Automated Essay Grader

Kartavya, Aditya, Shreyansh

Nov 27, 2019

BACK END:

Automated Essay Grader is essentially a usable wrapper for using machine learning models based on Essay grading. Our project runs on `Starlette server` hosted on `Heroku cloud` with a noSQL database hosted on `Firebase`. Using `MVC architecture` we have created a web interface powered by `Bootstrap 4.0`. Our project also has a fully featured `Android` interface.

To check a live web demo Check out

CHAPTER

ONE

PROJECT ENVIRONMENT SETUP

To start the project you first need to install all dependencies:

```
>>> pip install requirements.txt
```

After we have all the requirements set up, we will now create an environment

```
>>> conda env create -f Softlab.yml
```

After the environment is set up you can initialize the environment somehow

```
>>> conda activate Softlab
```

Yes that was it!! Now simply start the system by typing

```
>>> python3 main.py
```

CHAPTER

TWO

DEPLOYMENT STEPS USING DOCKER

1) You can simply build the the image file by running following in the directory where Dockerfile is present

```
>>> docker build --tag essay-grader .
```

```
>>> docker run --name essay-grader -p 8000:8000 essay-grader
```

CHAPTER

THREE

CHECKOUT SOME CODE NOW

3.1 MVC architecture connections

This is a short description of the code belonging to the plumbing between frontend and backend

The code in this file is mainly plumbing between the frontend and ML backend. It consists of routes which map to the browser url. We use the MVC architechture where we code the controller and modals. View manager in our code is Jinga2vec

async main.**contrbPage**(*request*)

This function gives people the option to contribute the essay set and make the model better.

```
>>> Essay: Hi, this is the festival of diwali and I like to enjoy it with my␣
↪friends and pet cow
```

```
>>> Score: 30/100
```

async main.**evaluate**(*request*)

The function evaluate: stores the user input essay and corresponding score We have two calls of function evaluate with the distinction of the routes with totally different route.

1) `@app.route('/evaluate',methods=["POST"])` This sends the essay to the server and recieves the score after some calculation.

2) `@app.route('/contribute',methods=["POST"])` This is to store the essay and score pair provided by the user and appreciate the wholsomeness of it

async main.**evaluateFile**(*request*)

This funicton gives us a method to get the text out of file directly. Intead of writing text, it is better to upload text.

We see the function on a route `@app.route('/evaluateFile',methods=["POST"])`

async main.**firebase_login**(*request*)

This function deals with authentication in the code

Param request

Return type

Templating engine for Python3 rendering

```
>>> cool@cool.com and coolcool
```

async main.**firebase_register**(*request*)

This function registers us on the firebase platform.

It accepts user input from UI, ie. The email and password

The route for the same would be `@app.route('/registration',methods=["POST"])`

async `main.`**`getEssay`**(*request*)

This function getEssay is here to give us access to the cloud hosted url whenever. We can now check about donated essays till now. This controller operates on route `@app.route('/{prompt}')`

async `main.`**`login`**(*request*)

For the controller Login, we have two different routes

1) `@app.route('/')`, This is invoked when we have a redirect to and from the default start state or direct home href links

2) `@app.route('/register')` We get here when it's the login link vi This function gives us the ability of logging in if our username and pass match

async `main.`**`show_index`**(*request*)

The function is a controller. It is invoked when we call `/auth`

3.2 Automated essay grader model

Here we have a very short description of the module prediction part of the parent project Automated essay grader model.

This is an efcient wrapper around a automated essay grading system

`prediction.`**`getAvgFeatureVecs`**(*essays*, *model*, *num_features*)

Main function to generate the word vectors for word2vec model.

Parameters

- **`essays`** – Input is essays
- **`model`** – This input specifies the model to be used to generate Feature vec
- **`num_features`** – This is a metric of number of columns in the matrix cols

Return type Returns the essay feature vector

`prediction.`**`makeFeatureVec`**(*words*, *model*, *num_features*)

This function generates feature vectors for each dimension ie. plane

Parameters

- **`words`** – We pass the words that we needed
- **`model`** – Model to decide on what basis police have been arrested

Return type `featureVec` is the result of the function

`prediction.`**`predict`**(*essay*)

This is the function which calculates the score given an essay

Parameters **`essay`** – This input is the essay that we got from the fronend.

Return type We are returning the y_pred values

`prediction.`**`tokenizeEssay`**(*essay*)

This funciton tokenizes the text provided as input in the from of essay

We remove `stopwords` from the essay

```
>>> This is america
>>> america
```

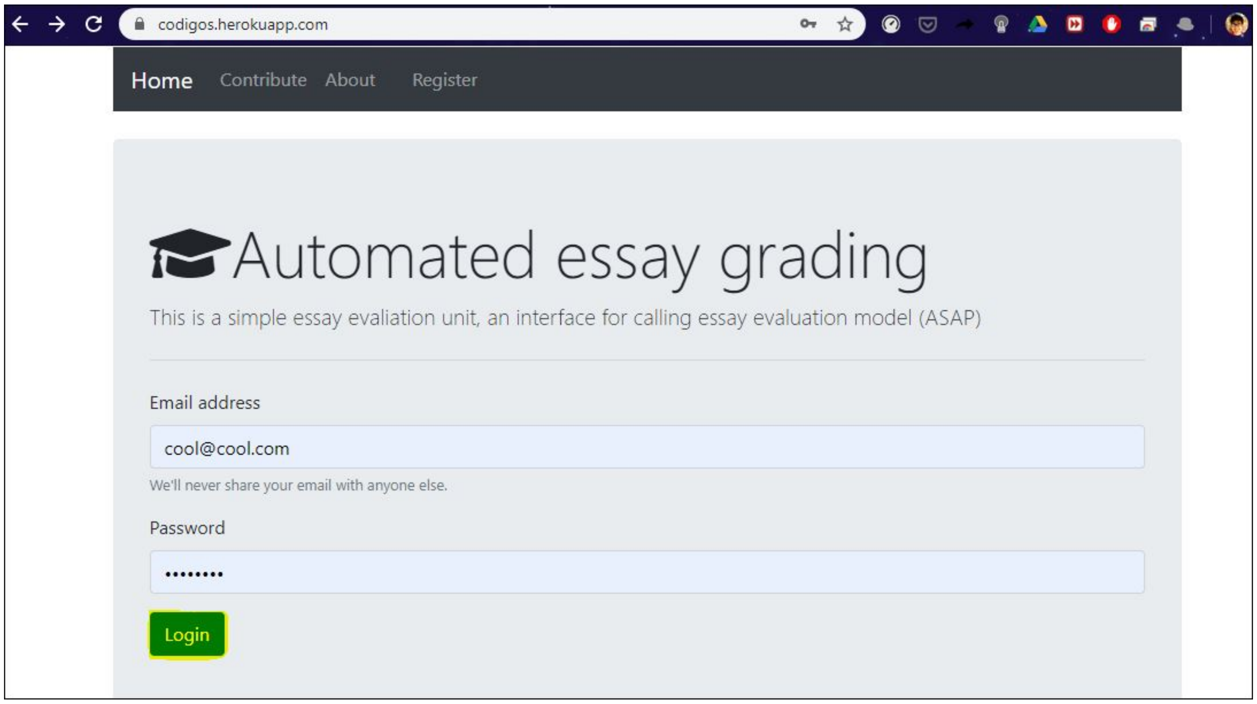
Parameters **essay** – Input in the form of text

Return type Tokenized words

3.3 Web views

This is the documentation for the web front end portal

`Login.html`


codigos.herokuapp.com

Automated essay grading

This is a simple essay evaliation unit, an interface for calling essay evaluation model (ASAP)

Email address

cool@cool.com

We'll never share your email with anyone else.

Password

••••••••

`Register.html`

`Index.html`

`Score.html`

`Contribute.html`

`Thanks.html`

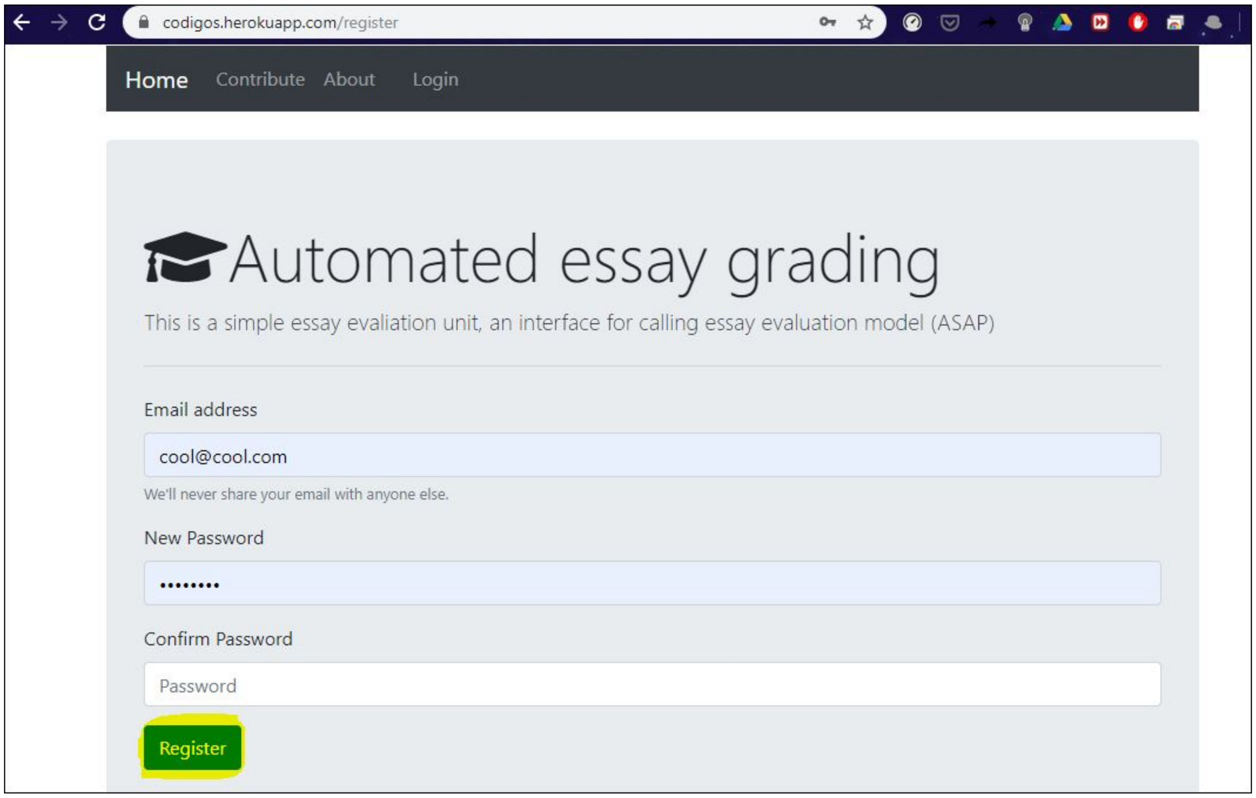
3.4 Android application

This is the documentation for android application

`Login`

`Register`

`Evaluate`



Automated essay grading

This is a simple essay evaliation unit, an interface for calling essay evaluation model (ASAP)

Email address

cool@cool.com

We'll never share your email with anyone else.

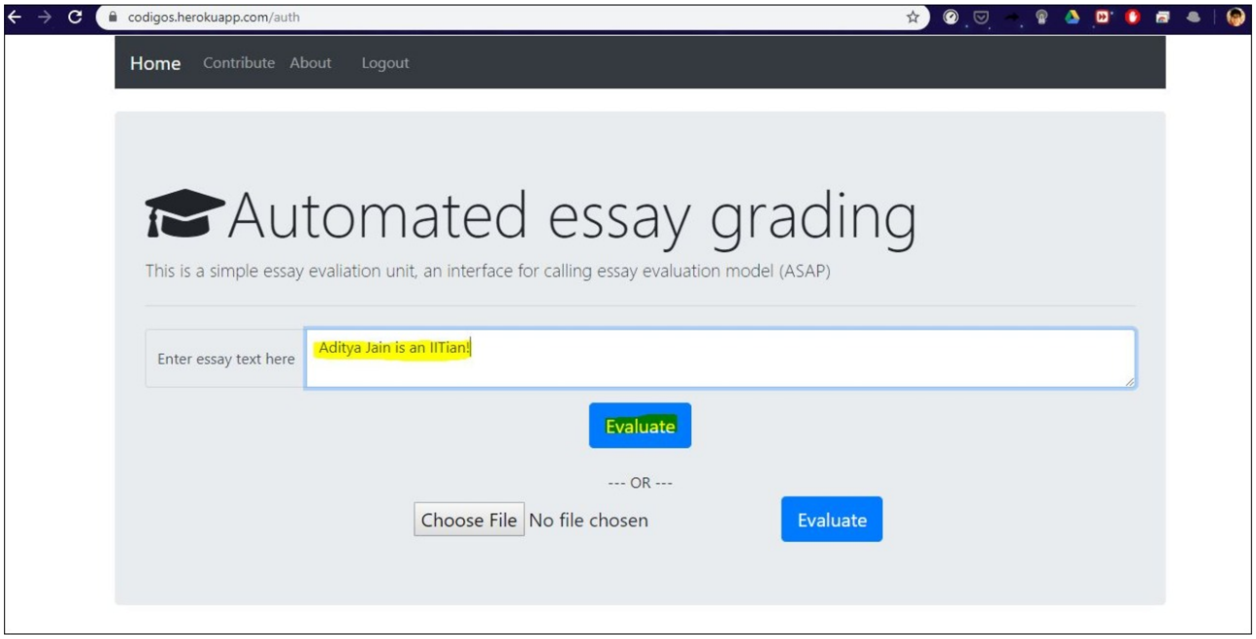
New Password

••••••••

Confirm Password

Password

Register



Automated essay grading

This is a simple essay evaliation unit, an interface for calling essay evaluation model (ASAP)

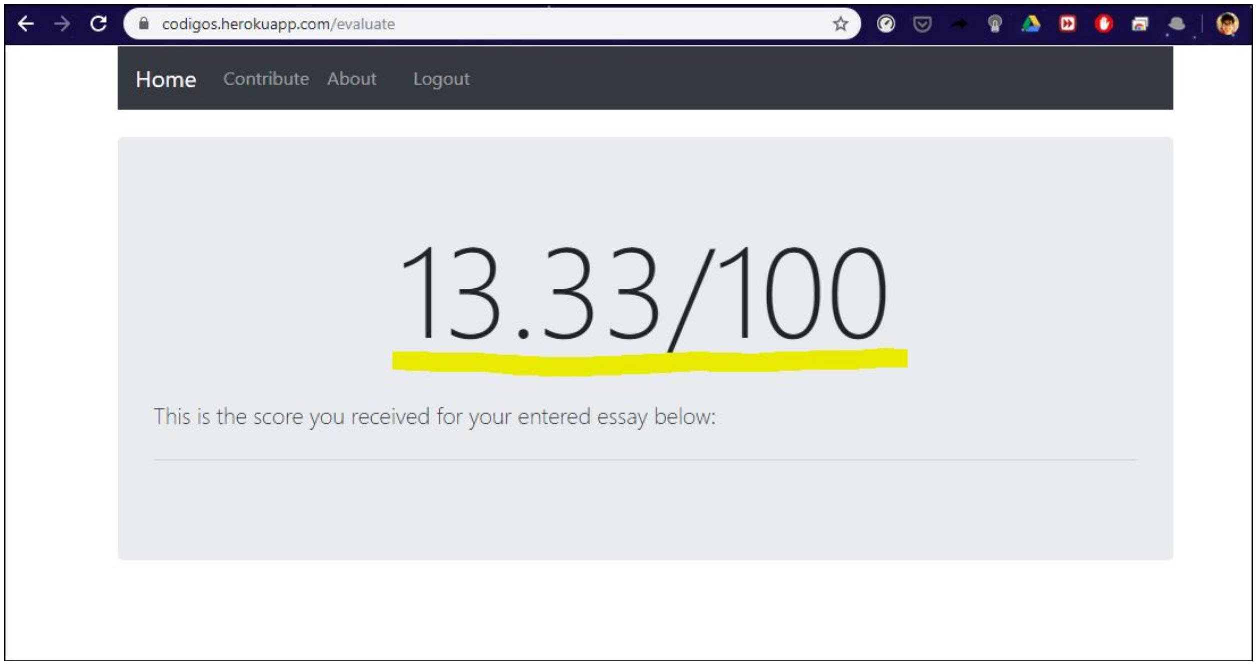
Enter essay text here: Aditya Jain is an IITian!

Evaluate

--- OR ---

Choose File No file chosen

Evaluate

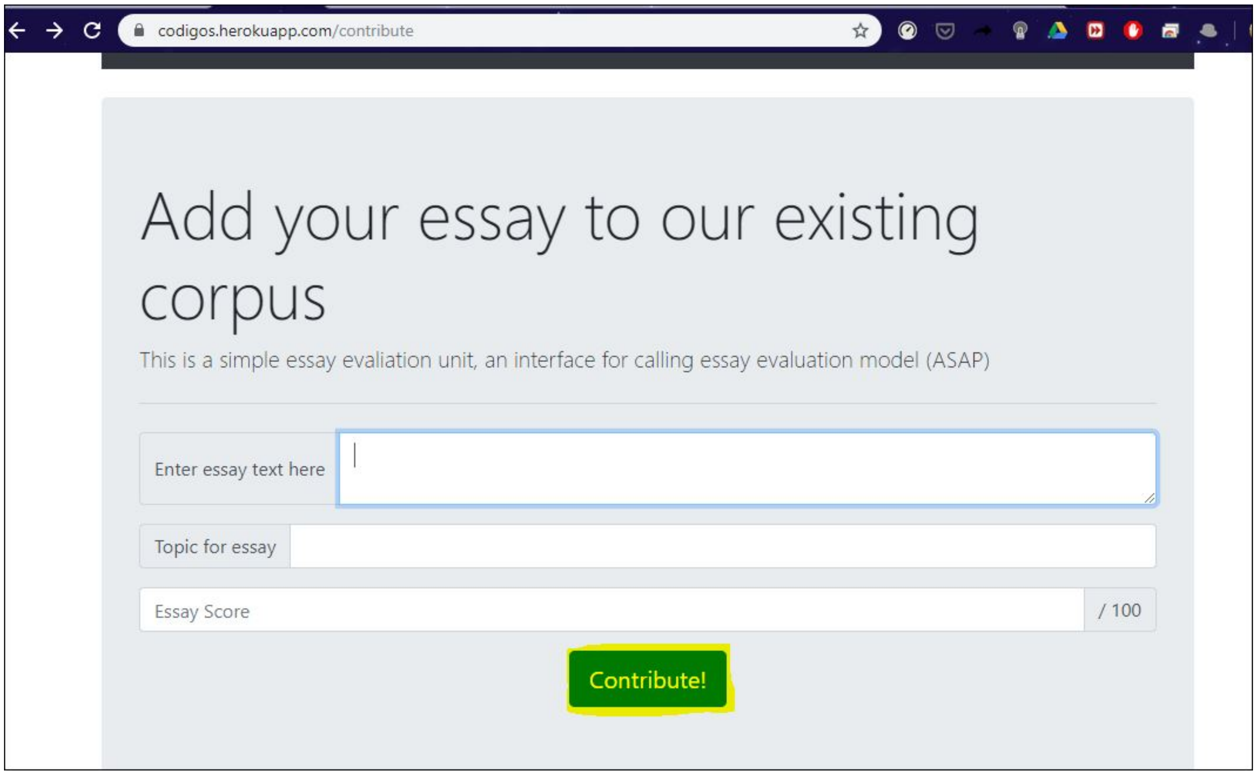


codigos.herokuapp.com/evaluate

Home Contribute About Logout

13.33/100

This is the score you received for your entered essay below:



codigos.herokuapp.com/contribute

Add your essay to our existing corpus

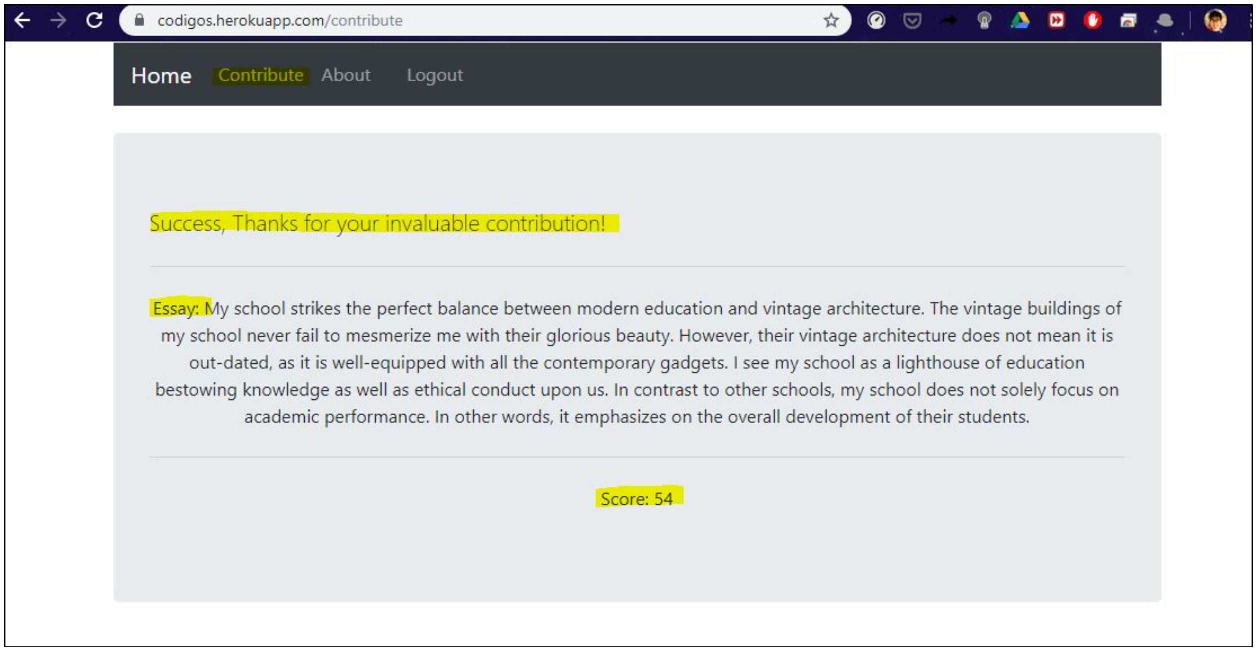
This is a simple essay evaliation unit, an interface for calling essay evaluation model (ASAP)

Enter essay text here

Topic for essay

Essay Score / 100

Contribute!



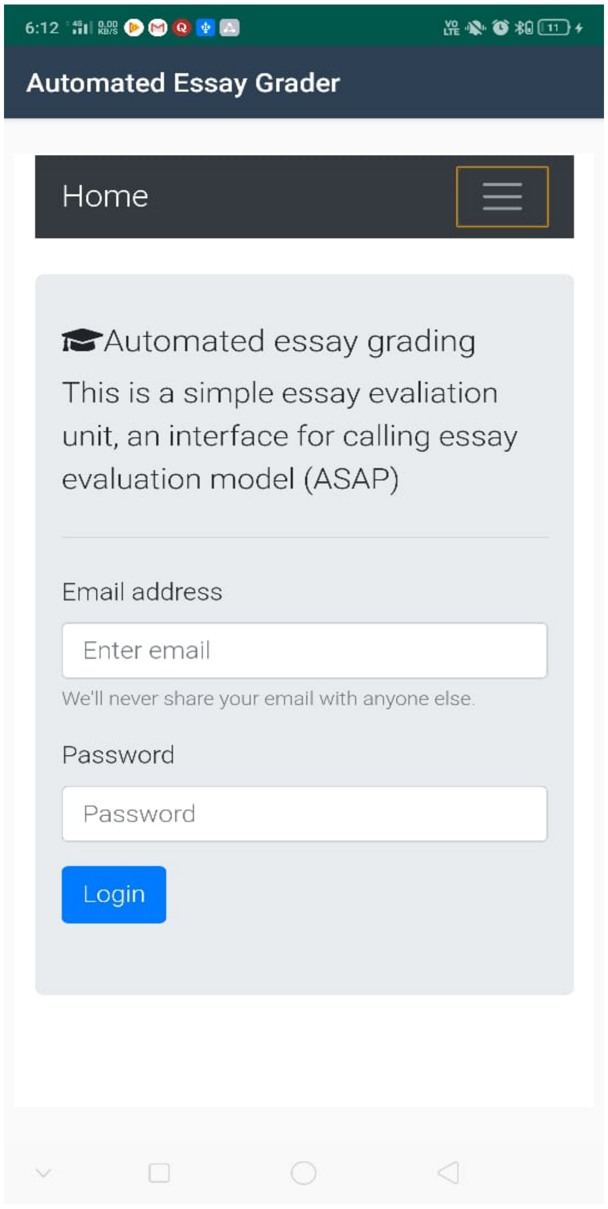
Success, Thanks for your invaluable contribution!

Essay: My school strikes the perfect balance between modern education and vintage architecture. The vintage buildings of my school never fail to mesmerize me with their glorious beauty. However, their vintage architecture does not mean it is out-dated, as it is well-equipped with all the contemporary gadgets. I see my school as a lighthouse of education bestowing knowledge as well as ethical conduct upon us. In contrast to other schools, my school does not solely focus on academic performance. In other words, it emphasizes on the overall development of their students.

Score: 54

`Score`

`Contribute`



Automated essay grading

This is a simple essay evaliation unit, an interface for calling essay evaluation model (ASAP)

Email address

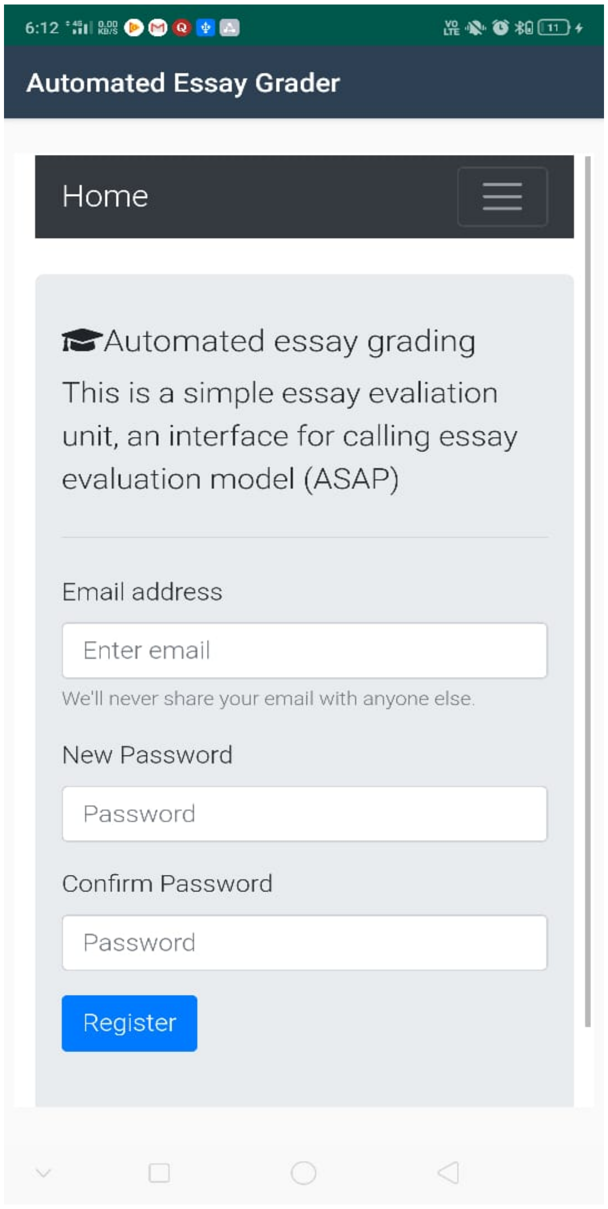
Enter email

We'll never share your email with anyone else.

Password

Password

Login


6:12

Automated Essay Grader

Home

Automated essay grading

This is a simple essay evaliation unit, an interface for calling essay evaluation model (ASAP)

Email address

Enter email

We'll never share your email with anyone else.

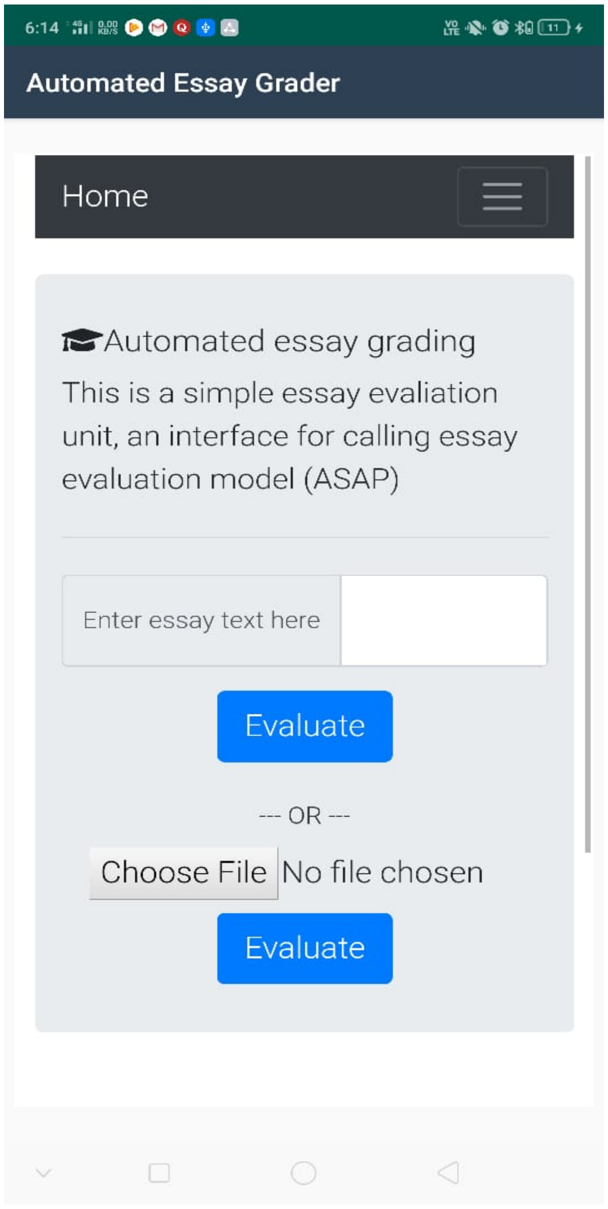
New Password

Password

Confirm Password

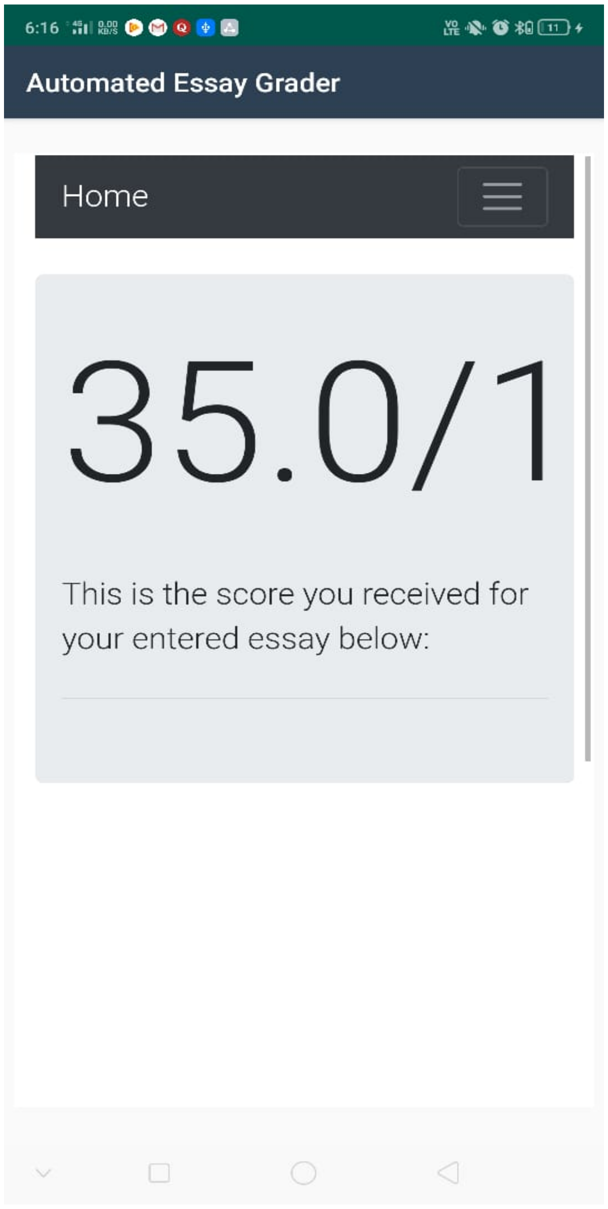
Password

Register



Automated essay grading

This is a simple essay evaliation unit, an interface for calling essay evaluation model (ASAP)

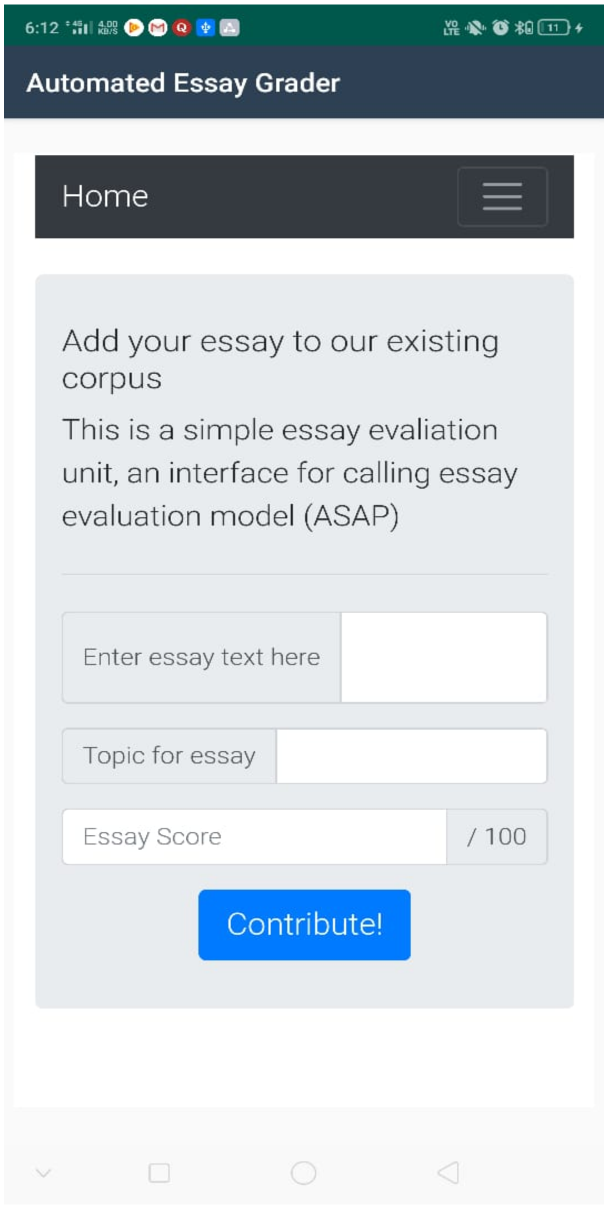


Automated Essay Grader

Home

35.0/1

This is the score you received for your entered essay below:



Automated Essay Grader

Home

Add your essay to our existing corpus

This is a simple essay evaliation unit, an interface for calling essay evaluation model (ASAP)

CHAPTER

FOUR

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

INDEX