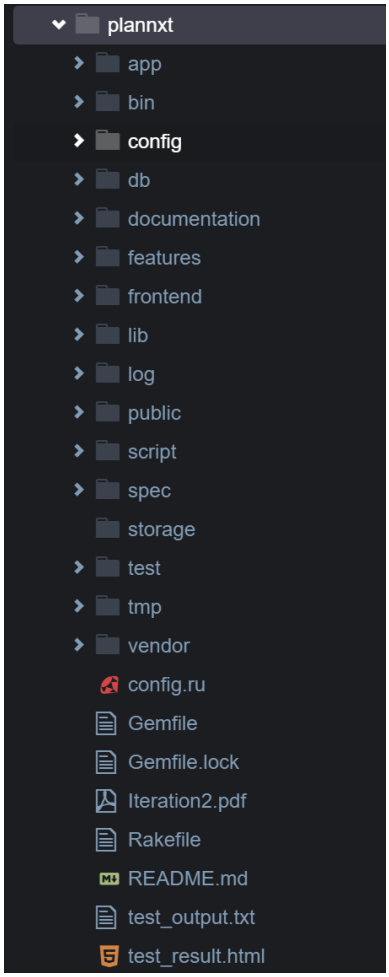
Code Architecture

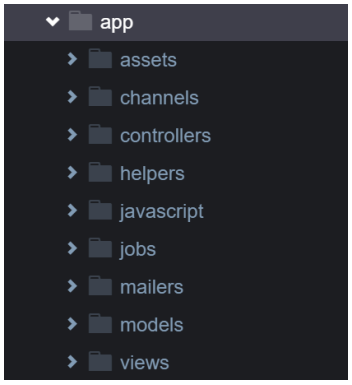
All the code files layout:

The project file folder “plannxt” has a structure as following.



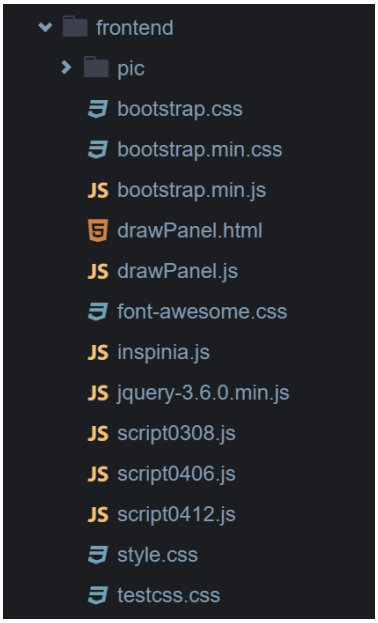
- App folder

  The “app” folder contains the functional modules of the project, including controllers, models and views.



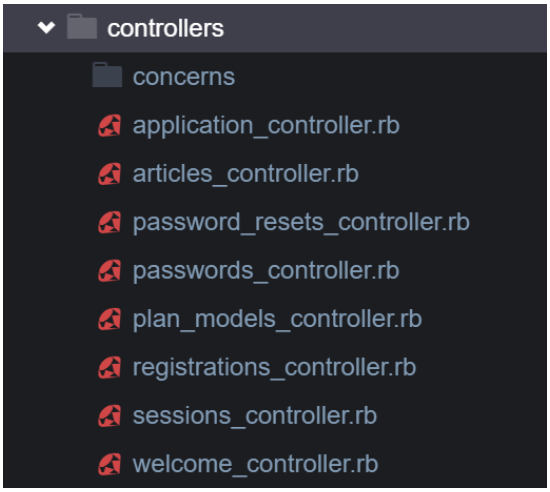
- Frontend folder

  The “frontend” folder contains the static frontend javascript and html files



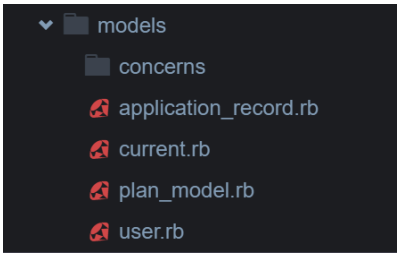
- Controllers:

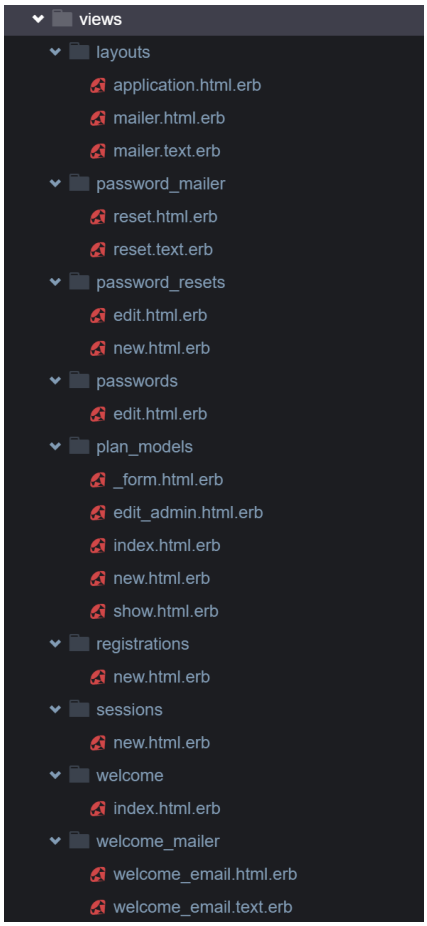
  The “controller” folder contains the controller Ruby code files.



- Models and Views:

  The “model” folder contains the models.
  The “view” folder contains the views associated with the controller.





Getting Started

1. How to run in the development environment?

a. Prepare rails environment, install necessary components

Before you install Rails, you should check to make sure that your system has the proper prerequisites installed. These include:

Ruby
Postgresql
Node.js
Yarn

The original development environment is based on:

Ruby version - 3.1.0
Rails version - v7.0.2.2
Command examples:

```
rvm get stable
rvm install "ruby-3.1.0"
```

You will also need an installation of the Postgresql database. The development environment is using:

PostgreSQL version - 10.19

```
sudo apt-get update
sudo apt install postgresql postgresql-contrib libpq-dev
```

PostgreSQL may require to create a role to allow rails to connect to the Postgre database. In AWS cloud9 ubuntu system, we executed

```
sudo -u postgres createuser --interactive ubuntu
```

For details of preparing rails environment, please refer to
[Getting Started with Rails]

b. Gti clone the codes from github

Cmd: *git clone https://github.com/LeviZhang1993/plannxt.git*

c. Bundle install

Cmd: *bundle install*
bin/rails db:environment:set RAILS_ENV=development

d. Migrate database

rake db:create
rake db:migrate

rake db:seed

e. Start server

rails server

2. How to deploy to the production environment(Heroku)?

a. Prepare rails environment and clone codes as development environment set up

b. Create and migrate production database

RAILS_ENV=production rake db:create db:migrate db:seed
heroku config:set RAILS_ENV=production

c. Start production server

RAILS_ENV=production rails s

Deploy to Heroku:

a. Register and login heroku account
Install heroku on the server.

b. Register and login heroku account
ssh-keygen -t rsa
heroku login -i
heroku keys:add
heroku create

c. Deploy our app to Heroku:

git push heroku main

d. first-time database creation on the Heroku side:

heroku run rake db:migrate

3. How to debug in the development environment?

a. Server log

Once a rails server starts, the console will print the server status, the request it received each time and its responses, as well as any errors and exceptions.
To add more information into console log, add
logger.info information_you_want
to where the codes to be executed.

b. Query database
We can use the rails console to query model databases.

rails console

To query database, for example,

Query the user database by a email address:

User.find_by(email: “name@xx.com”)

Query the user database by a user id:

User.find_by(id: 1)

Query the plan model database by plan id:

PlanModel.find_by(id: 1)

List all plan models:

PlanModel.all

Create a new user:

@user = User.new(:name => “username”, :email => “xxx@xx.com”, :level => 5, :password => “passwd”, :password_comfirmation => “passwd”)

@user.save

Features not implemented

- Automatically send verification email when registering or reset password.
  The mailer class has been created, but may need more modification to adapt to the environment. And also, users are allowed to use email to reset the password once they forget their password.

- Plan sharing and permission controlling
  Currently, users can only view and edit the plan they created, one feature needed is to allow users to share the plan they created to others, as well as to set the permissions for sharing. Some users may allow others to modify the plan, while some may only view the plan.

- Plan sharing frontend pages
  Once the permissions control is done, some web pages need to be created for sharing the plan, the sharing methods may include through user name, email link, or an encrypted url link.