Introduction to Web Science

Assignment 4

PD Dr. Matthias Thimm
thimm@uni-koblenz.de

Ipek Baris Schlicht
ibaris@uni-koblenz.de

Kenneth Skiba
kennethskiba@uni-koblenz.de

Institute of Web Science and Technologies
Department of Computer Science
University of Koblenz-Landau

Submission until: 08.12.2020, CEST 23:59

1 XML 26 Points

Consider the following XML document:

```
?xml version="1.1" encoding="UTF-8" ?>
<book>
    <author><publisher>Macmillan</author></publisher>
    <title>Alice Adventures in Wonderland</title>
    <year>1865</year>
</book>
```

1.1

Solution:
The above xml document is not valid for the below mentioned reasons:

1. In the declaration line, '<' is missing. The correct way is
   <?xml version="1.1" encoding="UTF-8" ?>
2. The element tags for author and publisher are not closed in order. The correct order should be
   <author><publisher>Macmillan</publisher></author>

1.2 Well formed XML

Consider the following information about the best-selling individual books[1]:

Name	Author(s)	Original language	First published	Approximate sales	Genre
The Communist Manifesto	Karl Marx and Friedrich Engels	German	1848	> 500 million	Political philosophy
A Tale of Two Cities	Charles Dickens	English	1859	200 million	Historical fiction
Harry Potter and the Philopher's Stone	J. K. Rowling	English	1997	120 million	Fantasy, mystery
The Lion, the Witch and the Wardrobe	C.S. Lewis	English	1950	85 million	Fantasy
The Adventures of Pinocchio	Carlo Collodi	Italian	1881	> 80 million	Fantasy

Your task is to encode the table above into a well formed XML document.

[1] Based on https://en.wikipedia.org/wiki/List_of_best-selling_books

2 HTML and CSS 27 points

The aim of this task to learn and practice the basics of HTML and CSS. You are required to implement a website that presents your resume. The layout of the website as seen Figure 1. The most left section is **Experiences**, the middle is the section for **Education** and the most right section is **Skills** followed by **References**.

Perform the following steps:

1. Create a HTML file called `index.html`, and a CSS file called `style_display`. In CSS, define the items referring the sections and header and use only `inline-block` for positioning the items.
2. Create a CSS file called `style_float.css`. In CSS, define the items referring the sections and header and use only `float` for positioning. Do not alter the contents in `index.html`.
3. Create a HTML file called `index_table.html`, and a CSS file called `style_table.css`. This time, you are supposed to implement the layout by using tables in HTML. However, for styling the cells and the properties of the table, you need to edit the `style_table.css`.

Three steps that you will implement should output same layout and content. However, small margins between the results could be neglected. The background color of the sections is `#ccc`.

For this task, don't use frameworks such as bootstrap. You are only allowed to use standard HTML and CSS items.

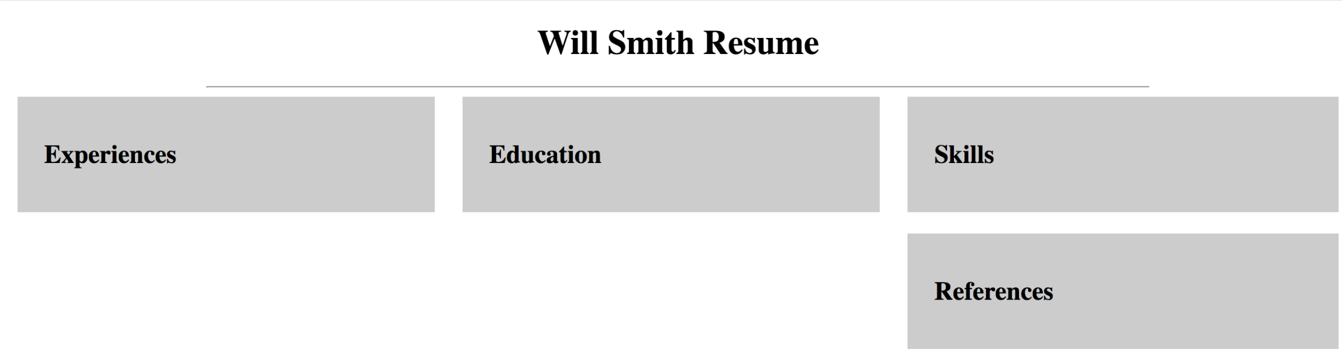
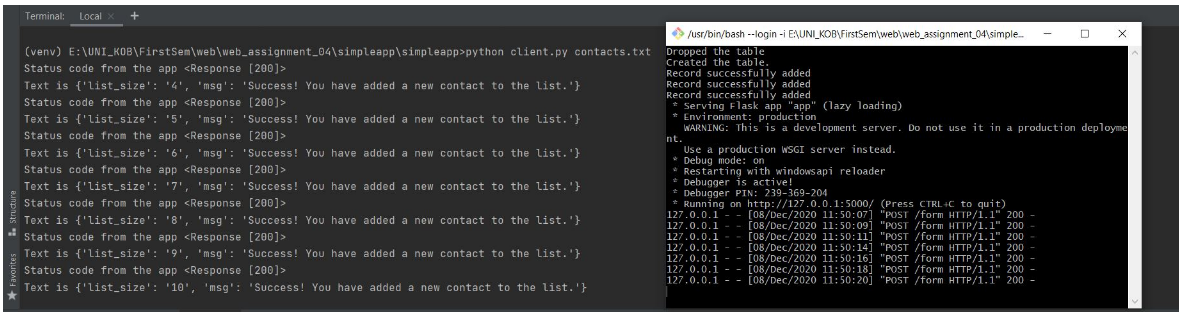


Figure 1: The website page you will implement

3 Dynamic Web Content 27 points

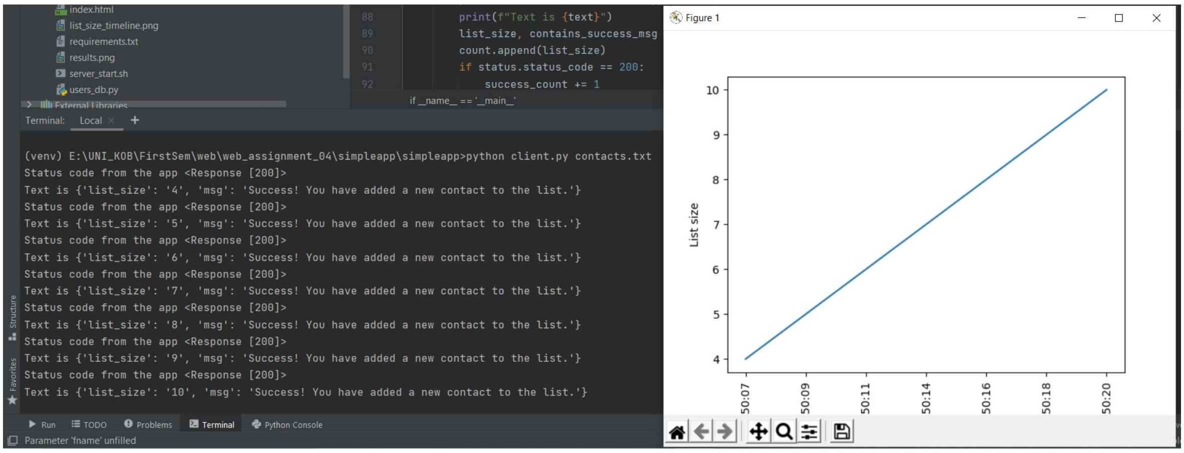


```
(venv) E:\UNI_KOB\FirstSem\web\web_assignment_04\simpleapp\simpleapp>python client.py contacts.txt
Status code from the app <Response [200]>
Text is {'list_size': '4', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '5', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '6', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '7', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '8', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '9', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '10', 'msg': 'Success! You have added a new contact to the list.'}
```

/usr/bin/bash --login -i E:\UNI_KOB\FirstSem\web\web_assignment_04\simple...

```
Dropped the table
Created the table.
Record successfully added
Record successfully added
Record successfully added
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployme
nt.
   Use a production WSGI server instead.
 * Debug mode: on
 * Restarting with windowsapi reloader
 * Debugger is active!
 * Debugger PIN: 239-369-204
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [08/Dec/2020 11:50:07] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:50:09] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:50:11] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:50:14] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:50:16] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:50:18] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:50:20] "POST /form HTTP/1.1" 200 -
```

Figure 2: Output : Contacts.txt



```
        print(f"Text is {text}")
        list_size, contains_success_msg
        count.append(list_size)
        if status.status_code == 200:
            success_count += 1
```

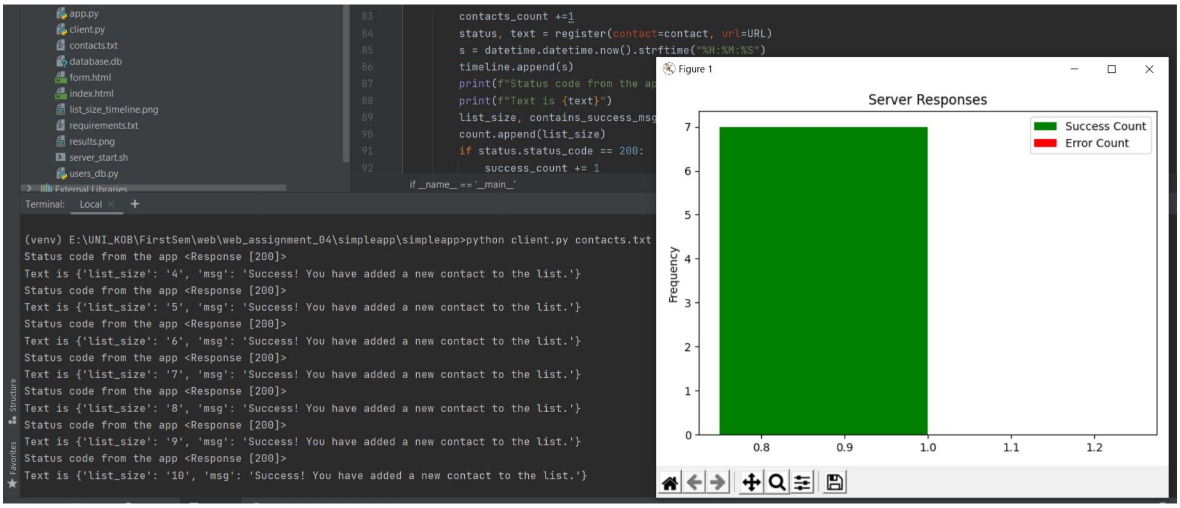
if __name__ == '__main__'

```
(venv) E:\UNI_KOB\FirstSem\web\web_assignment_04\simpleapp\simpleapp>python client.py contacts.txt
Status code from the app <Response [200]>
Text is {'list_size': '4', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '5', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '6', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '7', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '8', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '9', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '10', 'msg': 'Success! You have added a new contact to the list.'}
```

Figure 3: Output : Timeline Plot-Contacts.txt

1. The execute() for the query "SELECT * FROM clients WHERE (name='user[0]') AND (email='user[1]')" in user.db can execute only a single execute statement.

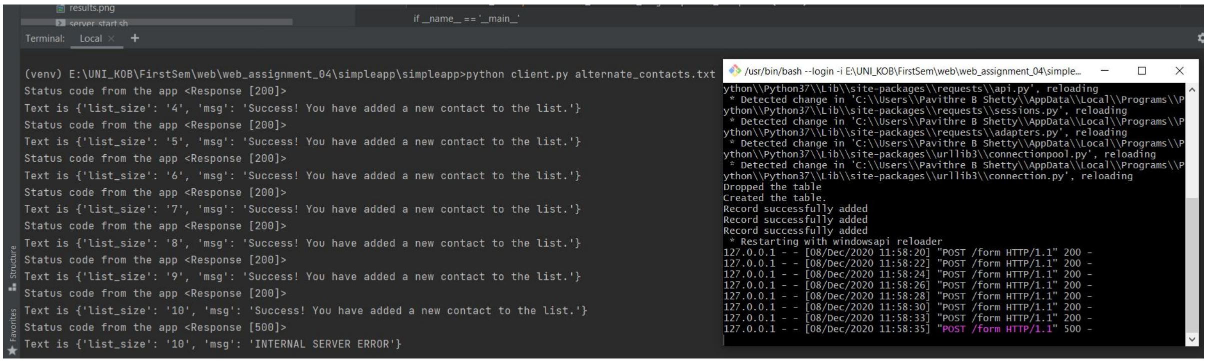
2. The input "Ipek, dummy@gmail.com'); DROP TABLE clients; –; INSERT INTO clients VALUES (Kenneth, dummy2@mail.com" has multiple lines which cannot be executed with excecute() for the above query. Thats why we get the warning as "You can execute only one statement at a time"



```
        contacts_count +=1
        status, text = register(contact=contact, url=URL)
        s = datetime.datetime.now().strftime("%H:%M:%S")
        timeline.append(s)
        print(f"Status code from the ap
        print(f"Text is {text}")
        list_size, contains_success_msg
        count.append(list_size)
        if status.status_code == 200:
            success_count += 1
if __name__ == '__main__'
```

```
(venv) E:\UNI_KOB\FirstSem\web\web_assignment_04\simpleapp\simpleapp>python client.py contacts.txt
Status code from the app <Response [200]>
Text is {'list_size': '4', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '5', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '6', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '7', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '8', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '9', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '10', 'msg': 'Success! You have added a new contact to the list.'}
```

Figure 4: Output : Bar Plot-Contact.txt

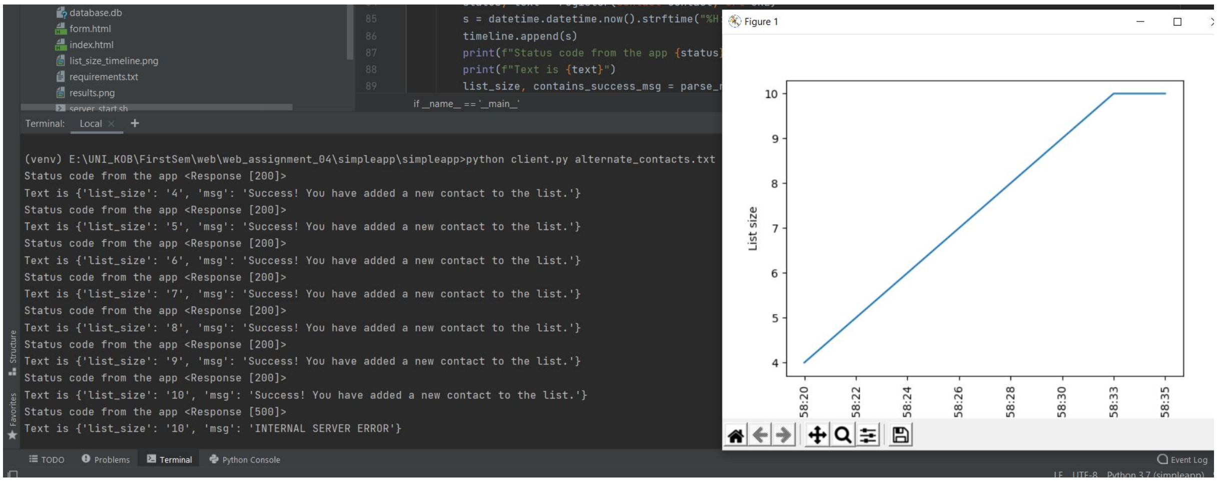

results.png
server_start.sh
Terminal: Local +
/usr/bin/bash --login -i E:\UNI_KOB\FirstSem\web\web_assignment_04\simple...
Structure
Favorites

```
if __name__ == '__main__'
```

```
(venv) E:\UNI_KOB\FirstSem\web\web_assignment_04\simpleapp\simpleapp>python client.py alternate_contacts.txt
Status code from the app <Response [200]>
Text is {'list_size': '4', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '5', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '6', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '7', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '8', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '9', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '10', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [500]>
Text is {'list_size': '10', 'msg': 'INTERNAL SERVER ERROR'}
```

```
ython\\Python37\\Lib\\site-packages\\requests\\api.py', reloading
 * Detected change in 'C:\\Users\\Pavithre B Shetty\\AppData\\Local\\Programs\\P
ython\\Python37\\Lib\\site-packages\\requests\\sessions.py', reloading
 * Detected change in 'C:\\Users\\Pavithre B Shetty\\AppData\\Local\\Programs\\P
ython\\Python37\\Lib\\site-packages\\requests\\adapters.py', reloading
 * Detected change in 'C:\\Users\\Pavithre B Shetty\\AppData\\Local\\Programs\\P
ython\\Python37\\Lib\\site-packages\\urllib3\\connectionpool.py', reloading
 * Detected change in 'C:\\Users\\Pavithre B Shetty\\AppData\\Local\\Programs\\P
ython\\Python37\\Lib\\site-packages\\urllib3\\connection.py', reloading
Dropped the table
Created the table.
Record successfully added
Record successfully added
Record successfully added
 * Restarting with windowsapi reloader
127.0.0.1 - - [08/Dec/2020 11:58:20] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:22] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:24] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:26] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:28] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:30] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:33] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:35] "POST /form HTTP/1.1" 500 -
```

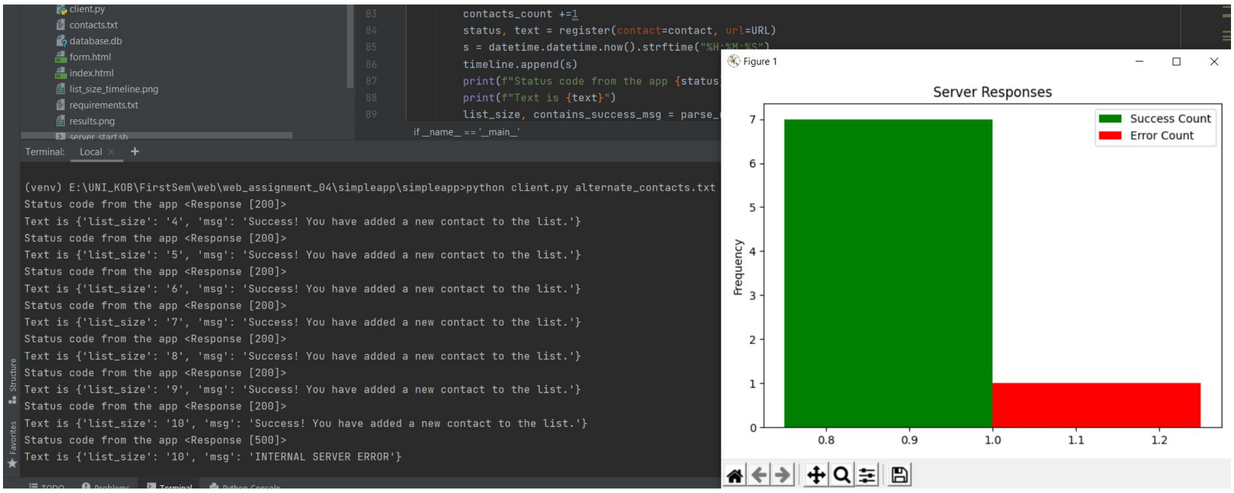
Figure 5: Output : AlternateContacts.txt



```
        s = datetime.datetime.now().strftime("%H
        timeline.append(s)
        print(f"Status code from the app {status
        print(f"Text is {text}")
        list_size, contains_success_msg = parse_
if __name__ == '__main__'
```

```
(venv) E:\UNI_KOB\FirstSem\web\web_assignment_04\simpleapp\simpleapp>python client.py alternate_contacts.txt
Status code from the app <Response [200]>
Text is {'list_size': '4', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '5', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '6', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '7', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '8', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '9', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '10', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [500]>
Text is {'list_size': '10', 'msg': 'INTERNAL SERVER ERROR'}
```

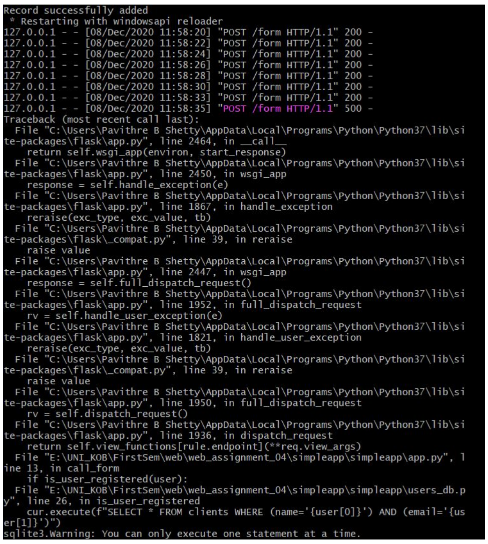
Figure 6: Output : Timeline Plot AlternateContacts.txt



```
            contacts_count +=1
            status, text = register(contact=contact, url=URL)
            s = datetime.datetime.now().strftime("%H:%M:%S")
            timeline.append(s)
            print(f"Status code from the app {status
            print(f"Text is {text}")
            list_size, contains_success_msg = parse_
    if __name__ == '__main__'
```

```
(venv) E:\UNI_KOB\FirstSem\web\web_assignment_04\simpleapp\simpleapp>python client.py alternate_contacts.txt
Status code from the app <Response [200]>
Text is {'list_size': '4', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '5', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '6', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '7', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '8', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '9', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [200]>
Text is {'list_size': '10', 'msg': 'Success! You have added a new contact to the list.'}
Status code from the app <Response [500]>
Text is {'list_size': '10', 'msg': 'INTERNAL SERVER ERROR'}
```

Figure 7: Output : Bar Plot - AlternateContacts.txt



```
Record successfully added
 * Restarting with windowsapi reloader
127.0.0.1 - - [08/Dec/2020 11:58:20] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:22] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:24] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:26] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:28] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:30] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:33] "POST /form HTTP/1.1" 200 -
127.0.0.1 - - [08/Dec/2020 11:58:35] "POST /form HTTP/1.1" 500 -
Traceback (most recent call last):
  File "C:\Users\Pavithre B Shetty\AppData\Local\Programs\Python\Python37\lib\si
te-packages\flask\app.py", line 2464, in __call__
    return self.wsgi_app(environ, start_response)
  File "C:\Users\Pavithre B Shetty\AppData\Local\Programs\Python\Python37\lib\si
te-packages\flask\app.py", line 2450, in wsgi_app
    response = self.handle_exception(e)
  File "C:\Users\Pavithre B Shetty\AppData\Local\Programs\Python\Python37\lib\si
te-packages\flask\app.py", line 1867, in handle_exception
    reraise(exc_type, exc_value, tb)
  File "C:\Users\Pavithre B Shetty\AppData\Local\Programs\Python\Python37\lib\si
te-packages\flask\_compat.py", line 39, in reraise
    raise value
  File "C:\Users\Pavithre B Shetty\AppData\Local\Programs\Python\Python37\lib\si
te-packages\flask\app.py", line 2447, in wsgi_app
    response = self.full_dispatch_request()
  File "C:\Users\Pavithre B Shetty\AppData\Local\Programs\Python\Python37\lib\si
te-packages\flask\app.py", line 1952, in full_dispatch_request
    rv = self.handle_user_exception(e)
  File "C:\Users\Pavithre B Shetty\AppData\Local\Programs\Python\Python37\lib\si
te-packages\flask\app.py", line 1821, in handle_user_exception
    reraise(exc_type, exc_value, tb)
  File "C:\Users\Pavithre B Shetty\AppData\Local\Programs\Python\Python37\lib\si
te-packages\flask\_compat.py", line 39, in reraise
    raise value
  File "C:\Users\Pavithre B Shetty\AppData\Local\Programs\Python\Python37\lib\si
te-packages\flask\app.py", line 1950, in full_dispatch_request
    rv = self.dispatch_request()
  File "C:\Users\Pavithre B Shetty\AppData\Local\Programs\Python\Python37\lib\si
te-packages\flask\app.py", line 1936, in dispatch_request
    return self.view_functions[rule.endpoint](**req.view_args)
  File "E:\UNI_KOB\FirstSem\web\web_assignment_04\simpleapp\simpleapp\app.py", l
ine 13, in call_form
    if is_user_registered(user):
  File "E:\UNI_KOB\FirstSem\web\web_assignment_04\simpleapp\simpleapp\users_db.p
y", line 26, in is_user_registered
    cur.execute(f"SELECT * FROM clients WHERE (name='{user[0]}') AND (email='{us
er[1]}')")
sqlite3.Warning: You can only execute one statement at a time.
```

Figure 8: Output : Server Error AlternateContacts.txt