

MINI PROJECT (2020-21)

Instagram Clone

MID-TERM REPORT



Institute of Engineering & Technology

Submitted by-

**Kritarth Sharma (181500335)
Priyanka Nigam (181500509)**

Supervised

By: - Mr. Amir Khan

Technical Trainer

Department of Computer Engineering & Applications

Contents

1. Abstract	3
2. Features	4
3. Objectives	35
4. Software and Hardware Requirements	37
5. Technology Used	38
6. References	52

Abstract

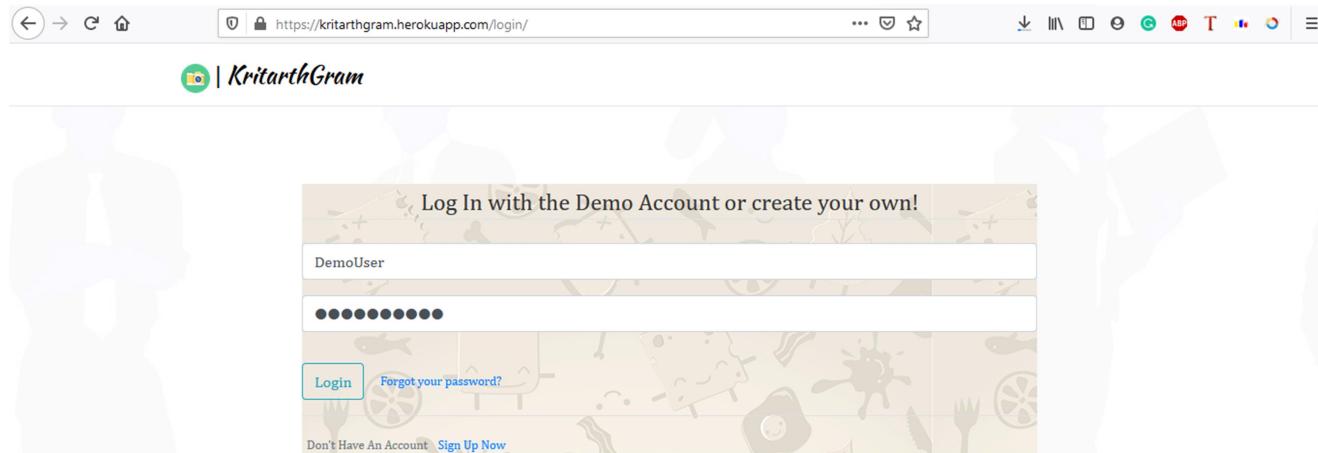
In this we will build a dynamic project similar to instagram that has innovative features. We are trying to add as many features as possible.

A mobile photo(and video) capturing and sharing service, has quickly emerged as a new medium in spotlight in the recent years. It provides users an instantaneous way to capture and share their life moments with friends through a series of (filter manipulated) pictures and videos.

Features

1) Login User

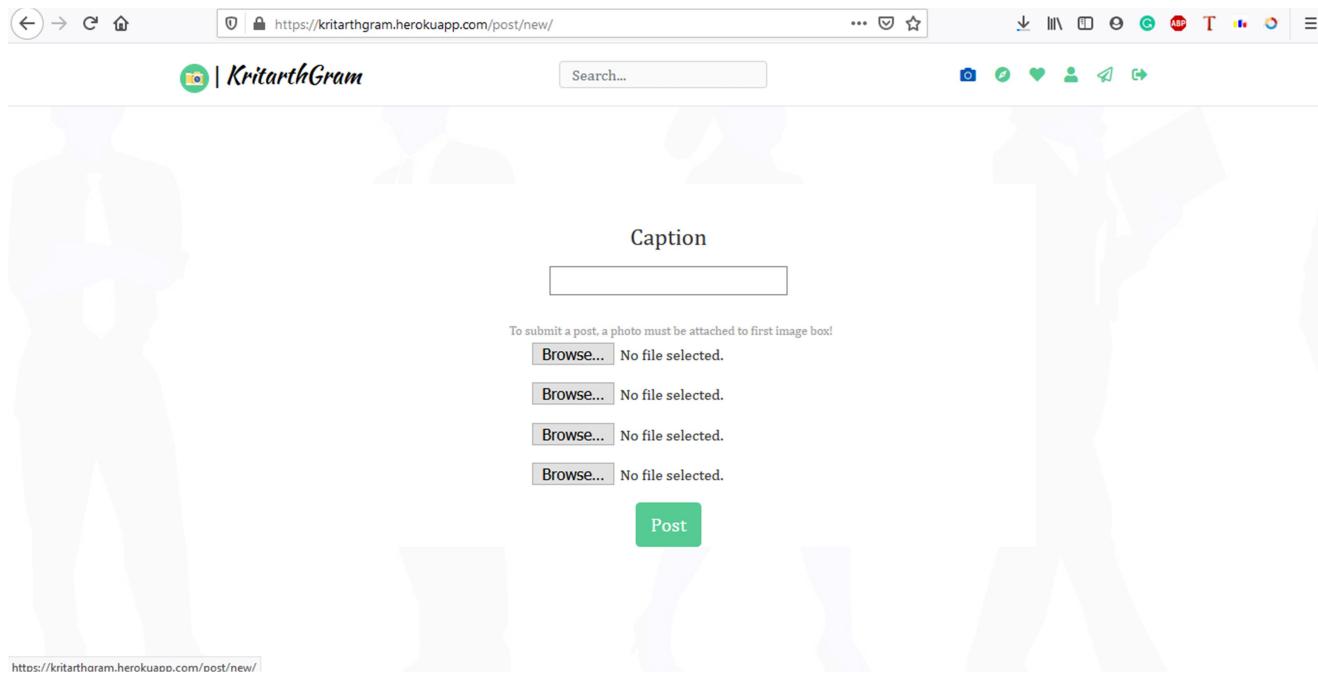
User can login we have used django authentication system for logging in users. Also a demo account is given for making it more useful for project purposes.



2) Adding Posts

Users can add post,(Upto 10 post at a time).These post are saved in amazon s3 bucket.We have used AWS here.

We can also add caption for image here



<https://kritarthram.herokuapp.com/post/new/>

3)Creating Account

Users can create account by filling their credentials like desired username,emailid,password.We have used regex for password validation and username validation



Join Today

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

default@email.com

- ❖ Your password can't be too similar to your other personal information.
- ❖ Your password must contain at least 8 characters.
- ❖ Your password can't be a commonly used password.
- ❖ Your password can't be entirely numeric.

Enter the same password as before, for verification.

[Sign Up](#)

Already Have An Account? [Sign In](#)

,If password doesn't matches desired regex sequence then a, error is thrown



Join Today

Please fill out this field.

default@email.com

•••••

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Enter the same password as before, for verification.

Sign Up

Already Have An Account? [Sign In](#)



Join Today

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

The two password fields didn't match.

Enter the same password as before, for verification.

Sign Up

Already Have An Account? [Sign In](#)



Join Today

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

This password is too short. It must contain at least 8 characters.

This password is too common.

Enter the same password as before, for verification.

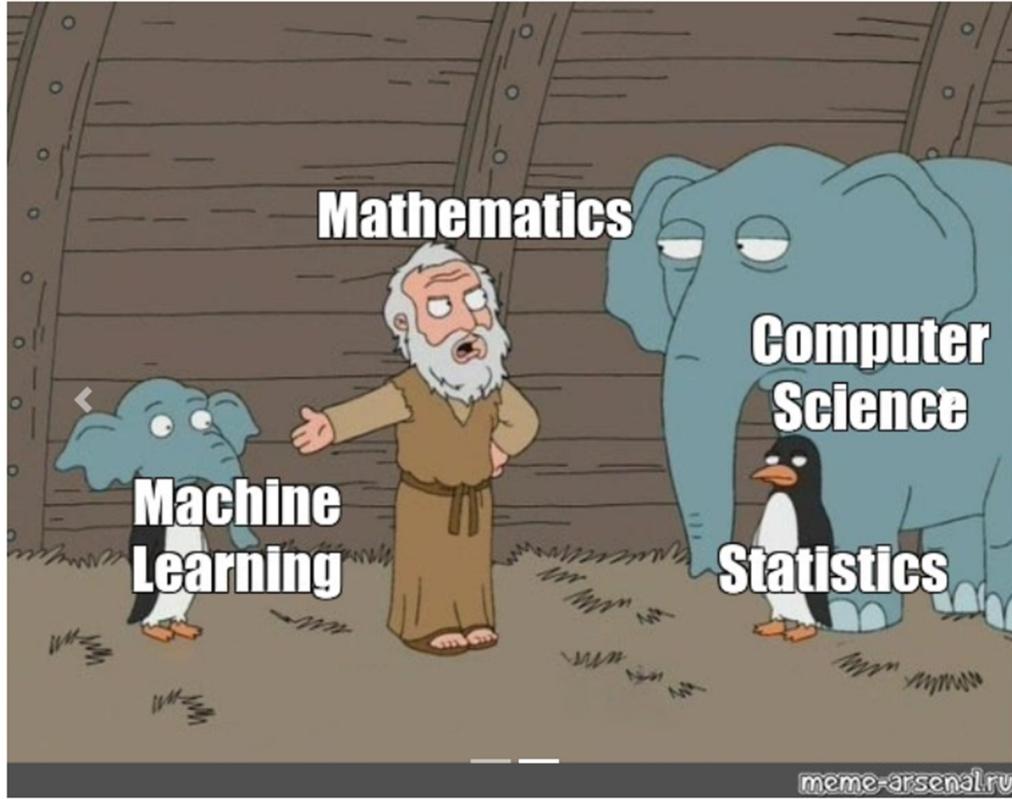
Already Have An Account? [Sign In](#)

3)Feed

Posts are displayed along with their like count and comments. One user can like the post only once, but a user can comment multiple times on the post.



DemoUser



meme-arsenal.ru



2 likes



heart comment more

2 likes

DemoUser This is an autogenerated post to help you see the sites functions right away

4 MONTHS AGO

DemoUser Nice NOW more

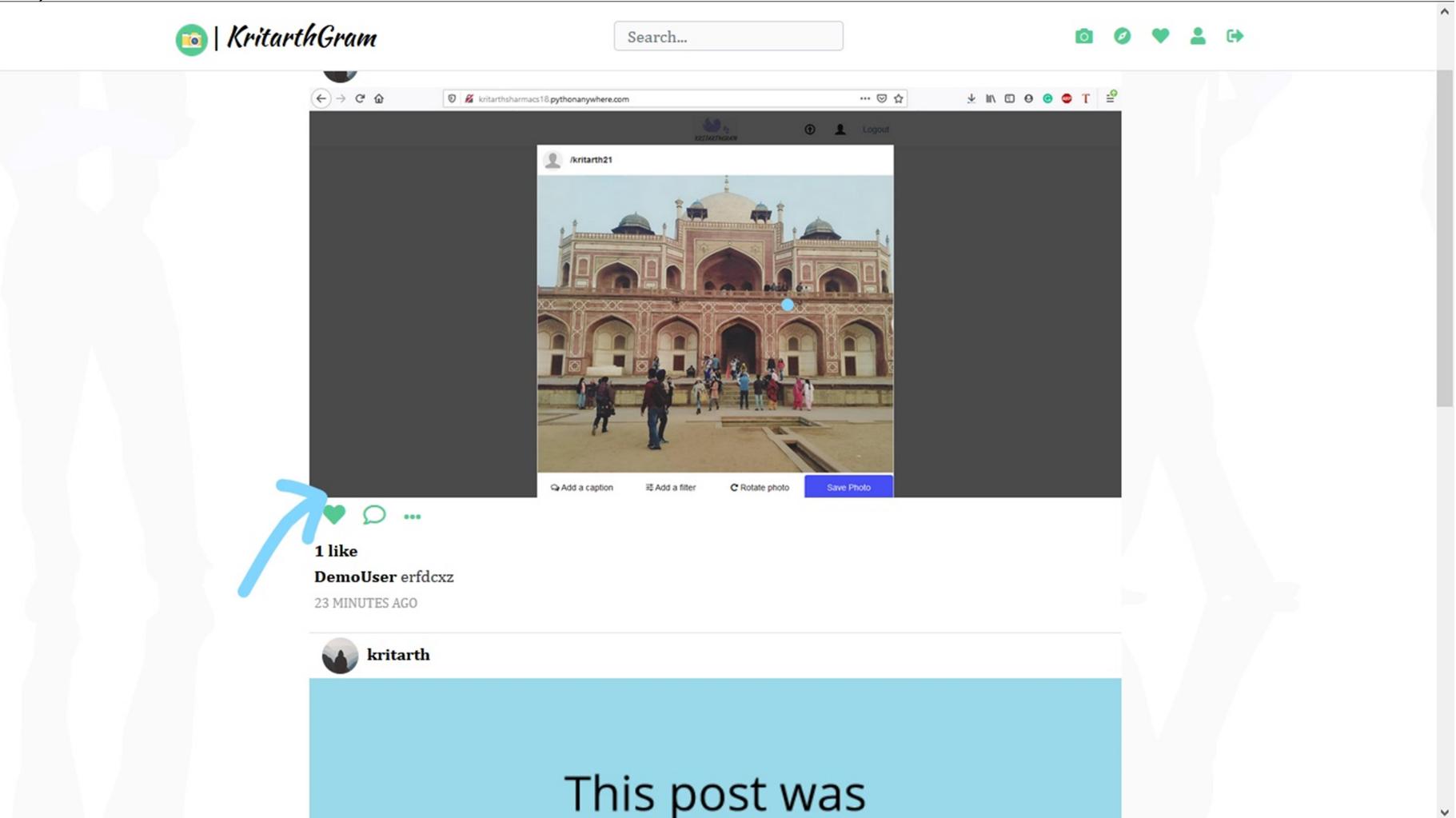
kritarthsharma oh bhai 5 HOURS AGO

kritarth Nice 4 MONTHS AGO

kritarth Here is your first comment! 4 MONTHS AGO

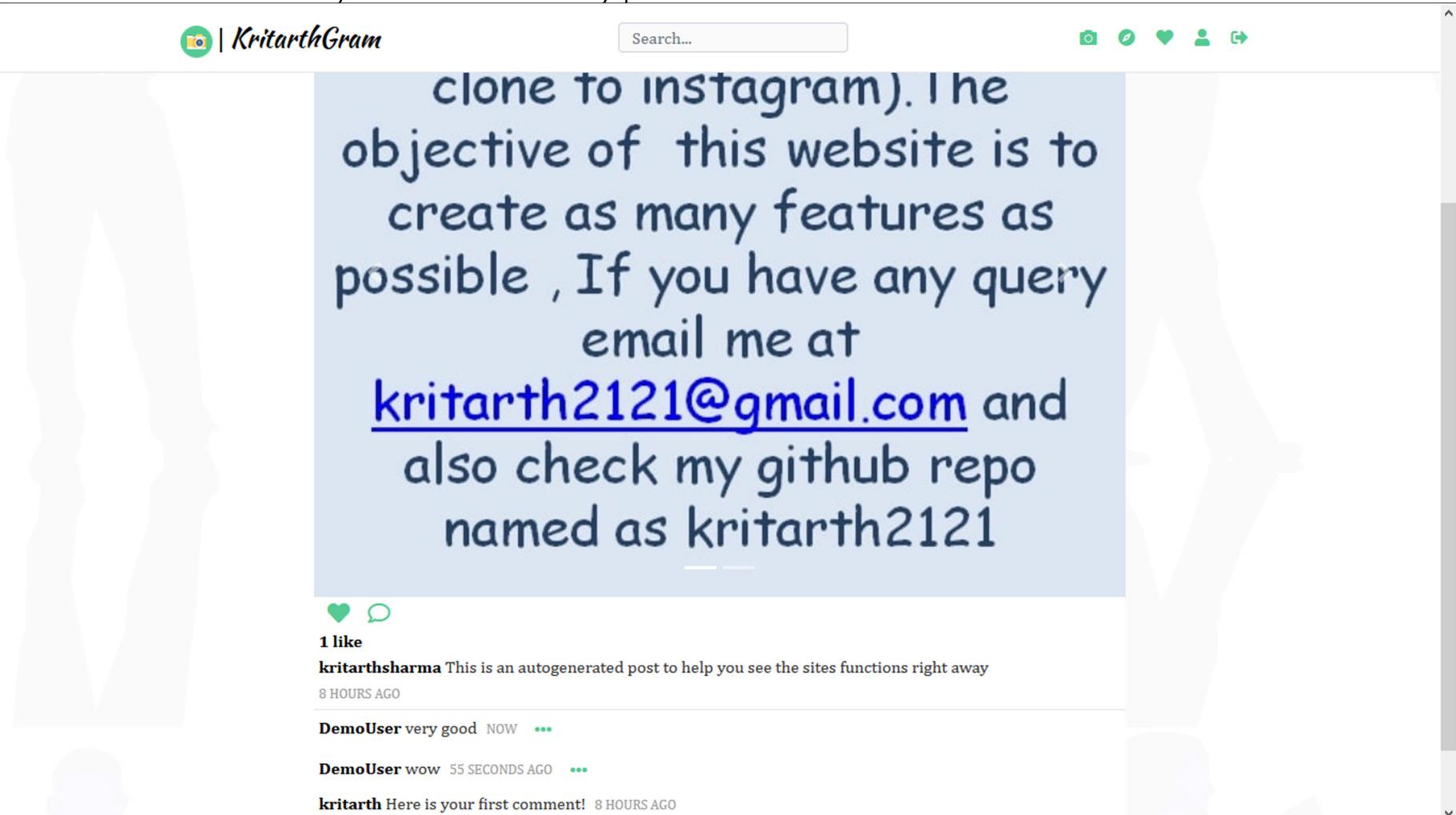
4)Like

Users can like on the post, without refreshing of the page the like is stored instantaneously using ajax request.



5)Comment

Users can comment any no of time on any post



The screenshot shows a user interface for a social media or content sharing platform. At the top, there's a header with a camera icon, the handle 'KritarthGram', a search bar containing 'Search...', and several small icons for camera, profile, heart, and share.

The main content area displays a post from 'KritarthGram'. The post text is as follows:

clone to instagram). The objective of this website is to create as many features as possible , If you have any query email me at kritarth2121@gmail.com and also check my github repo named as kritarth2121

Below the post, there are interaction metrics: 1 like and 1 comment. The comment is from 'kritarthsharma' stating: 'This is an autogenerated post to help you see the sites functions right away' posted 8 HOURS AGO.

Other visible comments include:

- A comment from 'DemoUser' saying 'very good' NOW
- A comment from 'DemoUser' saying 'wow' 55 SECONDS AGO
- A comment from 'kritarth' saying 'Here is your first comment!' 8 HOURS AGO

6)Search

We can search for any text present in any username,any post.
The search results are then fetched in list order.

Search Results for "th"

Users



kritarth



kritarthsharma

Posts



"This is an autogene..." by: kritarth



"This is an autogene..." by: kritarthsharma

7)Explore

There is a feature of exploring newly made accounts,newly made posts

Newest accounts



kkk



krishnaaaa



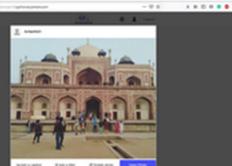
tvvjeknwdms



krishn

[Follow](#)[Follow](#)[Follow](#)[Follow](#)

Explore



Welcome to KritarthGram.(A clone to instagram).The objective of this website is to create as many features as possible . If you have any query email me at kritarth2121@gmail.com and also check my github repo named as kritarth2121

This post was auto generated to help you see functions of the site from the start. Explore your notifications by pressing the heart symbol. Also go check out your inbox!

8)Notification

Here user can view who have liked ,commented or followed them.



Search...



Notifications

No notifications!



Search...



Notifications



DemoUser commented "Nice" on your post! 0 MINUTES



DemoUser followed you! 1 MINUTE



9)Profile view.

Users can see their followers ,following list,photos,.They can also edit their profile picture and update and delete any posts.

The screenshot shows a user profile page for 'DemoUser'. At the top left is a camera icon followed by the text 'KritarthGram'. A search bar with the placeholder 'Search...' is positioned at the top center. On the right side of the header are five green icons: a camera, a person, a heart, a profile, and a share symbol. Below the header, the user's profile picture is a circular image of a landscape with mountains. To the right of the profile picture, the username 'DemoUser' is displayed with a gear icon indicating it's editable. Below the username, the following statistics are shown: '1 posts', '1 followers', and '1 following'. Underneath these stats is a thumbnail image of a post showing a building with a dome, identified as 'Agra'. The background of the page features a faint watermark of a person's silhouette.



Search...



Profile Info

DemoUser

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Currently: [default.jpg](#)

Change: No file selected.

 **DemoUser**

kritharsharma18.pythonanywhere.com

/kritarth21



Add a caption Add a filter Rotate photo Save Photo

1 like
DemoUser 45 MINUTES

Update Delete



Search...



erfdcxz

Update Post



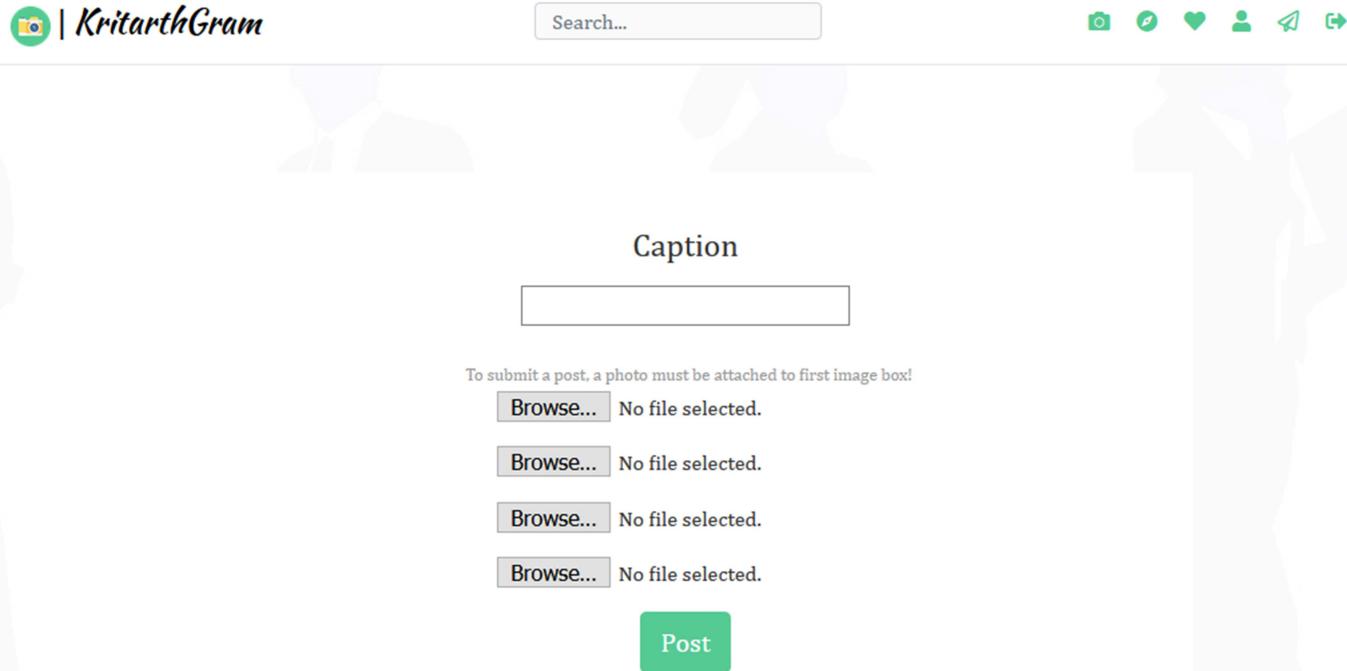
Search...



[Delete Post](#) [Cancel](#)

10)Adding posts

We can add post along with caption,we can add 4 photos at a time.



11)Logout

Users can logout from their account simply by clicking on the logout icon present on the corner of

page.

The image is a screenshot of a social media post from 'KritarthGram'. The post title is 'How to insert USB'. Below the title, there are two images of a yellow USB drive. The top image shows the USB drive inserted into a port with the text 'Wrong' next to it. The bottom image shows the USB drive inserted upside down with the text 'Wrong' next to it. A blue arrow points to the profile icon of the user 'DemoUser'.

KritarthGram

Search...

DemoUser

How to insert USB

Wrong

Wrong



You have been logged out

[Log In Again](#)

12)Amazon s3 for storage

We have used amazon s3 for storing media.

AWS Services ▾ kritarth21 Global Support

Bucket overview

Region Asia Pacific (Mumbai) ap-south-1	Amazon resource name (ARN) arn:aws:s3:::productivity21	Creation date June 18, 2020, 20:21 (UTC+05:30)	Access Bucket and objects not public
--	---	---	---

Objects Properties Permissions Metrics Management Access points

Drag and drop files and folders you want to upload here, or choose Upload.

Objects (3)

Objects are the fundamental entities stored in Amazon S3. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

List versions Delete Actions Create folder Upload

Find objects by prefix

Name	Type	Last modified	Size	Storage class
default.jpg	jpg	November 25, 2020, 21:14 (UTC+05:30)	10.7 KB	Standard
post_images/	Folder	-	-	-
static/	Folder	-	-	-

Feedback English (US) © 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

File Edit Selection View Go Run Terminal Help settings.py - django_project - Visual Studio Code

EXPLORER ... roj Mid term Report.docx test.jpg forms.py urls.py ...\\homepage list.html urls.py ...\\kriticgram settings.py

OPEN EDITORS

DJANGO_PROJECT

- forms.py
- models.py
- tests.py
- urls.py
- views.py
- kriticgram
 - _pycache_
 - __init__.py
 - settings.py M
 - urls.py
 - wsgi.py
- media
 - default.jpg
 - imageA.jpg U
 - imageB.png U
 - test.jpg
- static
 - staticfiles
 - admin
 - django_extensions
 - homepage
 - notifications
 - staticfiles.json M
 - templates

TERMINAL DEBUG CONSOLE PROBLEMS OUTPUT 2: python + ^ x

```
[25/Nov/2020 21:43:12] "GET /post/14/ HTTP/1.1" 200 11267
[25/Nov/2020 21:43:16] "GET /inbox/notifications/unread/ HTTP/1.1" 200 10186
[25/Nov/2020 21:44:24] "GET /profile/DemoUser/ HTTP/1.1" 200 10646
[25/Nov/2020 21:44:25] "GET /profile/edit/ HTTP/1.1" 200 12158
[25/Nov/2020 21:48:46] "GET /post/13/ HTTP/1.1" 200 10984
[25/Nov/2020 21:48:56] "GET /post/13/update/ HTTP/1.1" 200 9276
[25/Nov/2020 21:49:02] "GET /post/13/delete/ HTTP/1.1" 200 9229
```

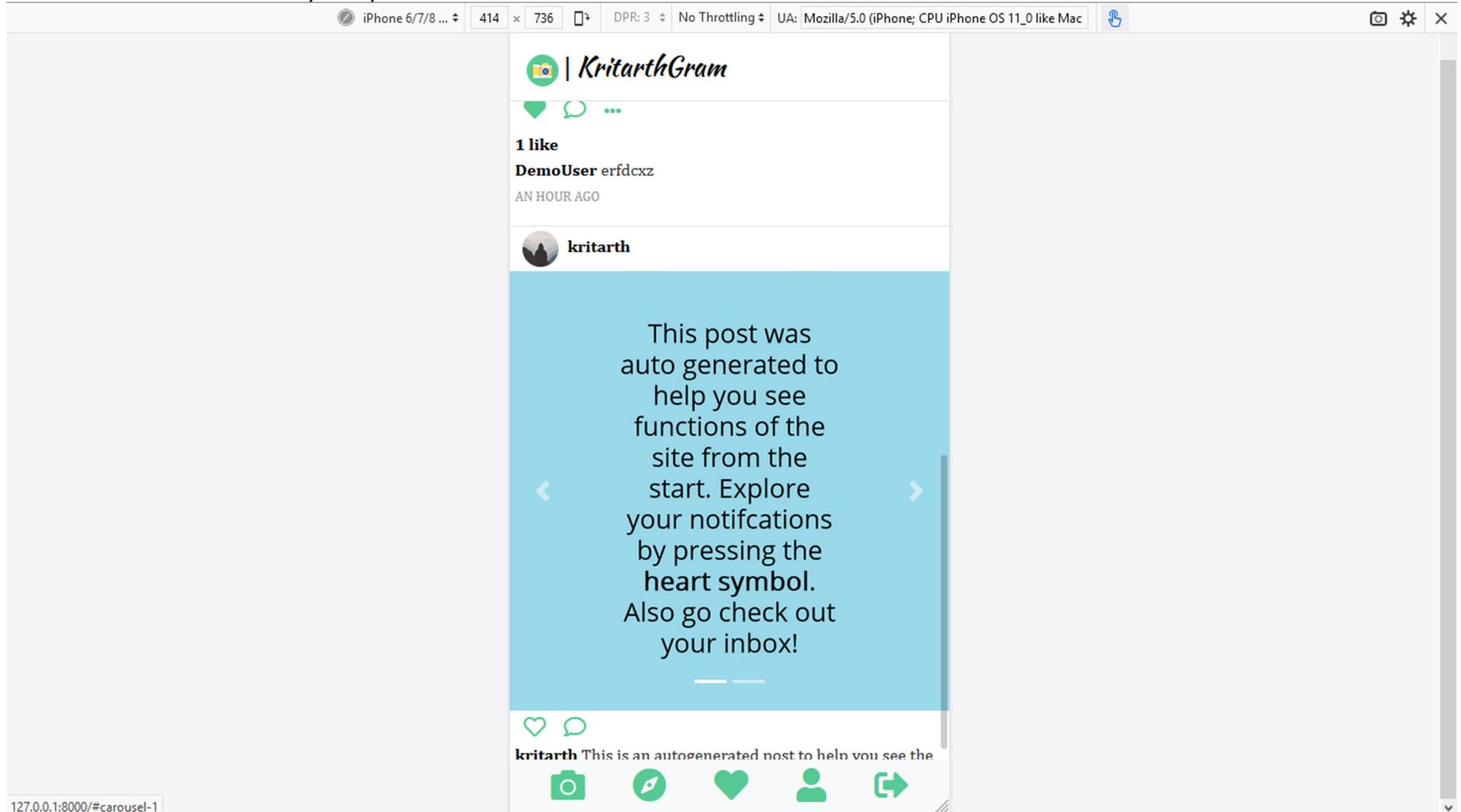
Ln 157, Col 2 Spaces: 4 UTF-8 CRLF Python ⚙ 9:58 PM 11/25/2020

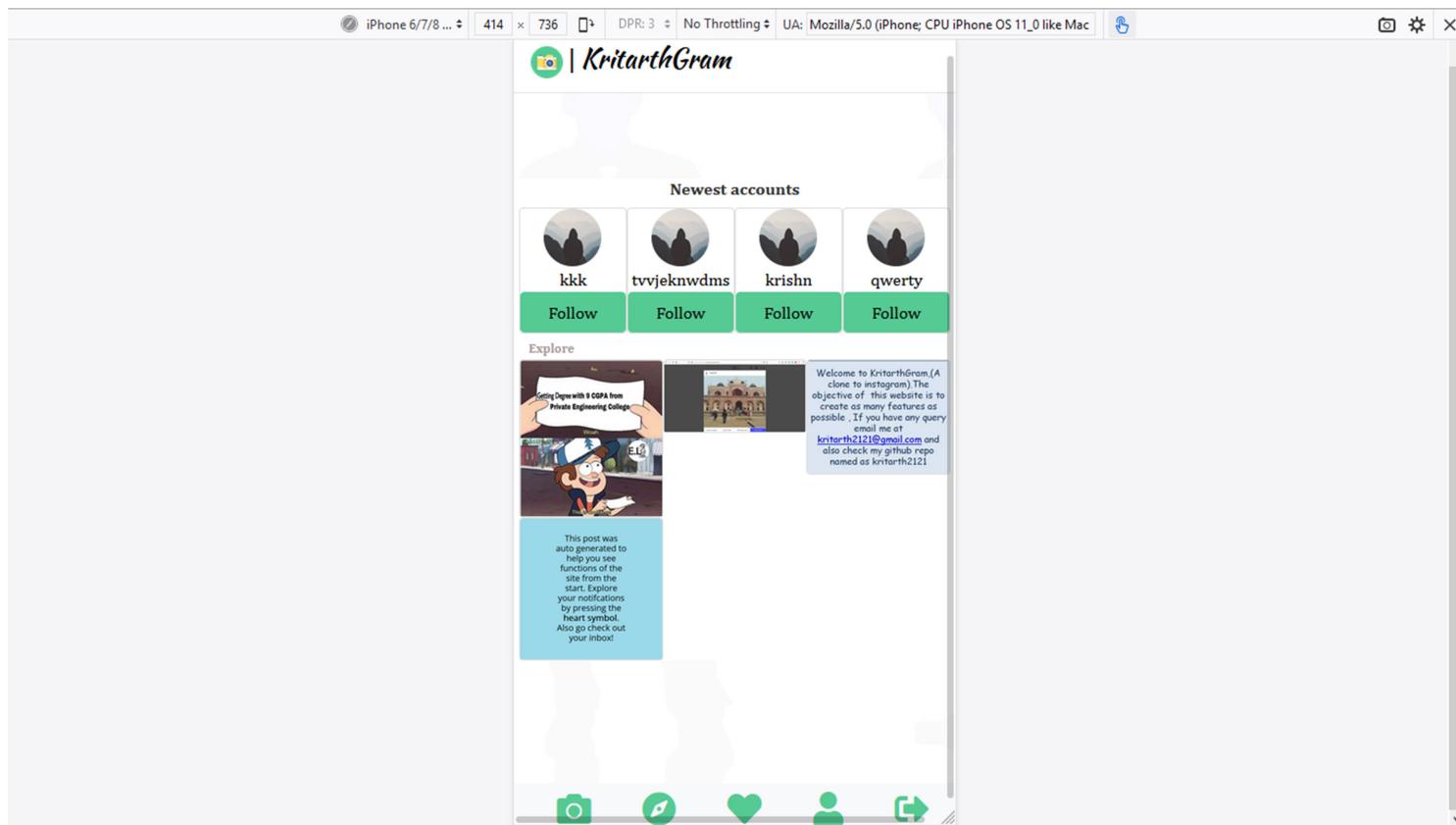
master* Run Testcases Python 3.7.7 64-bit ('myproject') 0 △ 0 Type here to search

Windows Start button Taskbar Icons

13)Responsiveness

We have made it fully responsive





14)Database

We have used django default database sqlite,as database

The screenshot shows the Django administration interface at 127.0.0.1:8000/admin/. The top navigation bar includes links for Site administration, Django site admin, and a plus sign icon. The header also displays the URL and a welcome message for user KRISHN.

The main content area is titled "Django administration" and "Site administration". It features a sidebar with several sections:

- AUTHENTICATION AND AUTHORIZATION**: Groups (+ Add, Change), Users (+ Add, Change)
- DIRECTMESSAGES**: Messages (+ Add, Change)
- HOME PAGE**: Comments (+ Add, Change), Likes (+ Add, Change), Posts (+ Add, Change)
- NOTIFICATIONS**: Notifications (+ Add, Change)
- USERS**: Followers (+ Add, Change)

On the right side, there are two panels:

- Recent actions**: A list of notifications from user kritarth and DemoUser.
- My actions**: A list of notifications from user kritarth and DemoUser.

The "Recent actions" panel contains the following items:

- kritarth followed you! ('kritarth', 'kritarth') follower-> followee 4 months, 2 weeks ago Notification
- kritarth sent you a This is your inbox! Click me to view our conversation. 4 months, 2 weeks ago Notification
- kritarth commented "Here is your first..." on your post! This is an autogenerated post to help you see the sites functions right away on Here is your first comment! 4 months, 2 weeks ago Notification
- DemoUser followed you! ('DemoUser', 'DemoUser') follower-> followee 4 months, 2 weeks ago Notification
- DemoUser followed you! ('DemoUser', 'kritarth') follower-> followee 4 months, 2 weeks ago Notification

The "My actions" panel contains the same list of notifications.

Select user to change | Django site X +

Home › Authentication and Authorization › Users

Select user to change

Action: 0 of 9 selected

<input type="checkbox"/>	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
<input type="checkbox"/>	DemoUser				✗
<input type="checkbox"/>	jiofekds	default@email.com			✗
<input type="checkbox"/>	kkk	kritarth2121@gmail.com			✗
<input type="checkbox"/>	krishn				✓
<input type="checkbox"/>	krishnaaaa	default@email.com			✗
<input type="checkbox"/>	kritarth				✓
<input type="checkbox"/>	kritarthsharma	default@email.com			✗
<input type="checkbox"/>	qwerty	default@email.com			✗
<input type="checkbox"/>	tvvjeknwdms	default@email.com			✗

9 users

FILTER

By staff status

- All
- Yes
- No

By superuser status

- All
- Yes
- No

By active

- All
- Yes
- No

ADD USER +

Type here to search

10:19 PM
11/25/2020

15)Hosting

The site is hosted on heroku,at <https://kritarthgram.herokuapp.com/>

The screenshot shows the Heroku dashboard for the 'kritarthgram' application. At the top, there's a navigation bar with the Heroku logo, a search bar labeled 'Jump to Favorites, Apps, Pipelines, Spaces...', and user account options. Below the header, the app name 'kritarthgram' is displayed along with its GitHub connection ('GitHub kt7756294/instagramclone master'). The main content area includes sections for 'Overview', 'Resources', 'Deploy', 'Metrics', 'Activity', 'Access', and 'Settings'. The 'Overview' section highlights a free Heroku Postgres add-on named 'Hobby Dev postgresql-graceful-40598'. The 'Activity' section shows a recent deployment by 'kt7756294@gmail.com' at 12:30 AM on July 10, followed by a build success message at 12:29 AM on the same day. Other activity items include provisioning completion and database attachment.

Personal > kritarthgram

GitHub kt7756294/instagramclone master

Overview Resources Deploy Metrics Activity Access Settings

Get a complete visualization of your app in a team-based continuous delivery environment with [Heroku Pipelines](#).

Installed add-ons \$0.00/month

Configure Add-ons

Heroku Postgres Hobby Dev postgresql-graceful-40598

Dyno formation \$0.00/month

Configure Dynos

This app is using free dynos

web gunicorn caseygram.wsgi ON

Collaborator activity

Manage Access

kt7756294@gmail.com 2 deploys

Latest activity

All Activity

kt7756294@gmail.com: Deployed 06082e2e Jul 10 at 12:30 AM · v6 · [Compare diff](#)

kt7756294@gmail.com: Build succeeded Jul 10 at 12:29 AM · [View build log](#)

kt7756294@gmail.com: Deployed c782ecde Jul 9 at 11:22 PM · v5

kt7756294@gmail.com: @ref:postgresql-graceful-40598 completed provisioning, setting DATABASE_URL. Jul 9 at 11:22 PM · v4

kt7756294@gmail.com: Attach DATABASE (@ref:postgresql-graceful-40598) Jul 9 at 11:22 PM · v3

kt7756294@gmail.com: Build succeeded Jul 9 at 11:20 PM · [View build log](#)

Objective

The main objectives of Social networking project is to make people connected as much as possible.

Here people can upload photos or videos to our service and share them with their followers or with a select group of friends. They can also view, comment and like posts shared by their friends on **Kritarthgram**.

As social networking sites are gaining popularity day by day, so we try to build it using Django, Ajax, HTML, CSS, Javascript, Amazon s3

It is an Indian based web-app hosted on Heroku.

Software Specification:

Technology Implemented : Django
Language Used : Python
Database : Mysql
User Interface Design : Html,CSS , javascript and Bootstrap
Web Browser : Microsoft edge/chrome

Hardware Requirements:

Processor : 64-bit/32 bit
Operating System : Any one (Preferrably above Windows 8)

RAM : 4-GB

Technology Used

HTML

Hypertext Markup Language (HTML) is the standard [markup language](#) for documents designed to be displayed in a [web browser](#). It can be assisted by technologies such as [Cascading Style Sheets](#) (CSS) and [scripting languages](#) such as [JavaScript](#).

[Web browsers](#) receive HTML documents from a [web server](#) or from local storage and [render](#) the documents into multimedia web pages. HTML describes the structure of a [web page semantically](#) and originally included cues for the appearance of the document.

[HTML elements](#) are the building blocks of HTML pages. With HTML constructs, [images](#) and other objects such as [interactive forms](#) may be embedded into the rendered page. HTML provides a means to create [structured documents](#) by denoting structural [semantics](#) for text such as headings, paragraphs, lists, [links](#), quotes and other items. HTML elements are delineated by [tags](#), written using [angle brackets](#). Tags such as `` and `<input />` directly introduce content into the page. Other tags such as `<p>` surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a [scripting language](#) such as [JavaScript](#), which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The [World Wide Web Consortium](#) (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997

HTML



In 1980, physicist [Tim Berners-Lee](#), a contractor at [CERN](#), proposed and prototyped [ENQUIRE](#), a system for CERN researchers to use and share documents. In 1989, Berners-Lee wrote a memo proposing an [Internet](#)-based [hypertext](#) system.^[3] Berners-Lee specified HTML and wrote the browser and server software in late 1990. That year, Berners-Lee and CERN data systems engineer [Robert Cailliau](#) collaborated on a joint request for funding, but the project was not formally adopted by CERN. In his personal notes^[4] from 1990 he listed^[5] "some of the many areas in which hypertext is used" and put an encyclopedia first.

HTML markup consists of several key components, including those called *tags* (and their *attributes*), character-based *data types*, *character references* and *entity references*. HTML tags most commonly come in pairs like `<h1>` and `</h1>`, although some represent *empty elements* and so are unpaired, for example ``. The first tag in such a pair is the *start tag*, and the second is the *end tag* (they are also called *opening tags* and *closing tags*).

Another important component is the HTML [document type declaration](#), which triggers [standards mode](#) rendering.

The following is an example of the classic ["Hello, World!" program](#):

```
<!DOCTYPE html>
<html>
  <head>
    <title>This is a title</title>
  </head>
  <body>
    <div>
      <p>Hello world!</p>
    </div>
  </body>
</html>
```

The text between `<html>` and `</html>` describes the web page, and the text between `<body>` and `</body>` is the visible page content. The markup text `<title>This is a title</title>` defines the browser page title, and the tag `<div>` defines a division of the page used for easy styling.

The Document Type Declaration `<!DOCTYPE html>` is for HTML5. If a declaration is not included, various browsers will revert to "[quirks mode](#)" for rendering.^[70]

AJAX

Ajax (also **AJAX** /'eɪdʒəks/; short for "Asynchronous [JavaScript](#) and [XML](#)")^{[1][2]} is a set of [web development](#) techniques using many web technologies on the [client side](#) to create [asynchronous web applications](#). With Ajax, web applications can send and retrieve data from a [server](#) asynchronously (in the background) without interfering with the display and behaviour of the existing page. By decoupling the data interchange layer from the presentation layer, Ajax allows web pages and, by extension, web applications, to change content dynamically without the need to reload the entire page.^[3] In practice, modern implementations commonly utilize [JSON](#) instead of XML.

Ajax is not a single technology, but rather a group of technologies. [HTML](#) and [CSS](#) can be used in combination to mark up and style information. The webpage can then be modified by JavaScript to dynamically display—and allow the user to interact with—the new information. The built-in [XMLHttpRequest](#) object, or since 2017 the new "fetch()" function within JavaScript, is commonly used to execute Ajax on webpages, allowing websites to load content onto the screen without refreshing the page. Ajax is not a new technology, or different language, just existing technologies used in new ways.

In the early-to-mid 1990s, most [Web](#) sites were based on complete HTML pages. Each user action required that a complete new page be loaded from the server. This process was inefficient, as reflected by the user experience: all page content disappeared, then the new page appeared. Each time the browser reloaded a page because of a partial change, all of the content had to be re-sent, even though only some of the information had changed. This placed additional load on the server and made [bandwidth](#) a limiting factor on performance.

In 1996, the [iframe](#) tag was introduced by [Internet Explorer](#); like the [object](#) element, it can load or fetch content asynchronously. In 1998, the Microsoft [Outlook Web Access](#) team developed the concept behind the [XMLHttpRequest](#) scripting object.^[4] It appeared as XMLHTTP in the second version of the [MSXML](#) library,^{[4][5]} which shipped with [Internet Explorer 5.0](#) in March 1999.^[6]

The functionality of the Windows XMLHTTP [ActiveX](#) control in IE 5 was later implemented by [Mozilla](#), [Safari](#), [Opera](#) and other browsers as the XMLHttpRequest [JavaScript](#) object.^[7] Microsoft adopted the native XMLHttpRequest model as of [Internet Explorer 7](#). The ActiveX version is still supported in Internet Explorer, but not in [Microsoft Edge](#). The utility of

these background [HTTP](#) requests and asynchronous Web technologies remained fairly obscure until it started appearing in large scale online applications such as Outlook Web Access (2000)^[8] and [Oddpost](#) (2002).

[Google](#) made a wide deployment of standards-compliant, [cross browser](#) Ajax with [Gmail](#) (2004) and [Google Maps](#) (2005).^[9] In October 2004 [Kayak.com](#)'s public beta release was among the first large-scale e-commerce uses of what their developers at that time called "the xml http thing".^[10] This increased interest in AJAX among web program developers.

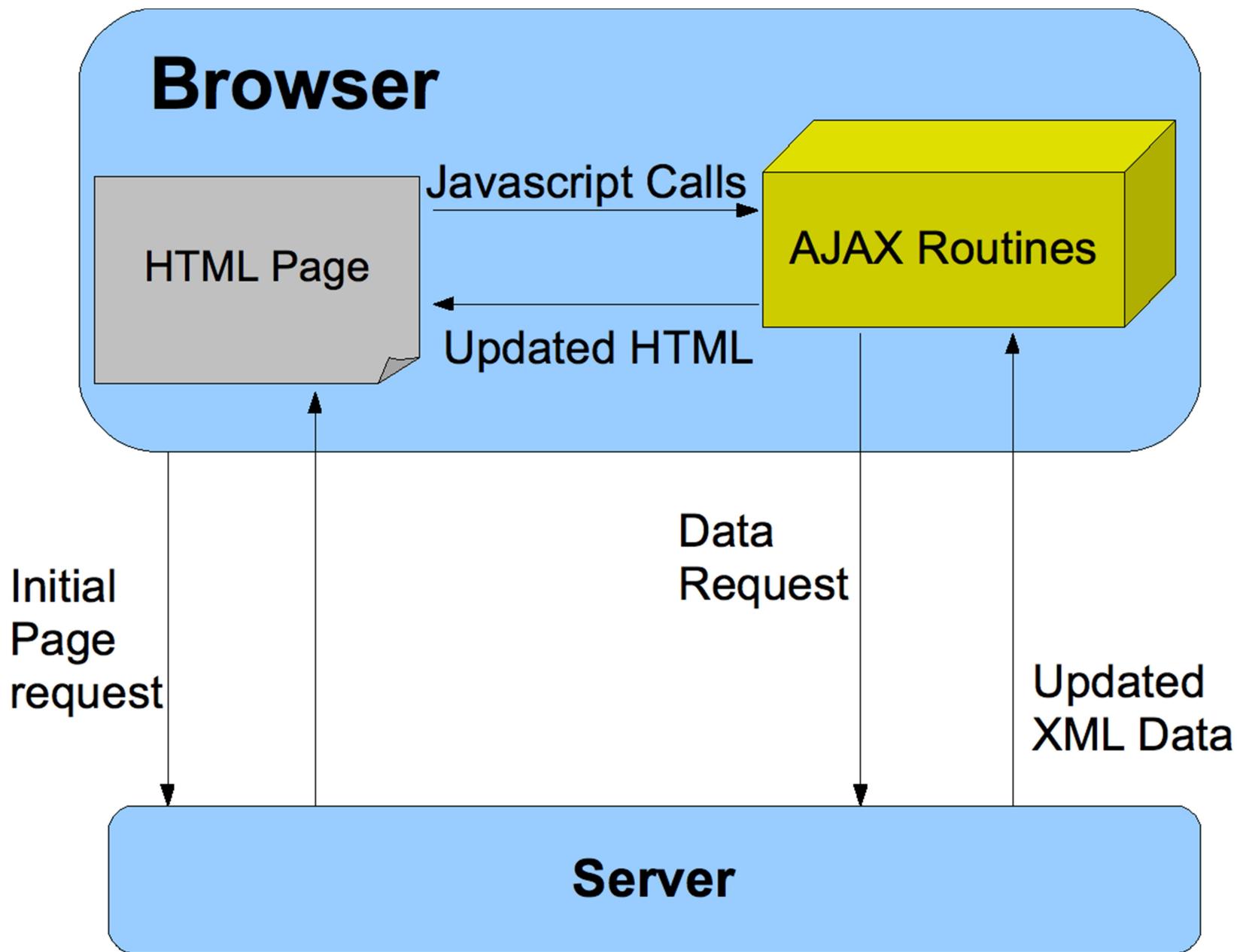
The term AJAX was publicly used on 18 February 2005 by [Jesse James Garrett](#) in an article titled Ajax: A New Approach to Web Applications, based on techniques used on Google pages.^[11]

On 5 April 2006, the [World Wide Web Consortium](#) (W3C) released the first draft specification for the XMLHttpRequest object in an attempt to create an official [Web standard](#).^[12] The latest draft of the XMLHttpRequest object was published on 6 October 2016,^[13] and the XMLHttpRequest specification is now a [living standard](#).^[13]

The term Ajax has come to represent a broad group of Web technologies that can be used to implement a Web application that communicates with a server in the background, without interfering with the current state of the page. In the article that coined the term Ajax,^{[11][13]} Jesse James Garrett explained that the following technologies are incorporated:

- [HTML](#) (or [XHTML](#)) and [CSS](#) for presentation
- The [Document Object Model](#) (DOM) for dynamic display of and interaction with data
- [JSON](#) or [XML](#) for the interchange of data, and [XSLT](#) for XML manipulation
- The [XMLHttpRequest](#) object for asynchronous communication
- [JavaScript](#) to bring these technologies together

Since then, however, there have been a number of developments in the technologies used in an Ajax application, and in the definition of the term Ajax itself. XML is no longer required for data interchange and, therefore, XSLT is no longer required for the manipulation of data. [JavaScript Object Notation](#) (JSON) is often used as an alternative format for data interchange,^[14] although other formats such as preformatted HTML or plain text can also be used.^[15] A variety of popular JavaScript libraries, including JQuery, include abstractions to assist in executing Ajax requests.





CSS

Cascading Style Sheets (CSS) is a [style sheet language](#) used for describing the [presentation](#) of a document written in a [markup language](#) such as [HTML](#).^[1] CSS is a cornerstone technology of the [World Wide Web](#), alongside HTML and [JavaScript](#).^[2]

CSS is designed to enable the separation of presentation and content, including [layout](#), [colors](#), and [fonts](#).^[3] This separation can improve content [accessibility](#), provide more flexibility and control in the specification of presentation characteristics, enable multiple [web pages](#) to share formatting by specifying the relevant CSS in a separate .css file which reduces complexity and repetition in the structural content as well as enabling the .css file to be [cached](#) to improve the page load speed between the pages that share the file and its formatting.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or [screen reader](#)), and on [Braille-based](#) tactile devices. CSS also has rules for alternate formatting if the content is accessed on a [mobile device](#).^[4]

The name cascading comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

The CSS specifications are maintained by the [World Wide Web Consortium](#) (W3C). Internet media type ([MIME type](#)) text/css is registered for use with CSS by [RFC 2318](#) (March 1998). The W3C operates a free [CSS validation service](#) for CSS documents.^[5]

In addition to HTML, other markup languages support the use of CSS including [XHTML](#), [plain XML](#), [SVG](#), and [XUL](#).

Before CSS, nearly all presentational attributes of HTML documents were contained within the HTML markup. All font colors, background styles, element alignments, borders and sizes had to be explicitly described, often repeatedly, within the HTML. CSS lets authors move much of that information to another file, the style sheet, resulting in considerably simpler HTML.

For example, headings (`h1` elements), sub-headings (`h2`), sub-sub-headings (`h3`), etc., are defined structurally using HTML. In print and on the screen, choice of [font](#), [size](#), [color](#) and [emphasis](#) for these elements is *presentational*.

Before CSS, document authors who wanted to assign such [typographic](#) characteristics to, say, all `h2` headings had to repeat HTML presentational markup for each occurrence of that heading type. This made documents more complex, larger, and more error-prone and difficult to maintain. CSS allows the separation of presentation from structure. CSS can define color, font, text alignment, size, borders, spacing, layout and many other typographic characteristics, and can do so independently for on-screen and printed views. CSS also defines non-visual styles, such as reading speed and emphasis for aural text readers. The [W3C](#) has now [deprecated](#) the use of all presentational HTML markup.[\[15\]](#)

CSS



For example, under pre-CSS HTML, a heading element defined with red text would be written as:

```
<h1><font color="red">Chapter 1.</font></h1>
```

Using CSS, the same element can be coded using style properties instead of HTML presentational attributes:

```
<h1 style="color: red;">Chapter 1.</h1>
```

The advantages of this may not be immediately clear but the power of CSS becomes more apparent when the style properties are placed in an internal style element or, even better, an external CSS file. For example, suppose the document contains the style element:

```
<style>
  h1 {
    color: red;
  }
</style>
```

All h1 elements in the document will then automatically become red without requiring any explicit code. If the author later wanted to make h1 elements blue instead, this could be done by changing the style element to:

```
<style>
  h1 {
    color: blue;
  }
</style>
```

rather than by laboriously going through the document and changing the color for each individual h1 element.

The styles can also be placed in an external CSS file, as described below, and loaded using syntax similar to:

```
<link href="path/to/file.css" rel="stylesheet" type="text/css">
```

This further decouples the styling from the HTML document and makes it possible to restyle multiple documents by simply editing a shared external CSS file.

JavaScript

JavaScript (/'dʒɑːvə'skript/),^[6] often abbreviated as **JS**, is a [programming language](#) that conforms to the [ECMAScript](#) specification.^[7] JavaScript is [high-level](#), often [just-in-time compiled](#), and [multi-paradigm](#). It has [curly-bracket syntax](#), [dynamic typing](#), [prototype-based object-orientation](#), and [first-class functions](#).

Alongside [HTML](#) and [CSS](#), JavaScript is one of the core technologies of the [World Wide Web](#).^[8] JavaScript enables interactive [web pages](#) and is an essential part of [web applications](#). The vast majority of [websites](#) use it for [client-side](#) page behavior,^[9] and all major [web browsers](#) have a dedicated [JavaScript engine](#) to execute it.

As a multi-paradigm language, JavaScript supports [event-driven](#), [functional](#), and [imperative programming styles](#). It has [application programming interfaces](#) (APIs) for working with text, dates, [regular expressions](#), standard [data structures](#), and the [Document Object Model](#) (DOM). However, the language itself does not include any [input/output](#) (I/O), such as [networking](#), [storage](#), or [graphics](#) facilities, as the host environment (usually a web browser) provides those APIs.

JavaScript engines were originally used only in web browsers, but they are now embedded in some [servers](#), usually via [Node.js](#). They are also embedded in a variety of applications created with [frameworks](#) such as [Electron](#) and [Cordova](#).

Although there are similarities between JavaScript and [Java](#), including language name, [syntax](#), and respective [standard libraries](#), the two languages are distinct and differ greatly in design.

The use of JavaScript has expanded beyond its [web browser](#) roots. [JavaScript engines](#) are now embedded in a variety of other software systems, both for [server-side](#) website deployments and non-browser [applications](#).

Initial attempts at promoting server-side JavaScript usage were [Netscape Enterprise Server](#) and [Microsoft's Internet Information Services](#),^{[30][31]} but they were small niches.^[32] Server-side usage eventually started to grow in the late-2000s, with the creation of [Node.js](#) and [other approaches](#).^[32]

JS



[Electron](#), [Cordova](#), and other [software frameworks](#) have been used to create many applications with behavior implemented in JavaScript. Other non-browser applications include [Adobe Acrobat](#) support for scripting [PDF](#) documents^[33] and [GNOME Shell](#) extensions written in JavaScript.^[34]

JavaScript has recently begun to appear in some [embedded systems](#), usually by leveraging Node.js.^[35]^[36]^[37]

DJANGO

Django (*/dʒængəʊ/ JANG-goh*; sometimes stylized as **django**)^[8] is a [Python-based free and open-source web framework](#) that follows the model-template-views (MTV) [architectural pattern](#).^[9]^[10] It is maintained by the [Django Software Foundation](#) (DSF), an American independent organization established as a [501\(c\)\(3\)](#) non-profit.

Django's primary goal is to ease the creation of complex, database-driven websites. The framework emphasizes [reusability](#) and "pluggability" of components, less code, low coupling, rapid development, and the principle of [don't repeat yourself](#).^[11] Python is used throughout, even for settings, files, and data models. Django also provides an optional administrative [create, read, update and delete](#) interface that is generated dynamically through [introspection](#) and configured via admin models.

Some well known sites that use Django include [PBS](#),^[12] [Instagram](#),^[13] [Mozilla](#),^[14] [The Washington Times](#),^[15] [Disqus](#),^[16] [Bitbucket](#),^[17] and [Nextdoor](#).^[18]

Django was created in the fall of 2003, when the [web programmers](#) at the [Lawrence Journal-World](#) newspaper, [Adrian Holovaty](#) and [Simon Willison](#), began using Python to build applications. [Jacob Kaplan-Moss](#) was hired early in Django's development shortly before Simon Willison's internship ended.^[19] It was released publicly under a [BSD license](#) in July 2005. The framework was named after guitarist [Django Reinhardt](#).^[20]

In June 2008, it was announced that a newly formed [Django Software Foundation](#) (DSF) would maintain Django in the future.^[21]



For developing a Django project, no special tools are necessary, since the source code can be edited with any conventional [text editor](#). Nevertheless, editors specialized on [computer programming](#) can help increase the productivity of development, e.g., with features such as [syntax highlighting](#). Since Django is written in Python, text editors which are aware of Python syntax are beneficial in this regard.

[Integrated development environments](#) (IDE) add further functionality, such as [debugging](#), [refactoring](#), and [unit testing](#). As with plain editors, IDEs with support for Python can be beneficial. Some IDEs that are specialized on Python additionally have integrated support for Django projects, so that using such an IDE when developing a Django project can help further increase productivity. For comparison of such Python IDEs, see the main article:

References

We have used wikipedia, youtube,google,vscode editor,github for creating this project.
Some of useful websites are

<https://docs.djangoproject.com/>

<https://github.com/>

<https://www.google.com/>

<https://www.youtube.com/>