**Project Overview**

This catalog application provides a list of catalog categories. Each category has a list of category items with description. The application provides a user registration and authentication system. Registered users will have the ability to post, edit and delete their own category items. Users log into the application using their google sign in credentials.
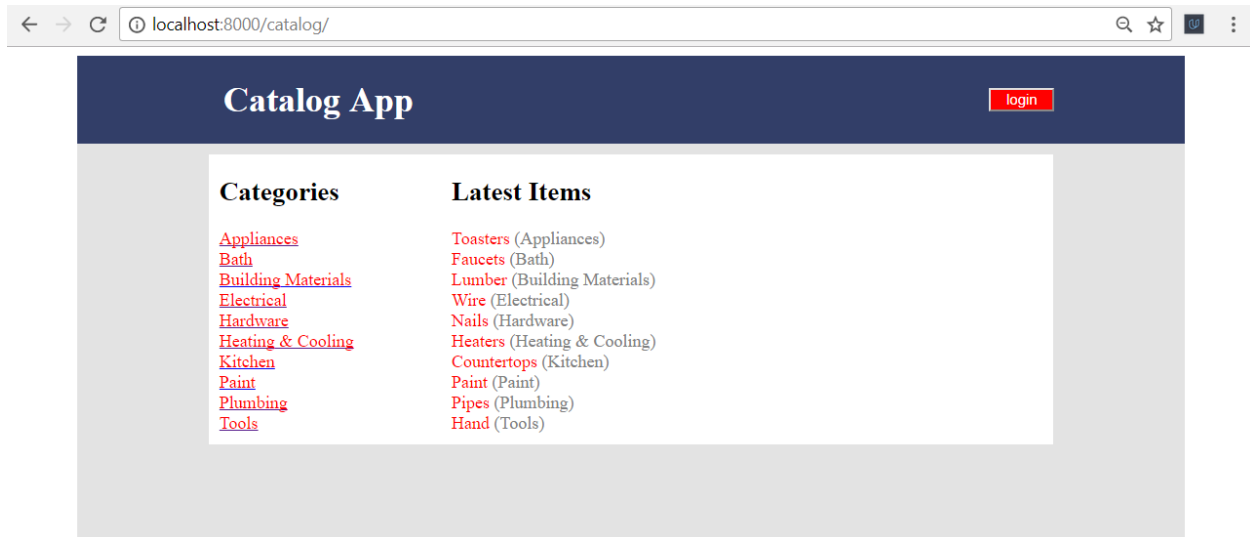
**Getting Started:**

Copy the db_setup.py, db_populator.py, and application.py files to the /vagrant/catalog directory (create the catalog directory under vagrant if necessary. Open a terminal.

```
$ cd <your_path>/vagrant
~ $ vagrant up
~ $ vagrant ssh
~ $ cd/vagrant/catalog
~ $ python db_setup.py (creates the db, and db tables)
~ $ python db_populator (populates the db)
~ $ python application.py (runs the application)
```

Access the application by visiting http://localhost:8000 locally on your browser.

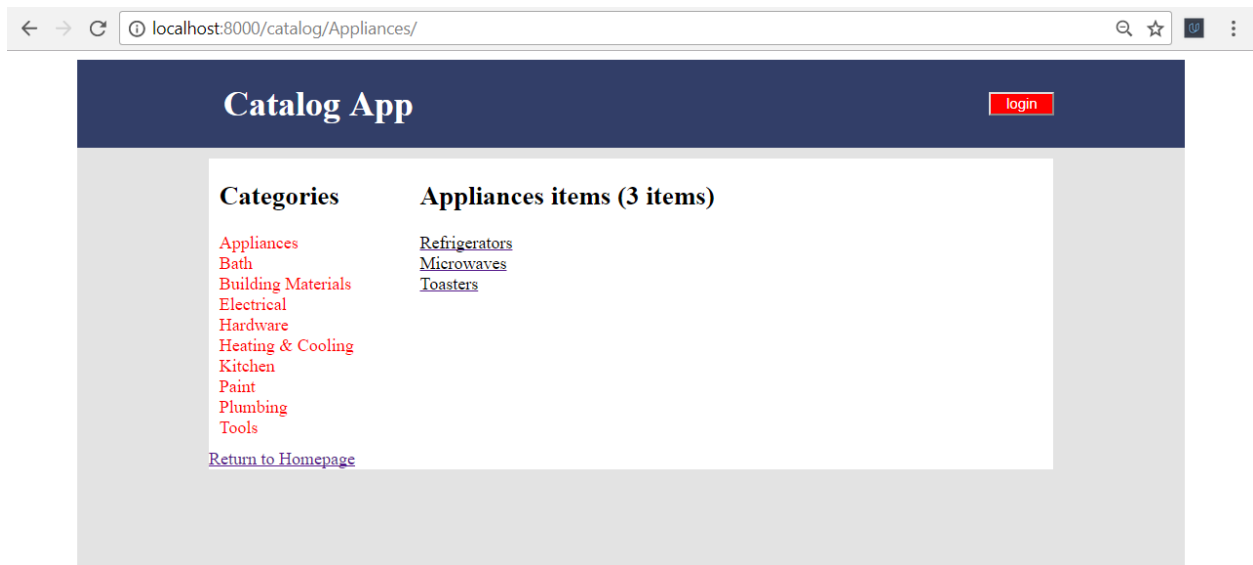**Catalog Display Examples**

The homepage displays all current categories with the latest added items.
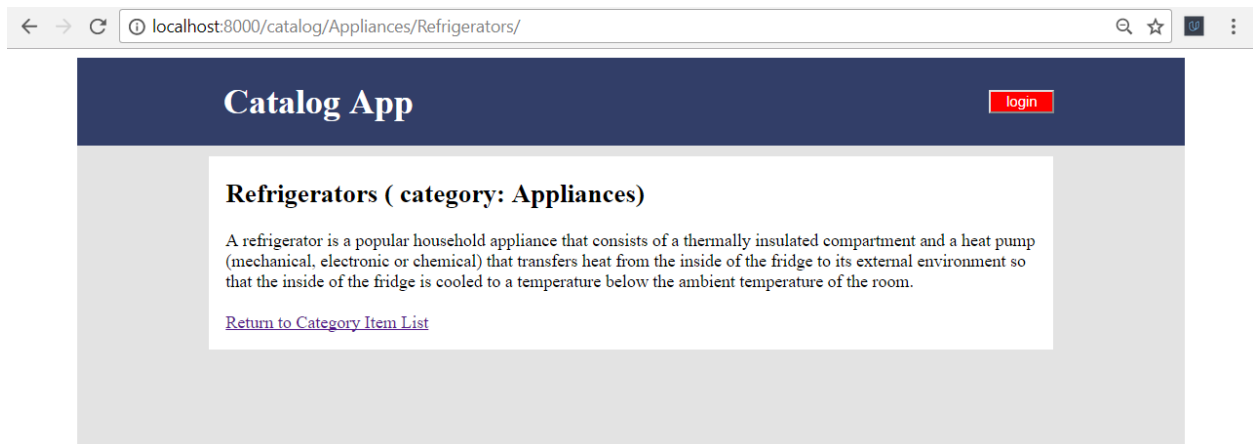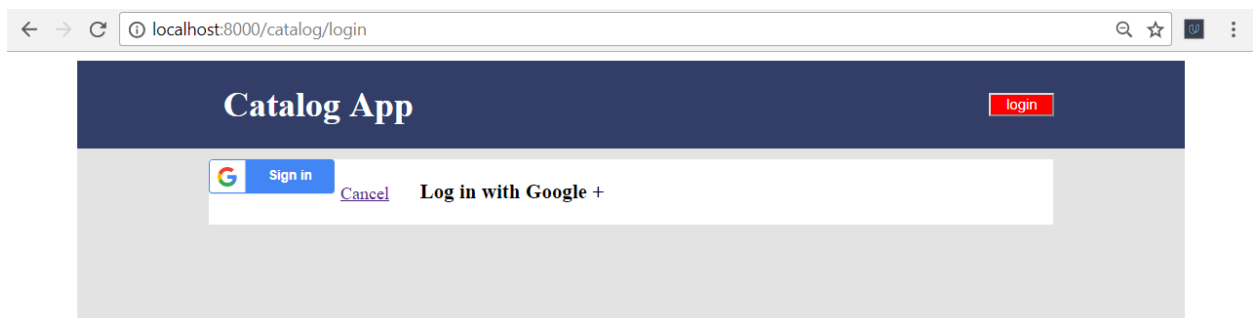
Selecting a specific category lists all items in that category.



Selecting a specific category item displays a description of that item.



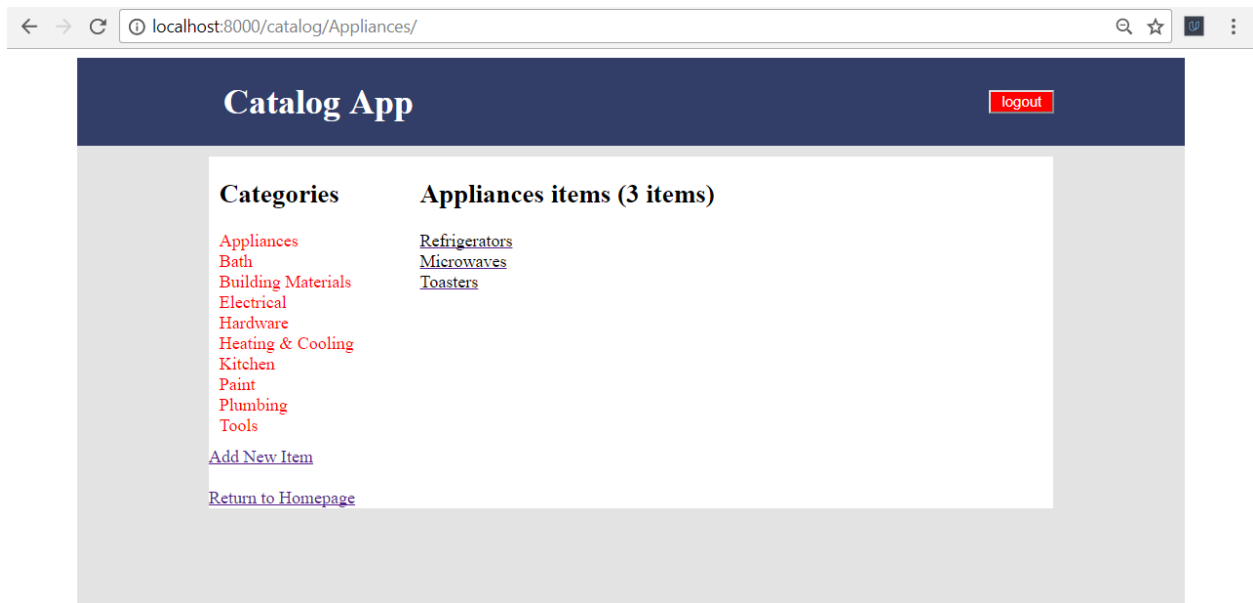Clicking the login button directs the user to the google sign-in page.  This application uses google sign-in to authenticate users.  After signing in the user is returned to the Homepage.
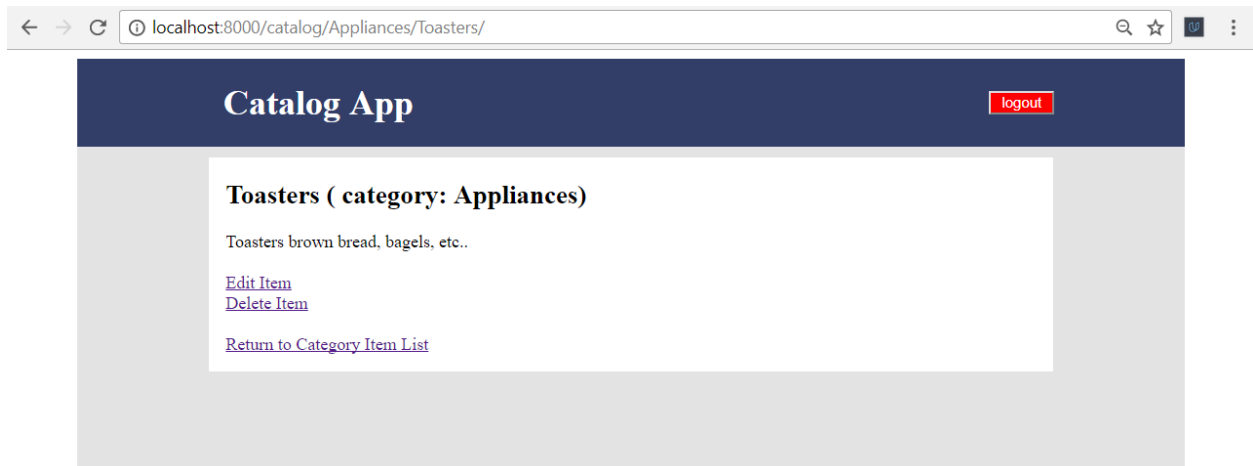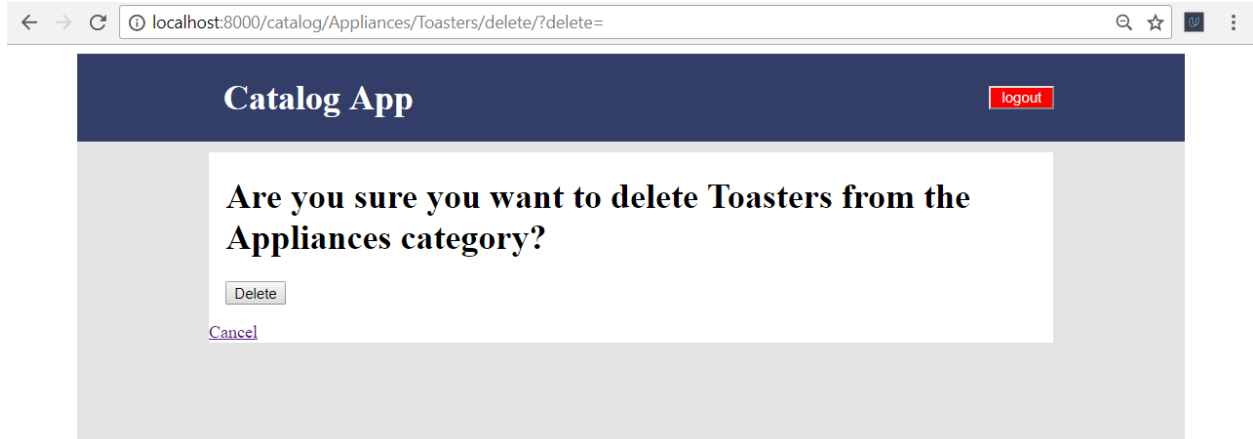
After logging in, a user has the right to add items to a category by clicking on the 'Add New Item' link.



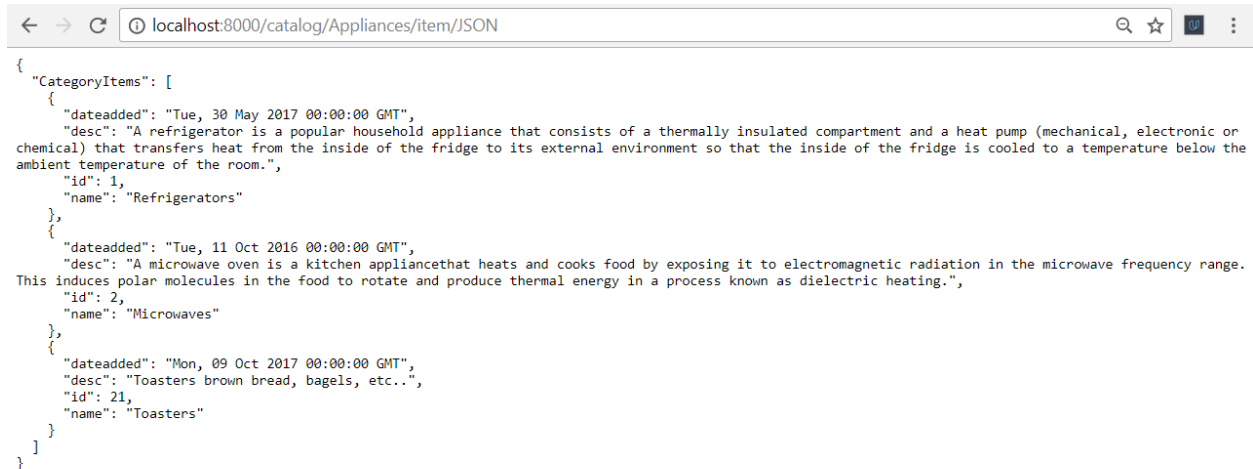After clicking a category item, a logged in user has the right to only edit and delete items that they have created.

← → C | ⓘ localhost:8000/catalog/Appliances/Toasters/delete/?delete= | 🔍 ☆ 🔲 ⋮

## Catalog App
logout

**Are you sure you want to delete Toasters from the Appliances category?**

Delete

Cancel

## JSON endpoint example

← → C | ⓘ localhost:8000/catalog/Appliances/item/JSON | 🔍 ☆ 🔲 ⋮

```
{
  "CategoryItems": [
    {
      "dateadded": "Tue, 30 May 2017 00:00:00 GMT",
      "desc": "A refrigerator is a popular household appliance that consists of a thermally insulated compartment and a heat pump (mechanical, electronic or chemical) that transfers heat from the inside of the fridge to its external environment so that the inside of the fridge is cooled to a temperature below the ambient temperature of the room.",
      "id": 1,
      "name": "Refrigerators"
    },
    {
      "dateadded": "Tue, 11 Oct 2016 00:00:00 GMT",
      "desc": "A microwave oven is a kitchen appliancethat heats and cooks food by exposing it to electromagnetic radiation in the microwave frequency range. This induces polar molecules in the food to rotate and produce thermal energy in a process known as dielectric heating.",
      "id": 2,
      "name": "Microwaves"
    },
    {
      "dateadded": "Mon, 09 Oct 2017 00:00:00 GMT",
      "desc": "Toasters brown bread, bagels, etc..",
      "id": 21,
      "name": "Toasters"
    }
  ]
}
```

## File Structure

| <your_path>/vagrant/catalog | <your_path>/vagrant/catalog/static |
|---|---|
| * application.py<br>* db_setup.py<br>* db_populator.py<br>* term_setup.py | * style.css |
| <your_path>/vagrant/catalog/templates<br>  * layout.html<br>  * category.html<br>  * items.html<br>  * newcatitem.html<br>  * edit_item.html<br>  * delete_item.html<br>  * descriptions.html<br>  * public_descriptions.html<br>  * login.html<br>  * logout.html | |

**Python Terminal Access to catalog db**

Open a python terminal.

$ cd <your_path>/vagrant

~ $ vagrant up

~ $ vagrant ssh

vagrant@vagrant:~$ cd /vagrant/catalog

vagrant@vagrant:/vagrant/catalog$ python

```
>>> from sqlalchemy import create_engine
from sqlalchemy import Column, ForeignKey, Integer, String
from sqlalchemy.ext.declarative import declarative_base
from sqlalchemy.orm import sessionmaker
from db_setup import Base, Category, Item, User
engine = create_engine('sqlite:///catalog.db')
Base = declarative_base()
Base.metadata.create_all(engine)
DBSession = sessionmaker(bind=engine)
session = DBSession()>>> from sqlalchemy.orm import relationship
>>> from sqlalchemy import Column, ForeignKey, Integer, String
>>> from sqlalchemy.ext.declarative import declarative_base
>>> from sqlalchemy.orm import sessionmaker
>>> from db_setup import Base, Category, Item, User
>>> engine = create_engine('sqlite:///catalog.db')
>>> Base = declarative_base()
>>> Base.metadata.create_all(engine)
>>> DBSession = sessionmaker(bind=engine)
>>> session = DBSession()
```