Implementation Details

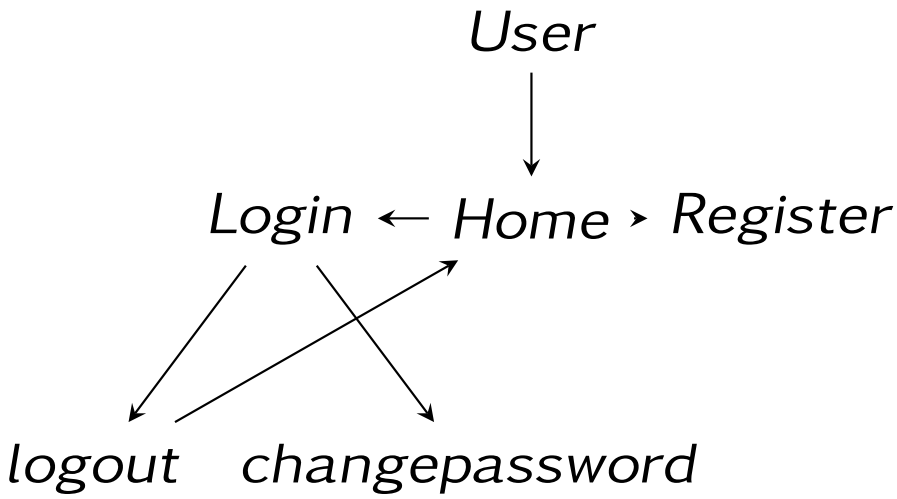
Database Design/Algorithm used

1. We mainly use two tables in our data base
2. One table corresponds to the default user model provided by django and stores the attributes required to register and login.
3. Another comprizes Profile picture, Nick name and ID(foreign key) which is inner joined with the first table(where it's ID is the primary key).
4. Algo for password management: We use *auth_password_validators* to validate the input password and flag appropriate error if the given requirements aren't met.

Implementation Details

Workflow

Most of the html files are extended from base.html



Reason for Choices Made

We choose the provided user authentication template to keep it simple for the user and ourself. Later created a new model to accomodate user details in the database.

Future Plan of Action

Future Timeline

1. Planning on creating new tables for Friends, Groups ,and recent activity by creating new models for them.
2. All the new models contains *Transaction_Id* as the primary key and *User_Id* as the foreign key.

Friends	Groups	Activity tab
User Id	User Id	User Id
Transaction Id	Transaction Id	Description
Friends Id	Friends Id	Transaction Id
Money	Money for friend	
Transaction time	Transaction time	
Description	Group name	
	Description	

Table: Models for new database

Future Plan of Action

Reasoning for the additional feature selection

Planning on taking up the second additional feature i.e notification management,when thers a settle up option. For this, we are planning on creating another tab in the frontend and a model in the backend for notifications.

Detailed plan

1. Existing tables are to be extensively used.
2. For example, date ranges can be accessed from Friends and Groups tab and we can list out the transactions pertaining to that range.
3. to get the bar graphs, pie chart, time series plot, Matplotlib in python can be used.