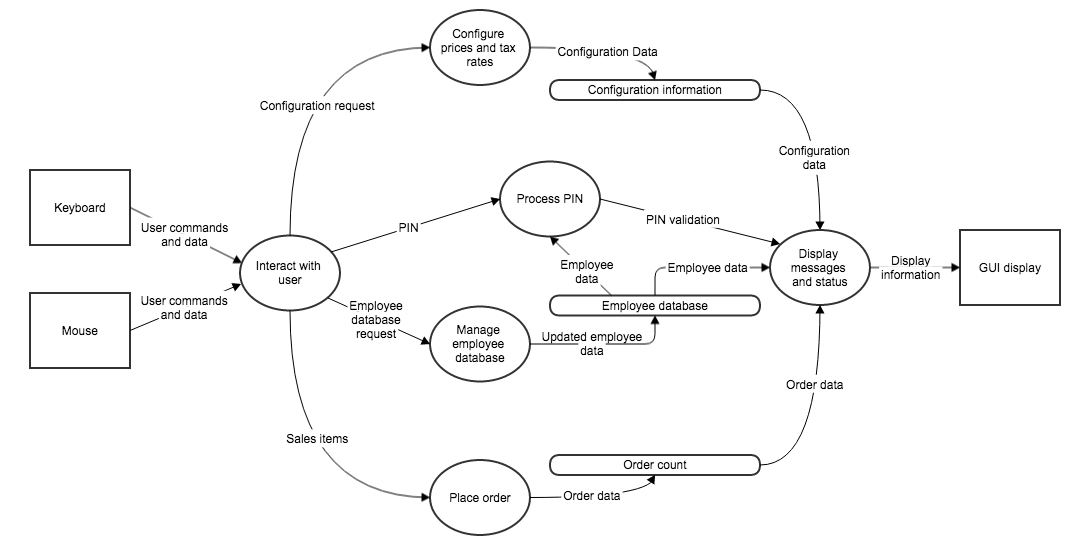


## Data Flow Diagrams

Level 0 Data Flow Diagram



Level 1 Data Flow Diagram



## Data Dictionary

* The employee list will be stored as an array list of employees.
* The PINs will be stored as four digit integers.
* The employee names will be stored as strings.
* The subtotal will be stored as a double with only two digits after the decimal point.
* The grand total will be stored as a double with only two digits after the decimal point.
* The tax rate will stored as a double.
* The tax amount will be stored as a double with only two digits after the decimal point.
* The order number will be stored as an integer.
* The toppings for a pizza will be stored as an array of enumerated topping types.
* The sizes of the regular and specialty pizzas will be stored in an enumeration.
* The prices of each item be stored in an array of doubles with only two digits after the decimal point.
* The items ordered will be stored as an array list of items for sale.

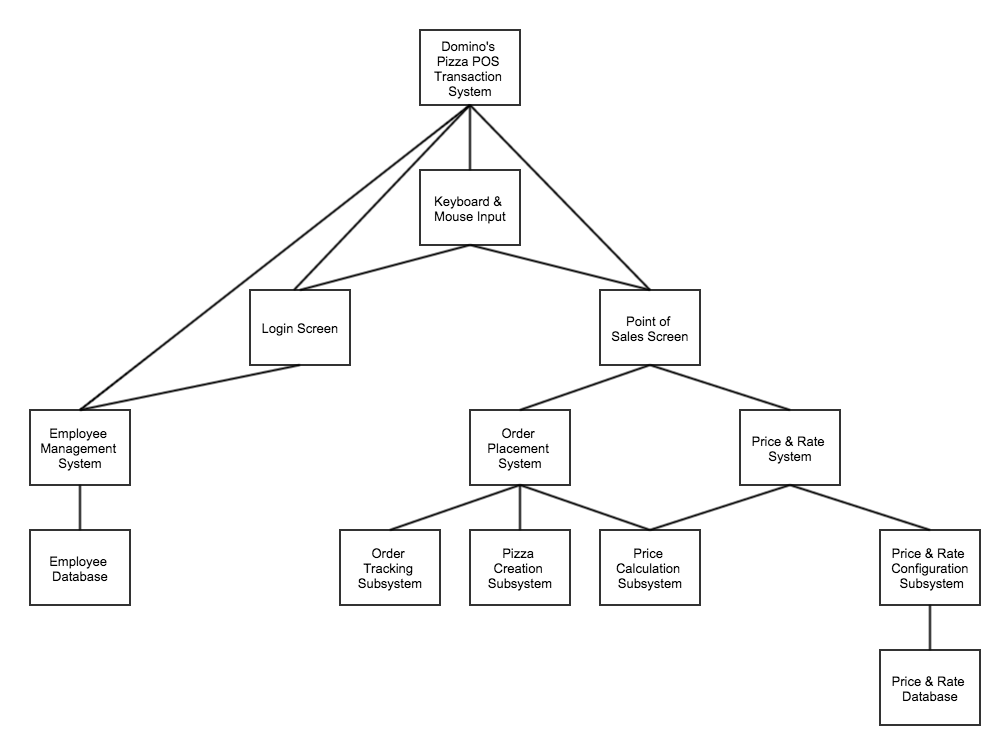
# Architecture Design

This section describes the design of the entire software system, including the hierarchy of the modules present in the system.

## 

## Program Structure

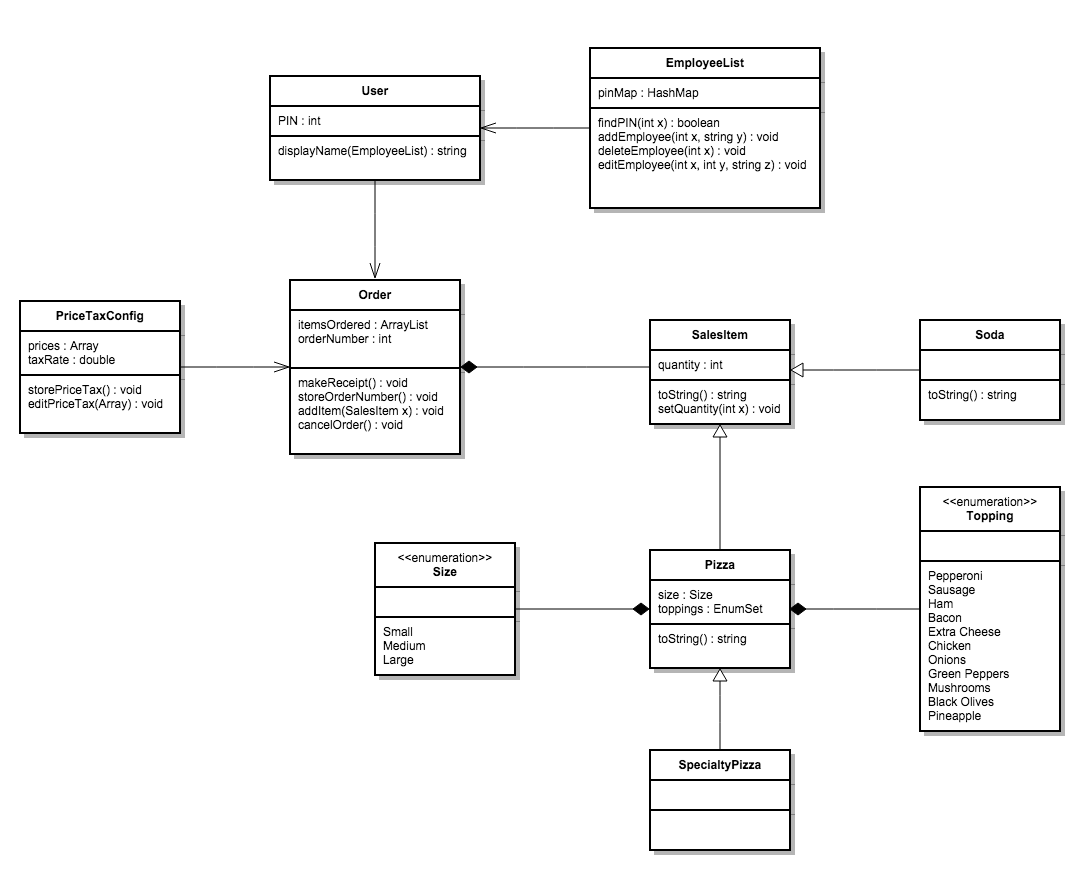
Program Architecture Diagram



# Interface Design

This section describes internal and external program interfaces in addition to the design of human interface.

## Internal System Interfaces



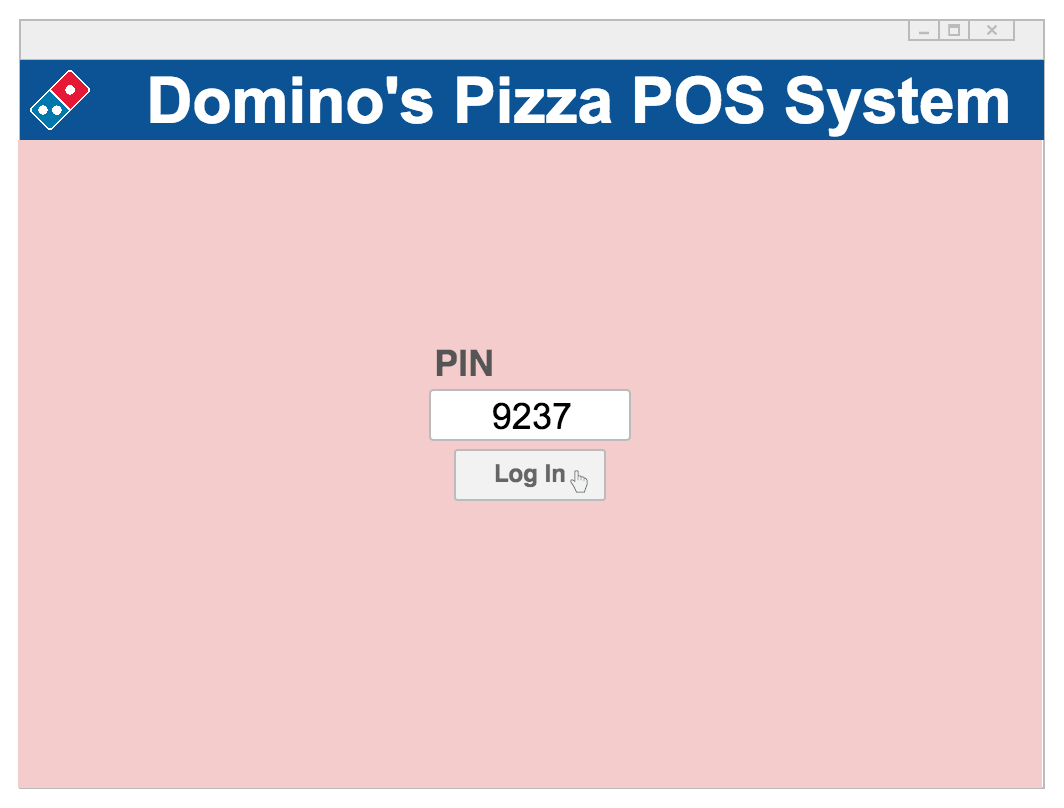
## External System Interfaces

The external system interfaces for this project are:

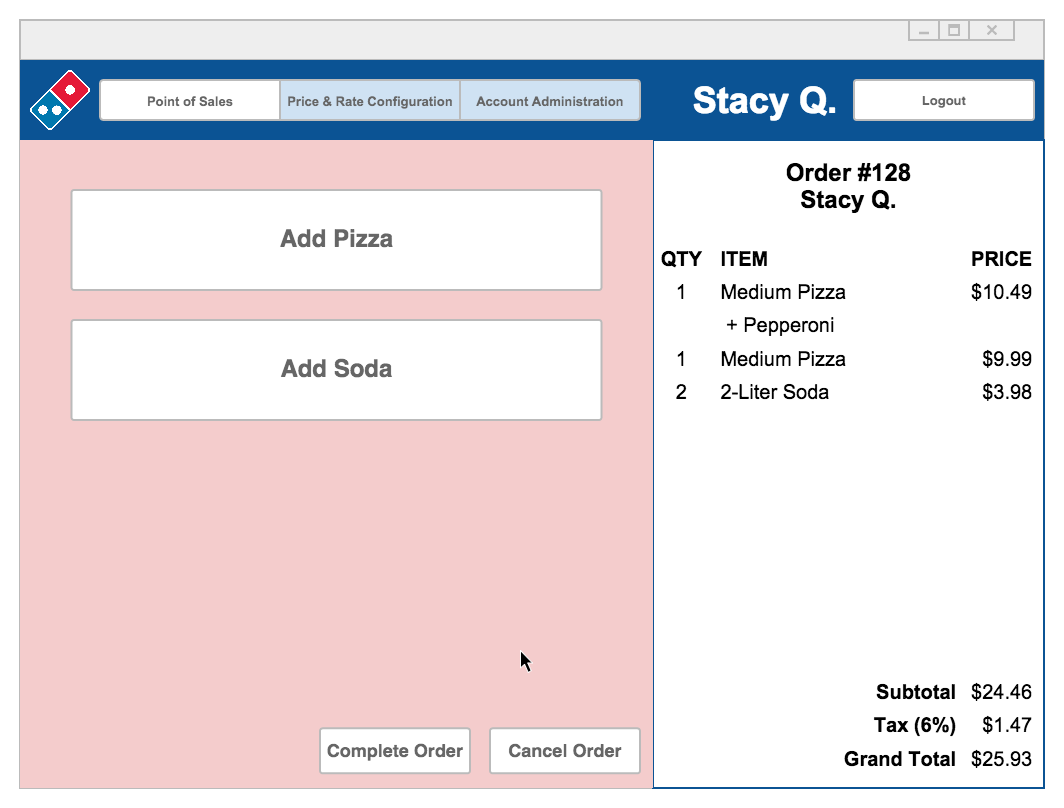
* The computer’s Operating System
* Display
* Keyboard
* Mouse

## User Interfaces

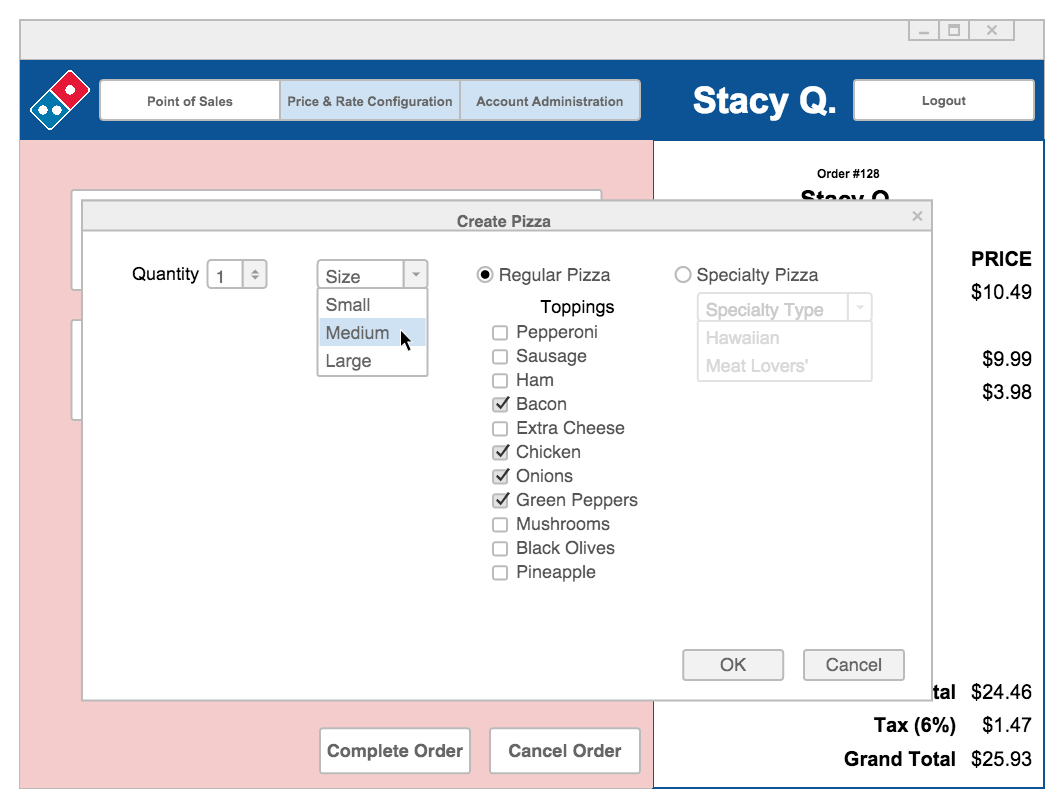
### Login Screen



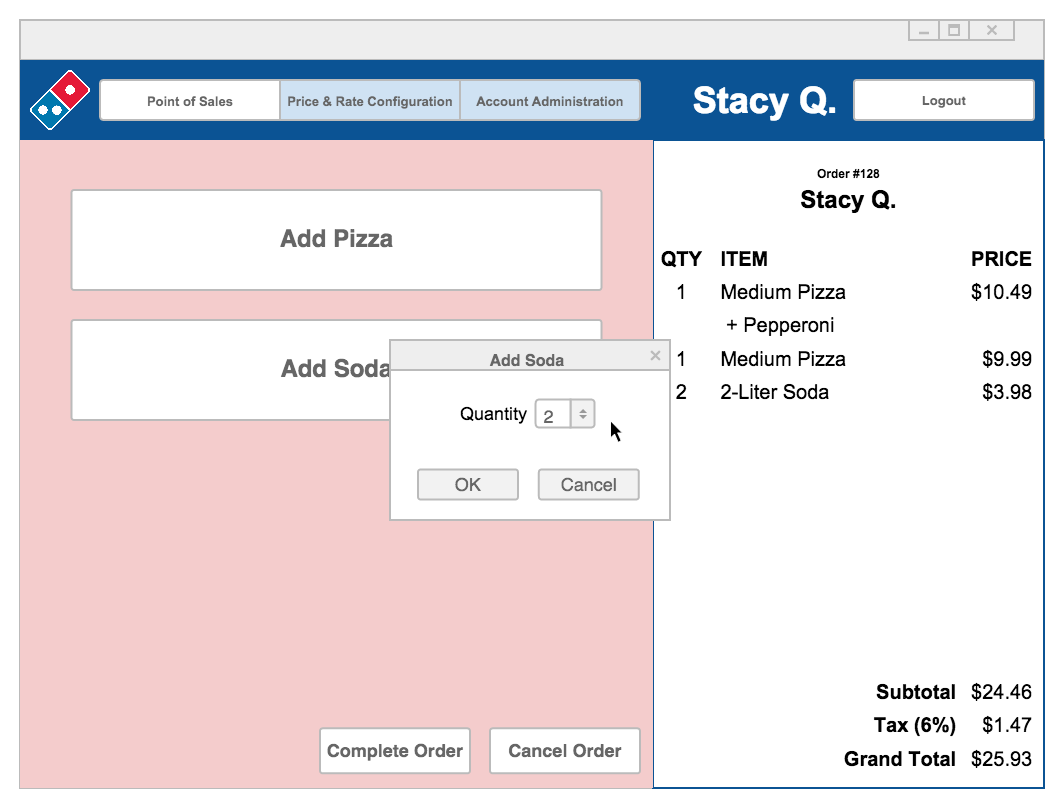
### POS Screen



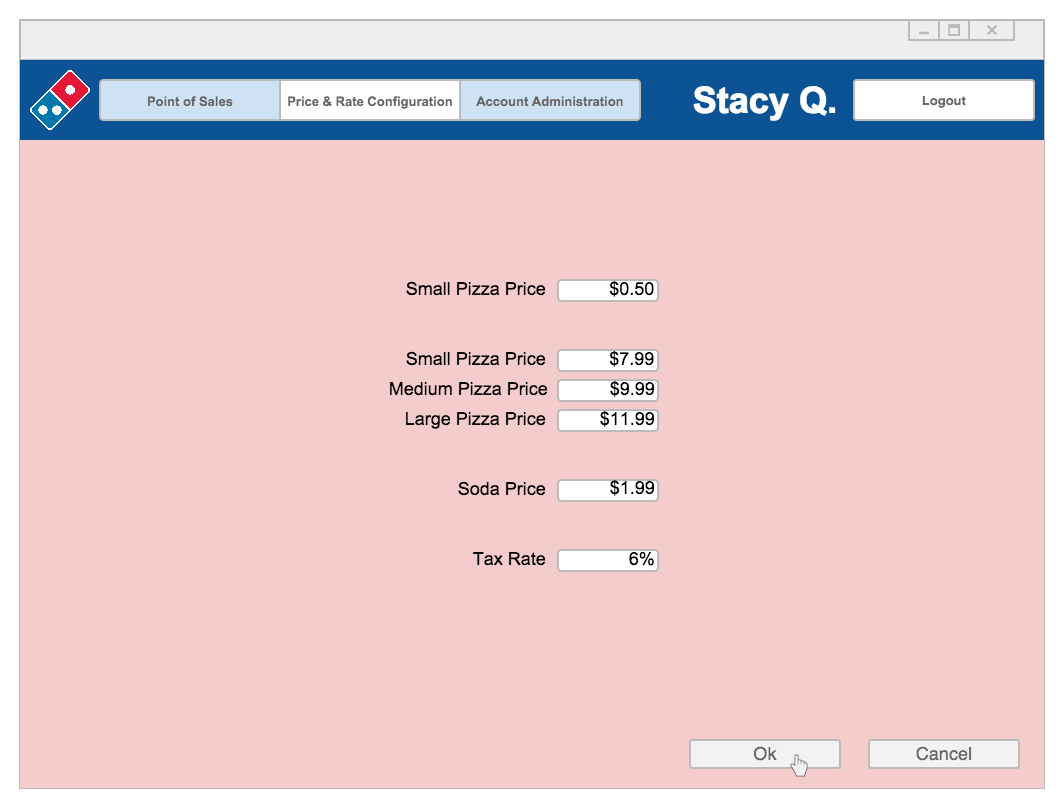
### Pizza Creation Dialog



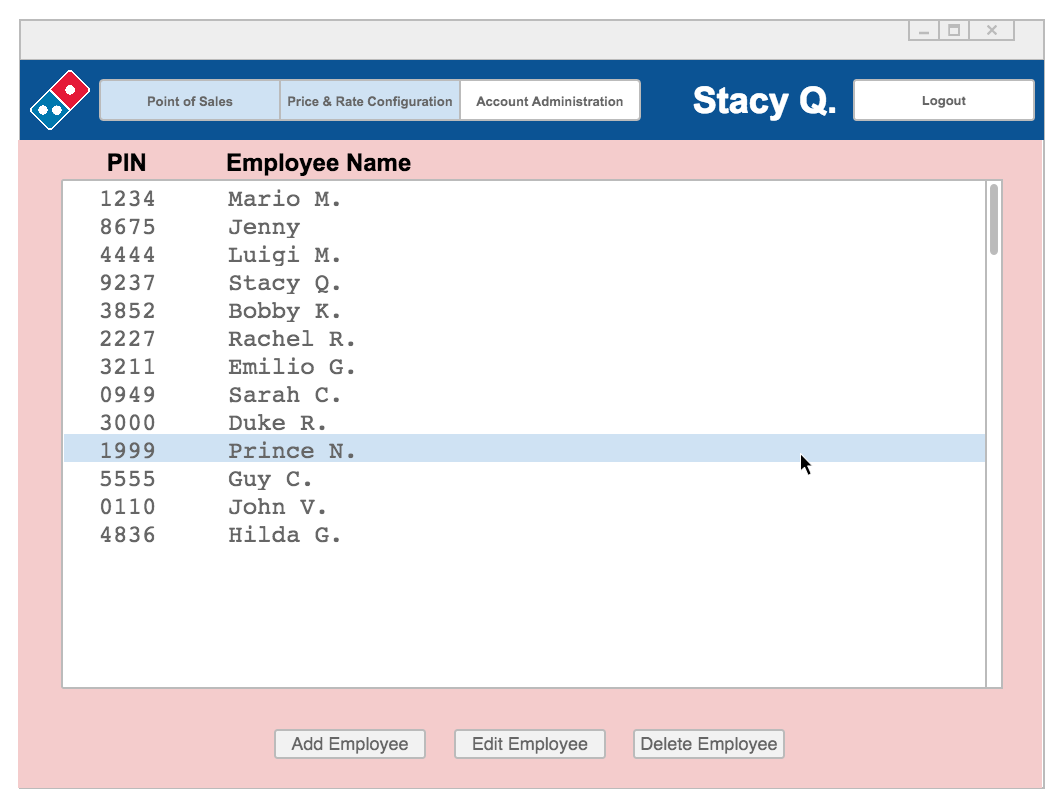
## Add Soda Dialog



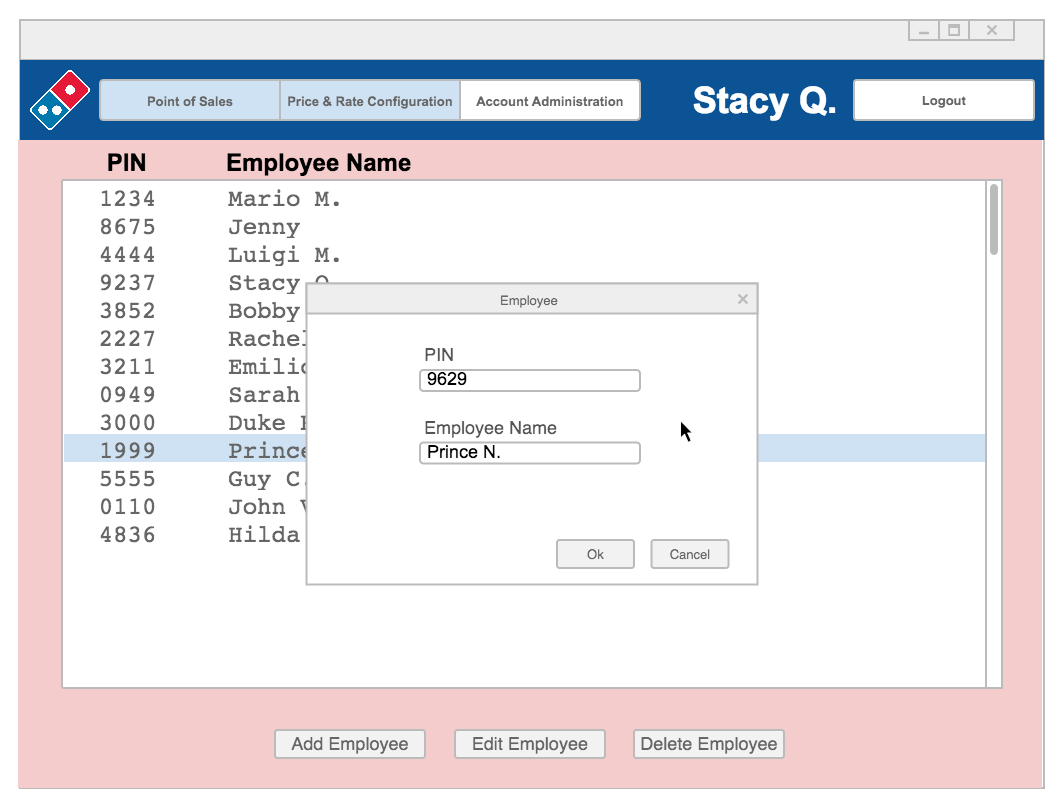
### Price & Tax Rate Configuration Screen



### Employee Account Administration Screen

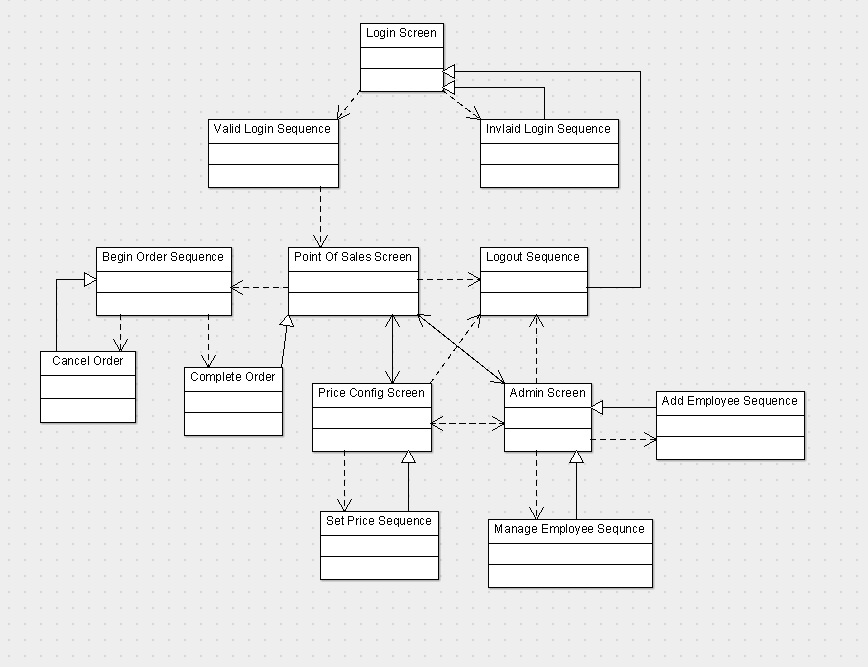


### Add/Edit Employee Dialog



# Procedural Design

This section describes structured programming concepts using graphical, tabular, and textual notations that enable the designers to represent procedural detail facilitating translation to code. This blueprint for implementation will form the basis for all subsequent software engineering work.



# Miscellaneous

There are no miscellaneous designs for this project.