**WEB PROGRAMMING TEAM PROJECT : MyShoppingMall**

Dept : Electronic and Electrical Engineering

Student ID : 20164542

Name : Seonjae Kim

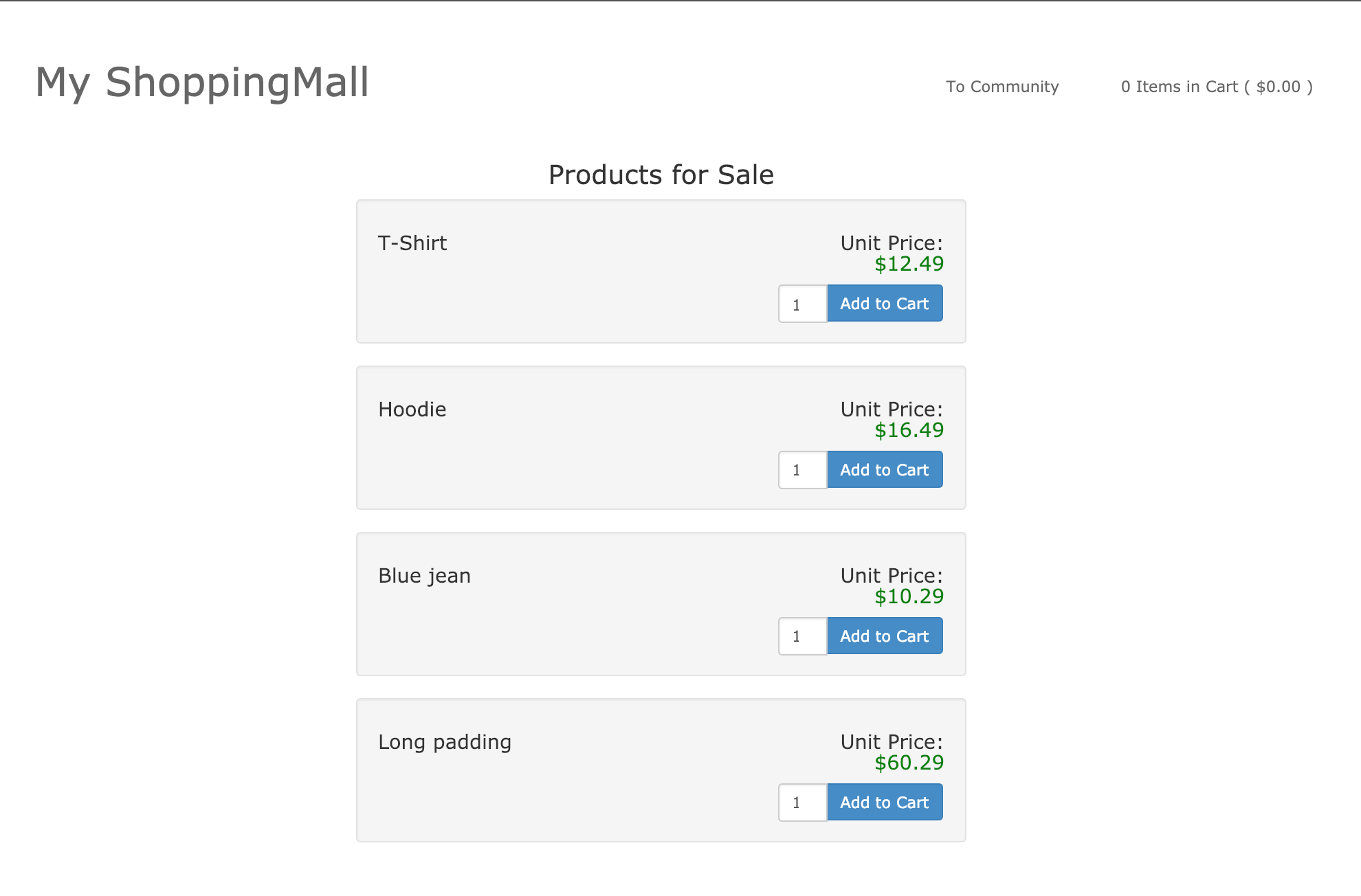
**1. User Guide**

# Brief description of the product

This site is a simple shopping mall solution.

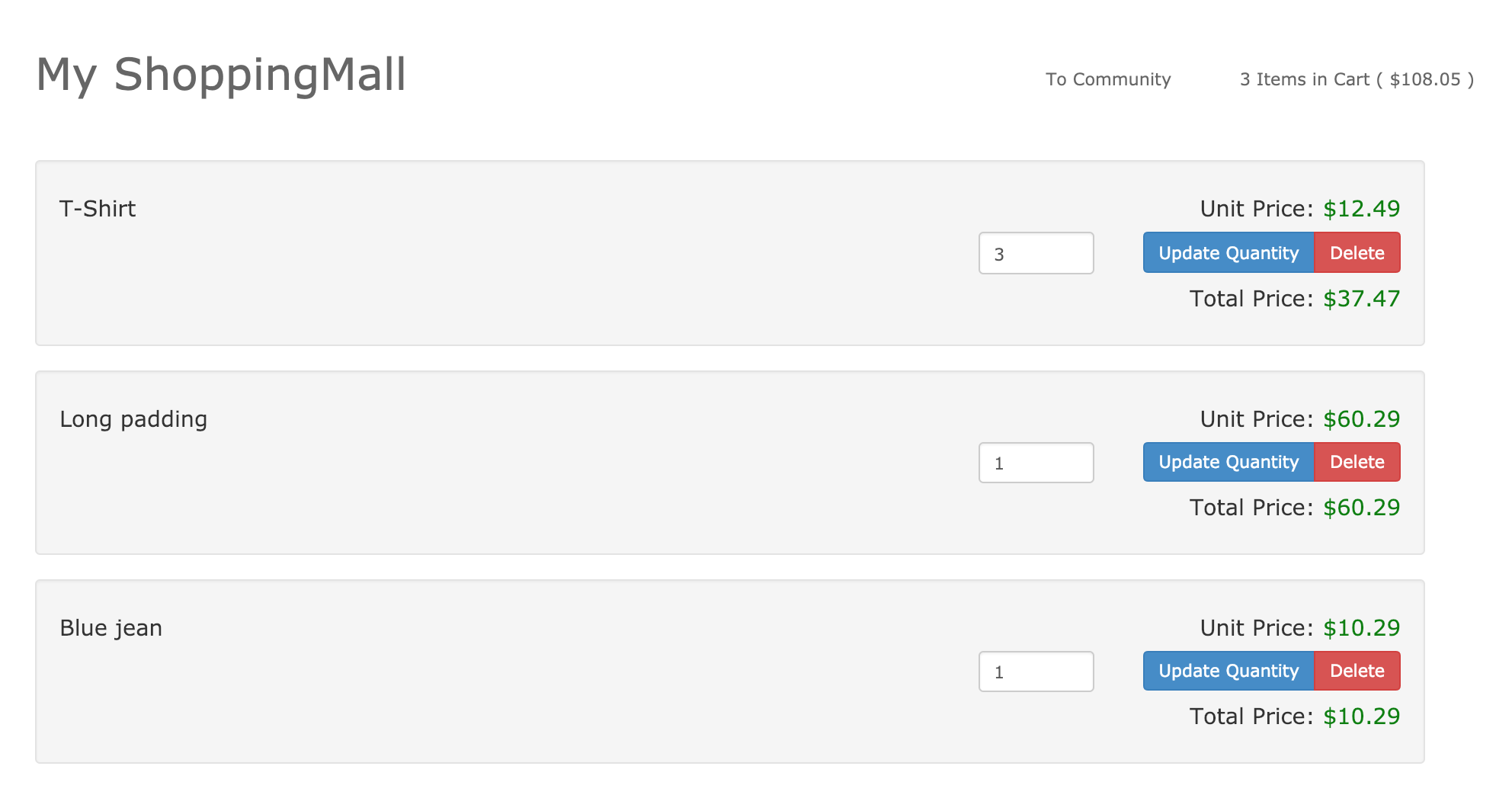
# How to use the software

The main page is below.



< Fig 1. Main Page >

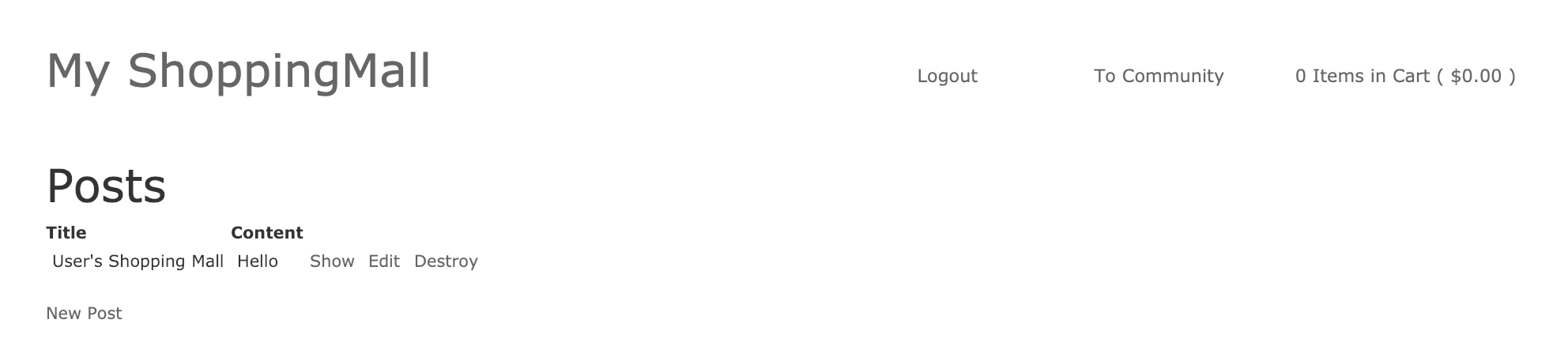
Users can look around the product list, like a T-Shirt, a Hoodie, a Blue jean and a long padding. Anyway they can click the 'Add to Cart' to push into their cart list, with any quantity. When they click the button, the cart list page is below.



< Fig 2. Cart Page >

Users can look their selected products. They can update the quantity for their products. And anytime they can click the delete button. They can see the total price of products. Users notice the "3 Items in Cart ( $108.05 )" at the top-right text.

There is a place where the users can be chatty. When clickking the 'To Community'



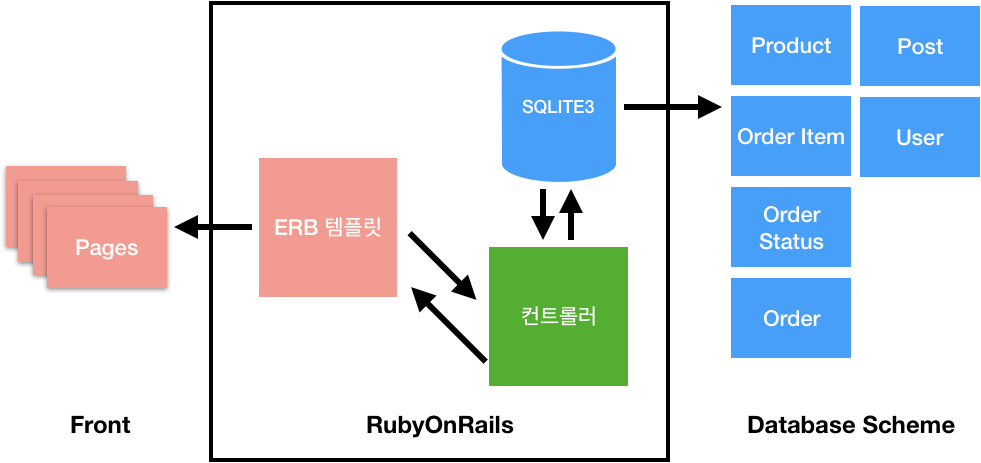
< Fig 3. Community Page >

Users can post anything articles about shopping.

**2. Developer Guide**

# System Architecture

The system architecture is below.



In RubyOnRails, There are three parts, Models, Views and Controllers. At the time, the model can be SQLITE3. The model consist of a product model and post model. The product model is about the shopping cart. On the other hand, the post model is about the community page.

The controllers are bridged between views and models. also The views consist of product pages and post pages. Product views need to get the product models. And Community views need to get the post models.

# Source code structure

The main source code folders are below

* app/controllers/\* : Controllers are in charge of business logics.
* app/views/\* : Views are in chare of showing what clients see.
* app/models/\* : Models are defined relationships with oher models.
* config/routes.rb : Define the routing addresses.
* db/migrate/\* : Define the database schemes.
* db/seeds.rb : Set the initial necessary model data.

# How to build

In Windows,

1. Install Ruby 2.4.0
2. Install Debkit into the ruby folder
3. In command line, type 'gem install rails'
4. type 'bundle install' in this project file, to install dependency components
5. Type 'rails s' in this project file, to run the web server

In Mac OS X,

1. Install HomeBrew
2. type 'brew install rbenv', the ruby version manager package
3. type 'rbenv install 2.4.0'
4. type 'gem install rails'
5. type 'sudo bundle install' in this project file, to install dependency components
6. type 'rails s' in this project file, to run the web server