

Chat based news assistant

CS691 : CHANDNI SHAH, GUANNAN ZOU, KINNARI JASOLIYA, LONGYUAN CHU,
PRADEEP PRAKASH, WAHIDUR RAHMAN

Objective

- * To develop a website to display news, scraping it from different news websites
- * To develop a chatbot that would assist the user to find news according to his preference and suggest latest news according to his preference after a certain desired time intervals
- * To train chatbot to smart enough to pick out news based on user preferences

Web scraping

- * Web Scraping is also termed Screen Scraping, Web Data Extraction, Web Harvesting etc
- * It is a technique employed to extract large amounts of data from websites whereby the data is extracted and saved to a local file in your computer or to a database in table (spreadsheet) format.

Modules used for web scraping

- * The modules used by us for web scraping is:
- * BeautifulSoup



Accomplishment

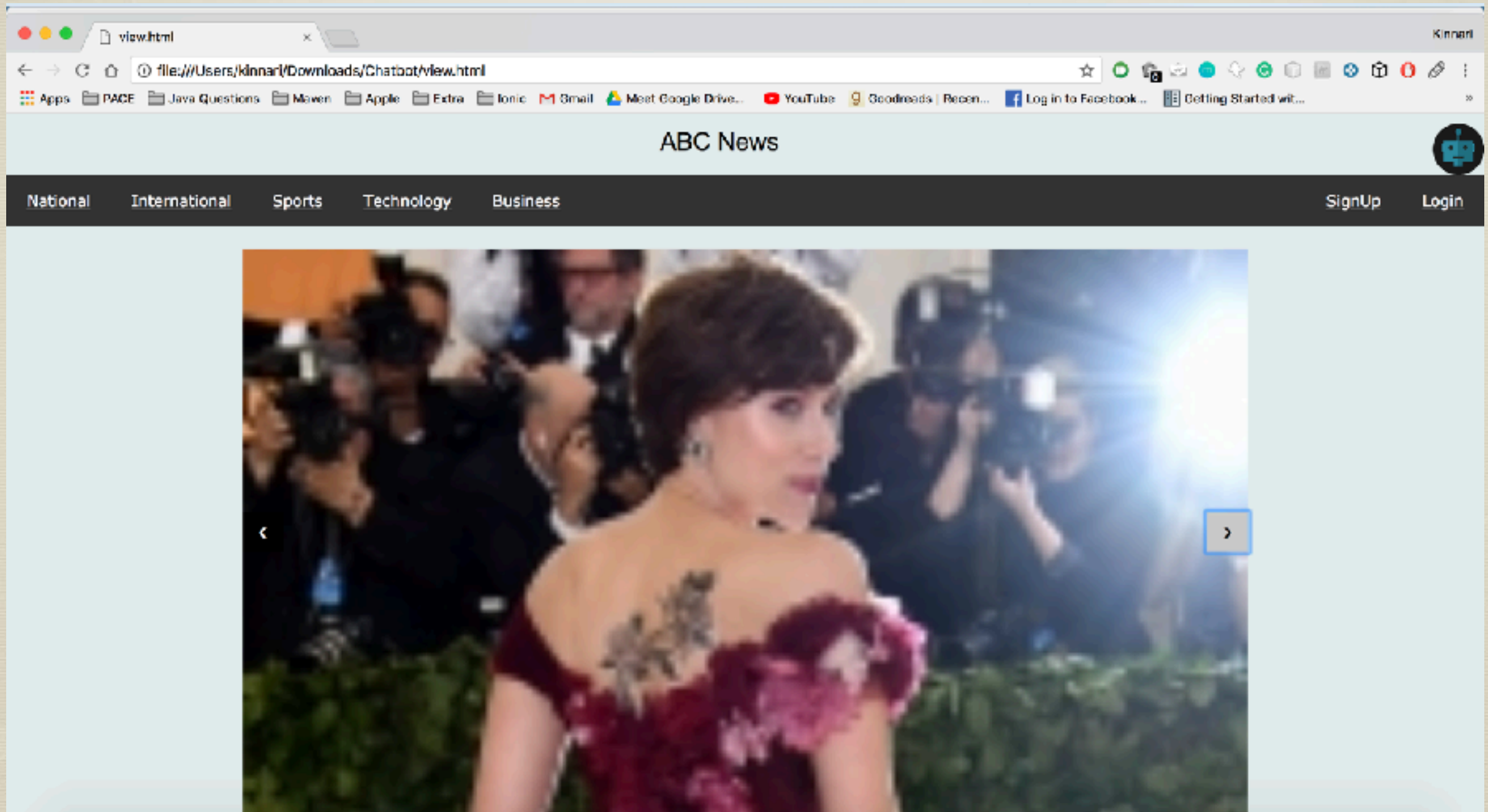
- * Using the python module, scraped news data from websites and stored it in a csv file for now
- * A simple UI to display news and a simple UI for a chatbot, separate for now


```

main.py x
1  # import libraries
2  import urllib2
3  import bs4
4  import csv
5  from datetime import datetime
6  data = []
7
8  # specify the url
9  quote_page = 'https://www.washingtonpost.com/news/powerpost/?utm_term=.46e8da5f7e3b'
10
11 # query the website and return the html to the variable 'page'
12 page = urllib2.urlopen(quote_page)
13
14 # parse the html using beautiful soup and store in variable 'soup'
15 soup = bs4.BeautifulSoup(page, 'html.parser')
16
17 # Take out the <div> of name and get its value
18 name_box = soup.find_all(['h3'])
19
20
21 # print datetime.now()
22 for i in name_box:
23     # name = name_box.text.strip() # strip() is used to remove starting and trailing
24     # print i.text.strip()
25     name = i.text.encode('utf-8').strip()
26     data.append(name)
27     # print data
28
29
30 # specify the url
31 quote_page = 'https://www.cnn.com/us'
32
33 # query the website and return the html to the variable 'page'
34 page = urllib2.urlopen(quote_page)
35
36 # parse the html using beautiful soup and store in variable 'soup'
37 soup = bs4.BeautifulSoup(page, 'html.parser')
38
39 # Take out the <div> of name and get its value
40 name_box = soup.find_all('h3', attrs={'class': 'cd__headline'})
41
42
43 # print datetime.now()
44 for i in name_box:
45     # name = name_box.text.strip() # strip() is used to remove starting and trailing

```

UI Static pages




UI Static pages

signup.html

file:///Users/kinnari/Downloads/Chatbot/signup.html

Apps PACE Java Questions Maven Apple Extra Ionic Gmail Meet Google Drive... YouTube Goodreads | Recen... Log in to Facebook... Getting Started wit...

SignUp Form



Username

Firstname

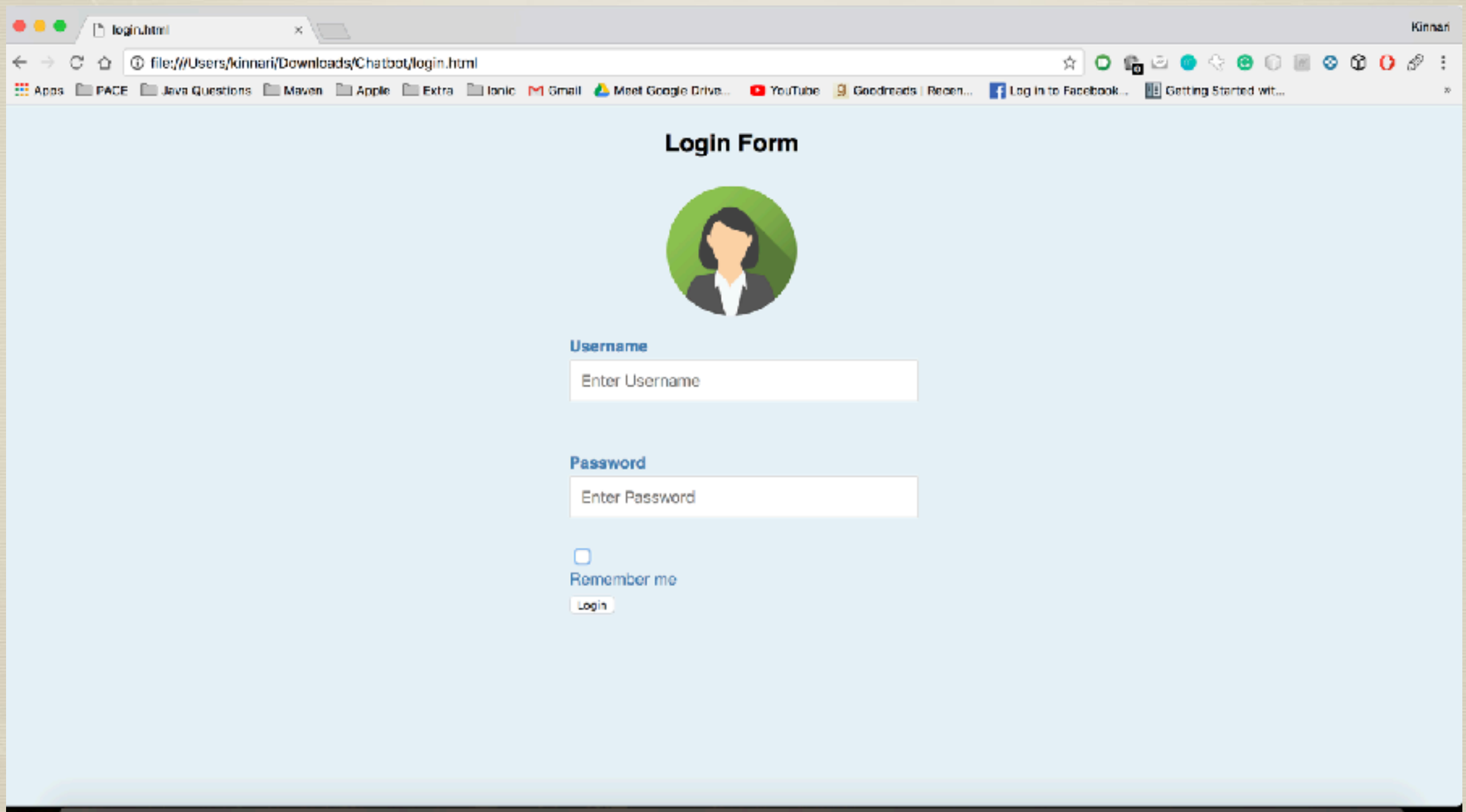
Lastname

Email

Password


Confirm Password

UI Static pages



The screenshot shows a web browser window with a single tab titled 'login.html'. The address bar displays the file path 'file:///Users/kinnari/Downloads/Chatbot/login.html'. The browser's toolbar includes various icons for navigation and extensions. The login form itself is centered on a light blue background. It features a circular profile picture placeholder with a green background and a dark silhouette of a person. Below the profile picture, there are two input fields: one for 'Username' and one for 'Password', both with placeholder text 'Enter Username' and 'Enter Password' respectively. A 'Remember me' checkbox is located below the password field, and a 'Login' button is positioned at the bottom of the form.

Login Form



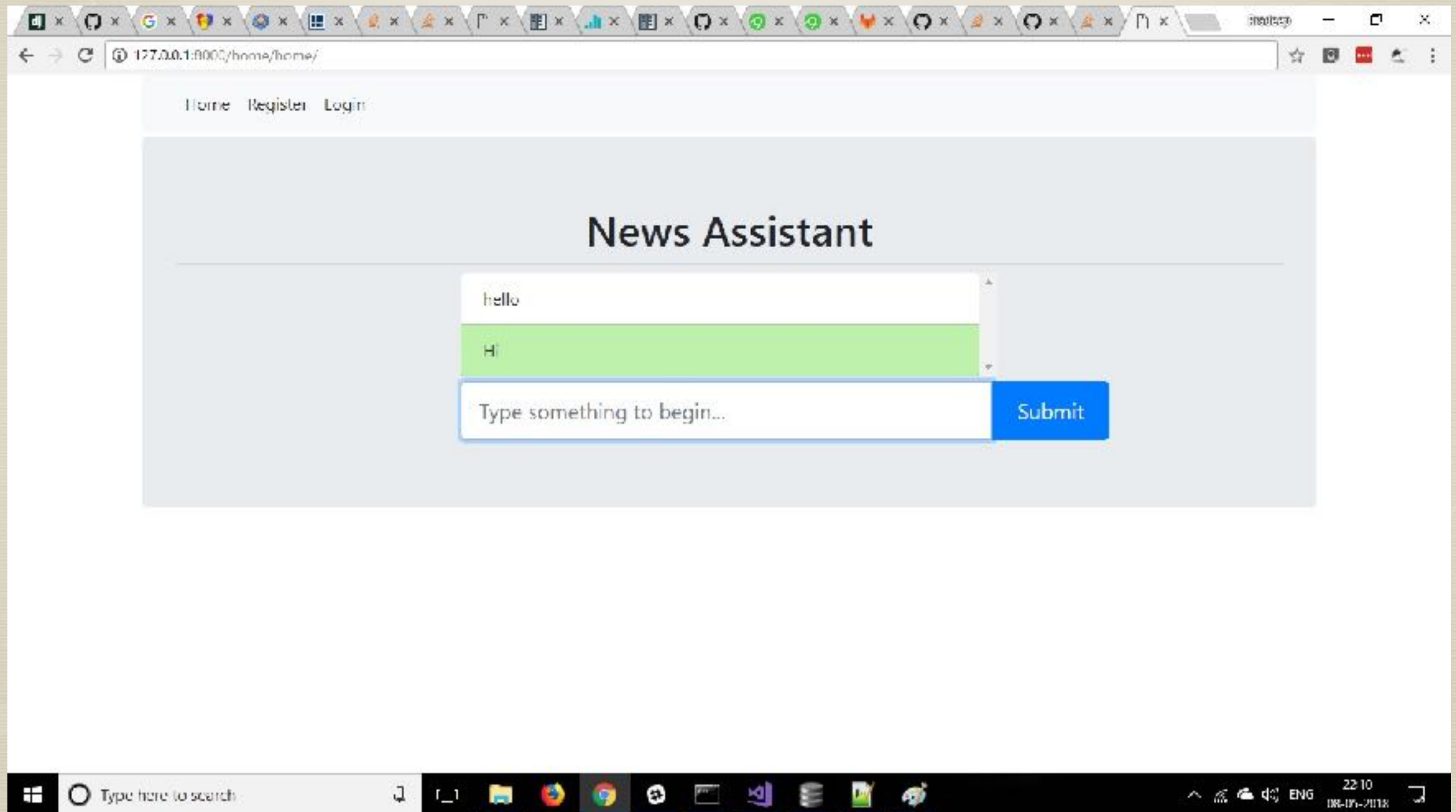
Username
Enter Username

Password
Enter Password

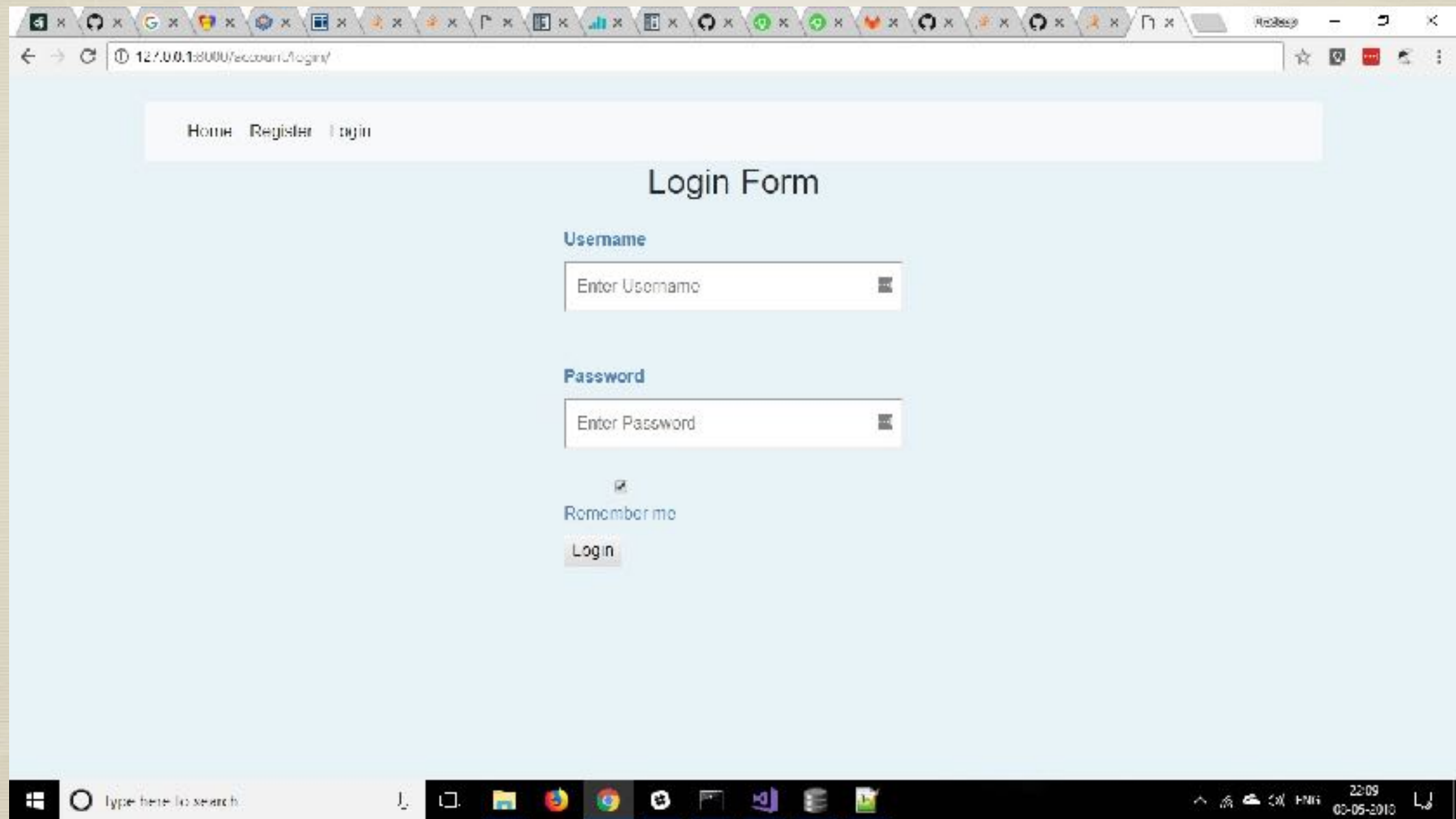
☐ Remember me

Login

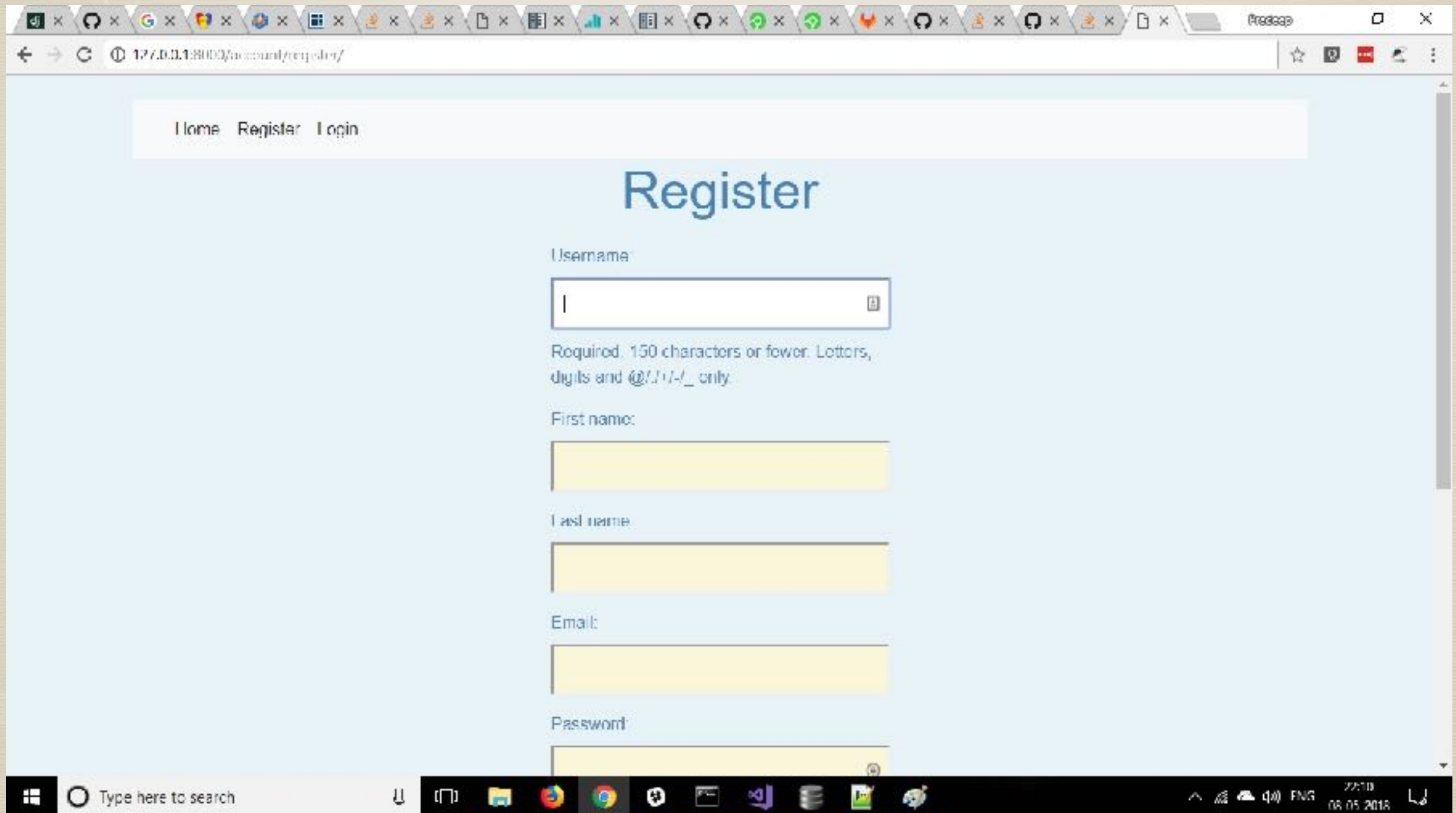
Django web UI with chatbot



Django web UI with chatbot



Django web UI with chatbot

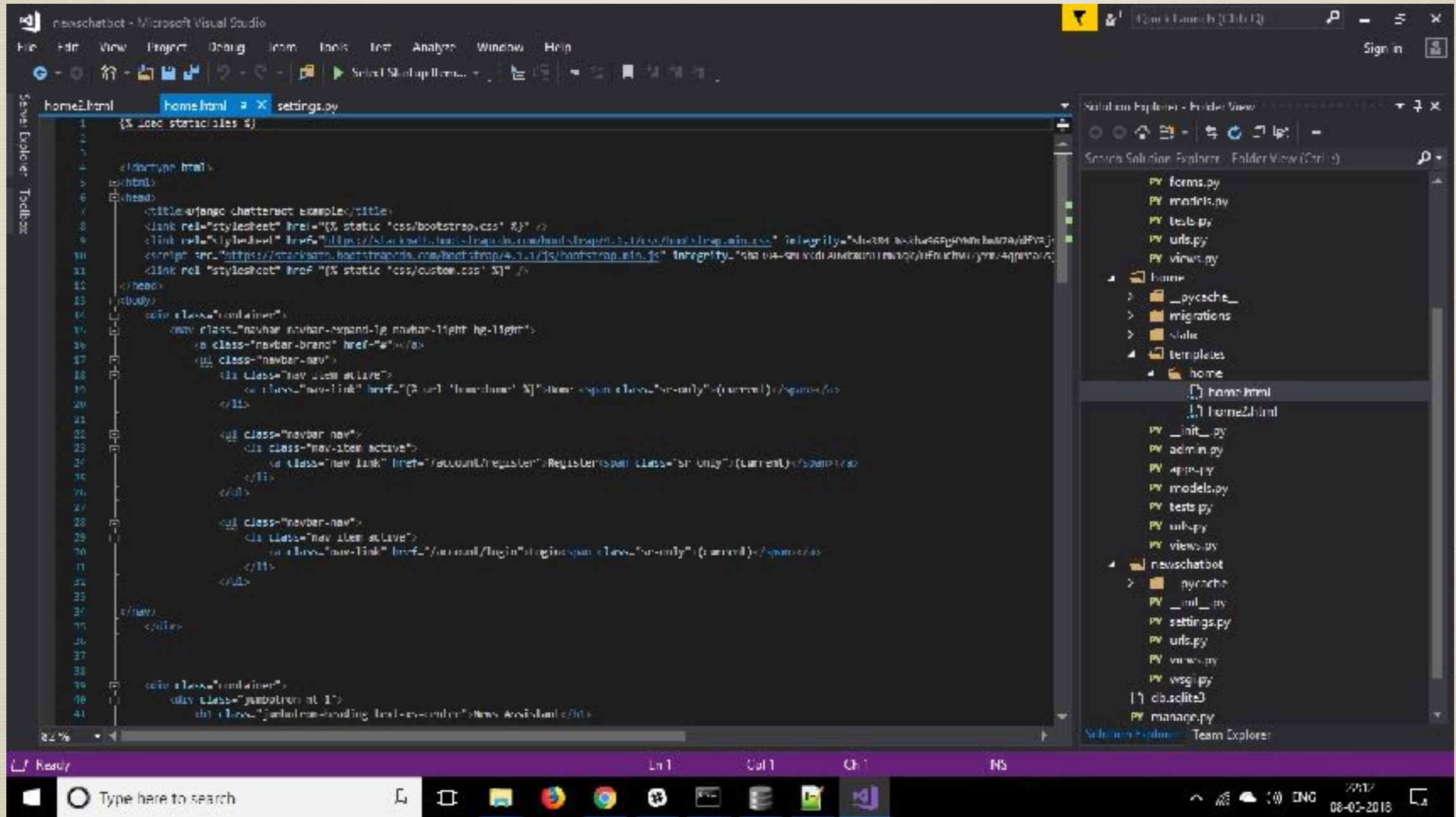


The screenshot shows a web browser window with the URL `177.0.0.1:8000/account/register/`. The page has a light blue background and a white navigation bar at the top with links for [Home](#), [Register](#), and [Login](#). The main heading is

Register

. Below the heading, there are five form fields:
1. **Username:** A text input field with a small icon on the right. Below it, a message reads: "Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only."
2. **First name:** A yellow text input field.
3. **Last name:** A yellow text input field.
4. **Email:** A yellow text input field.
5. **Password:** A yellow text input field with a small icon on the right.
At the bottom of the browser window, the Windows taskbar is visible, showing the search bar and several application icons. The system clock in the bottom right corner indicates the time is 22:10 on 05.05.2018.

Django code snippet



Django database

DD Browser for SQLite - C:\Users\pradeep\Desktop(newschatbot)\newschatbot\db.sqlite3

File Edit View Help

New Database Open Database Write Changes Revert Changes

Database Structure Browser Data Edit Pragma Execute SQL

Tables: auth_user

id password last_login is_superuser username first_name email is_staff is_active

1	1	pbkdf2_sha256...	2018-05-07 2...	1	pradeep		pradeep@n...	1	1
2	2	pbkdf2_sha256...	NULL	0	kinan	kinan	kinan@n...	0	1
3	3	pbkdf2_sha256...	NULL	0	chandan	chandan	chandan@n...	0	1
4	4	pbkdf2_sha256...	NULL	0	wahidur	wahidur	wahidur@n...	0	1
5	5	pbkdf2_sha256...	NULL	0	guannan	guannan	guannan@n...	0	1
6	6	pbkdf2_sha256...	NULL	0	longyu	longyu	longyu@n...	0	1

Full Database Cell

Mode: Image Import Export Set as NULL

Type of data currently in cell: NULL
0 byte(s)

Apply

DB Schema

Name	Type
auth_user_groups_group_id_user_id	ForeignKey
auth_user_groups_user_id_group_id	ForeignKey
auth_user_groups_user_id_group_id_91093...	ForeignKey
auth_user_permissions_permission_id...	ForeignKey
auth_user_permissions_user_id_a95a...	ForeignKey
auth_user_permissions_user_id_permi...	ForeignKey
django_admin_log_content_type_id_48c...	ForeignKey
django_admin_log_user_id_c56fda6	ForeignKey
django_chatterbot_conversation_response...	ForeignKey
django_chatterbot_conversation_response...	ForeignKey
django_chatterbot_conversation_response...	ForeignKey
django_chatterbot_response_response_id...	ForeignKey
django_chatterbot_response_response_id...	ForeignKey

SQL Log Print DB Schema Results

UTF-8

Type here to search

20:17 03-05-2018

Future Enhancement

- * Update UX for usability
- * Add more interaction using JS scripts
- * Update HTML and CSS
- * Implement push notifications
- * Add Machine learning models for better NLU and responses

Future Enhancements

- * Implement storage procedures and user roles
- * Implement Merge for data transfer for deferent servers
- * Update the Database Architecture
- * Implement proper test cases for testing

Github wiki

* [https://github.com/paceuniversityCS691TeamB/
NewsAssistant/wiki](https://github.com/paceuniversityCS691TeamB/NewsAssistant/wiki)

Thank you