# 4.architecture

**4.1 Development and deployment architecture**

There are 3 applications used to control development and to deploy to server

**4.1.1 Trello**

Trello is used for separating tasks to each member to do in each sprint.

**4.1.2 GitHub** Github is a source control application and also has powerful functions to connect other applications such as Heroku, which is used for deploying in this project. There are 3 important tasks that Github help the team to collaborate.

1. Source control: After separating tasks to each team member, each member develop their task on their own branch, then, merge to a sprint branch which is checked before delivering to the client every week.

2. Each sprint brach is sent to Heroku application to deploy on reviewing server to review the web application’s workflow.

3. After each sprint approved, they are merged to master brach and deploy to staging server on Heroku.

**4.1.3 Heroku**

Heroku is a cloud platform that lets developers build, deliver, monitor and scale apps. There are 2 main roles for develop and deployment.

1. Review and test web application.

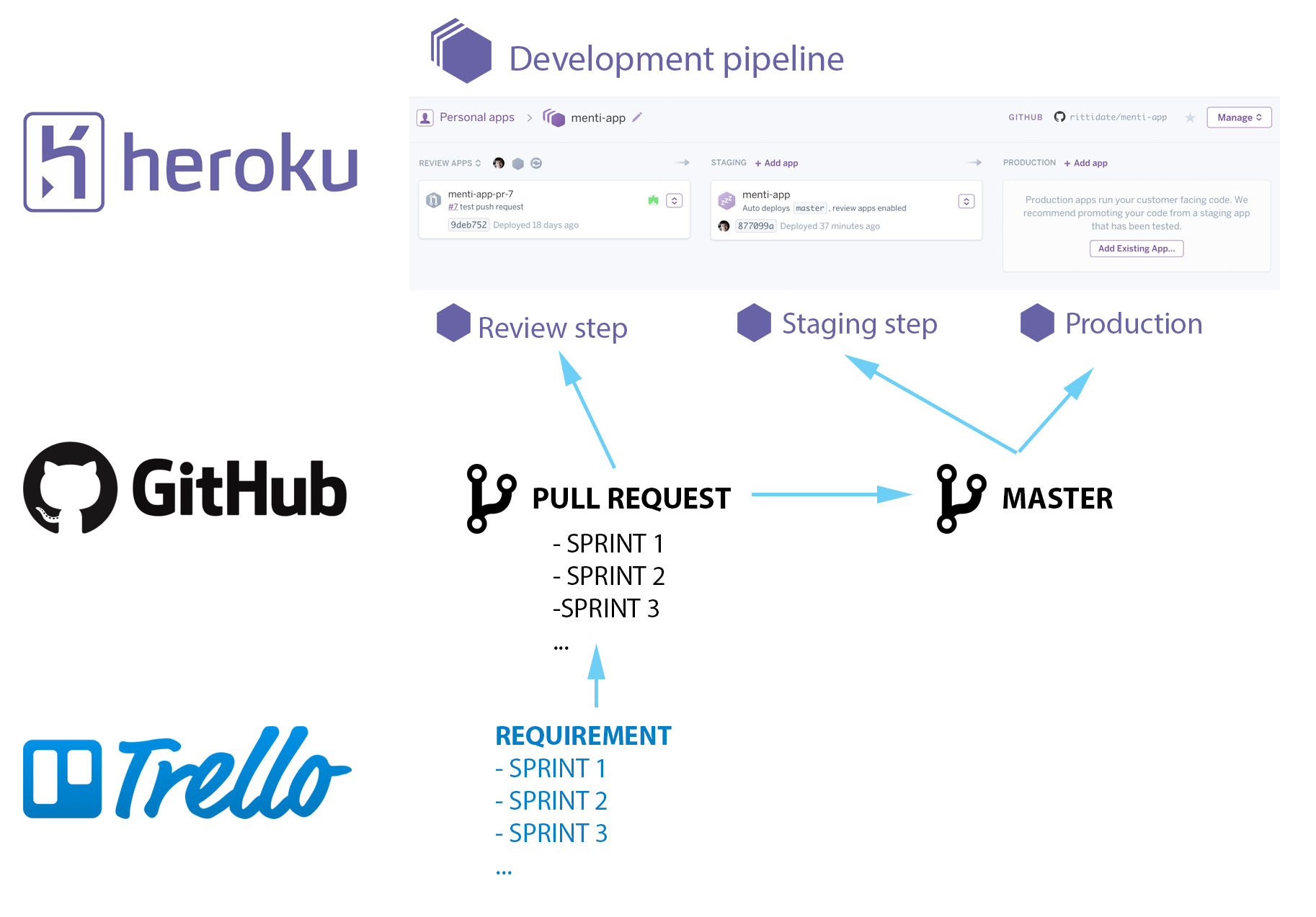
2. Deploy on real environment.

Figure 4.1. Development and deployment architecture

**4.2 Framework architecture**

Architecture of Ruby on Rails framework has 3 main parts.

**4.2.1 Model**

**T**he model relates to data in database which this project use PostgreSQL as database engine.

**4.2.2 Controller**

The controller can send parameters to the model to update the model's state. It can also send parameters to its associated view to change the view's presentation of the model. It also connect to outside applications’s api which there 3 applications used in this project.

1. Facebook Login

2. Braintree

3. Amazon S3

**4.2.3 View**

The view generates new output to the user based on changes in the model. It also interact with a user. There are many extensions that facilitate the developers to display on screen.

1. Sass is one of CSS component to compile programatic CSS code.

2. Materialise CSS is a library to help the view to display in many platforms such as PCs and mobiles.

3. CoffeeScript is a little language that compiles into JavaScript.

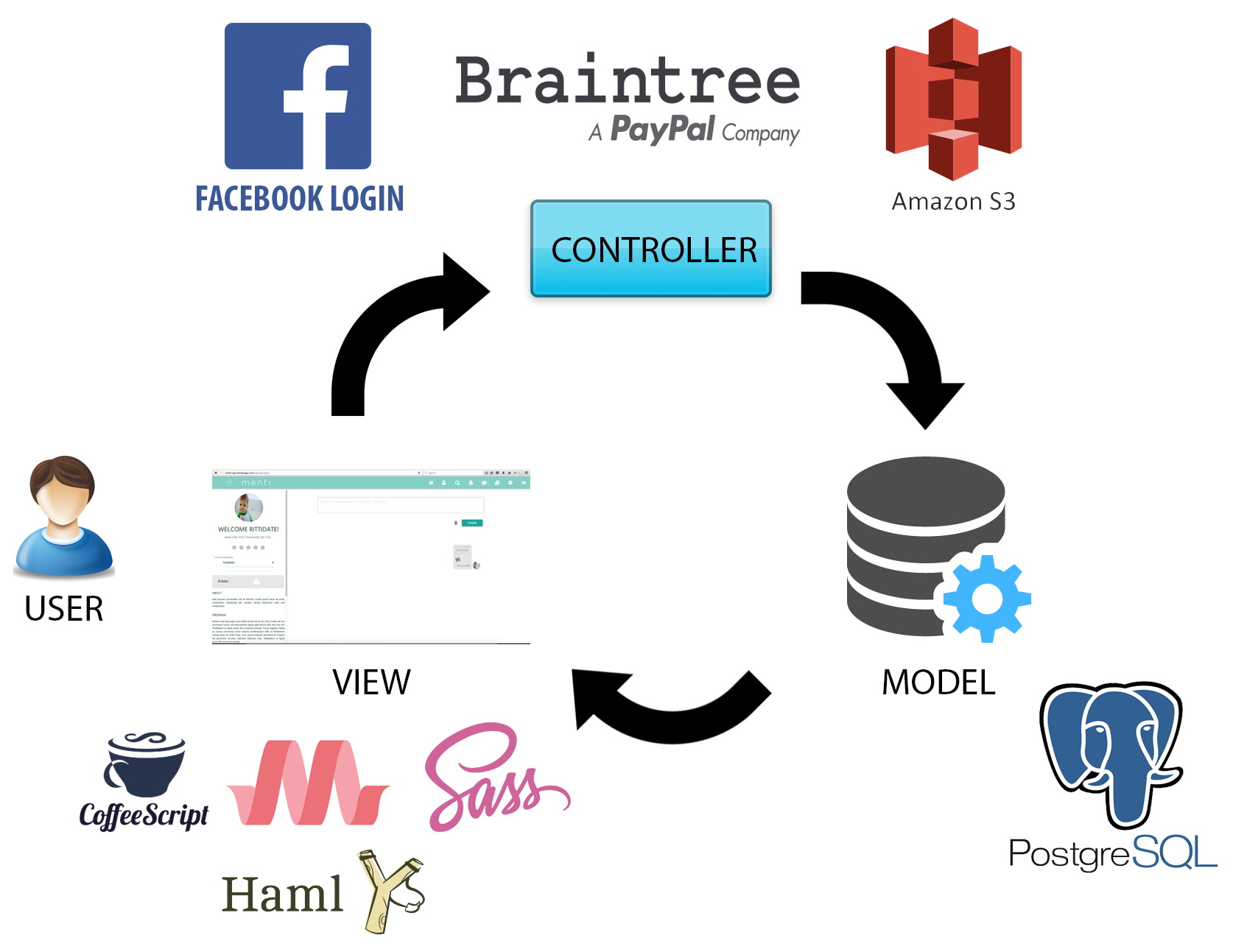
4. Haml is a language to complies into html. It can compile both html and other programatic language such as ruby and PHP.

Figure 4.2. Framework architecture

**4.3 Website application architecture**

In figure 4.3, There are two types of block, dark blocks and white blocks. The dark blocks are display pages which users can see. The white blocks are pages that get parameters to process with database and return data as JSON.

**4.3.1. Display pages**

* + - **Homepage**: the main page of the website
    - **Registration**: the users’ registration page
    - **Login**: sign in page
    - **Dashboard**: a page show after sign in
    - **My profile**: a page shows user’s information, desired categories and feed message.
    - **Search**: a page used to search mentors and mentees matching a keyword
    - **Notification**: a centre page to show notifications that notify interacting between mentors and mentees.
    - **Message**: send message between mentors and mentees.
    - **Resource**: a page gather all documentation and upload.
    - **Setting**
    1. User info: update a user information.
    2. Change password: update password.
    3. Default payment: update credit card information.
    4. Category: update desired categories of mentees,
    5. Mentor’s setting: update course and price of mentors.
    - **Logout**: Sign out from user’s authorisation.
    - **Mentor’s** **profile**: mentor’s information, following, feed message and contact.
    - **Course’s agreement**: agree mentee's request in notification page.
    - **Add credit card**: add a credit card similar to default payment page.
    - **Wait**: a page that a user wait for a mentor’s response.1

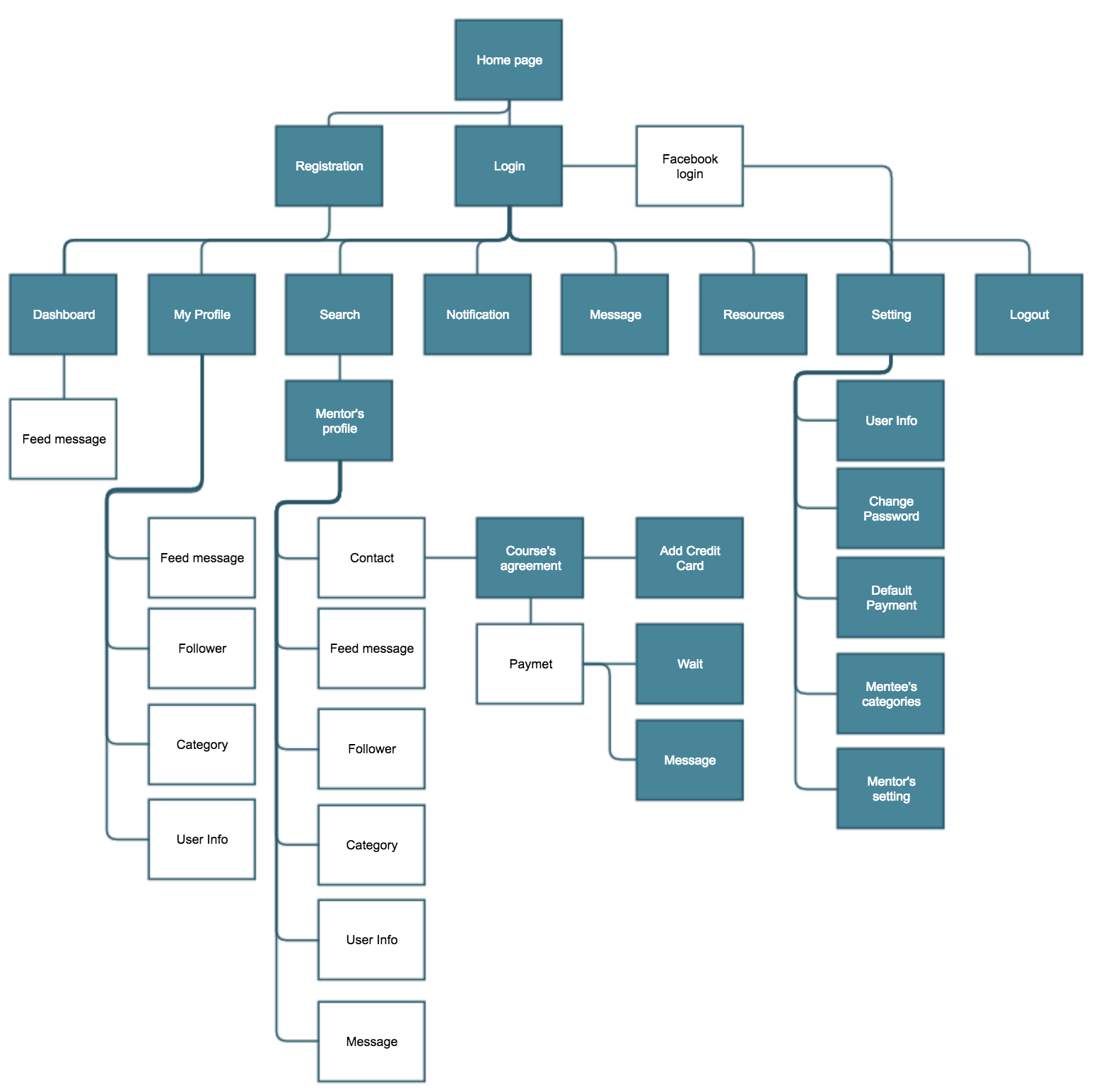


Figure 4.2. Site map

**4.4 Database architecture**

The database has the following tables:

* + **U****sers table**

- username

- password

- email

- user’s info such as address education and preferred language

- user’s type such as mentor, mentee or both

- user profile’s image

- mentor status

* + **Categories table**
  + **Categories users relations table**

This table shows a user relates to categories

* + **Conversations table**

This table shows a link between two users that can send message to each other

* + **Conversations replies table**

This table shows that a user sends message to a conversation’s room.

* + **Conversations histories table**

This table shows that a user deleted latest message from a conversation’s room.

* + **Courses table**

This table shows that a mentor create a course links to a category and define a price.

* + **Courses user relations table**  
     This table shows that a mentee applies a course of a mentor.
  + **Feed messages table**

This table is used to create and share a message in a user’s feed

* + **Follows table**

This table shows that links between followers and a user.

* + **Identities table**

This table is used to store data from Facebook login api.

* + **Notifications table**

This table shows that notifications are sent from conversations, apply courses and resource functions.

* + **Payment table**

This table is used to store client’s number which return from braintree api.

* + **Ratings table**

This table shows other users give rating to a mentor.

* + **Resources table**

This table is used to store file name and link to show on view.

* + **Transactions table**

This table shows referred number getting braintree transactions.

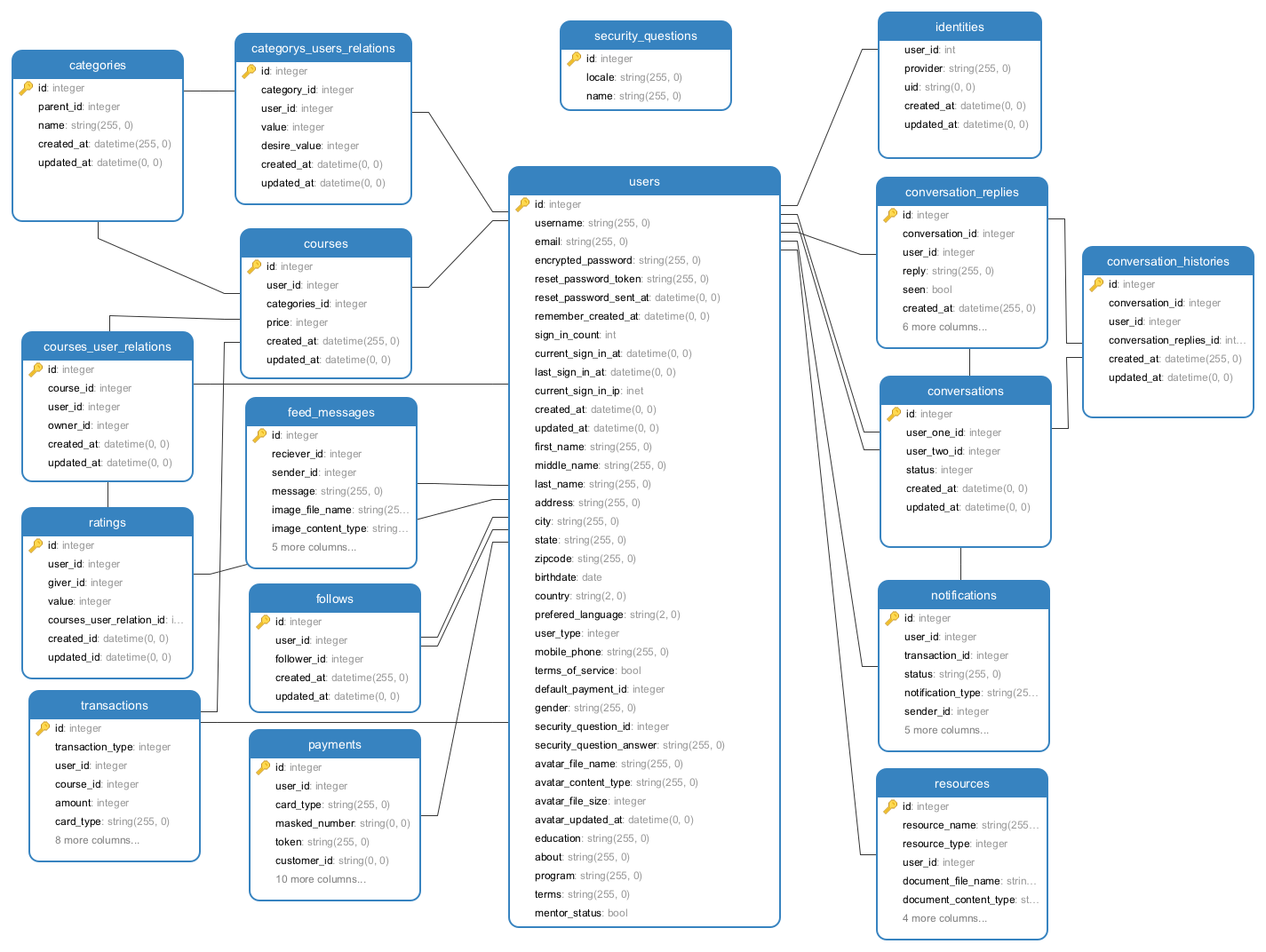


Figure 4.4. Database diagram