Heroku workflow

The workflow for heroku is the same as it is on the github. Develop code, test it and push it to heroku, so it can be deployed using git.

However, there are several heroku specific things that have to be noted.

1. requirements.txt

Whenever a new library is added and used in django using pip install command, those libraries have to be included in requirements.txt file.

An example:

Template:

```
library_name == version_of_the_library
```

An example from requirements.txt is “Django==1.5.4”

The requirements.txt is an important file for heroku, as it the server on heroku reads it first, so it can install required libraries, before it can run the code. Missing entries for installed libraries in the file will cause the server to fail.

2. Procfile

If scalability becomes an important part for the project, it can be achieved in two ways. One way is to include new nodes in the Profile for the project. More about Procfile can be read here: https://devcenter.heroku.com/articles/procfile

The second way is to use to change settings on the heroku website for the project. This can be done by going to the project page on heroku. In resources tab there is a slider that responsible for adding new nodes.

This feature is not free, so the project owner will have to pay, before new nodes can be deployed.

3. Storage on Heroku

The project right now is using a Postgress Database that is only used for development purposes. Therefore, all the uploaded pictures get deleted after 3-4 hours to keep the server from overloading. One of the solutions is to used a persistent database that are hosted by a third-party company. An example of such company is Amazon, which can be read more about here: https://devcenter.heroku.com/articles/s3

Android Deployment

-Install android SDK from https://developer.android.com/sdk/installing/index.html
-Connect the android phone to the computer using the micro usb connector - usb 2.0
-If the eclipse is not available then use adb tool. Upload the app on your android phone by running this command in your terminal: adb -d Path_To_Your_APK_File
- The APK file is your compiled app.
-If eclipse is available, then run the application in the android emulator
-Export the package and sign it.
-Done

Webpage Deployment
PgAdmin III

Installation instruction of pgAdmin III

Assuming you have Django 1.5.4 installed - Has to be

1. Google PostGres and download and run the installer.
When it prompts for different settings, you can use whatever you'd like.
Be sure to remember what password you provided, though.
2. Open up pgadmin.
3. On the tree view, navigate to Servers->PostGreSQL <<blah blah blah>>. Right click PostGreSQL <<>> and click connect.
4. Create a New Login Role by right clicking PostGreSQL <<blah blah blah>>, and clicking New Object->New Login Role
5. Set the user name (role name) of the new user to django_login and the password to 12345. Click OK
6. Navigate to Servers->PostGreSQL <<blah blah blah>>->Databases
7. Right click on the Databases node and click New Database
8. Set up the New Database with the following settings: Name: django_db, Owner: django_login.
9. Try to call manage.py syncdb
10. Try call to manage.py runserver like usual.(“manage.py runserver”)

If you get an exception in 9 or 10 like that a module can't be imported, you should do the following:

Go to http://www.lfd.uci.edu/~gohlke/pythonlibs/#pip

1. Download/Install Setuptools and pip (you might also need to install psycopg2 if pip refuses to install it).
2. Ensure <<python 2.7 directory>/scripts is in your path
Restart cmd/shell
3. on the command line, use: pip install <<whatever-it-says-is-missing>
4. Repeat (3) until sqldb and runserver work.

Installation Instruction of Webpage

There is no installation required for the website, the only thing the user needs to do getting a stable host to support the website if the user wants to make the website work continuously in the future.

Heroku - Getting started

Before you can deploy to heroku, you will need to install Heroku toolbelt.

1. Make an account on heroku.com
2. For Windows: download the toolbelt from here: https://toolbelt.heroku.com/windows
   For Mac: https://toolbelt.heroku.com/osx
   For Linux: https://toolbelt.heroku.com/osx
3. After installing it, you will have to login to heroku via command line
   - In command line, type: heroku login
     - It will ask for your email and password. Please enter them carefully
   - Then it will look for your public key on your machine
     - If the toolbelt can find it, it will upload it to your account on server
     - if it can't find one, it will ask for your permission to create one and send it to your account. The process is automatic, so give it a few seconds.
4. After registering on Heroku, the owner of the webapp will have to include you in the project. After getting access to the app, go to Settings and check the Info section.
6. In the info section, you will find Git URL. Copy the link and clone the project into the folder where you will be working on it.
5. After the previous step, you are set to go. Since the Ultra-Remote-Medicine app is already configured to work on heroku, there is nothing else to do.

More info is located here: https://devcenter.heroku.com/articles/quickstart