

# Django DBMS Project Report

Software Engineering Team Project



Team members:  
Chao Chen, Saheed Adepoju, Yihang Zhao

Fall 2016

# Content

1. Project Commitment	3
2. User Stories	4
2.1 Use case card	4
2.3 Architecture Diagram	10
2.4 Activity Diagram	11
2.5 Class and Components Diagram	15
2.6 Database Diagram	16
3. Working Code	17
3.1 Code Sample	17
3.2 Software Engineering Design Module Matrix	18
3.3 User Interface Screens	20
4. Project Repository and Tools	23
4.1 Repository	23
4.2 SCRUM Tool	24
5. Weekly SCRUM Sheets	25

# 1. Project Commitment

The target of Django DBMS project is to develop a web-based platform for small store owner or manager to manage resources and for the staffs to conduct their regular works.

This project is built on top of Django-based web service, therefore it only need to be deployed on specified server instead of to be deployed on different working environment. All the user need are just the internet and web browser.

Below figure shows the project commitment form:

**Prairie View A&M University  
Department of Computer Science**

**COMP 5423 Software Engineering Project Commitment Form  
FALL 2016**

This form should be filled by the students and signed by ALL the students. The completed form should be returned to Dr. A. Lodger by September 21, 2016.

**Student Names:** Chao Chen Saheed Adepoju Tihang Zhao

**Project Title:** Django DBMS

**Title and Brief Description of the Project:**  
This project targets on delivering a django-based website for small store owner, manager and staff to manage resources and conducting regular works.

**Commitment to the Project by Student:**

Student Name	Student Signature	Date	Email	Phone
<u>Chao Chen</u>	<u>Chao Chen</u>	9/18/2016	cohen-rough@gmail.com	8329104612
<u>Zhao Tihang</u>	<u>Zhao Tihang</u>	9/20/2016	tony92320@gmail.com	8324741068
<u>Saheed Adepoju</u>	<u>Saheed Adepoju</u>	9/20/2016	sadebukola@gmail.com	7134729263

By signing above, we certify that we will do original work for the project and will do the best to complete the above project on time. Our individual failure to work on the project may result in a grade that may prevent us individually and our team members from getting a satisfactory grade for the class. We have indicated our team contact information (Email and phone) and we individually will use this as our primary communication channel, and respond immediately to the communication.

The Team leader for this team is : Chao Chen

## 2. User Stories

### 2.1 Use case card

Scenario 1: Account Creation.

Given that user doesn't exist

And can not use the System.

When the user tries to use the Application

The Application allows user to create  
Account.(username and password)

The Application allows the user to log  
onto the system.

Scenario 2: Logging out.

Given that a user has logged  
onto the System.

When a user clicks log-out

The Application logs out the user.

### Scenario 3: Account login.

Given that there is no current user logged in.

i7.8

Given that there already exist accounts in the system.

Given that the Username and Password are correctly entered into the Application.

When they are entered

The Application logs them into the system.

### Scenario 4: Add New Stock to the Application.

Given that a user is logged in.

Select the workflow menu.

Select the expenditure sub-menu.

Select the Stock sub-menu.

Enter the Name, the Price, count and Payment.

Click Submit

The Application will outline that the process was successful.

### Scenario 5: Add a new membership record.

Given that an admin user is already logged in.

Select the workflow menu.

Select the Membership sub-menu.

Select the New-membership sub-menu.

Enter the Name, Phone number, gender, Birthdate, balance amount, PinCode, Address.

Click Submit to submit the details.

The application should respond with that the record has been entered.

### Scenario 6: Purchasing an item.

Given that an admin user has logged into the system.

Select the workflow menu.

Select the Purchasing sub-menu.

Select the Purchasing sub-menu.

Select the Stock from the drop-down-menu

Select the Quantity needed.

Enter the Quantity needed.

The application calculates the total amount due.

Enter the membership ID of the customer.

Select the Submit button to complete the transaction.

Scenario 7: @@@@ AND @@@@ Query a customer  
Given that an admin user is logged onto the system.

Select the Query menu.  
Select the Query and Edit sub-menu.  
Select the Customer item from the class drop down menu.  
Select the attribute you would like to use to search for a customer.  
Click on Search.  
The application displays all users who meet the criteria entered.

Scenario 8: @@@@ AND @@@@ Edit a customer

Given that an admin user is logged onto the system.

Given that the customer details exist with the system.

Select the Query menu.  
Select the Customer item from the class drop-down menu.  
Select the attribute you would like to use to search for a customer.  
Search for a customer.  
Click on Search.  
The application displays all users who meet the criteria entered.

Scenario 9

Select Workflow Menu.

Select Activity Function.

Create Discount For specific product.

Select Valid date for discount.

Select Target For membership or birthday.

Submit button to complete.

Scenario 10

Select workflow menu

Select Activity Function.

Select Create promotion.

Select Valid date for discount.

Select product for promotion.

Give value for Buy N get one.

Submit button for complete.

## Scenario 11

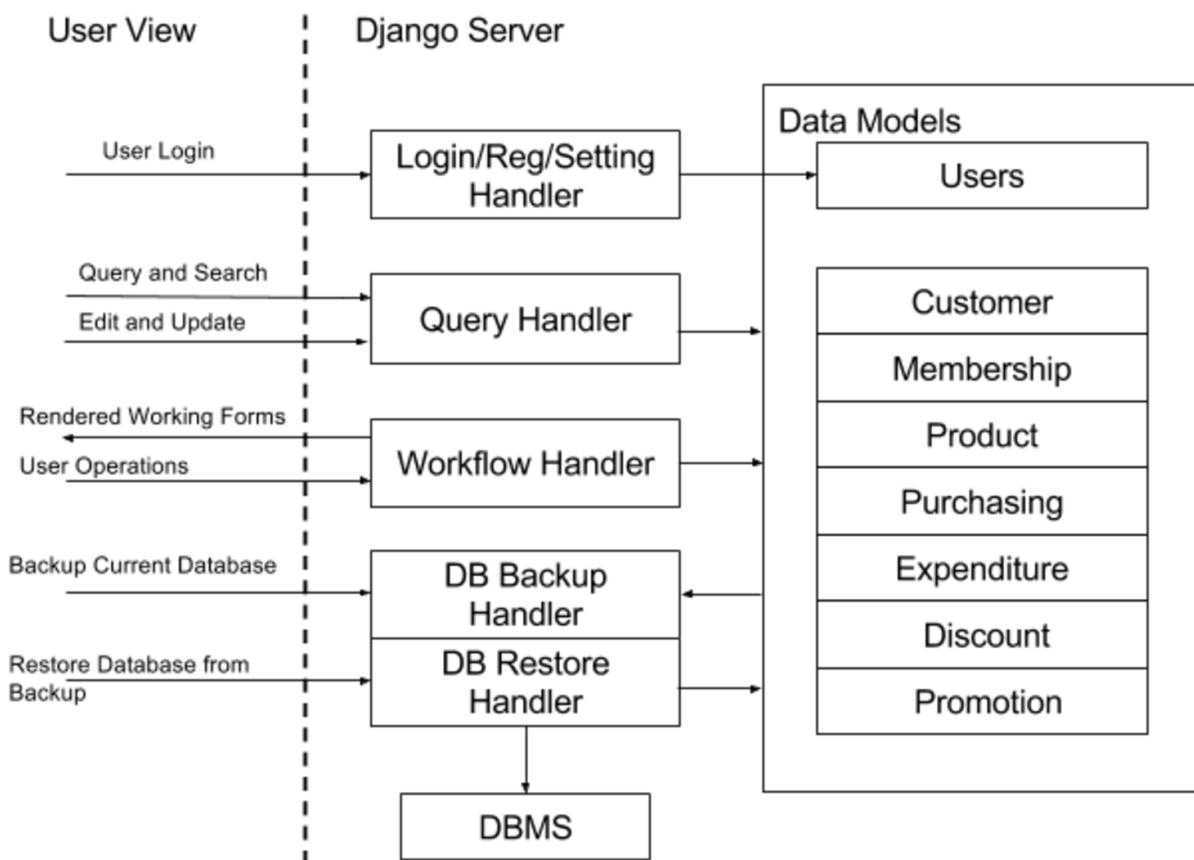
select Query Menu  
select Backup and Restore Menu.  
select backup.  
backup ~~introduction~~ table of database  
cover previous date.  
Exit function.

## Scenario 12

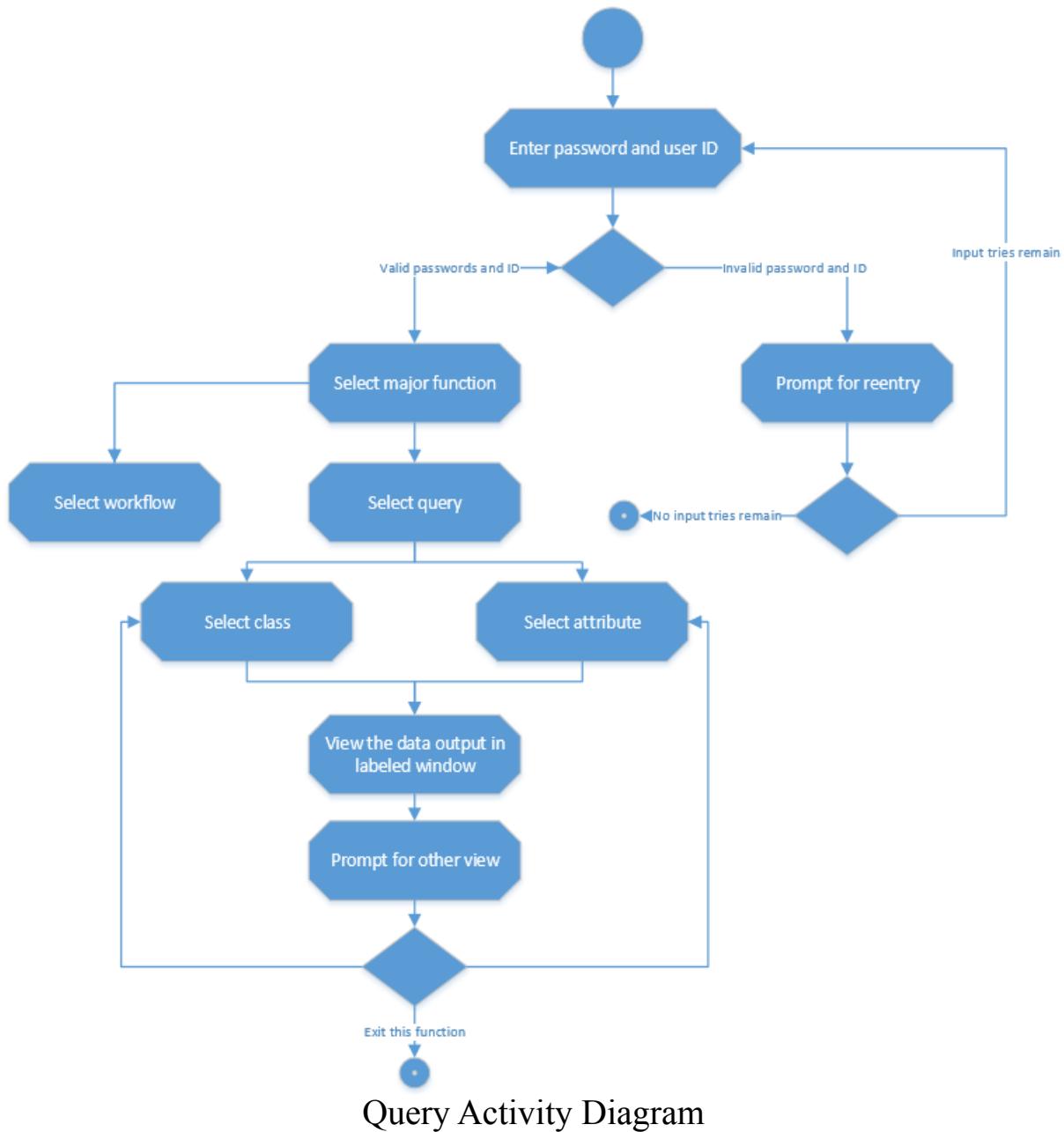
select Query Menu  
select Backup and Restore Menu  
select restore  
do query and specify the date.  
submit query get the table.  
Exit function.

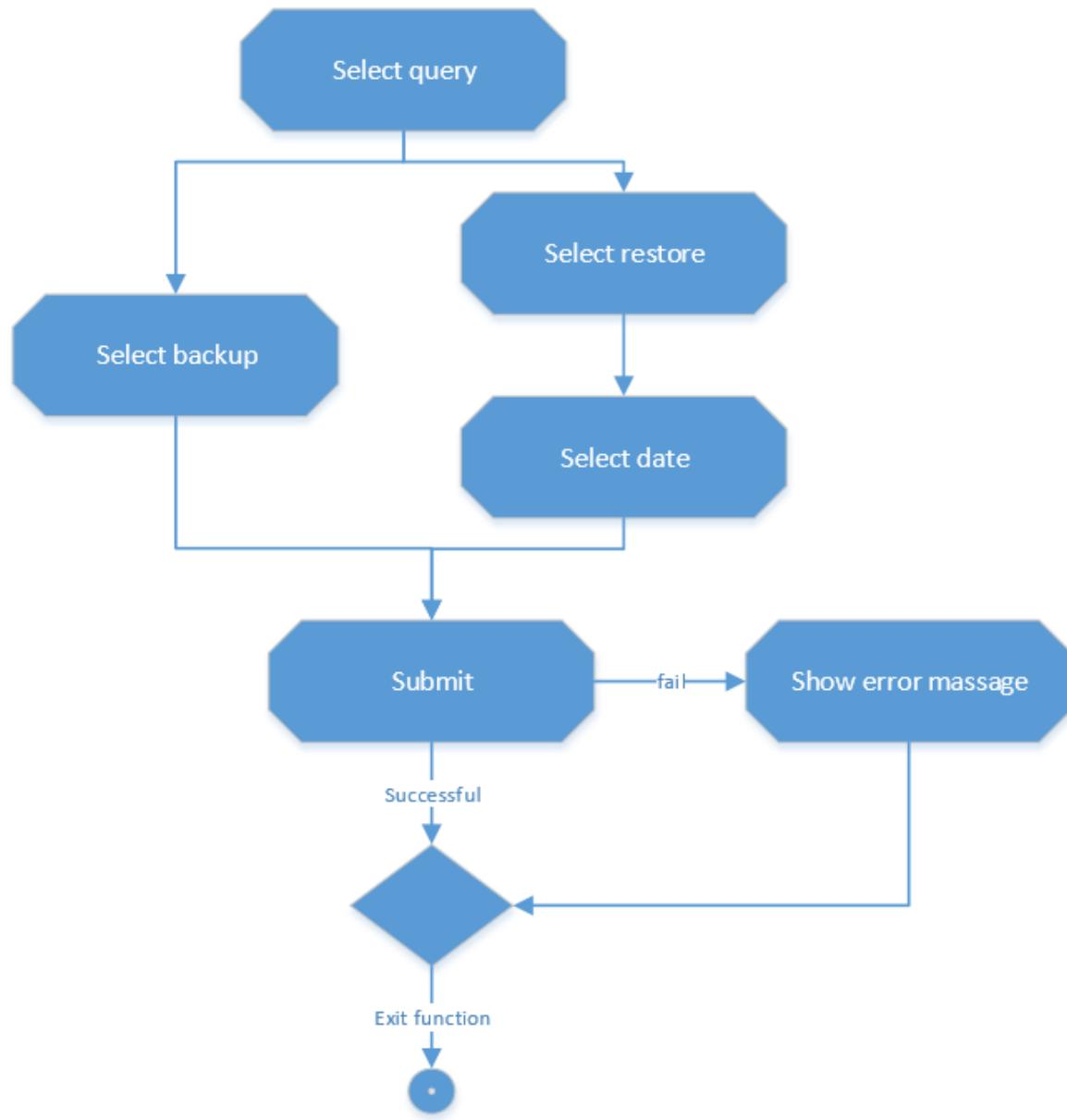
## 2.3 Architecture Diagram

Below figure shows the overview of the web application's framework, it also demonstrate how the web view handler interact with the user view an data models. In this sprint, we add the function of system database backup and restore. The related components of the completed system are shown as below.

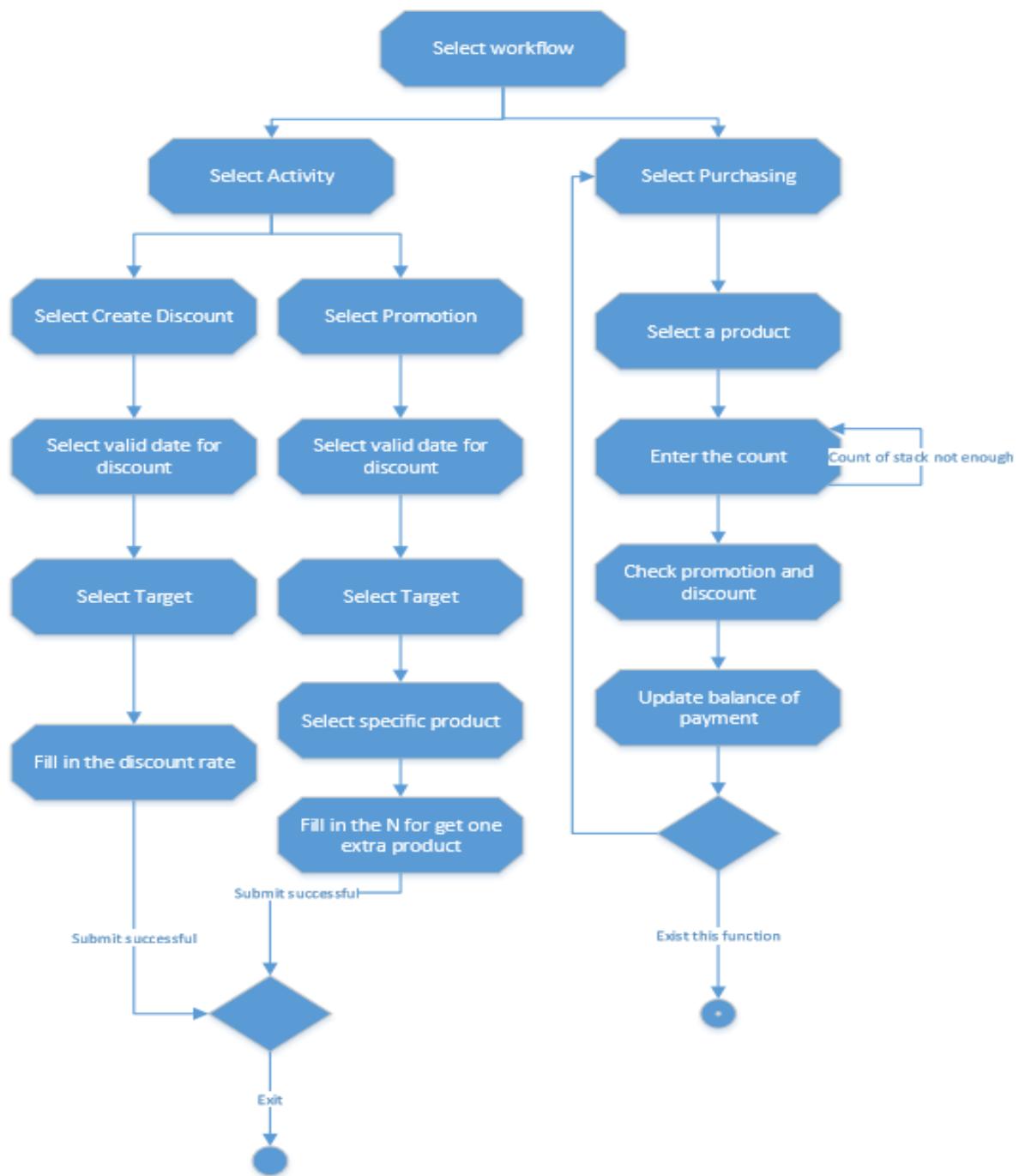


## 2.4 Activity Diagram

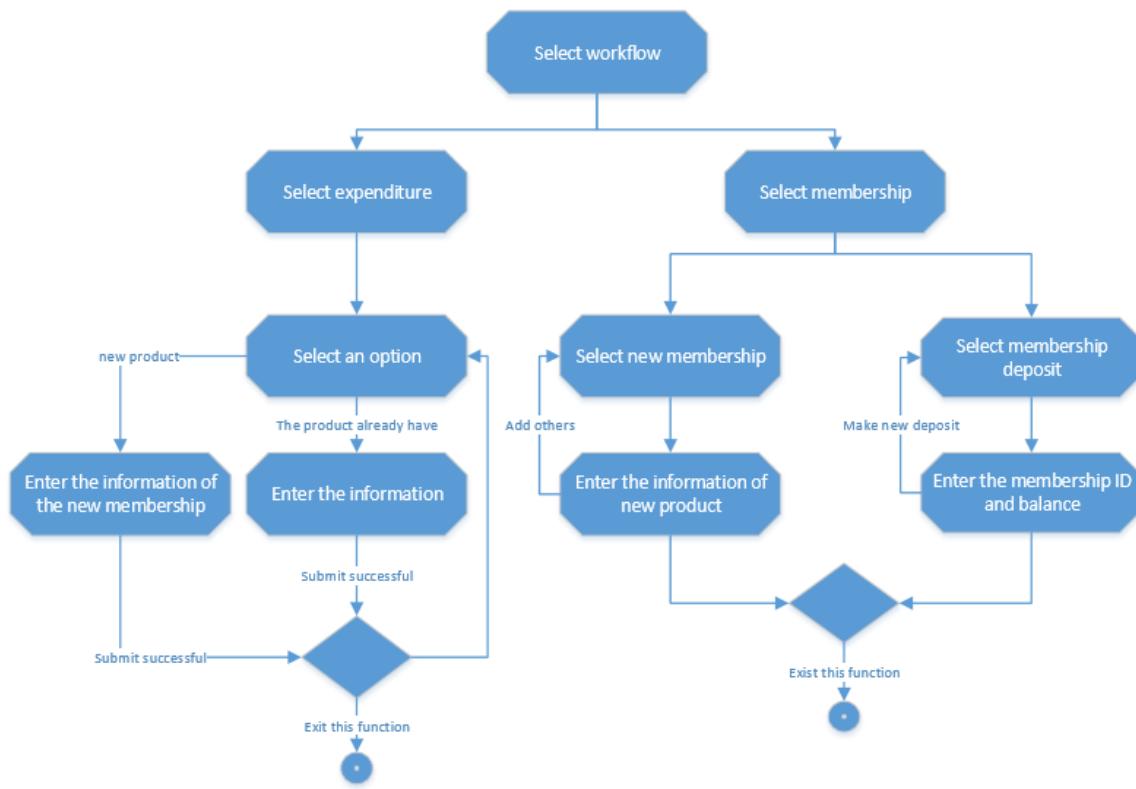




Back and Restore Activity Diagram

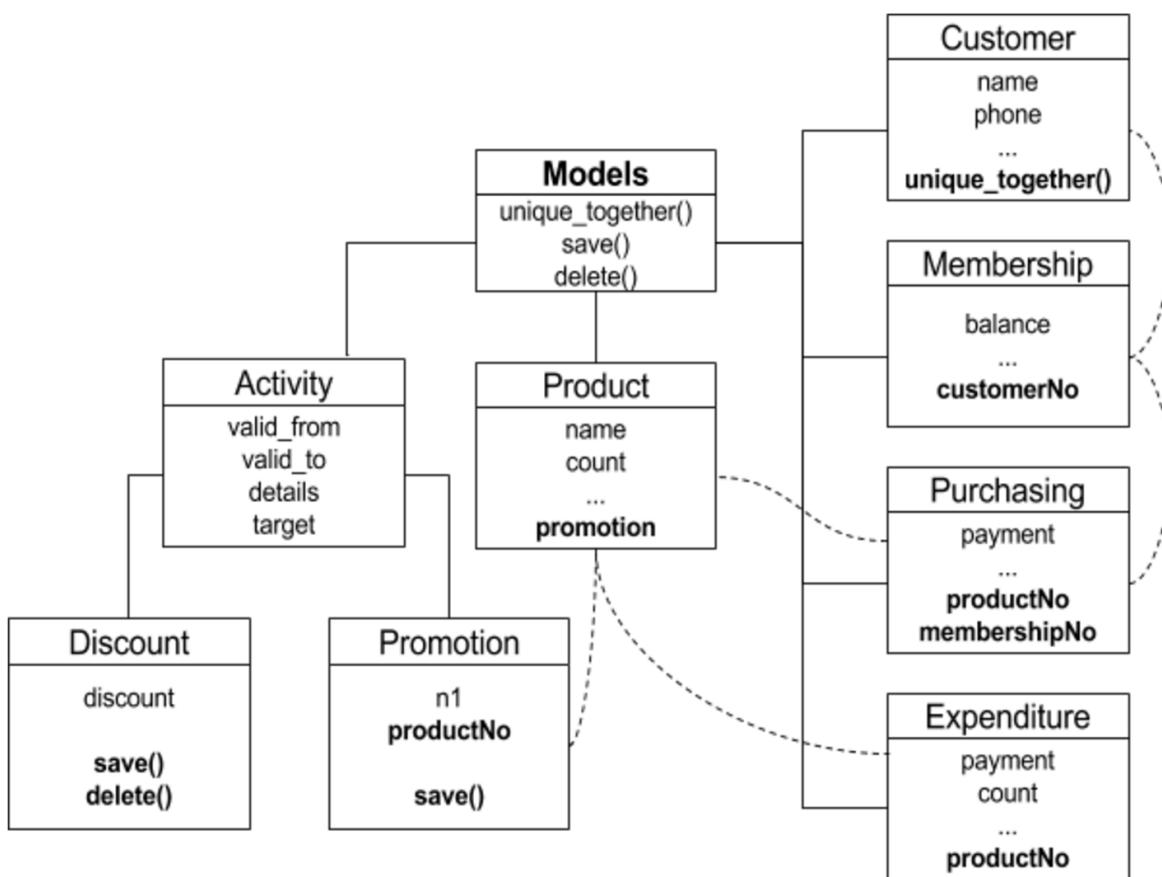


Discount and Promotion Activity Diagram

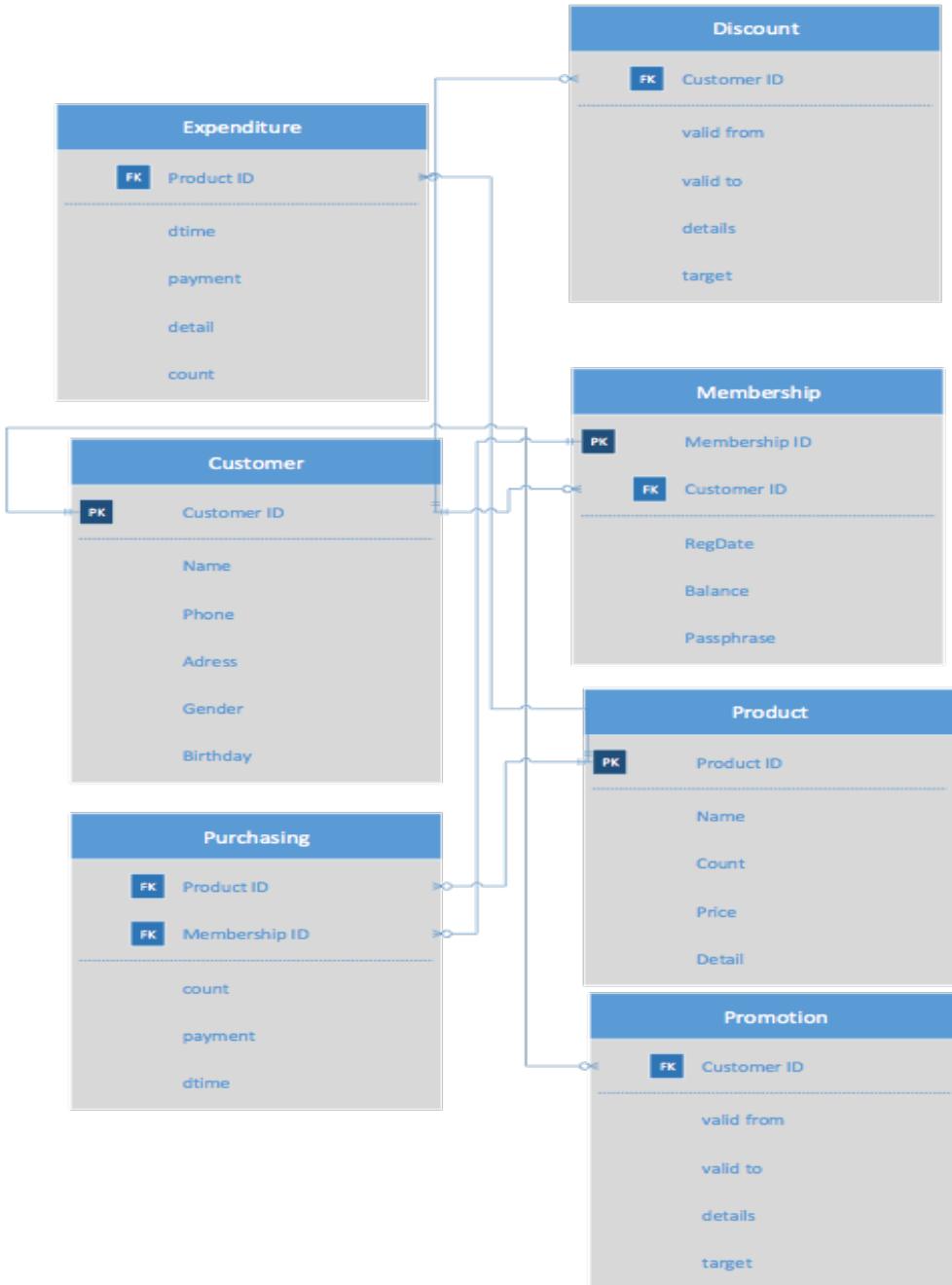


Workflow Activity Diagram

## 2.5 Class and Components Diagram



## 2.6 Database Diagram



### 3. Working Code

#### 3.1 Code Sample

Below code section demonstrate the unit test code of data models. The unit test code is implemented by using standard Python library unittest, which defines a class-based testing approach.

```
from django.test import TestCase
from query.models import *

class QueryTestCase(TestCase):
    def setUp(self):
        today = datetime.datetime.now().date()

        c1 = Customer(name="tony", phone="1234567890", address="address_tony", gender="Male", birthday=today)
        c2 = Customer(name="jamail", phone="2234567890", address="address_jamail", gender="Male", birthday=today)
        c3 = Customer(name="hanbing", phone="3234567890", address="address_hanbing", gender="Female", birthday=today)
        c1.save()
        c2.save()
        c3.save()

        m1 = Membership(balance=100, passphrase='pwd_tony', customerNo=c1)
        m2 = Membership(balance=100, passphrase='pwd_hanbing', customerNo=c3)
        m1.save()
        m2.save()

    def test_membership_check(self):
        c1 = Customer.objects.get(name="tony")
        c2 = Customer.objects.get(name="jamail")
        c3 = Customer.objects.get(name="hanbing")

        self.assertTrue(len(Membership.objects.filter(customerNo=c1)) >= 1, msg="Customer1 is a member.")
        self.assertFalse(len(Membership.objects.filter(customerNo=c2)) >= 1, msg="Customer2 is not a member.")
        self.assertTrue(len(Membership.objects.filter(customerNo=c3)) >= 1, msg="Customer3 is a member.")
```

### 3.2 Software Engineering Design Module Matrix

Module Name	PIC	S	O	L	I	D	Cohesion	Coupling	Reason of Ranks
Login/Register	Chao	8	8	6	6	10	8	8	Login and register should be separated interfaces
Membership	Chao	10	8	8	10	10	10	9	Simple logic and strong cohesion.
Purchase	Yihang	10	8	8	10	10	10	9	Simple logic and strong cohesion.
Product	Saheed	10	8	8	10	10	10	9	Simple logic and strong cohesion.
Activity	Chao	10	10	8	10	10	10	9	Decorator Class
Promotion	Yihang	10	9	8	10	10	9	8	Using decorator pattern.
Discount	Saheed	10	9	8	10	10	9	8	Using decorator pattern.
Query Handler	Chao	10	9	7	10	10	8	8	Few coupling with data models.
Workflow Handler	Chao	10	9	8	10	10	8	7	Few coupling with data models and UI.
Promotion Handler	Yihang	10	9	8	10	10	8	7	Few coupling with data models and Purchase
Discount Handler	Saheed	10	9	8	10	10	8	7	Few coupling with data models and Purchase

Backup/ Restore Handler	Chao	10	9	7	10	10	9	9	Very few coupling with server environment.
Query JS	Saheed	8	7	6	8	10	8	8	Few coupling with UI.
Workflow JS	Yihang Chao	7	6	6	7	10	7	6	Few coupling with data models and UI
Login JS	Chao	8	7	5	6	10	8	8	Few coupling with UI
Backup/ Restore JS	Yihang	8	7	5	8	10	9	9	Very few coupling with Backup/Restore handler.

### 3.3 User Interface Screens

This project finally delivered a website that could be deployed on web server or local computer. Below screenshots show some user interface of this application:

The screenshot shows a web browser window for the URL `localhost:8000/query/init/`. The page title is "Waka". The navigation bar includes links for Home, Query (highlighted in red), Workflow, and About. A user profile icon for "cchen" is also present. On the left, there's a sidebar with "Query and Edit" and a blue button labeled "Backup and Restore". The main content area has a large input field with the placeholder "Backup to Date". Below it, a message says "Backup successes!" in red. A modal dialog titled "Restore from Date" is open, showing a date picker set to "11/16/2016". The calendar for November 2016 is displayed, with the 16th highlighted in blue.

The screenshot shows a web browser window for the URL `localhost:8000/purchasing/`. The page title is "Waka". The navigation bar includes links for Home, Query, Workflow (highlighted in red), and About. A user profile icon for "Tony Zhao" is shown. On the left, there's a sidebar with "Purchasing" (highlighted in blue) and other options: Membership, Expenditure, and Activity. The main content area has a "Customer Purchasing" form. It includes fields for "Stock" (a dropdown menu with the option "... select an product ..."), "Price" (a text input with "0.00"), "Count" (a text input with "0"), "Membership ID" (a text input with "Membership ID"), "Payment" (a text input with "0.00"), and a "Submit" button.

Waka

Home Query Workflow About Tony Zhao

Purchasing

**Membership**

Expenditure

Activity

New membership

Name: Zhao Yihang

Phone: 8324741068

Gender: Male

Birthday: mm/dd/yyyy

Balance: 0.00

Pin code: Pin code

Address: Address

Waka

Home Query Workflow About Tony Zhao

Purchasing

Membership

**Expenditure**

Activity

Stock

Others

**Stock In**

Stock

New Stock

Count: Count

Payment: 0.00

Details: Details

Waka

Home Query Workflow About Tony Zhao

Purchasing

Membership

Expenditure

**Activity**

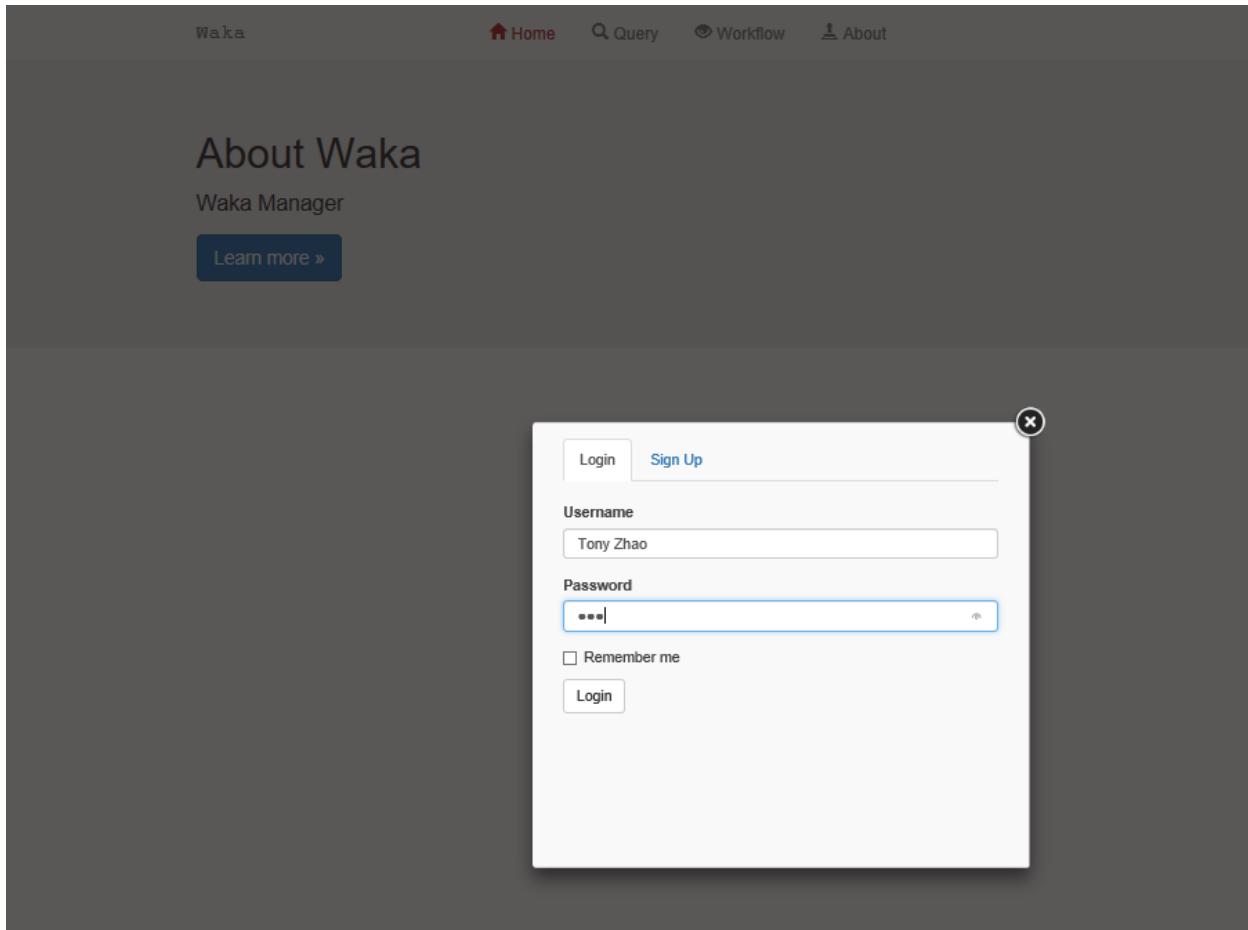
Create Discount

Create Promotion

**New Discount**

Valid From: mm/dd/yyyy

Valid To:	mm/dd/yyyy	YY
July	12	2012
August	13	2013
September	14	2014
October	15	2015
<b>November</b>	<b>16</b>	<b>2016</b>
December	17	2017
January	18	2018
February	19	2019
March	20	2020
April	21	2021



## 4. Project Repository and Tools

### 4.1 Repository

This project uses GitHub as the source code control tool. Below figure shows the screenshot of the source code repository:

The screenshot shows a GitHub repository page. At the top, there are buttons for 'Branch: master ▾', 'New pull request', 'Create new file', 'Upload files', 'Find file', and a large green 'Clone or download ▾' button. The main area displays a list of commits from user 'cchen1983' with the message 'Fix minor bugs.'.

File / Commit Message	Description	Time Ago
contacts	Add first stage development source.	2 months ago
home	Add first stage development source.	2 months ago
login	Update code comments	a month ago
query	Fix minor bugs.	6 hours ago
static	Add Activity/Discount/Promotion Modules.	18 hours ago
templates	Implement discount and promotion applying	7 hours ago
waka	Add first stage development source.	2 months ago
workflow	Fix minor bugs.	6 hours ago
.gitignore	Add first stage development source.	2 months ago
ReadMe.txt	Initialize the django project.	2 months ago
get-pip.py	Add first stage development source.	2 months ago
manage.py	Add first stage development source.	2 months ago
mysqlUtils.py	Add first stage development source.	2 months ago
waka.sql	Add first stage development source.	2 months ago
wakalnit.sql	Add first stage development source.	2 months ago
wakaTest.sql	Add first stage development source.	2 months ago

Below the commit list is a single file entry for 'ReadMe.txt'.

## 4.2 SCRUM Tool

For project management purpose, we use Scrumwise, an online SCRUM platform , to manage all the tasks and resources by evenly distributed them to multiple sprints. Below screenshot shows the situation in one of the sprints:

The screenshot displays the Scrumwise interface with the following details:

**Sprint 5:** Completed (11 days completed)

- DJTeam
  - Activity public interface Implementation (3 d, Sprint completed)
  - Promotion Class Implementation (4 d, Sprint completed)
  - Discount Class implementation (4 d, Sprint completed)

**Sprint 6:** Completed (18 days completed)

- DJTeam
  - Implement Discount/Promotion Info Query/Response. (5 d, Sprint completed)
  - Integrated test and debug (3 d, Sprint completed)
  - Event handler of create new promotion implementation. (5 d, Sprint completed)
  - Event handler of create new discount implementation. (5 d, Sprint completed)

At the bottom left is a button to "Add a sprint". At the bottom right are buttons for "Show All sprints" and a dropdown menu.

## 5. Weekly SCRUM Sheets

COMP 5423 Software Engineering Processes, CS Dept, PVAMU Fall 2016 SCRUM MASTER DAILY/WEEKLY FORM							
This form must list the date of each week, tasks assigned to each team member for the week, and the answers to the three questions that the Scrum Master must ask each team member every day. All members MUST sign the form. Entries in this form WILL contribute towards individual member grade changes. Incorrect information WILL lead to grade degradation of entire team. This form MUST be submitted to the instructor EVERY week by the SM to the instructor at beginning of class. Each team will project this form in class, to present their progress in class each week.							
Project Name:	Django DBMS						
Date of each week:	1/1/16	1/1/16	1/1/16	1/1/16	1/1/16	1/1/16	1/1/16
Master Name:	Chao Chen		SM: All tasks accomplished?: Yes No	SM Sign: Chao Chen			
Tasks Assigned for the week:	SOLID Principle Refactoring. Unit test						
What did you do?	Code Refactory	Code R-factory		Code refactoring		Unit test tool setup	
What obstacles you had?							
What do you plan to achieve?	Fix problem Fix Problem of dependency of Style Refactoring Fix some coupling. Install Unit test tool and Junit test framework						
Date of each week:	1/1/16	1/1/16	1/1/16	1/1/16	1/1/16	1/1/16	1/1/16
Member 2 Name:	Yihong Zhao		SM: All tasks accomplished?: Yes No	Mem 2 Sign: Yihong Zhao			
Tasks Assigned for the week:	SOLID principle Refactoring. Debug						
What did you do?	Code Refactory	Code Refactory	Code Refactor	Code Refactoring & Debug			
What obstacles you had?							
What do you plan to achieve?	Fix dependency issue. Remove multiple fix correlations Solve Refactoring problem Fix UI issue when commit code without form						
Date of each week:	1/1/16	1/1/16	1/1/16	1/1/16	1/1/16	1/1/16	1/1/16
Member 3 Name:	Soham Aditya		SM: All tasks accomplished?: Yes No	Mem 3 Sign: Soham Aditya			
Tasks Assigned for the week:	Solid principle Refactoring Debug						
What did you do?	Code Refactory	Code Refactory	Debug	Code Refactor		Debug	
What obstacles you had?							
What do you plan to achieve?	Fix dependency problem in source code multiple times. Fix ajax error in unused script and variables Clear code issues. Fix static issue in styling.						