GRUM

Team

Jonathan Pesso Jonathan Gau Derek Phan Leonard Ma Kelvin Yang

Introduction

Our goal is to make this web application as secure as possible in order to show and teach other companies and application owners how to better secure their products. This sample web application that we are building will be used to share, and host various files online.

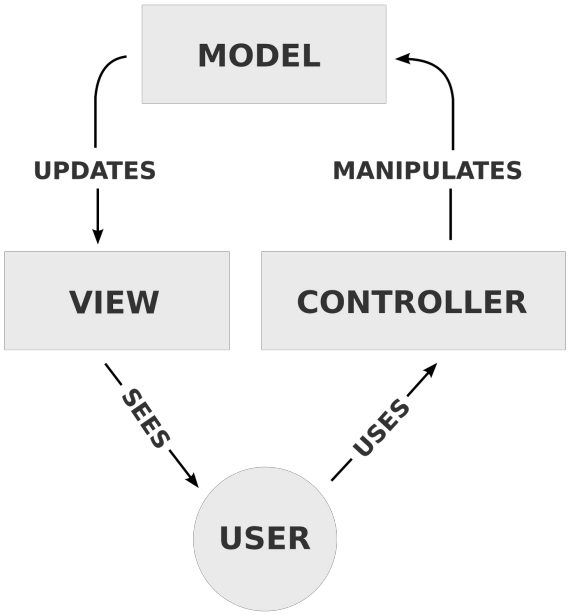
Glossary of Terms

Admin - Has complete access to all files that are public
Moderator - Specific users with elevated permissions
Registered User- Any person that has created an account. Will be able to upload files.
New User- Will be forced to make an account in order to access any of the application resources.

System Architecture Overview

MVC Framework (Model, View, Controller)

- Model: The model is the data that is retrieved by the controller for populating the view.
- View: The view is the html code generated as requested.
- Controller: The controller maps web requests to web pages, which are populated with data necessary for display.



User Database (auth_user)

- id - id of the user
- name - name of the user
- email - email address used for registration
- password - password used for login
- register_date - date account was created
- birthday - birthdate of user
- display_pic - display picture of user
- ip_address - list of whitelisted ip addresses for the account
- banned - whether or not the account is banned

File Database (files)

-Accesses User database

- id - id of the file
- upload_user - reference to the user entry in user db
- upload_modified_by - user who modifies the file
- file_name - name of the file
- file_tags - keywords for searching file
- upload_file - file that is uploaded
- upload_time - time of upload
- upoad_modified_on - time user modified the file
- file_visibility - visibility of files: public, private, pri
- file_score - score of a given file

Comment Database (comments)

-Accesses User and files databases

- id - id of the comment
- comment_user - reference to the user entry in user db
- comment_modified_by - user who modified the comment
- comments - comments
- comment_time - timestamp of comment (can be used to sort)
- comment_modified_on - time user modified the comment
- comment_score - score of a given comment

File Upvote/Downvote Database(file_votes)

-Accesses files databases

- id - id of the upvote/downvote for a file
- upvote/downvote - up or down votes a file
- auth.signature - contains information regarding who upvoted/downvoted a file, and when it was done

Comment Upvote/Downvote Database(comment_votes)

-Accesses comments databases

- id - id of the upvote/downvote for a comment
- upvote/downvote - up or down votes a comment
- auth.signature - contains information regarding who upvoted/downvoted a comment, and when it was done

Mail Database(mail)

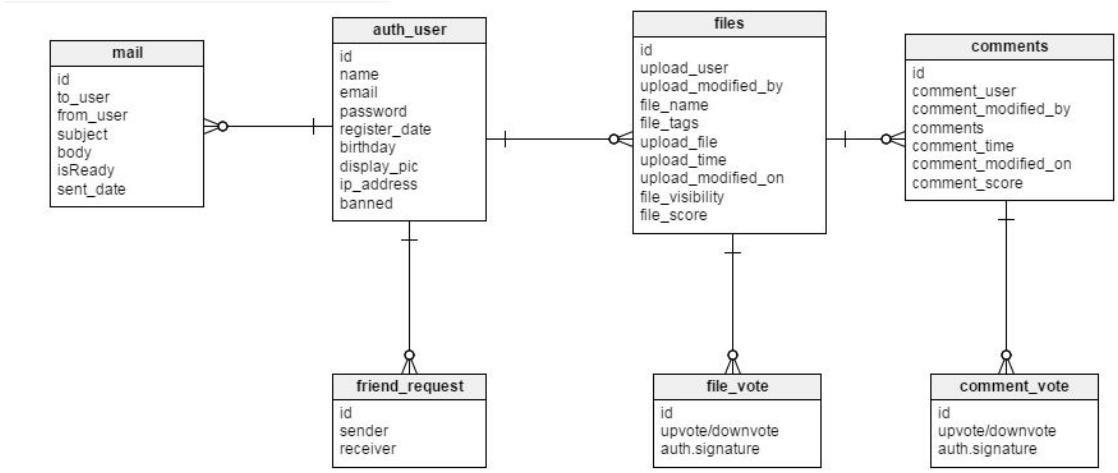
-Accesses user database

- id - id of the mail sent/received
- to_user - user mail is sent to
- from_user - user mail is from
- subject - subject of mail
- body - message
- isRead - tells if mail has been read or not
- sent_date - date mail is sent

Friend Request Database(friend_request)

-Accesses user database

- id - id of friend request sent/received
- sender - user who sends friend request
- receiver - user who receives friend request



Groups and Permissions

- All users will be part of a rank group (Admin, Moderator, Member). These are mutually exclusive.

- Each user with be a part of his/her own group. Any other users added to this group will be a friend of the owner, and will have read access to posts that a user has shared with friends, and a user's detailed profile.
- Each file will have its own group. If the visibility of a file is set to public, nobody will be in that group and all registered users should be able to see the file. If a file's visibility is set to friends, all the user's friends will be added to the group and all the user's friends should be able to see the file. If a file's visibility is set to private, only the user will be added to the group and only the user who uploaded the file should be able to see the file.

Requirements

1. As a guest, I am directed to a sign-in/log-in page.
2. As a guest, I am not allowed to post files.
3. As a guest, I can view the description of the website.
4. As a new user, I must register a new account.
5. As a registered user, I can upload pictures to share.
6. As a registered user, I can upload videos to share.
7. As a registered user, I can change my password.
8. As a registered user, I can comment on other people's posts.
9. As a registered user, I can view public files.
10. As a registered user, I can view my own files.
11. As a registered user, I can view my friends' files.
12. As a registered user, I can search for files by tags.
13. As a registered user, I can add and remove friends.
14. As a registered user, I can flag inappropriate content.
15. As a registered user, I can manage the visibility of my files.
16. As a registered user, I can modify my uploaded files.
17. As a registered user, I can set my display picture.
18. As a registered user, I can receive notifications through mailing system.
19. As a registered user, I can view my news feed on my home page.
20. As a registered user, I can view how many users and files are currently registered.
21. As a registered user, I can send and receive private mail w/inbox support.
22. As a registered user, I can upload audio to share.
23. As a registered user, I can upload text files to share.
24. As a registered user, I can upvote/downvote posts and comments.
25. As a registered user, I can view a directory of all registered users.
26. As a registered user, I can view which users are currently online.
27. As a registered user, I can see how many users viewed my post.
28. As a moderator, I can remove public uploaded files.
29. As an admin I can ban users.
30. As an admin I can modify and remove public uploaded files.

Test Cases
https://github.com/dphan94/Hopefully_Unhackable

Guest Tests:

1. Is a guest directed to a sign-in/log-in page?
2. Is a guest allowed to post files?
3. Can a guest see the description of the website?

New Users Test:

4. Are new users required to register a new account?

Registered Users Tests:

5. Can users upload pictures to share?
6. Can users upload videos to share?
7. Can users change their password?
8. Can users comment on other people's posts?
9. Can users view public files?
10. Can users view their own files?
11. Can users view their friend's files?
12. Can users search for files by tag?
13. Can users add or remove friends?
14. Can users flag inappropriate content?
15. Can users manage the visibility of their files?
16. Can users modify their uploaded files?
17. Can users set their display picture?
18. Can users receive notifications through the mailing system?
19. Can users view their news feed on the home page?
20. Can users see how many users and files are currently registered?
21. Can users send and receive private mail w/inbox support?
22. Can users upload audio to share?
23. Can users upload text files to share?
24. Can users upvote/downvote posts and comments?
25. Can users view a directory of all registered users?
26. Can users view which users are currently online?
27. Can users view how many users have seen their post?

Moderator Test:

28. Can moderators remove public uploaded files?

Admin Tests:

29. Can admins ban users?
30. Can admins modify and remove public uploaded files?

Access Ranks and Permissions

Public Posts	Member	Moderator	Admin
Read	Yes	Yes	Yes
Comment	Yes	Yes	Yes
Create	Yes	Yes	Yes
Delete	No	Yes	Yes
Modify	No	No	Yes

Private Posts	Owner	Friends	Others
Read	Yes	Yes	No

Comment	Yes	Yes	No
Delete	Yes	No	No
Modify	Yes	No	No

Appendices:

Technologies

- Bootstrap
- Javascript
- SQLite
- web2py
- Python 2.12.3
- HTML/CSS