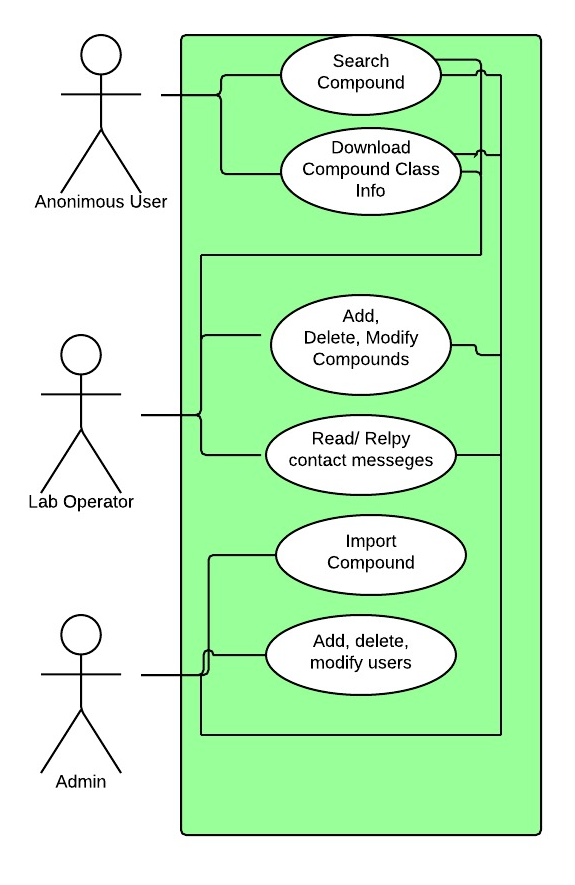
## User Story

# 

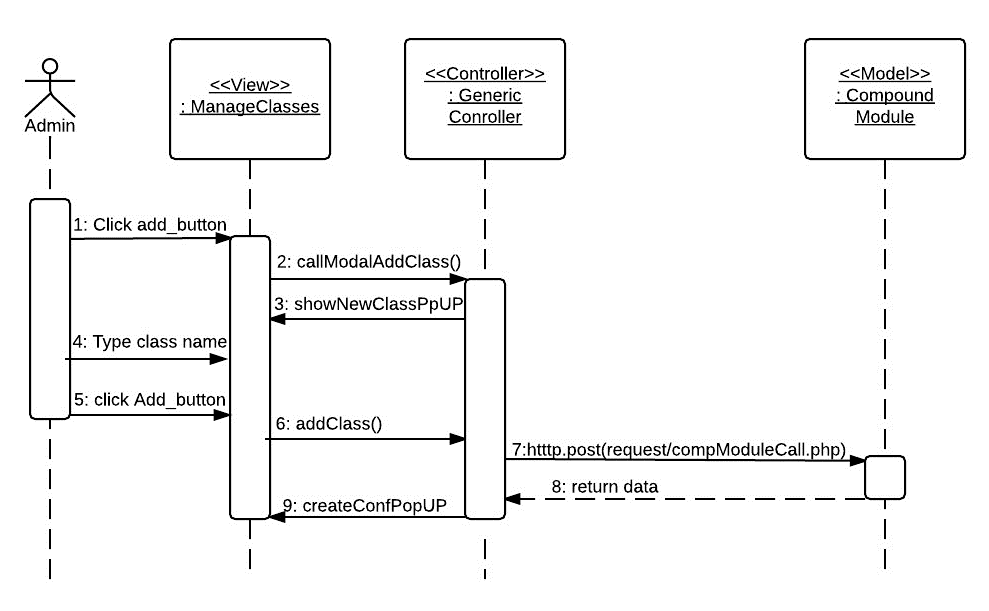
## Use Case

|  |
| --- |
| Name: 223\_create\_class |
| Participating actor: admin user or labOP |
| Entry condition:   * User is logged in the system * User is in their manageClasses page. * User has admin or LabOP rights |
| Exit condition: User invite is sent |
| Event flow:   1. User click on the Add Class button on manageClasses view 2. Function on controller callModalAddClass is called 3. Contoller shows a popup on the manageClasses view 4. User enter the name of the class on the pop up text box 5. User clicks the add button on the pop up. 6. Function on controller addClass() is called 7. Model is called with an http request calling the compound module 8. Model return data confirmation to Controller. 9. Controller shows pop up in view confirming that class was added or not. |

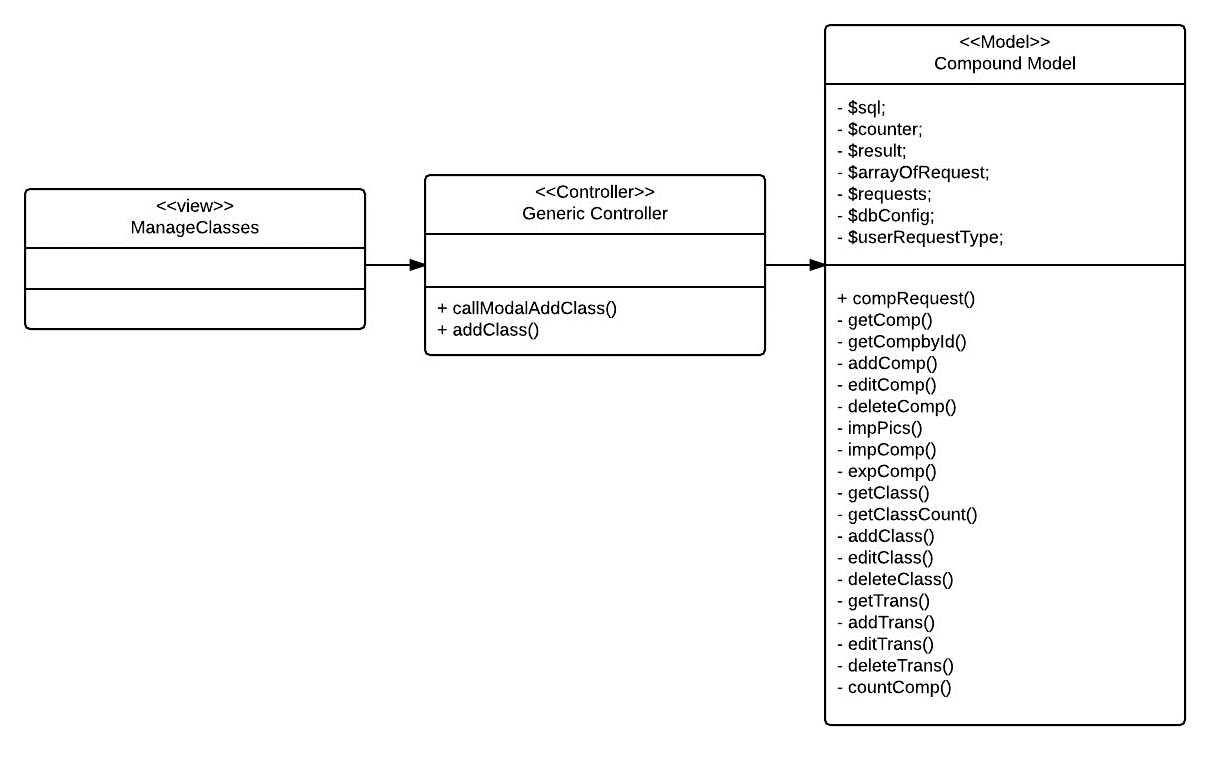
# Use Case Diagram



# Sequence Diagram

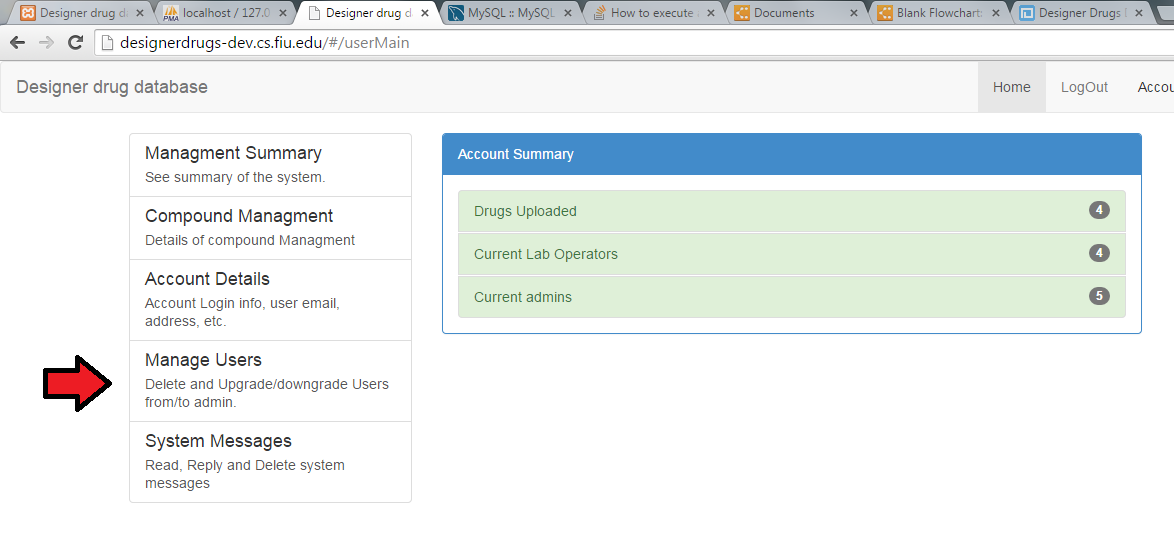
****

# Class Diagram

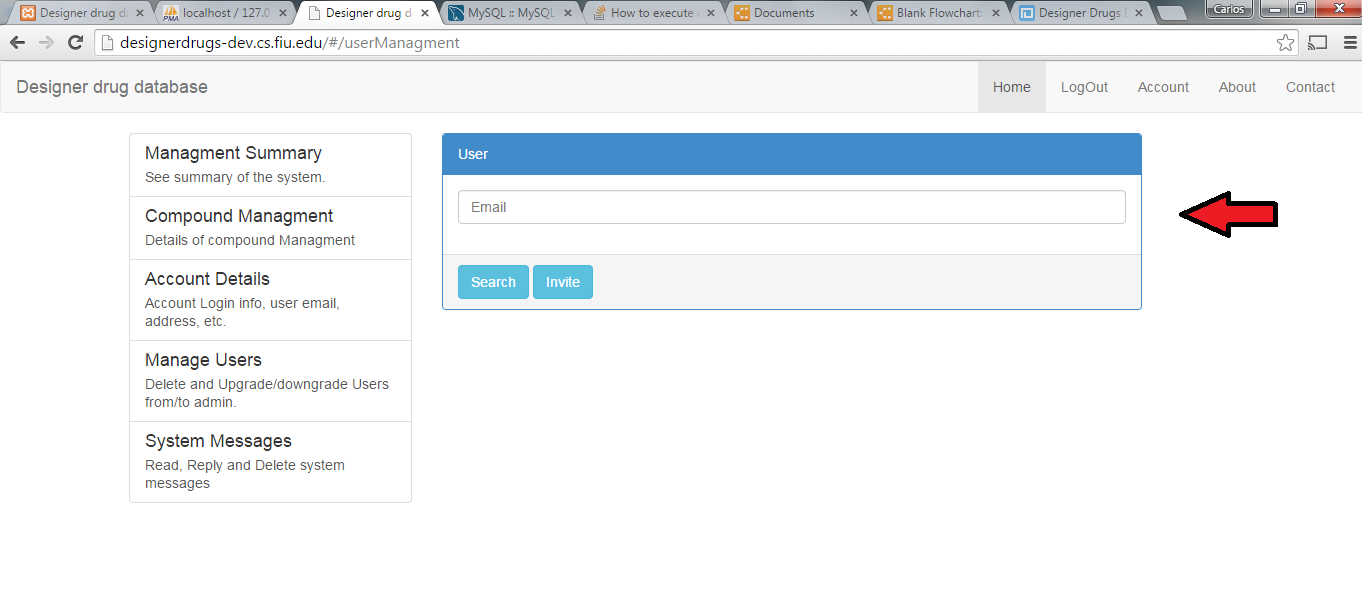


# User Manual

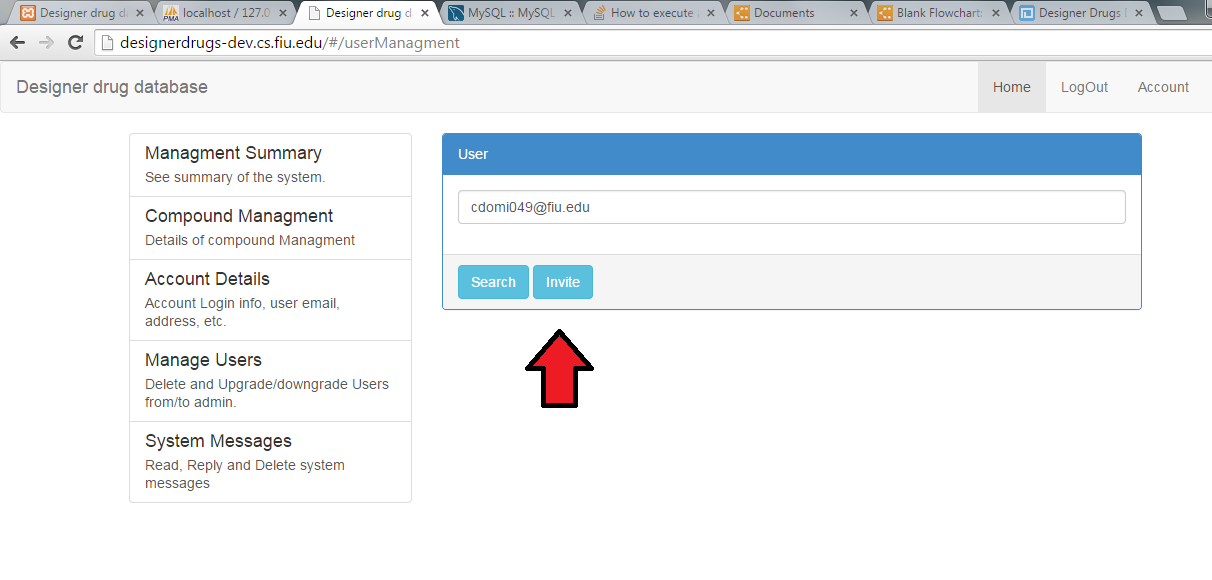
To create a new user, after you logged in click on the user Management Menu



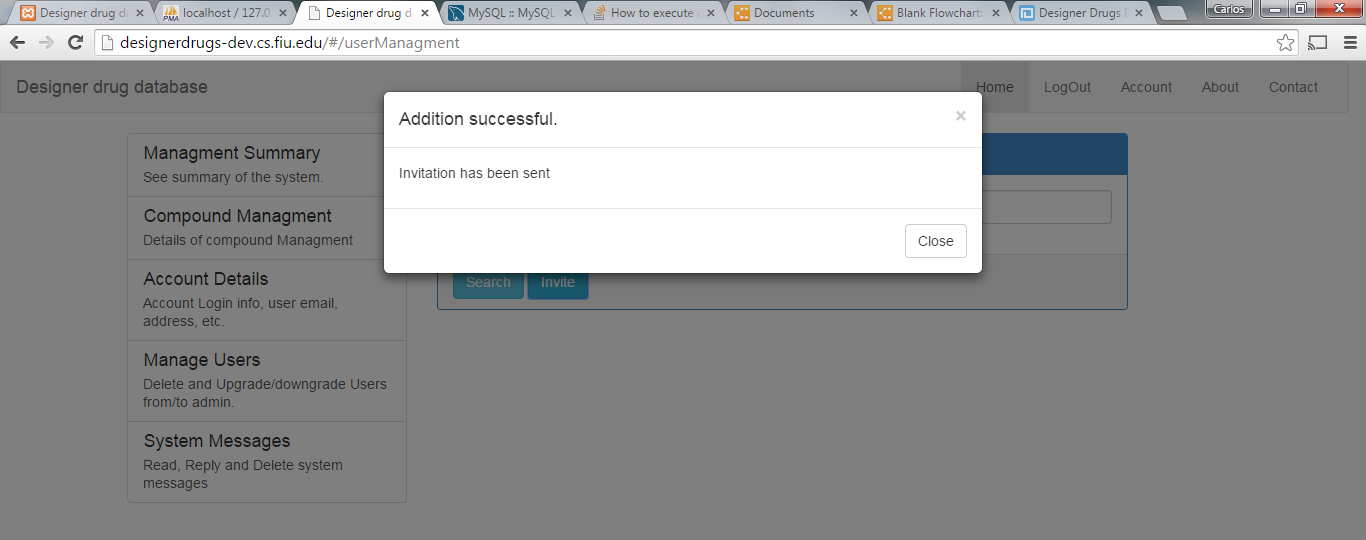
Then enter the email of the person you would like to invite in the email box



Click Invite button



Then you will get a confirmation pop up to confirm that the invitation was successfully sent.



Installation Manual

first install Apache2, Apache is the web server, According to .....

to install apache, use:

sudo apt-get install apache2

------------------------------------------

Second install php5, php is .....

Note: because you have already install Apache, the php will detect that Apache is already installed and it will install the php module for Apache. This is the reason why it is important to keep the order

to install php use:

sudo apt-get install php5

-------------------------------------------

PHP Additional module

third install mysql client for php

Note: PDO is been used in the application, but it is a feature that is installed by default in the php core. However, in this step is installed PDO for mysql.

to install this additional module use:

sudo apt-get install php5-mysql

--------------------------------------------

fourth install mysql server

to install mysql server use:

sudo apt-get install mysql-server

----------------------------------------------

fifth install mysql client:

Note: this step is optional although very helpful. This step is to be able to access the mysql server on the console.

to install mysql client use:

sudo apt-get install mysql-client

----------------------------------------------

sixth install git to access github server which is the versioning system for the code.

Note: this step is also optional to synchronize the code with the github repository in case changes will be made to it. To check how to use git, check manuals online.

to install git use:

sudo apt-get install git

-------------------------------------------

seventh install sendMail, which is a MTA, which is a Mail Transport Agent.

Note: this is not an email server, is just an MTA that allow us using php in linux to send emails. This feature is used in the program because creating users is by invitation.

to install sendMail use:

sudo apt-get install sendmail

--------------------------------------------

eigth download or copy the application files in your Apache designated folder

Note: it does not have to exactly be but in my case the folder was located into /var/www/html

--------------------------------------------

finally create the database using the file in Code/database

need to login with super user permision, to do this use:

sudo -su

then enter your password if needed

then go to the folder that contains the .sql file and enetr:

mysql -u root -p < drugsdb.sql

then enter the password when requested

--------------------------------------------

## Test Cases Suites

|  |  |
| --- | --- |
| Test Case ID | 175InviteUser\_01 |
| Purpose | To test that a user email has been sent when an invitation is sent |
| Preconditions | 1. User is logged in as an admin 2. User is in the User Management Page 3. User “cdomi049@gmail.com” has never been invited |
| Input | 1. On the email textbox enter “Cdomi049@gmail.com” 2. Click on the Invite Button |
| Expected Output | Confirmation that invitation has been sent |
| Actual Results | A pop up with confirmation that invitation has been sent is showed |

|  |  |
| --- | --- |
| Test Case ID | 175InviteUser\_02 |
| Purpose | To test that an invitation is not sent when email field is empty |
| Preconditions | 1. User is logged in as an admin 2. User is in the User Management Page |
| Input | 1. On the email textbox enter “” (nothing) 2. Click on the Invite Button |
| Expected Output | Pop up saying email cannot be empty. |
| Actual Results | A pop up saying that email cannot be empty is showed. |

|  |  |
| --- | --- |
| Test Case ID | 175InviteUser\_03 |
| Purpose | To test that invitation is not sent when user email has already been invited |
| Preconditions | 1. User is logged in as an admin 2. User is in the User Management Page 3. User “cdomi049@gmail.com” has never been invited |
| Input | 1. On the email textbox enter “Cdomi049@gmail.com” 2. Click on the Invite Button 3. Click Ok to the pop Up saying that the invite was sent 4. On the email textbox enter “Cdomi049@gmail.com” 5. Click on the Invite Button |
| Expected Output | Pop up saying user has already been invited. |
| Actual Results | A pop up saying user has already been invited is showed. |