VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA INFORMACIONE I KOMUNIKACIONE TEHNOLOGIJE

INTERNET TEHNOLOGIJE

DOKUMENTACIJA

TABLE FINDER

Modul: Web programiranje
Predmet: Web programiranje PHP 2

Student: Katarina Kalanj 89/18

Beograd, mart 2022

Sadržaj

1. Uvod

1.1 Korišćeni programski jezici

Sajt je pravljen korišćenjem jezika PHP, Laravel, HTML, SCSS, Javascript, jQuery.

1.2 Template

Nije korišćen template.

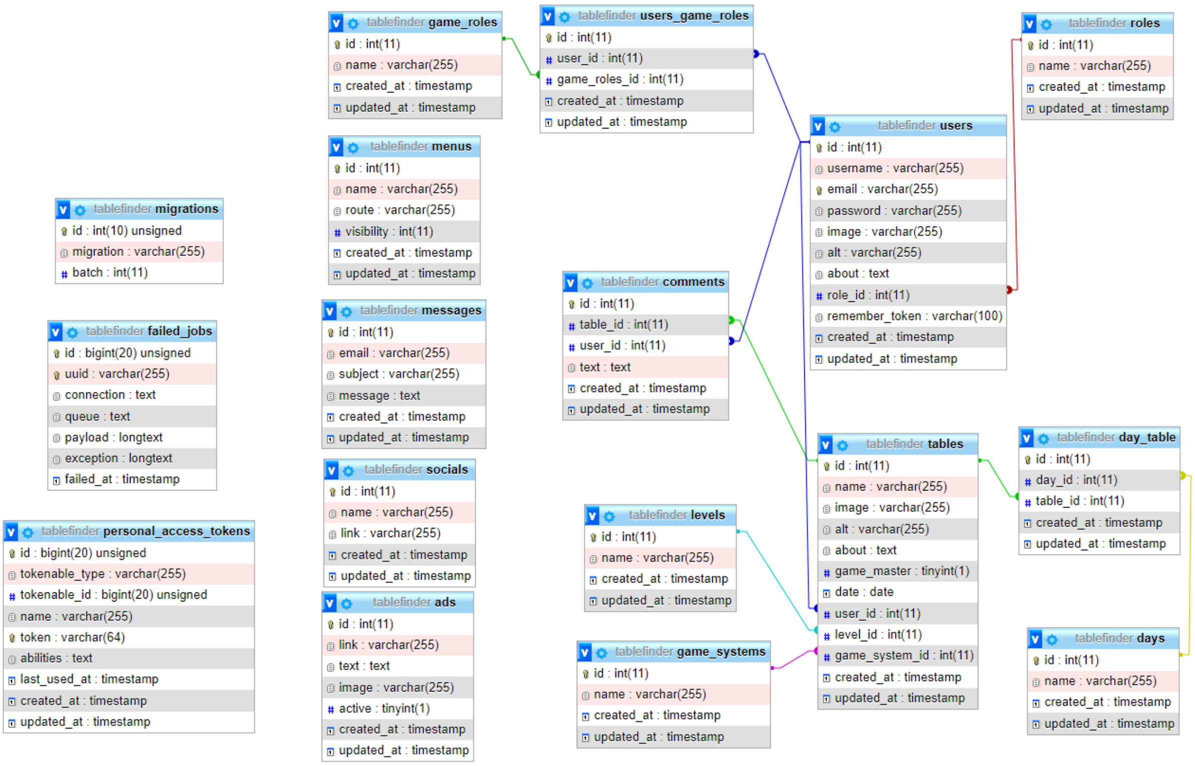
1.3 Opis funkcionalnosti

Sajt je zamišljen kao blog za pronalaženje grupa ili igrača za tabletop igre.

- Logovanje i registracija sa validacijom podataka na serveru.
- Kontakt forma namenjena adminu sajta, provera podataka ajaxom, slanje podataka ajaxom, serverska provera. Nakon uspešnog slanja, podaci se skladište u tabelu messages u bazi. Za poštu je korišćen Mailtrap.
- Na stranici Tables su inicijalno izlistani osnovni podaci o stolovima uz paginaciju putem ajaxa. Pretraga se takođe vrši putem ajaxa. Moguće je resetovati pretragu. Ispod se nalazi div sa reklamama koje su dinamički učitane iz baze.
- Klikom na see more(donji desni ugao stolova) otvara se stranica sa informacijama o specifičnom stolu. Ukoliko je korisnik vlasnik stola, prikazuju se dugmad za edit i delete.
- Komentari se učitavaju ajaxom, postoji opcija za brisanje ili editovanje svog komentara, dok vlasnik stola ima mogućnost brisanja tuđih komentara. Mogu se posetiti profili drugih korisnika.
- Account stranica je profil korisnika. Po registraciji dobija default sliku. Klikom na Edit Account ima mogućnost menjanja game role-a, dodavanje slike i menjanje about sekcije.
- Ucitani su stolovi koji pripadaju korisniku, kao i tabela sa njegovim komentarima, datumom i vremenom, i linkom ka stolu na kom je komentar ostavljen. Klikom na New Table prelazi na stranicu za kreiranje novog stola, svi podaci osim slike su obavezni.
- Admin stranica sadrži podatke o ostatku sajta, sve se dohvata i prosleđuje ajaxom i ispisuje u centralnom divu.
  - Activity – izlistane aktivnosti korisnika iz log.txt fajla, mogućnost filtriranja po izabranom datumu. Beleži se log in, logout, CRUD komentara, CRUD stolova, editovanje profila.
  - Users – izlistani svi korisnici uz mogućnost brisanja i editovanja. Moguće je promeniti username, password, ulogu(user ili admin). Front i back provera, kontroler proverava da li je request ajax ili ne. Brisanje ajaxom. Mogućnost kreiranja novog korisnika sa dodelom uloge.
  - Tables – izlistani stolovi uz mogućnost brisanja i editovanja. Moguće je editovanje naslova i opisa.
  - Comments – svi komentari poređani od novijih, sa stolovima i username-om korisnika. Omogućeno brisanje.
  - Game Systems – izlistani svi sistemi, mogućnost editovanja kao i kreiranja novih.
  - Messages – tabela iz baze sa mailovima koji su poslati adminu, moguće brisanje.
  - Ads – izlistane reklame sa linkovima, tekstom, slikama i statusom. Ako reklama nije aktivna, ne učitava se u slideshow na Tables stranici. Moguća izmena teksta, statusa i slike, kao i brisanje i kreiranje novih reklama.
  - Socials – lista linkova ka socijalnim mrežama koje su ispisane na Home stranici, moguće dodavanje novih, izmena i brisanje postojećih.

2. Organizacija

2.1 Baza podataka



tablefinder migrations
id : int(10) unsigned
migration : varchar(255)
batch : int(11)

tablefinder failed_jobs
id : bigint(20) unsigned
uuid : varchar(255)
connection : text
queue : text
payload : longtext
exception : longtext
failed_at : timestamp

tablefinder personal_access_tokens
id : bigint(20) unsigned
tokenable_type : varchar(255)
tokenable_id : bigint(20) unsigned
name : varchar(255)
token : varchar(64)
abilities : text
last_used_at : timestamp
created_at : timestamp
updated_at : timestamp

tablefinder game_roles
id : int(11)
name : varchar(255)
created_at : timestamp
updated_at : timestamp

tablefinder menus
id : int(11)
name : varchar(255)
route : varchar(255)
visibility : int(11)
created_at : timestamp
updated_at : timestamp

tablefinder messages
id : int(11)
email : varchar(255)
subject : varchar(255)
message : text
created_at : timestamp
updated_at : timestamp

tablefinder socials
id : int(11)
name : varchar(255)
link : varchar(255)
created_at : timestamp
updated_at : timestamp

tablefinder ads
id : int(11)
link : varchar(255)
text : text
image : varchar(255)
active : tinyint(1)
created_at : timestamp
updated_at : timestamp

tablefinder users_game_roles
id : int(11)
user_id : int(11)
game_roles_id : int(11)
created_at : timestamp
updated_at : timestamp

tablefinder comments
id : int(11)
table_id : int(11)
user_id : int(11)
text : text
created_at : timestamp
updated_at : timestamp

tablefinder levels
id : int(11)
name : varchar(255)
created_at : timestamp
updated_at : timestamp

tablefinder game_systems
id : int(11)
name : varchar(255)
created_at : timestamp
updated_at : timestamp

tablefinder users
id : int(11)
username : varchar(255)
email : varchar(255)
password : varchar(255)
image : varchar(255)
alt : varchar(255)
about : text
role_id : int(11)
remember_token : varchar(100)
created_at : timestamp
updated_at : timestamp

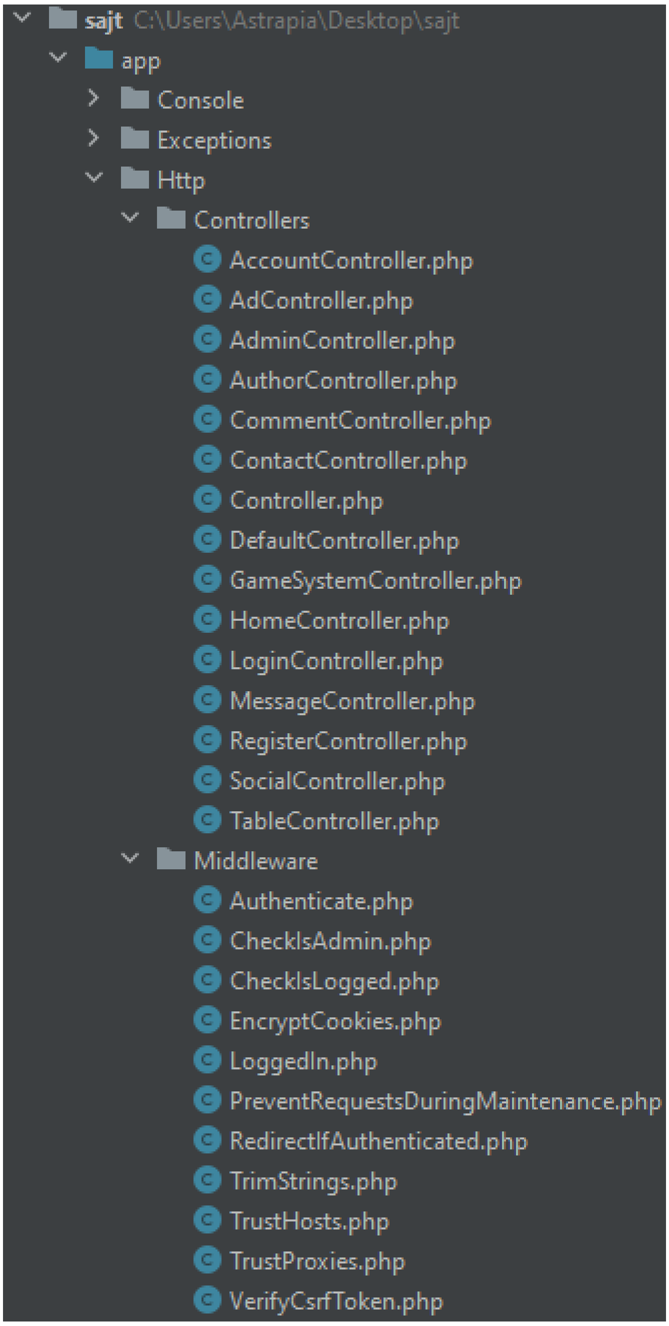
tablefinder tables
id : int(11)
name : varchar(255)
image : varchar(255)
alt : varchar(255)
about : text
game_master : tinyint(1)
date : date
user_id : int(11)
level_id : int(11)
game_system_id : int(11)
created_at : timestamp
updated_at : timestamp

tablefinder roles
id : int(11)
name : varchar(255)
created_at : timestamp
updated_at : timestamp

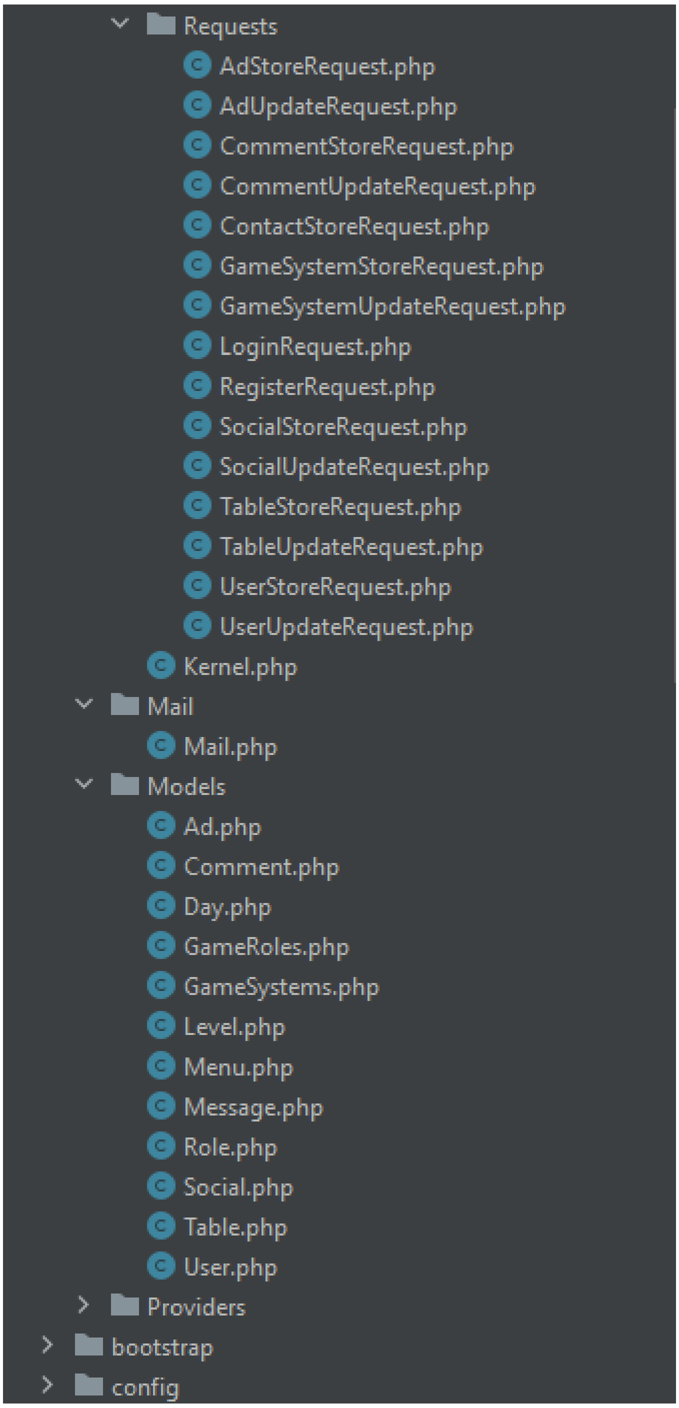
tablefinder day_table
id : int(11)
day_id : int(11)
table_id : int(11)
created_at : timestamp
updated_at : timestamp

tablefinder days
id : int(11)
name : varchar(255)
created_at : timestamp
updated_at : timestamp

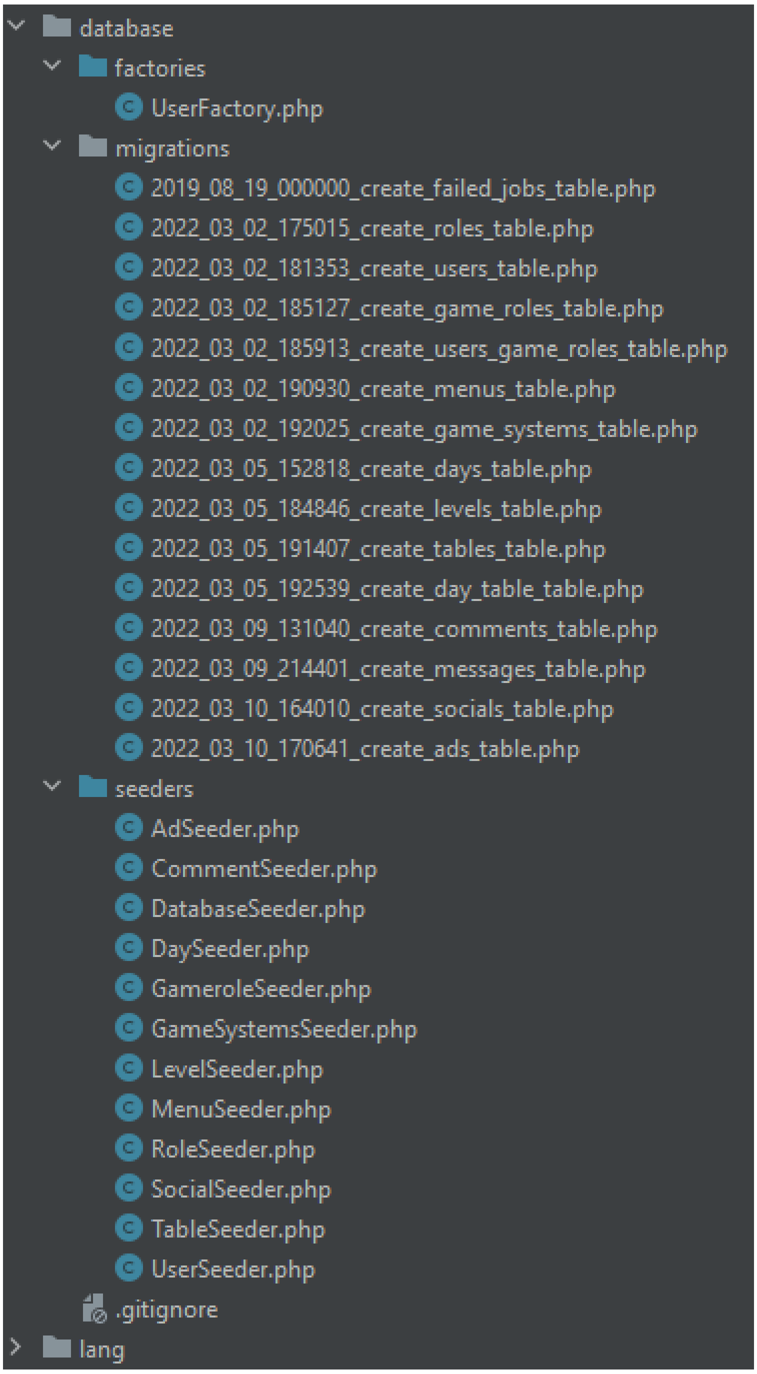
2.2 Struktura projekta



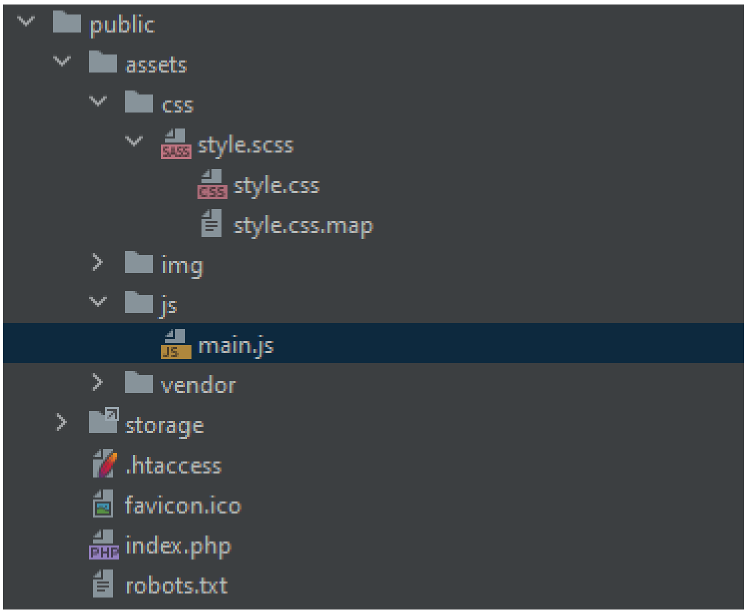
```
sajt C:\Users\Astrapia\Desktop\sajt
  app
    Console
    Exceptions
    Http
      Controllers
        AccountController.php
        AdController.php
        AdminController.php
        AuthorController.php
        CommentController.php
        ContactController.php
        Controller.php
        DefaultController.php
        GameSystemController.php
        HomeController.php
        LoginController.php
        MessageController.php
        RegisterController.php
        SocialController.php
        TableController.php
      Middleware
        Authenticate.php
        CheckIsAdmin.php
        CheckIsLogged.php
        EncryptCookies.php
        LoggedIn.php
        PreventRequestsDuringMaintenance.php
        RedirectIfAuthenticated.php
        TrimStrings.php
        TrustHosts.php
        TrustProxies.php
        VerifyCsrfToken.php
```



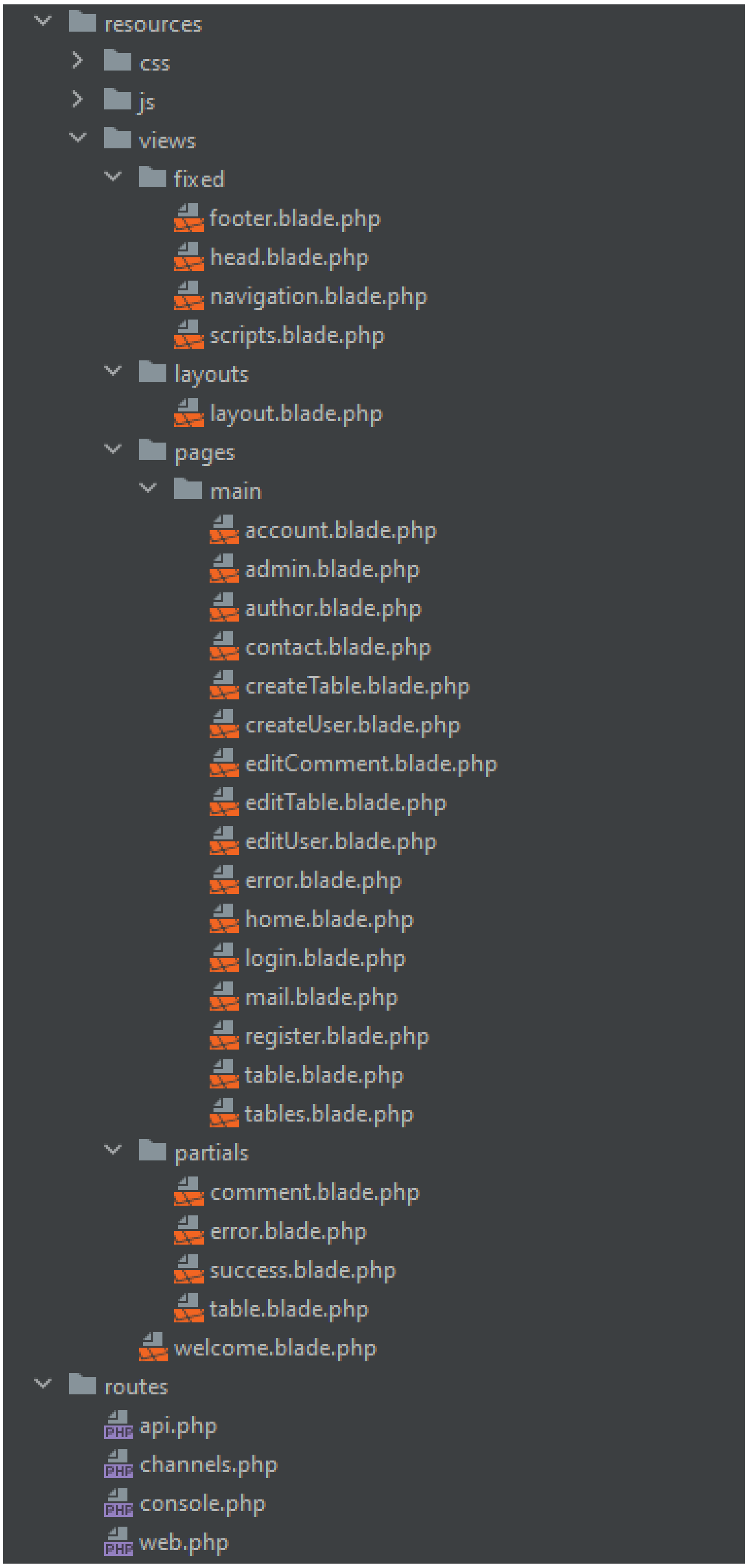
```
      ⌄ Requests
          AdStoreRequest.php
          AdUpdateRequest.php
          CommentStoreRequest.php
          CommentUpdateRequest.php
          ContactStoreRequest.php
          GameSystemStoreRequest.php
          GameSystemUpdateRequest.php
          LoginRequest.php
          RegisterRequest.php
          SocialStoreRequest.php
          SocialUpdateRequest.php
          TableStoreRequest.php
          TableUpdateRequest.php
          UserStoreRequest.php
          UserUpdateRequest.php
        Kernel.php
    ⌄ Mail
        Mail.php
    ⌄ Models
        Ad.php
        Comment.php
        Day.php
        GameRoles.php
        GameSystems.php
        Level.php
        Menu.php
        Message.php
        Role.php
        Social.php
        Table.php
        User.php
    › Providers
  › bootstrap
  › config
```



```
database
  factories
    UserFactory.php
  migrations
    2019_08_19_000000_create_failed_jobs_table.php
    2022_03_02_175015_create_roles_table.php
    2022_03_02_181353_create_users_table.php
    2022_03_02_185127_create_game_roles_table.php
    2022_03_02_185913_create_users_game_roles_table.php
    2022_03_02_190930_create_menus_table.php
    2022_03_02_192025_create_game_systems_table.php
    2022_03_05_152818_create_days_table.php
    2022_03_05_184846_create_levels_table.php
    2022_03_05_191407_create_tables_table.php
    2022_03_05_192539_create_day_table_table.php
    2022_03_09_131040_create_comments_table.php
    2022_03_09_214401_create_messages_table.php
    2022_03_10_164010_create_socials_table.php
    2022_03_10_170641_create_ads_table.php
  seeders
    AdSeeder.php
    CommentSeeder.php
    DatabaseSeeder.php
    DaySeeder.php
    GameroleSeeder.php
    GameSystemsSeeder.php
    LevelSeeder.php
    MenuSeeder.php
    RoleSeeder.php
    SocialSeeder.php
    TableSeeder.php
    UserSeeder.php
  .gitignore
lang
```



```
public
  assets
    css
      style.scss
        style.css
        style.css.map
    img
    js
      main.js
    vendor
  storage
  .htaccess
  favicon.ico
  index.php
  robots.txt
```



```
resources
    css
    js
    views
        fixed
            footer.blade.php
            head.blade.php
            navigation.blade.php
            scripts.blade.php
        layouts
            layout.blade.php
        pages
            main
                account.blade.php
                admin.blade.php
                author.blade.php
                contact.blade.php
                createTable.blade.php
                createUser.blade.php
                editComment.blade.php
                editTable.blade.php
                editUser.blade.php
                error.blade.php
                home.blade.php
                login.blade.php
                mail.blade.php
                register.blade.php
                table.blade.php
                tables.blade.php
        partials
            comment.blade.php
            error.blade.php
            success.blade.php
            table.blade.php
        welcome.blade.php
routes
    api.php
    channels.php
    console.php
    web.php
```

3. Slike stranica

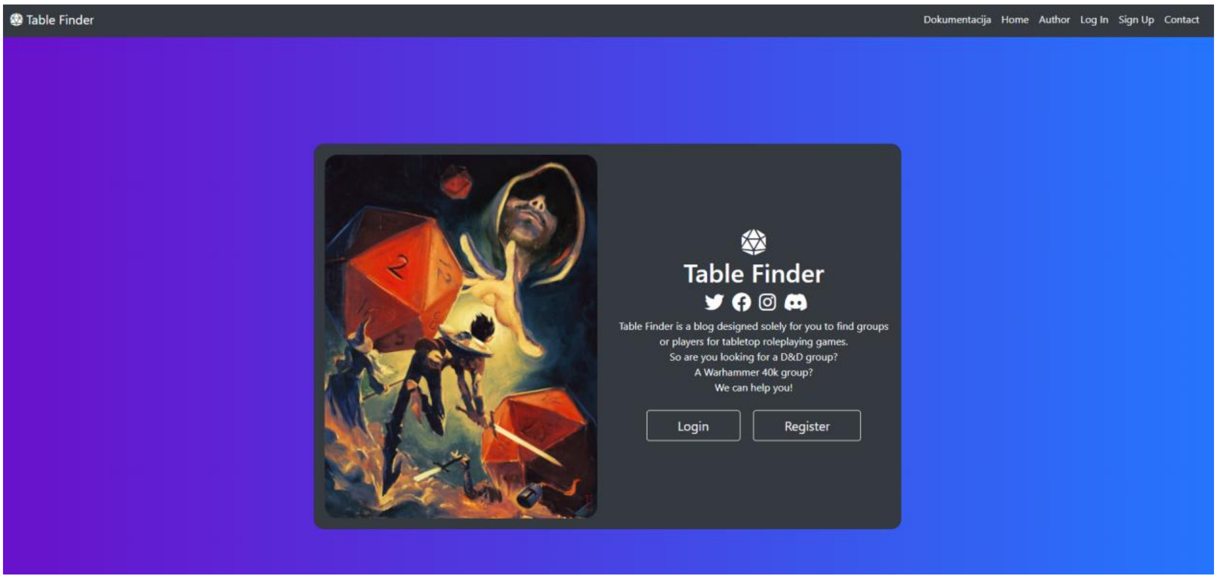
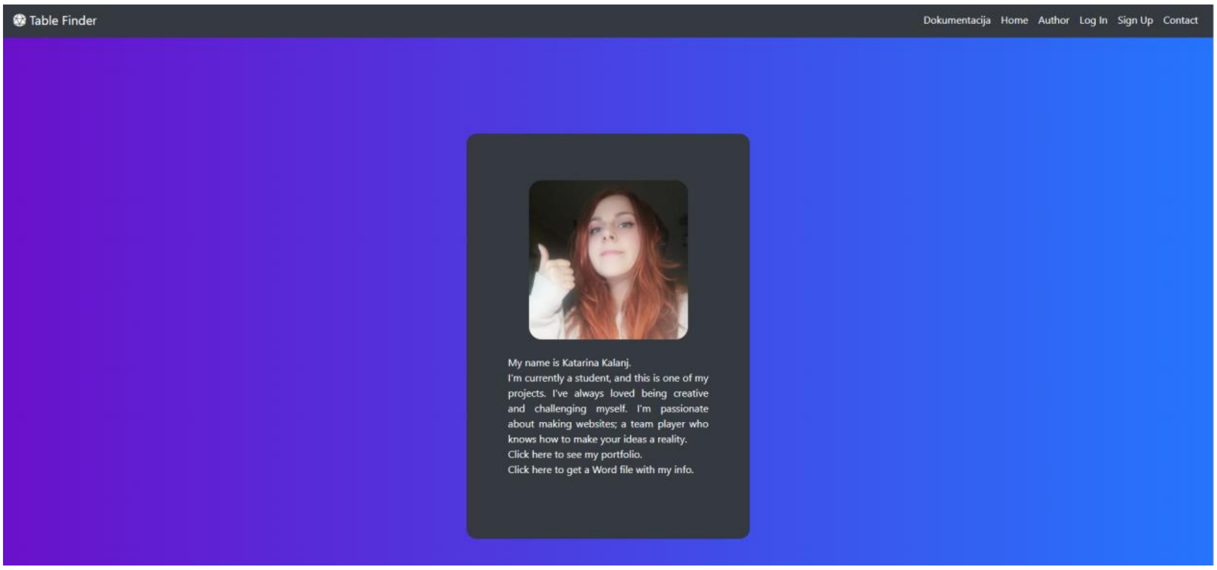
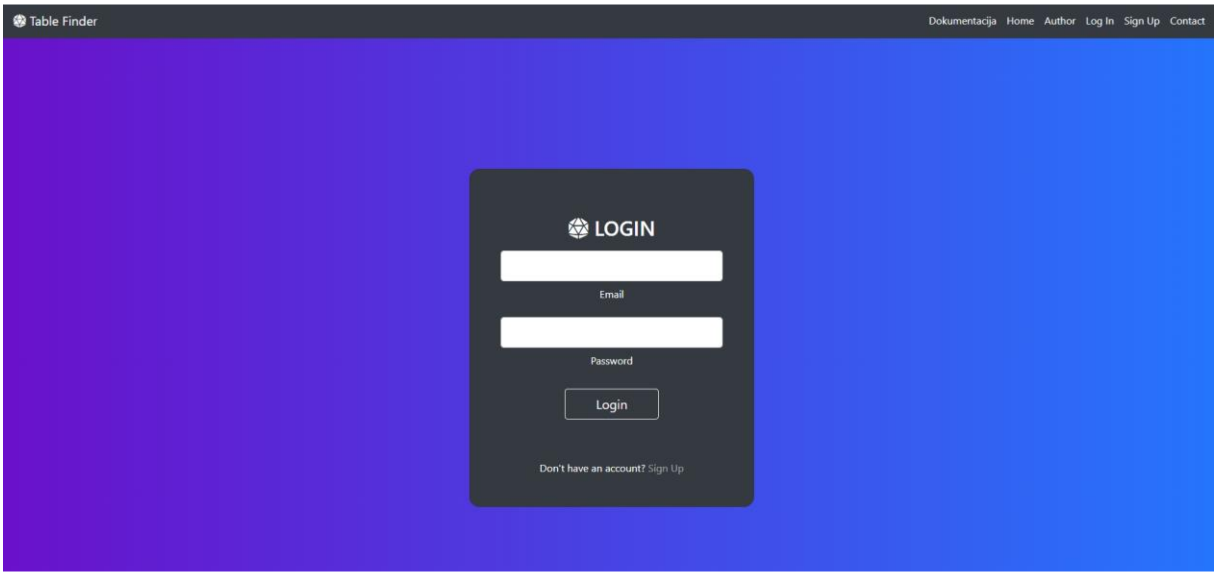


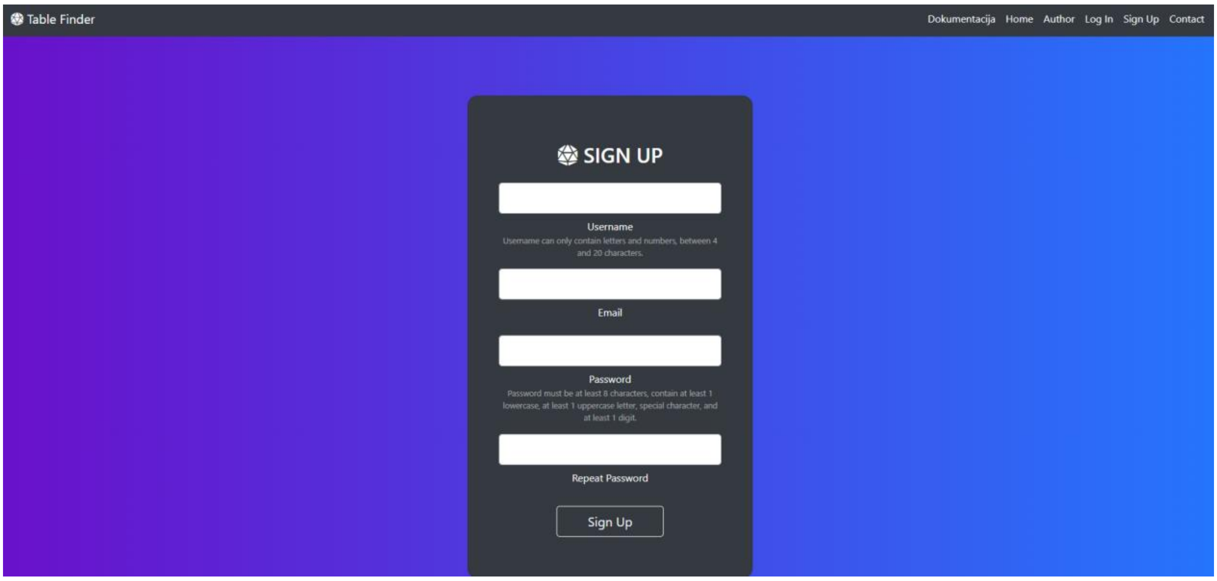
Table Finder is a blog designed solely for you to find groups
or players for tabletop roleplaying games.
So are you looking for a D&D group?
A Warhammer 40k group?
We can help you!



My name is Katarina Kalanj.
I'm currently a student, and this is one of my projects. I've always loved being creative and challenging myself. I'm passionate about making websites; a team player who knows how to make your ideas a reality.
Click here to see my portfolio.
Click here to get a Word file with my info.

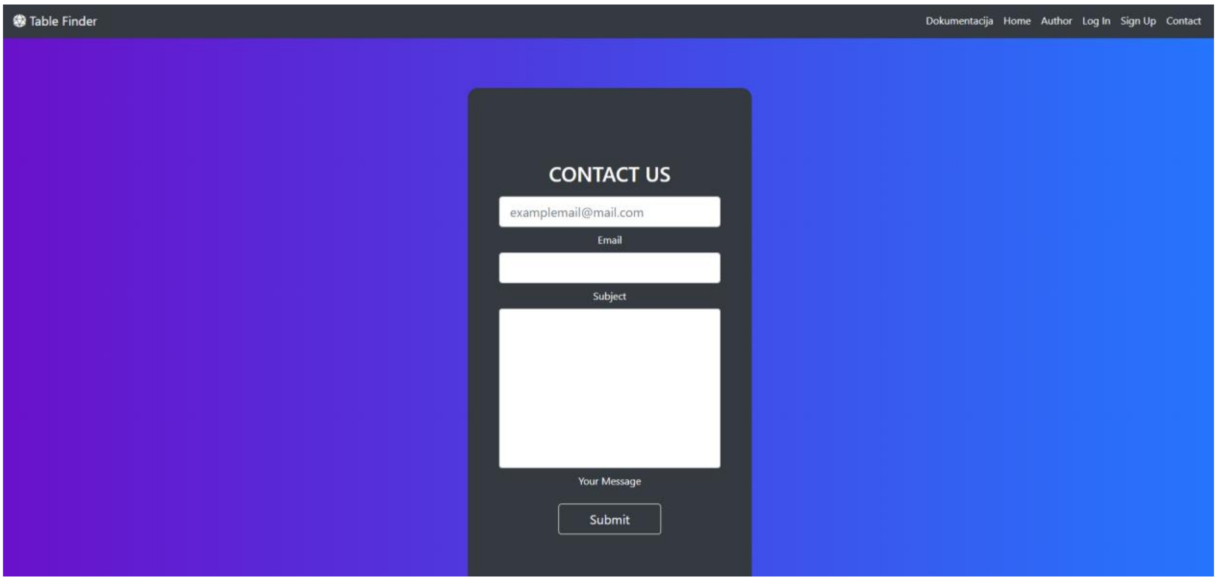


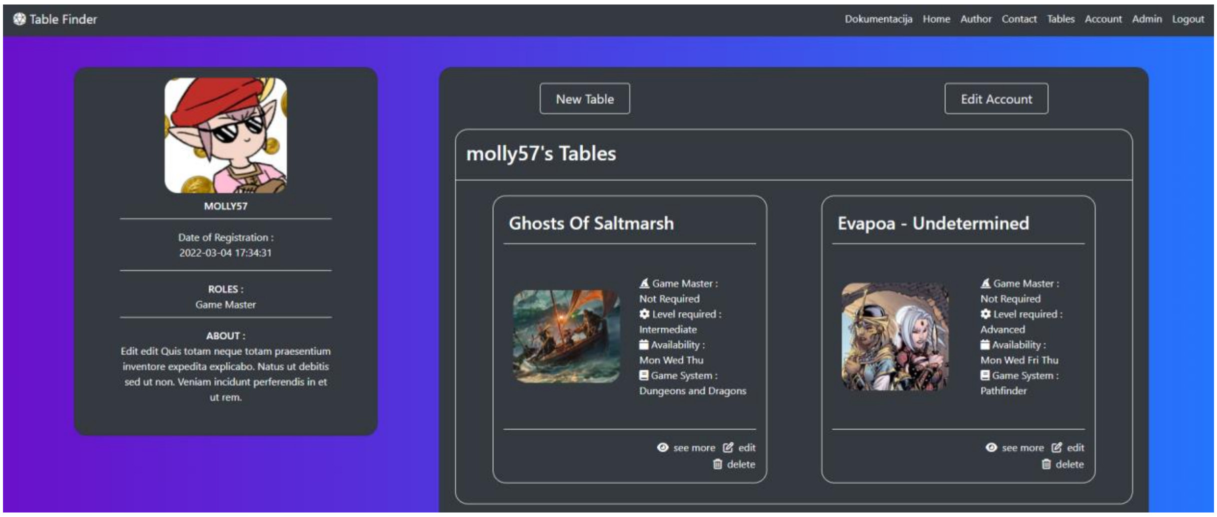
Don't have an account? Sign Up



Username can only contain letters and numbers, between 4 and 20 characters.

Password must be at least 8 characters, contain at least 1 lowercase, at least 1 uppercase letter, special character, and at least 1 digit.





MOLLY57

Date of Registration :
2022-03-04 17:34:31

ROLES :
Game Master

ABOUT :
Edit edit Quis totam neque totam praesentium inventore expedita explicabo. Natus ut debitis sed ut non. Veniam incidunt perferendis in et ut rem.

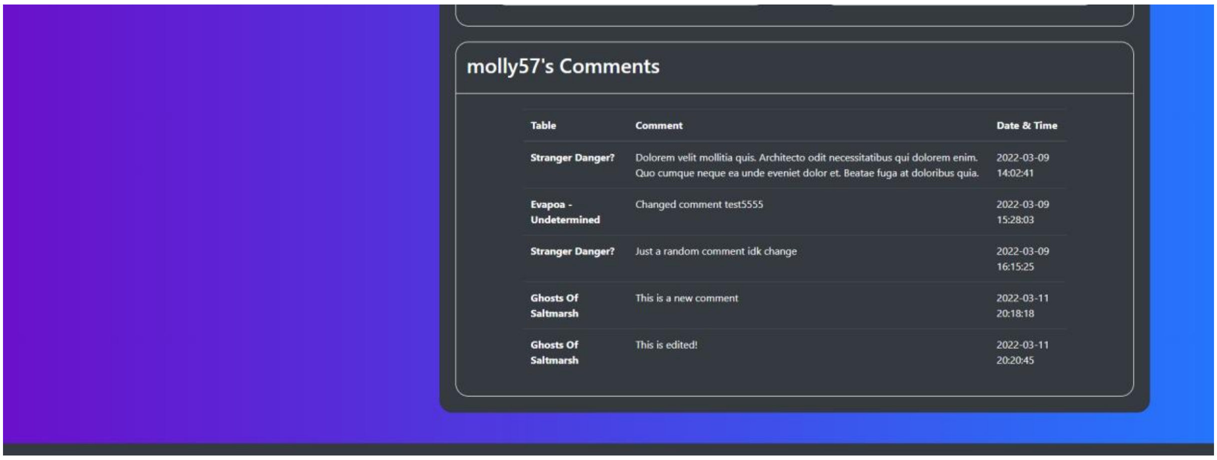
molly57's Tables

Ghosts Of Saltmarsh

Game Master :
Not Required
Level required :
Intermediate
Availability :
Mon Wed Thu
Game System :
Dungeons and Dragons

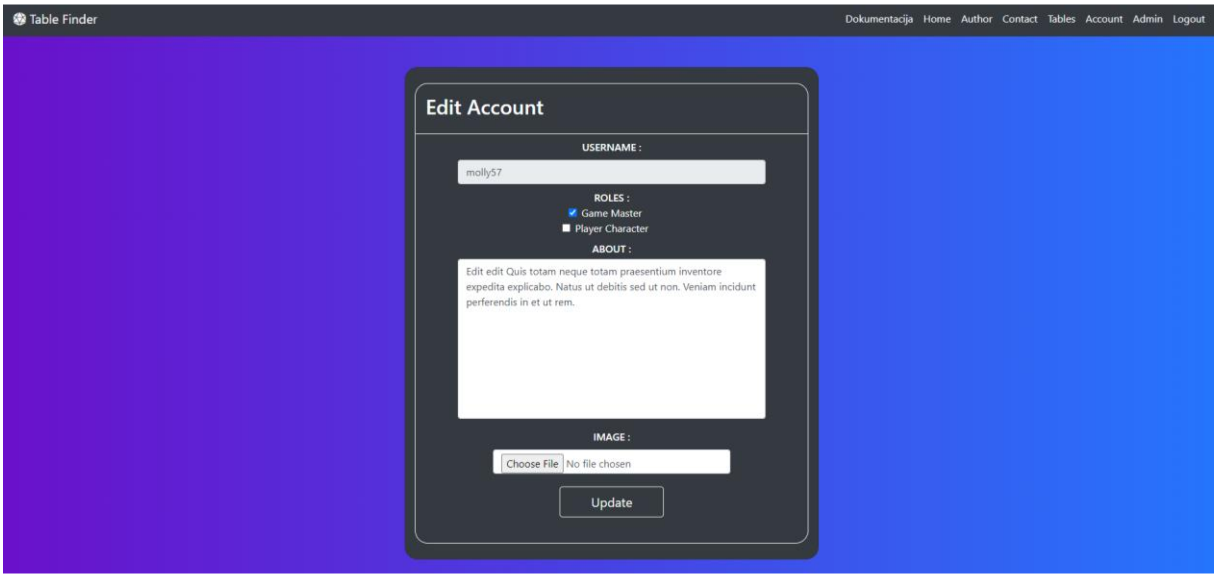
Evapoa - Undetermined

Game Master :
Not Required
Level required :
Advanced
Availability :
Mon Wed Fri Thu
Game System :
Pathfinder



molly57's Comments

Table	Comment	Date & Time
Stranger Danger?	Dolorem velit mollitia quis. Architecto odit necessitatibus qui dolorem enim. Quo cumque neque ea unde eveniet dolor et. Beatae fuga at doloribus quia.	2022-03-09 14:02:41
Evapoa - Undetermined	Changed comment test5555	2022-03-09 15:28:03
Stranger Danger?	Just a random comment idk change	2022-03-09 16:15:25
Ghosts Of Saltmarsh	This is a new comment	2022-03-11 20:18:18
Ghosts Of Saltmarsh	This is edited!	2022-03-11 20:20:45



Edit Account

USERNAME :

molly57

ROLES :

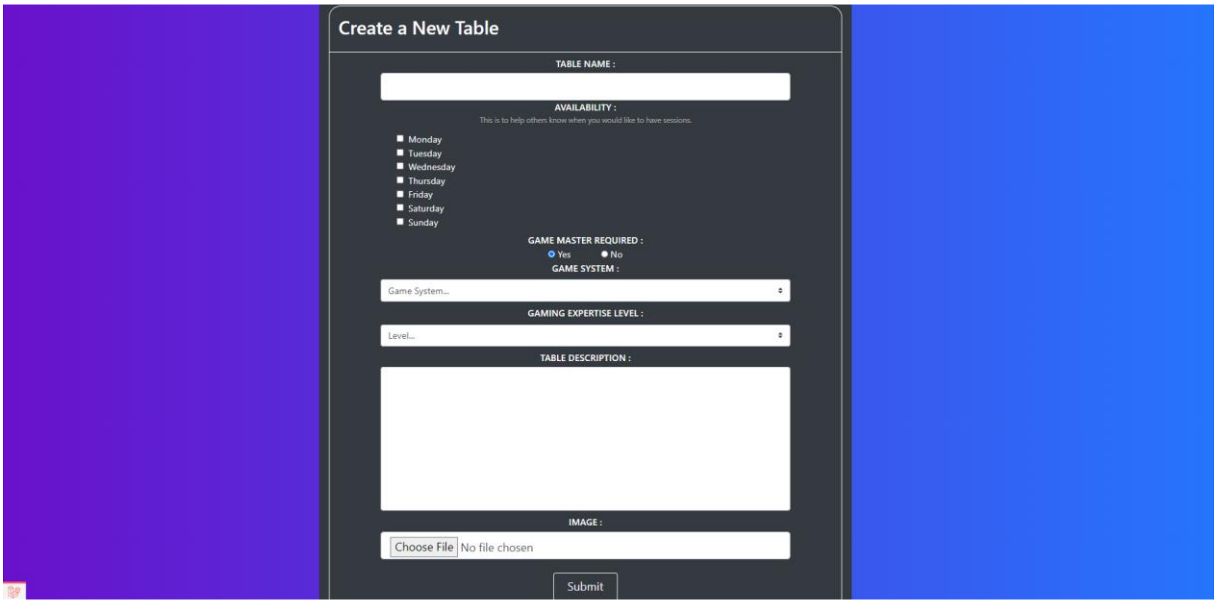
☑ Game Master
□ Player Character

ABOUT :

Edit edit Quis totam neque totam praesentium inventore expedita explicabo. Natus ut debitis sed ut non. Veniam incidunt perferendis in et ut rem.

IMAGE :

No file chosen



Create a New Table

TABLE NAME :

AVAILABILITY :

This is to help others know when you would like to have sessions.

□ Monday
□ Tuesday
□ Wednesday
□ Thursday
□ Friday
□ Saturday
□ Sunday

GAME MASTER REQUIRED :

● Yes ○ No

GAME SYSTEM :

Game System...

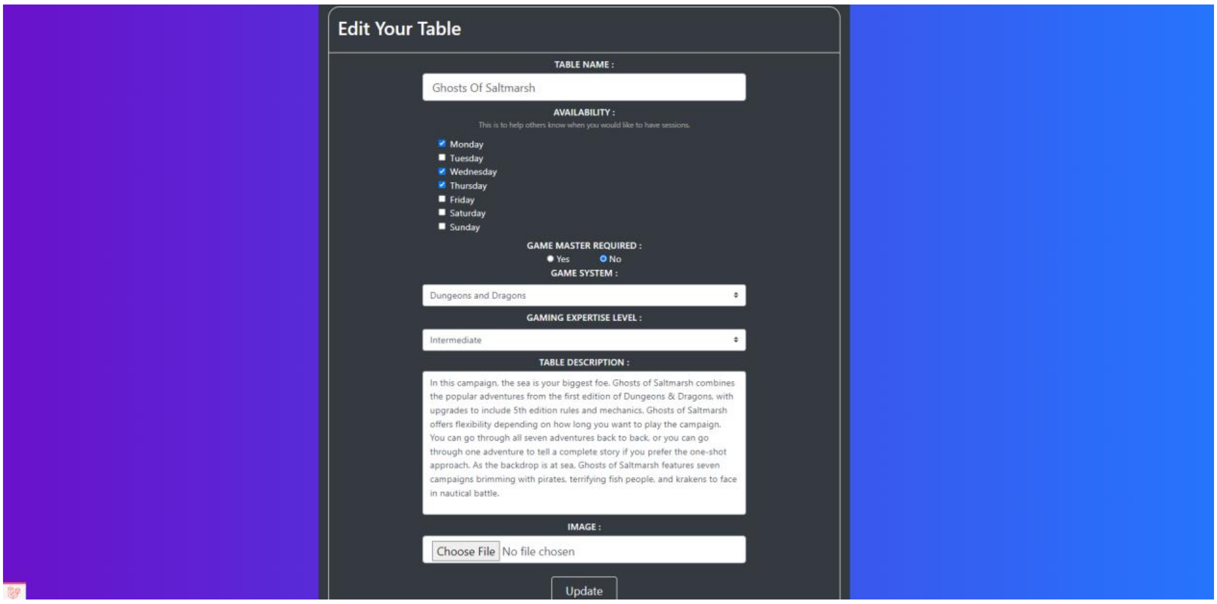
GAMING EXPERTISE LEVEL :

Level...

TABLE DESCRIPTION :

IMAGE :

No file chosen



Edit Your Table

TABLE NAME :

Ghosts Of Saltmarsh

AVAILABILITY :

This is to help others know when you would like to have sessions.

☑ Monday
□ Tuesday
☑ Wednesday
☑ Thursday
□ Friday
□ Saturday
□ Sunday

GAME MASTER REQUIRED :

○ Yes ● No

GAME SYSTEM :

Dungeons and Dragons

GAMING EXPERTISE LEVEL :

Intermediate

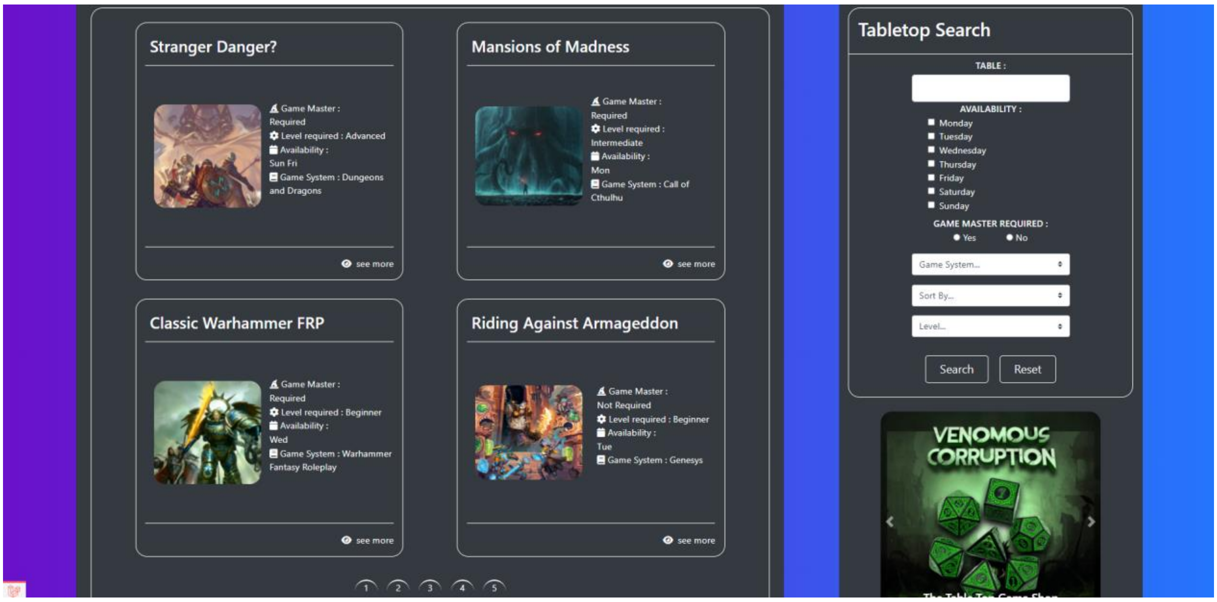
TABLE DESCRIPTION :

In this campaign, the sea is your biggest foe. Ghosts of Saltmarsh combines the popular adventures from the first edition of Dungeons & Dragons, with upgrades to include 5th edition rules and mechanics. Ghosts of Saltmarsh offers flexibility depending on how long you want to play the campaign. You can go through all seven adventures back to back, or you can go through one adventure to tell a complete story if you prefer the one-shot approach. As the backdrop is at sea, Ghosts of Saltmarsh features seven campaigns brimming with pirates, terrifying fish people, and krakens to face in nautical battle.

IMAGE :

Choose File No file chosen

Update



Stranger Danger?

Game Master : Required
Level required : Advanced
Availability : Sun Fri
Game System : Dungeons and Dragons

see more

Mansions of Madness

Game Master : Required
Level required : Intermediate
Availability : Mon
Game System : Call of Cthulhu

see more

Classic Warhammer FRP

Game Master : Required
Level required : Beginner
Availability : Wed
Game System : Warhammer Fantasy Roleplay

see more

Riding Against Armageddon

Game Master : Not Required
Level required : Beginner
Availability : Tue
Game System : Genesys

see more

1 2 3 4 5

Tabletop Search

TABLE :

AVAILABILITY :

□ Monday
□ Tuesday
□ Wednesday
□ Thursday
□ Friday
□ Saturday
□ Sunday

GAME MASTER REQUIRED :

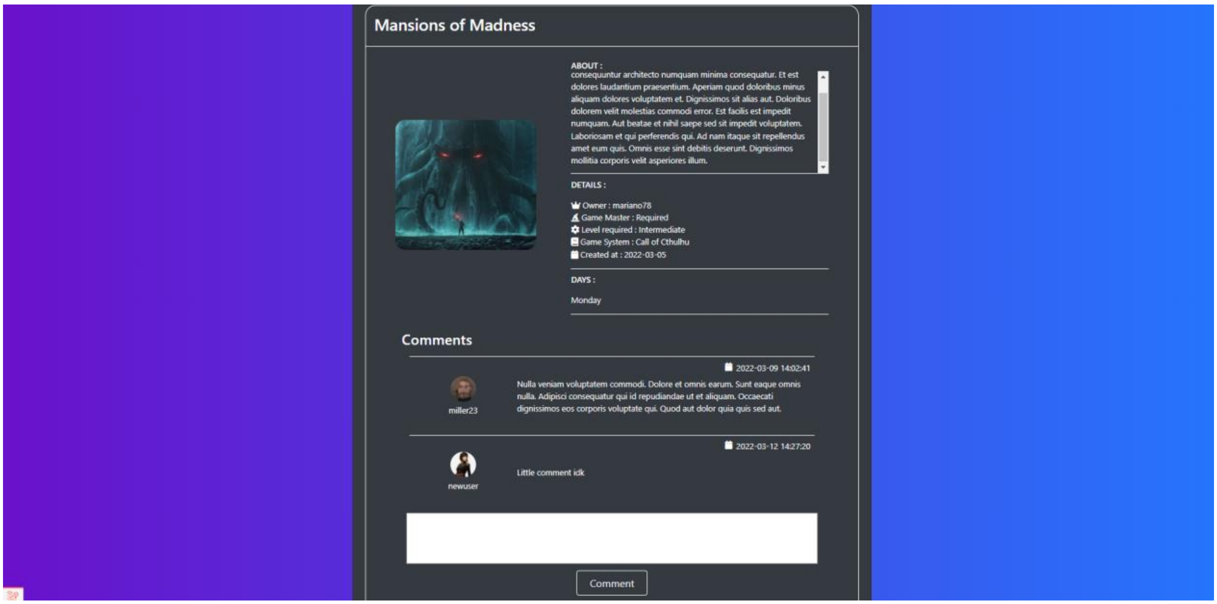
○ Yes ○ No

Game System...

Sort By...

Level...

Search Reset



Mansions of Madness

ABOUT :
consequuntur architecto numquam minima consequatur. Et est dolores laudantium praesentium. Aperiam quod doloribus minus aliquam dolores voluptatem et. Dignissimos sit alias aut. Doloribus dolorem velit molestias commodi error. Est facilis est impedit numquam. Aut beatae et nihil saepe sed sit impedit voluptatem. Laboriosam et qui perferendis qui. Ad nam itaque sit repellendus amet eum quis. Omnis esse sint debitis deserunt. Dignissimos mollitia corporis velit asperiores illum.

DETAILS :

Owner : mariano78
Game Master : Required
Level required : Intermediate
Game System : Call of Cthulhu
Created at : 2022-03-05

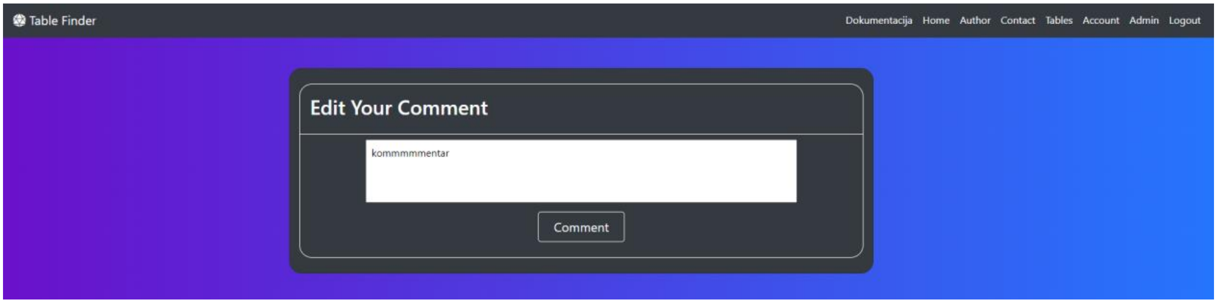
DAYS :

Monday

Comments

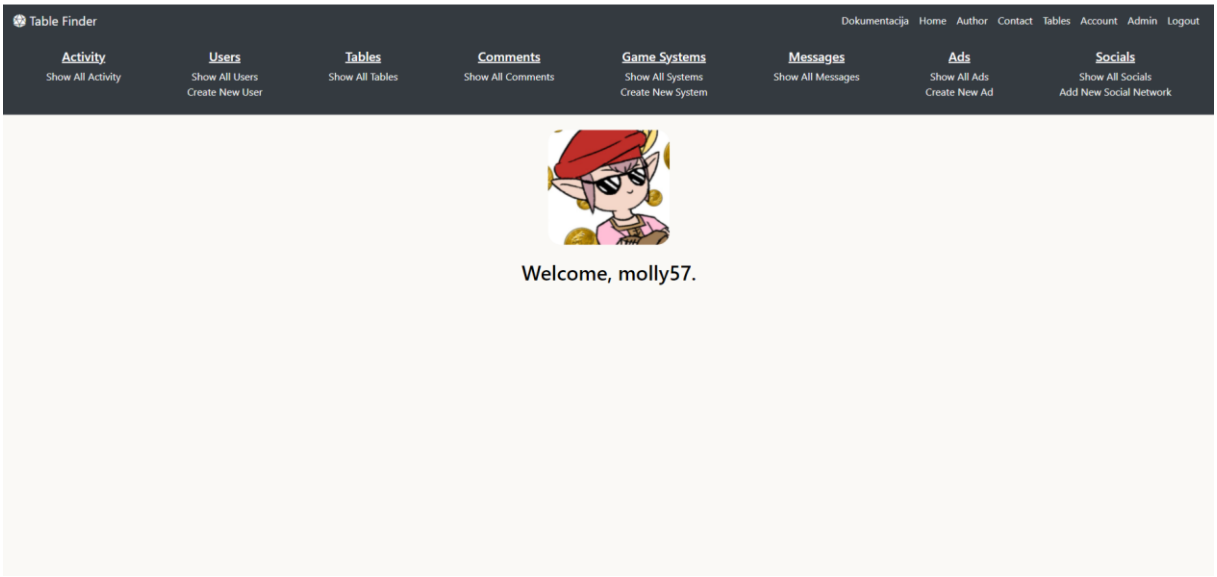
2022-03-09 14:02:41
miller23
Nulla veniam voluptatem commodi. Dolore et omnis earum. Sunt eaque omnis nulla. Adipisci consequatur qui id repudiandae ut et aliquam. Occaecati dignissimos eos corporis voluptate qui. Quod aut dolor quia quis sed aut.

2022-03-12 14:27:20
newuser
Little comment idk

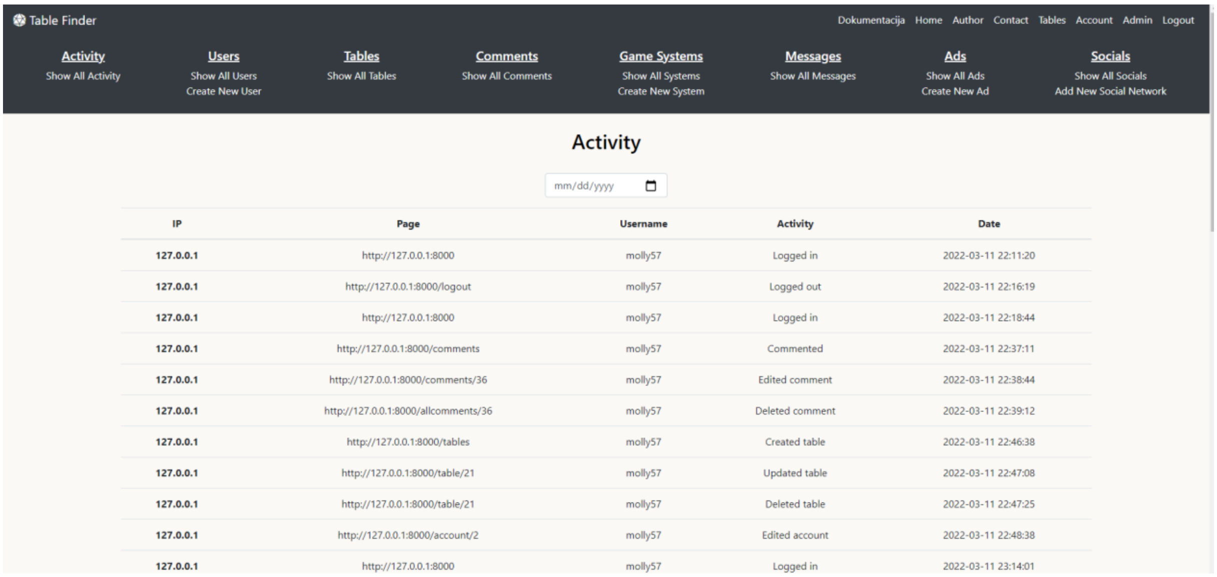


Edit Your Comment

kommmmmentar

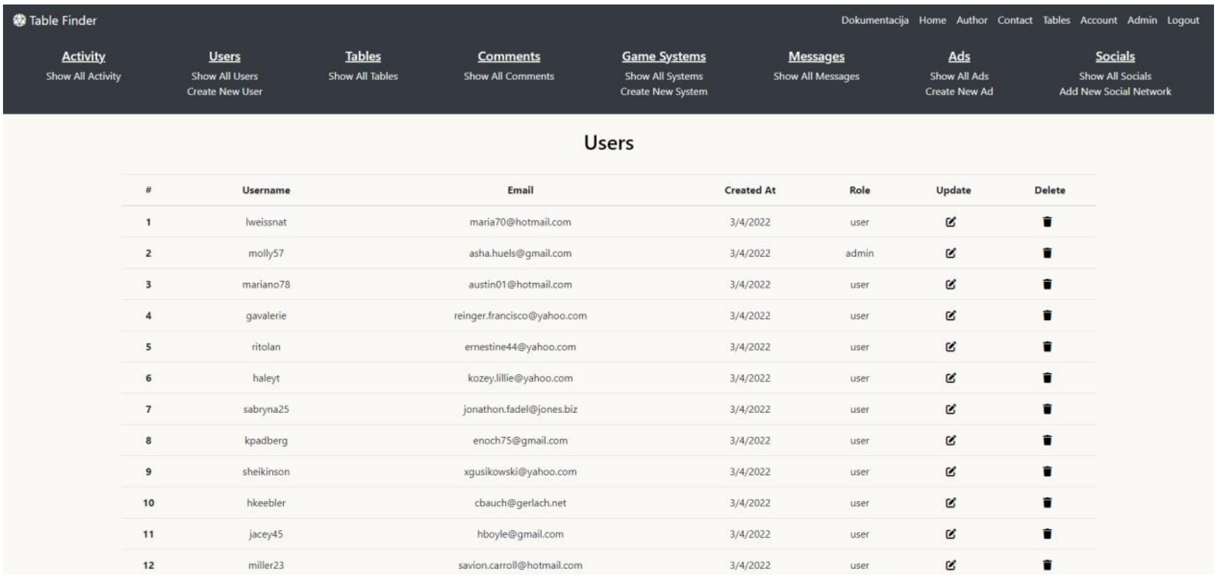


Welcome, molly57.



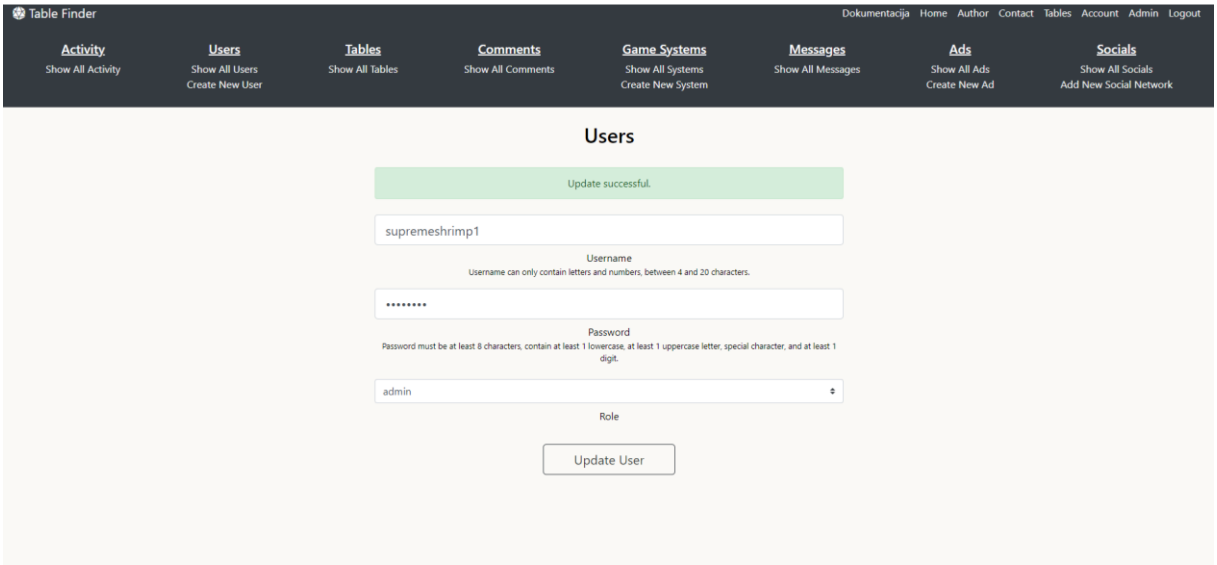
Activity

IP	Page	Username	Activity	Date
127.0.0.1	http://127.0.0.1:8000	molly57	Logged in	2022-03-11 22:11:20
127.0.0.1	http://127.0.0.1:8000/logout	molly57	Logged out	2022-03-11 22:16:19
127.0.0.1	http://127.0.0.1:8000	molly57	Logged in	2022-03-11 22:18:44
127.0.0.1	http://127.0.0.1:8000/comments	molly57	Commented	2022-03-11 22:37:11
127.0.0.1	http://127.0.0.1:8000/comments/36	molly57	Edited comment	2022-03-11 22:38:44
127.0.0.1	http://127.0.0.1:8000/allcomments/36	molly57	Deleted comment	2022-03-11 22:39:12
127.0.0.1	http://127.0.0.1:8000/tables	molly57	Created table	2022-03-11 22:46:38
127.0.0.1	http://127.0.0.1:8000/table/21	molly57	Updated table	2022-03-11 22:47:08
127.0.0.1	http://127.0.0.1:8000/table/21	molly57	Deleted table	2022-03-11 22:47:25
127.0.0.1	http://127.0.0.1:8000/account/2	molly57	Edited account	2022-03-11 22:48:38
127.0.0.1	http://127.0.0.1:8000	molly57	Logged in	2022-03-11 23:14:01



Users

#	Username	Email	Created At	Role	Update	Delete
1	lweissnat	maria70@hotmail.com	3/4/2022	user	✎	🗑
2	molly57	asha.huels@gmail.com	3/4/2022	admin	✎	🗑
3	mariano78	austin01@hotmail.com	3/4/2022	user	✎	🗑
4	gavalerie	reinger.francisco@yahoo.com	3/4/2022	user	✎	🗑
5	ritolan	ernestine44@yahoo.com	3/4/2022	user	✎	🗑
6	haleyt	kozey.lillie@yahoo.com	3/4/2022	user	✎	🗑
7	sabryna25	jonathon.fadel@jones.biz	3/4/2022	user	✎	🗑
8	kpadberg	enoch75@gmail.com	3/4/2022	user	✎	🗑
9	sheikinson	xgusikowski@yahoo.com	3/4/2022	user	✎	🗑
10	hkeebler	cbauch@gerlach.net	3/4/2022	user	✎	🗑
11	jacey45	hboyle@gmail.com	3/4/2022	user	✎	🗑
12	miller23	savion.carroll@hotmail.com	3/4/2022	user	✎	🗑



Users

Update successful.

supremeshrimp1

Username

Username can only contain letters and numbers, between 4 and 20 characters.

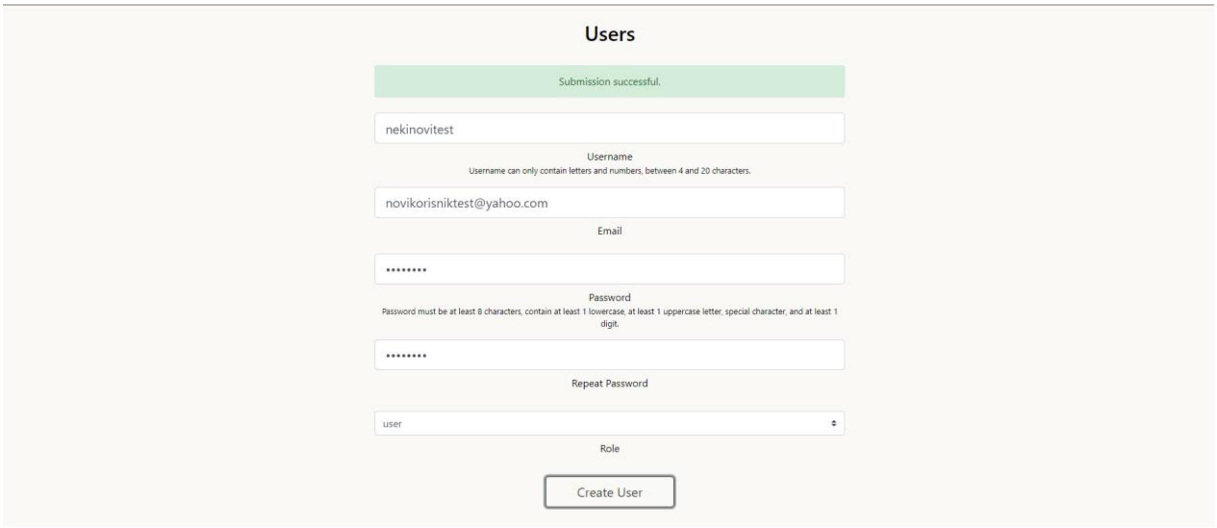
••••••••

Password

Password must be at least 8 characters, contain at least 1 lowercase, at least 1 uppercase letter, special character, and at least 1 digit.

admin

Role



Users

Submission successful.

nekinovitest

Username

Username can only contain letters and numbers, between 4 and 20 characters.

novikorisniktest@yahoo.com

Email

••••••••

Password

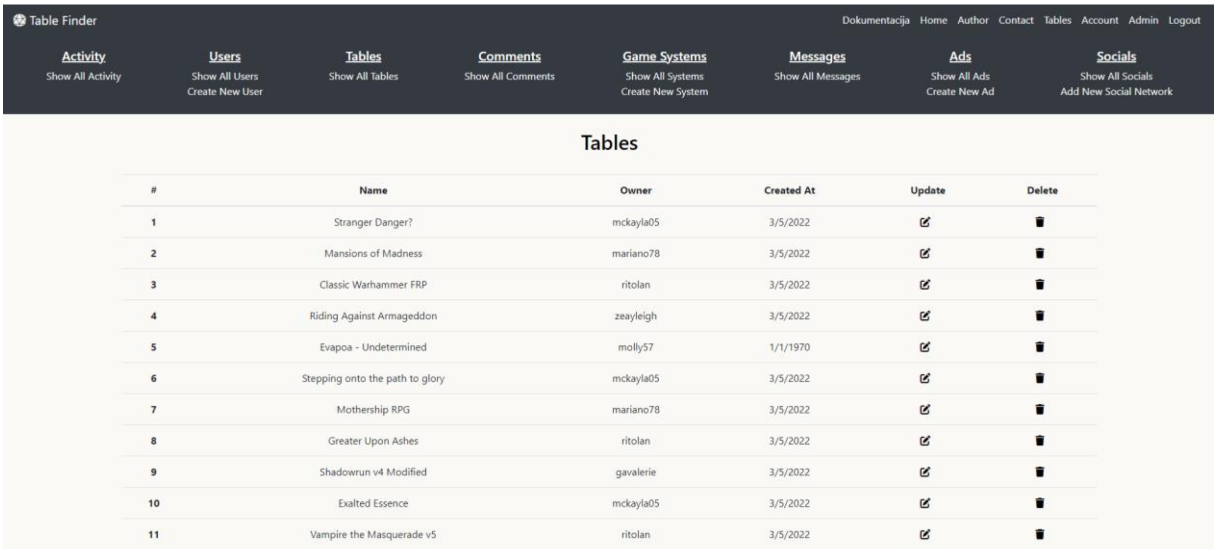
Password must be at least 8 characters, contain at least 1 lowercase, at least 1 uppercase letter, special character, and at least 1 digit.

••••••••

Repeat Password

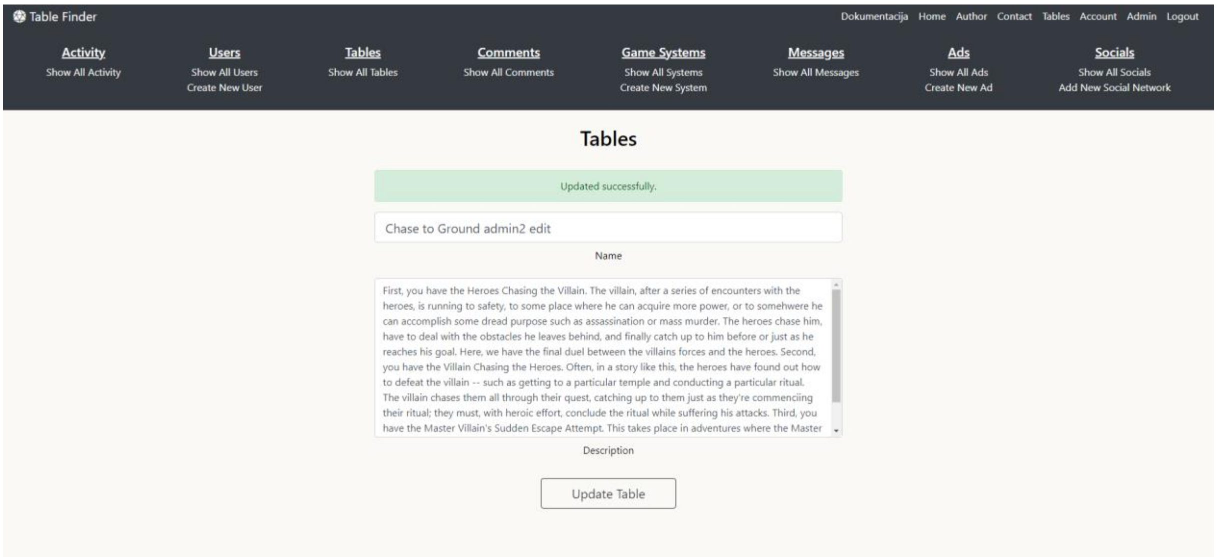
user

Role



Tables

#	Name	Owner	Created At	Update	Delete
1	Stranger Danger?	mckayla05	3/5/2022	✎	🗑
2	Mansions of Madness	mariano78	3/5/2022	✎	🗑
3	Classic Warhammer FRP	ritolan	3/5/2022	✎	🗑
4	Riding Against Armageddon	zeayleigh	3/5/2022	✎	🗑
5	Evapoa - Undetermined	molly57	1/1/1970	✎	🗑
6	Stepping onto the path to glory	mckayla05	3/5/2022	✎	🗑
7	Mothership RPG	mariano78	3/5/2022	✎	🗑
8	Greater Upon Ashes	ritolan	3/5/2022	✎	🗑
9	Shadowrun v4 Modified	gavalerie	3/5/2022	✎	🗑
10	Exalted Essence	mckayla05	3/5/2022	✎	🗑
11	Vampire the Masquerade v5	ritolan	3/5/2022	✎	🗑



Tables

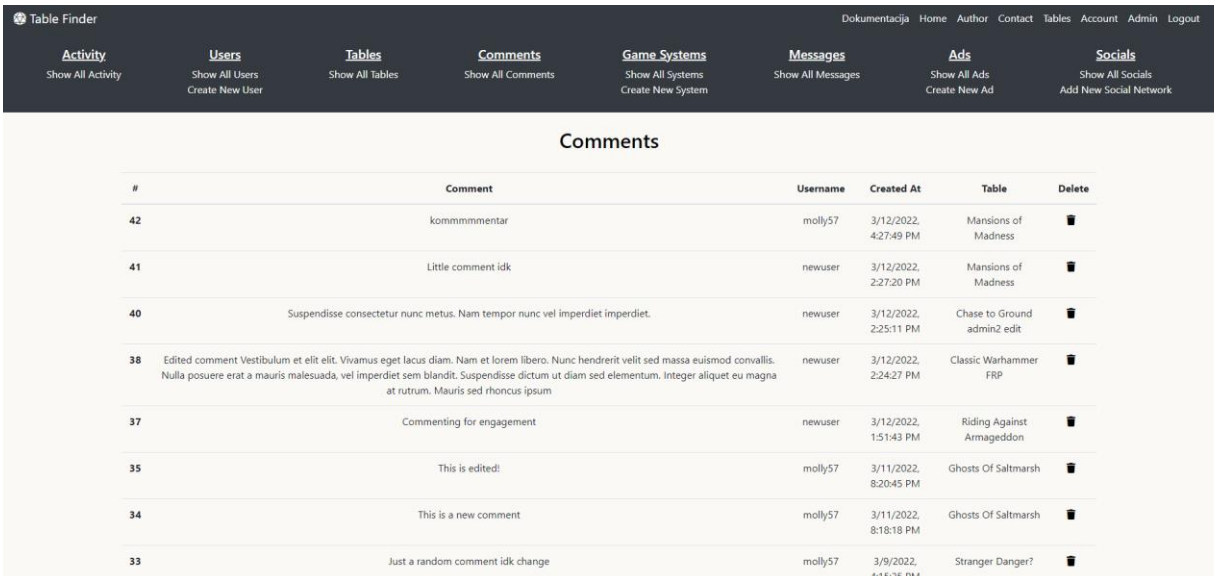
Updated successfully.

Chase to Ground admin2 edit

Name

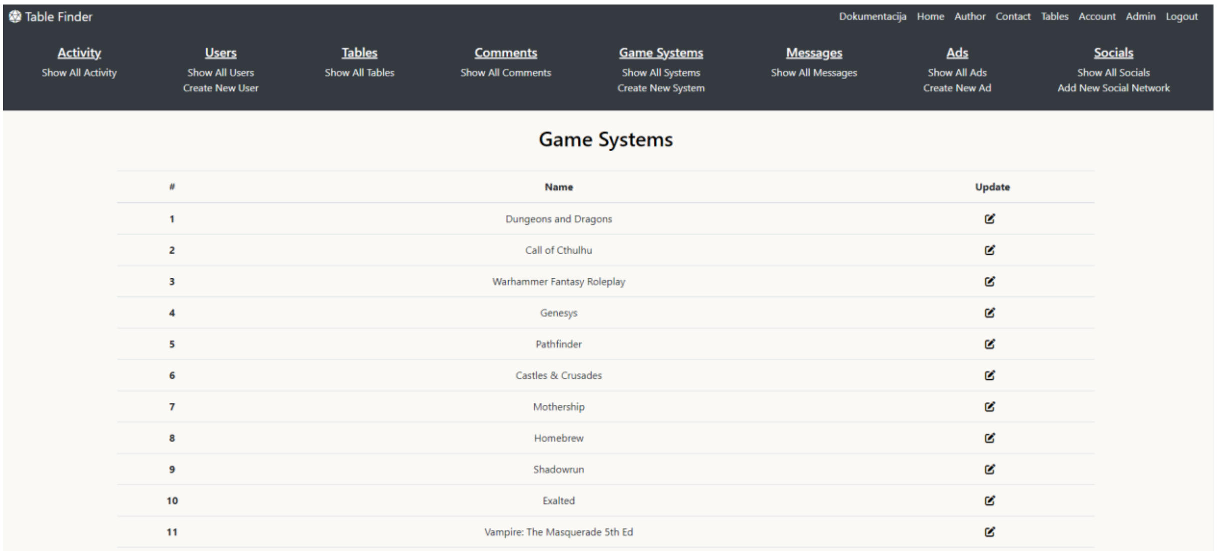
First, you have the Heroes Chasing the Villain. The villain, after a series of encounters with the heroes, is running to safety, to some place where he can acquire more power, or to somehwere he can accomplish some dread purpose such as assassination or mass murder. The heroes chase him, have to deal with the obstacles he leaves behind, and finally catch up to him before or just as he reaches his goal. Here, we have the final duel between the villains forces and the heroes. Second, you have the Villain Chasing the Heroes. Often, in a story like this, the heroes have found out how to defeat the villain -- such as getting to a particular temple and conducting a particular ritual. The villain chases them all through their quest, catching up to them just as they're commenciing their ritual; they must, with heroic effort, conclude the ritual while suffering his attacks. Third, you have the Master Villain's Sudden Escape Attempt. This takes place in adventures where the Master

Description



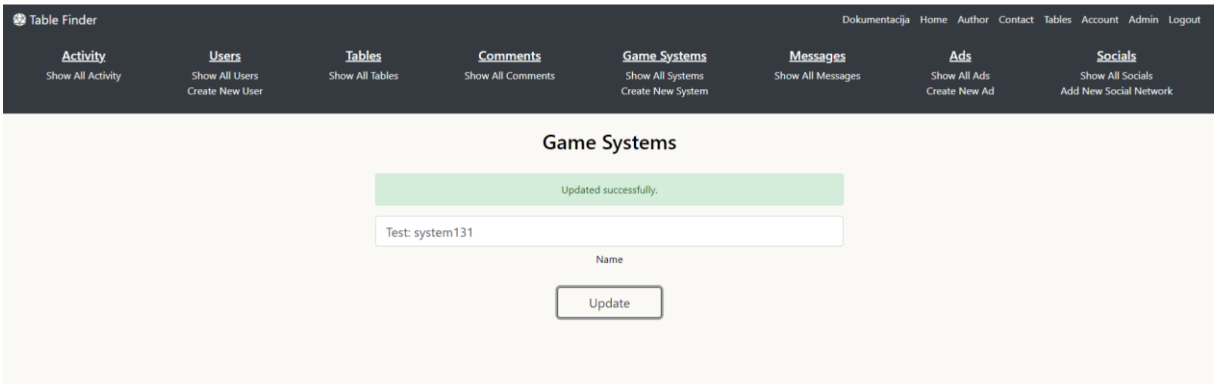
Comments

#	Comment	Username	Created At	Table	Delete
42	kommmmmentar	molly57	3/12/2022, 4:27:49 PM	Mansions of Madness	🗑
41	Little comment idk	newuser	3/12/2022, 2:27:20 PM	Mansions of Madness	🗑
40	Suspendisse consectetur nunc metus. Nam tempor nunc vel imperdiet imperdiet.	newuser	3/12/2022, 2:25:11 PM	Chase to Ground admin2 edit	🗑
38	Edited comment Vestibulum et elit elit. Vivamus eget lacus diam. Nam et lorem libero. Nunc hendrerit velit sed massa euismod convallis. Nulla posuere erat a mauris malesuada, vel imperdiet sem blandit. Suspendisse dictum ut diam sed elementum. Integer aliquet eu magna at rutrum. Mauris sed rhoncus ipsum	newuser	3/12/2022, 2:24:27 PM	Classic Warhammer FRP	🗑
37	Commenting for engagement	newuser	3/12/2022, 1:51:43 PM	Riding Against Armageddon	🗑
35	This is edited!	molly57	3/11/2022, 8:20:45 PM	Ghosts Of Saltmarsh	🗑
34	This is a new comment	molly57	3/11/2022, 8:18:18 PM	Ghosts Of Saltmarsh	🗑
33	Just a random comment idk change	molly57	3/9/2022,	Stranger Danger?	🗑



Game Systems

#	Name	Update
1	Dungeons and Dragons	✎
2	Call of Cthulhu	✎
3	Warhammer Fantasy Roleplay	✎
4	Genesys	✎
5	Pathfinder	✎
6	Castles & Crusades	✎
7	Mothership	✎
8	Homebrew	✎
9	Shadowrun	✎
10	Exalted	✎
11	Vampire: The Masquerade 5th Ed	✎

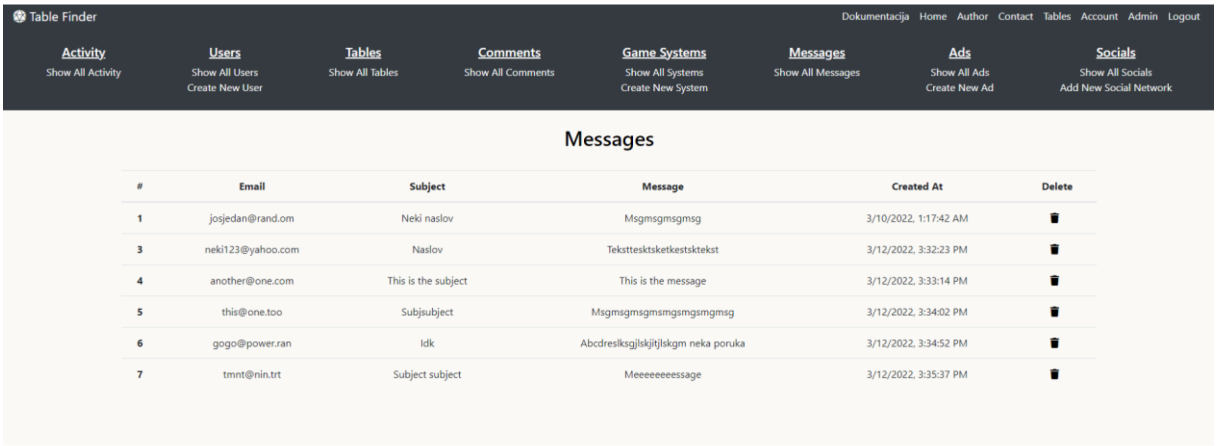


Game Systems

Updated successfully.

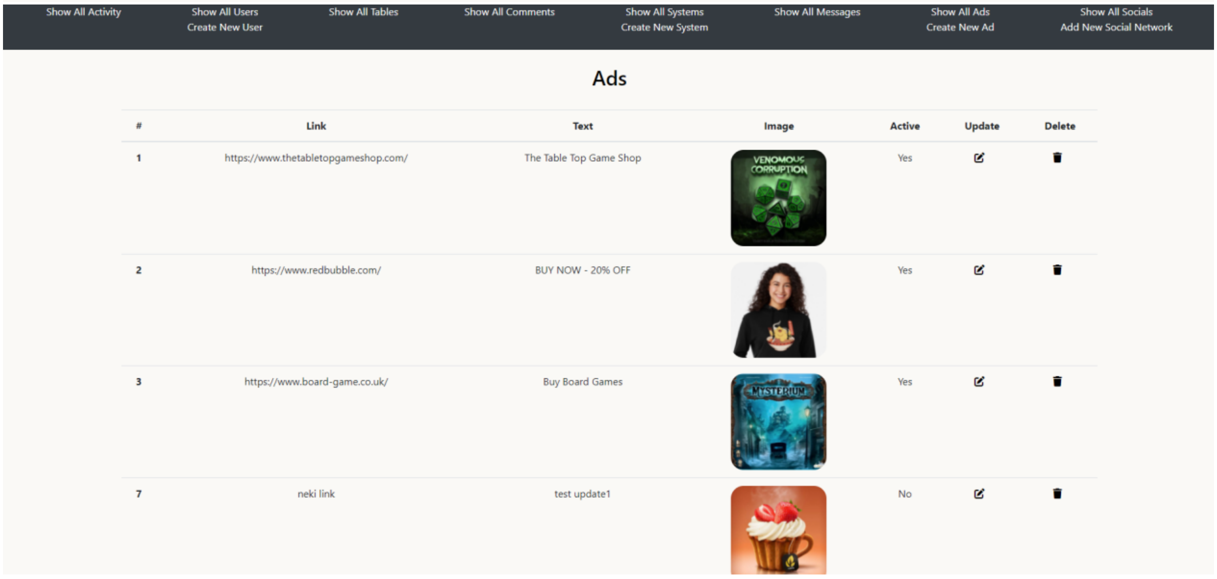
Test: system131

Name



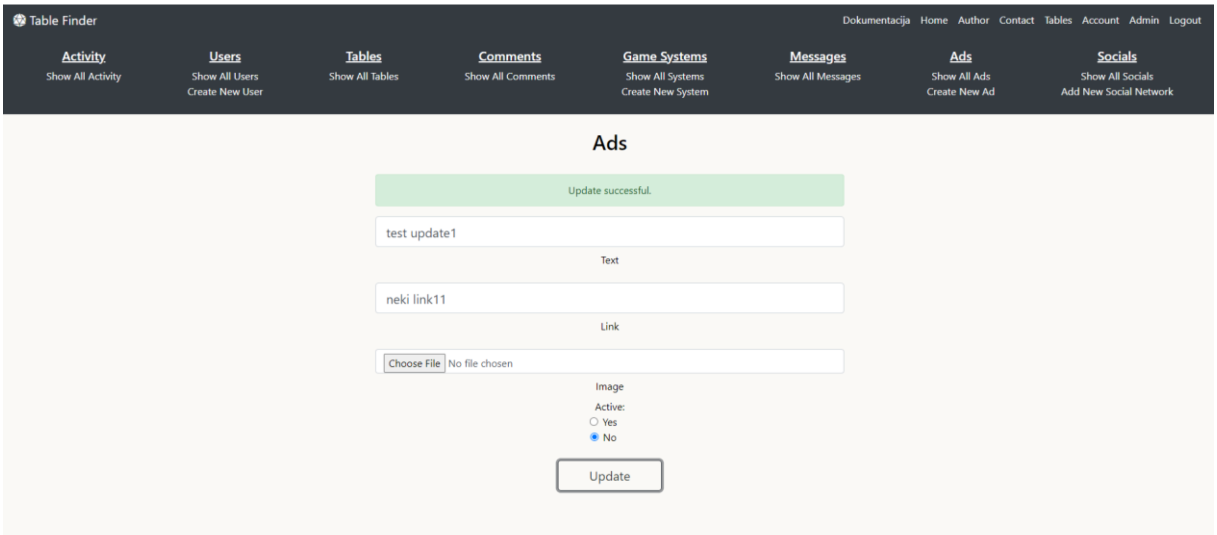
Messages

#	Email	Subject	Message	Created At	Delete
1	josjedan@rand.om	Neki naslov	Msgmsgmsgmsg	3/10/2022, 1:17:42 AM	🗑
3	neki123@yahoo.com	Naslov	Tekstteskstsketkestsktekst	3/12/2022, 3:32:23 PM	🗑
4	another@one.com	This is the subject	This is the message	3/12/2022, 3:33:14 PM	🗑
5	this@one.too	Subjsubject	Msgmsgmsgmsmgsmgsmgmsg	3/12/2022, 3:34:02 PM	🗑
6	gogo@power.ran	Idk	Abcdreslksgjlskjitjlskgm neka poruka	3/12/2022, 3:34:52 PM	🗑
7	tmnt@nin.trt	Subject subject	Meeeeeeeessage	3/12/2022, 3:35:37 PM	🗑



Ads

#	Link	Text	Image	Active	Update	Delete
1	https://www.thetabletopgameshop.com/	The Table Top Game Shop		Yes	✎	🗑
2	https://www.redbubble.com/	BUY NOW - 20% OFF		Yes	✎	🗑
3	https://www.board-game.co.uk/	Buy Board Games		Yes	✎	🗑
7	neki link	test update1		No	✎	🗑



Ads

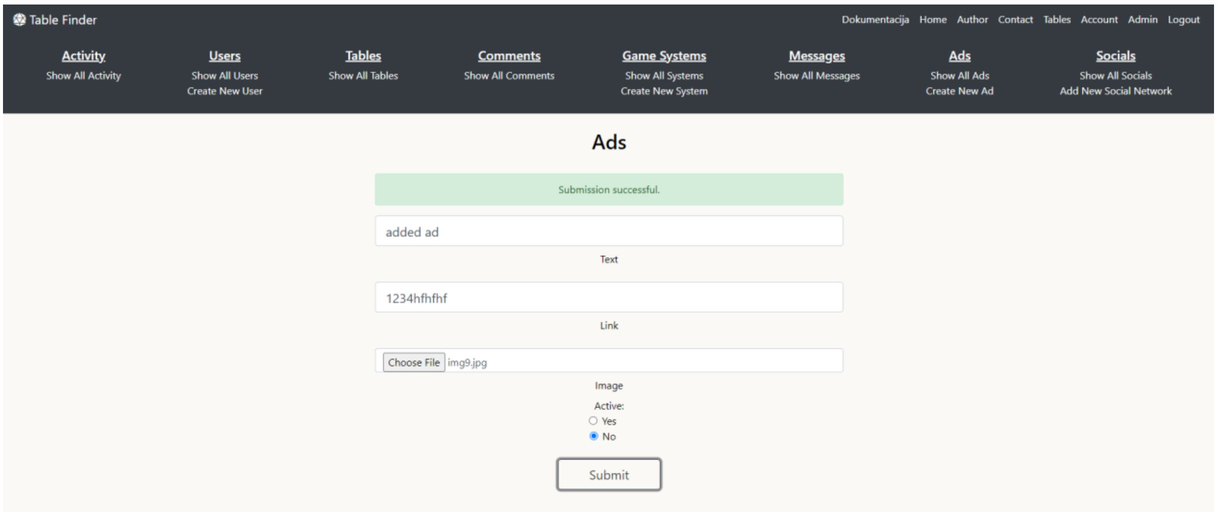
Update successful.

Text: test update1

Link: neki link11

Image: No file chosen

Active:
○ Yes
● No



Ads

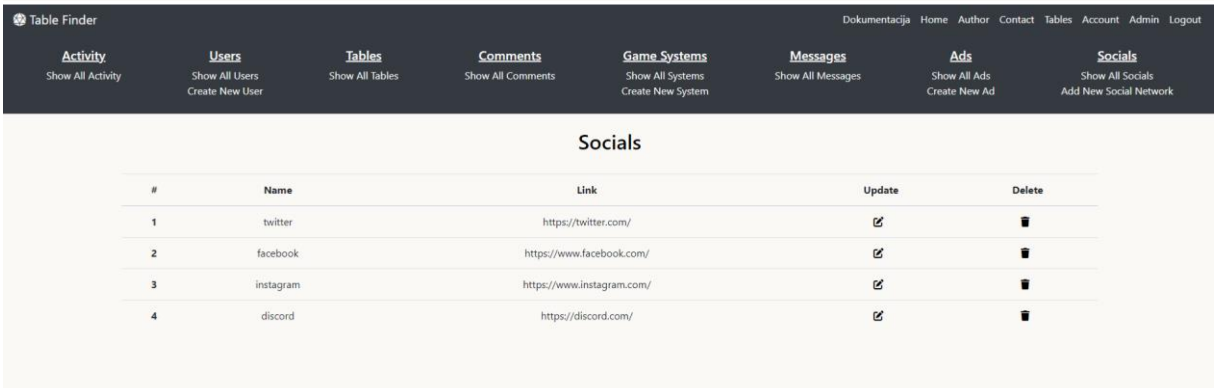
Submission successful.

Text: added ad

Link: 1234hfhfhf

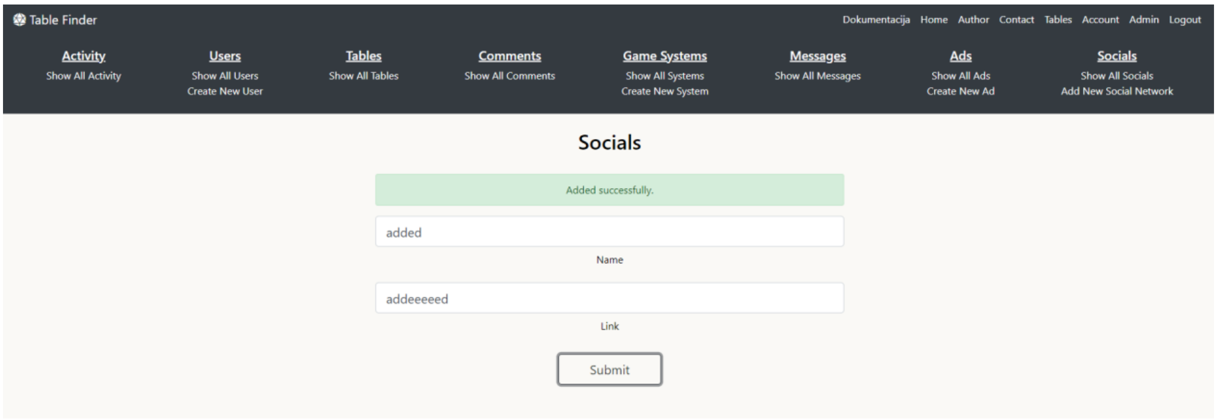
Image: img9.jpg

Active:
○ Yes
● No



Socials

#	Name	Link	Update	Delete
1	twitter	https://twitter.com/	✎	🗑
2	facebook	https://www.facebook.com/	✎	🗑
3	instagram	https://www.instagram.com/	✎	🗑
4	discord	https://discord.com/	✎	🗑

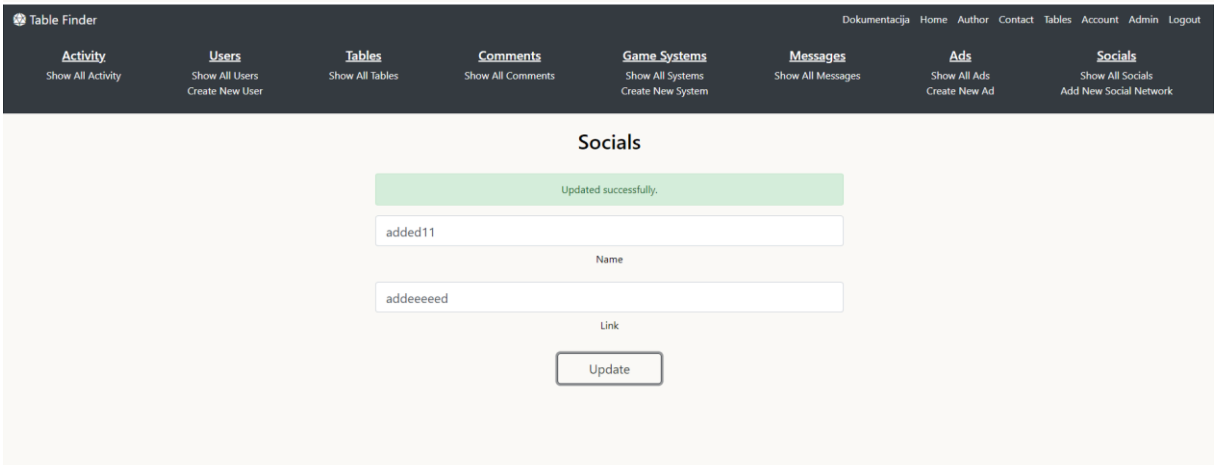


Socials

Added successfully.

Name: added

Link: addeeeeed



Socials

Updated successfully.

Name: added11

Link: addeeeeed

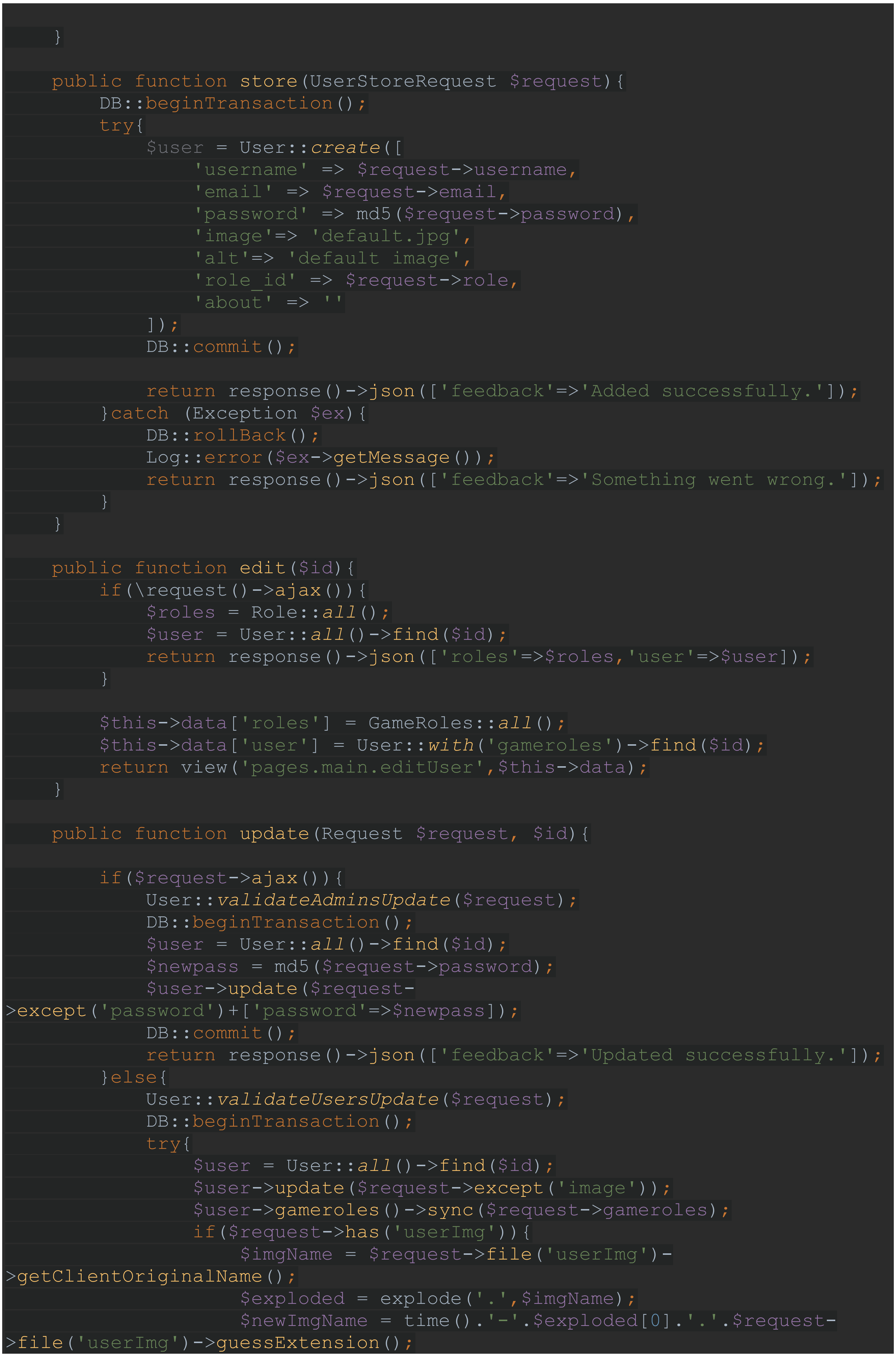
4. Kodovi

4.1 PHP

4.1.1 Controllers

4.1.1.1 AccountController.php

```php
<?php

namespace App\Http\Controllers;

use App\Http\Requests\UserStoreRequest;
use App\Http\Requests\UserUpdateRequest;
use App\Models\Comment;
use App\Models\Day;
use App\Models\GameRoles;
use App\Models\Table;
use App\Models\User;
use App\Models\Role;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use Illuminate\Support\Facades\Log;
use Exception;
use Illuminate\Support\Facades\DB;

class AccountController extends DefaultController
{
    protected $user;

    private $table;
    public function __construct()
    {
        parent::__construct();
        $this->table = new Table();
    }

    public function index(){
        $users = User::getAllUsers();
        return $users;
    }

    public function show($id){
        $testUser = new User();
        $comment = new Comment();
        $comments = $comment->getUsersComments($id);
        $user = User::with('gameroles')->find($id);
        $this->data['user'] = $user;
        $this->data['tables'] = $this->table->getAllTablesFromUser($id);
        $this->data['comments'] = $comments;
        return view('pages.main.account',$this->data);
    }

    public function create(Request $request){
        if($request->ajax()){
            $roles = Role::all();
            return response()->json(['roles'=>$roles]);
        }else{
            $this->data['roles'] = Role::all();
            return view('pages.main.createUser',$this->data);
        }
```

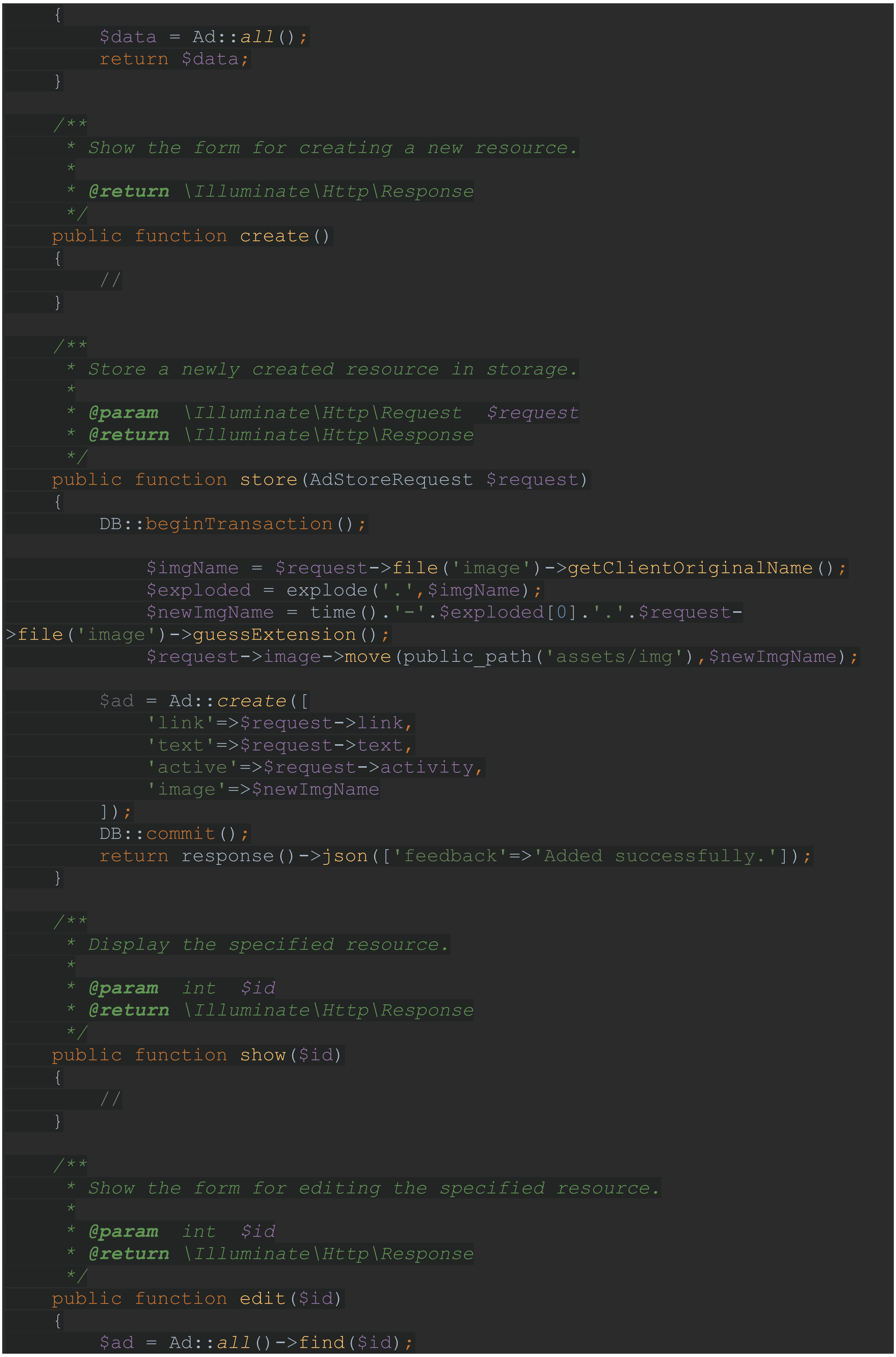


```
    }

    public function store(UserStoreRequest $request){
        DB::beginTransaction();
        try{
            $user = User::create([
                'username' => $request->username,
                'email' => $request->email,
                'password' => md5($request->password),
                'image'=> 'default.jpg',
                'alt'=> 'default image',
                'role_id' => $request->role,
                'about' => ''
            ]);
            DB::commit();

            return response()->json(['feedback'=>'Added successfully.']);
        }catch (Exception $ex){
            DB::rollBack();
            Log::error($ex->getMessage());
            return response()->json(['feedback'=>'Something went wrong.']);
        }
    }

    public function edit($id){
        if(\request()->ajax()){
            $roles = Role::all();
            $user = User::all()->find($id);
            return response()->json(['roles'=>$roles,'user'=>$user]);
        }

        $this->data['roles'] = GameRoles::all();
        $this->data['user'] = User::with('gameroles')->find($id);
        return view('pages.main.editUser',$this->data);
    }

    public function update(Request $request, $id){

        if($request->ajax()){
            User::validateAdminsUpdate($request);
            DB::beginTransaction();
            $user = User::all()->find($id);
            $newpass = md5($request->password);
            $user->update($request-
>except('password')+['password'=>$newpass]);
            DB::commit();
            return response()->json(['feedback'=>'Updated successfully.']);
        }else{
            User::validateUsersUpdate($request);
            DB::beginTransaction();
            try{
                $user = User::all()->find($id);
                $user->update($request->except('image'));
                $user->gameroles()->sync($request->gameroles);
                if($request->has('userImg')){
                    $imgName = $request->file('userImg')-
>getClientOriginalName();
                    $exploded = explode('.',$imgName);
                    $newImgName = time().'-'.$exploded[0].'.'.$request-
>file('userImg')->guessExtension();
```

```
                    $request->userImg-
>move(public_path('assets/img'),$newImgName);
                    $user->image = $newImgName;
                    $user->save();
                }
                ///////////
                $model = new User();
                $text=\request()->ip()."\t".\request()-
>url()."\t".session()->get('user')->username."\t"."Edited
account"."\t".date("Y-m-d H:i:s")."\n";
                $model->log($text);
                ///////////
                DB::commit();
                return redirect()->route('account',$id)-
>with('successMsg','Account updated.');
            }catch (Exception $ex){
                DB::rollBack();
                Log::error($ex->getMessage());
                return redirect()->route('account.update',$id)-
>with('errorMsg','Something went wrong.');
            }
        }

    }

    public function getAllUsers(){
        return $this->data['users'] = User::getAllUsers();
    }

    public function destroy($id){

        $user = User::all()->find($id);
        $user->gameroles()->detach();
        $user->delete();

        return response()->json(['feedback'=>'Deleted successfully.']);

    }

}
```

4.1.1.2 AdController.php

```
<?php

namespace App\Http\Controllers;

use App\Http\Requests\AdStoreRequest;
use App\Http\Requests\AdUpdateRequest;
use App\Models\Ad;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;

class AdController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
```



```
    {
        $data = Ad::all();
        return $data;
    }

    /**
     * Show the form for creating a new resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        //
    }

    /**
     * Store a newly created resource in storage.
     *
     * @param  \Illuminate\Http\Request  $request
     * @return \Illuminate\Http\Response
     */
    public function store(AdStoreRequest $request)
    {
        DB::beginTransaction();

            $imgName = $request->file('image')->getClientOriginalName();
            $exploded = explode('.',$imgName);
            $newImgName = time().'-'.$exploded[0].'.'.$request-
>file('image')->guessExtension();
            $request->image->move(public_path('assets/img'),$newImgName);

        $ad = Ad::create([
            'link'=>$request->link,
            'text'=>$request->text,
            'active'=>$request->activity,
            'image'=>$newImgName
        ]);
        DB::commit();
        return response()->json(['feedback'=>'Added successfully.']);
    }

    /**
     * Display the specified resource.
     *
     * @param  int  $id
     * @return \Illuminate\Http\Response
     */
    public function show($id)
    {
        //
    }

    /**
     * Show the form for editing the specified resource.
     *
     * @param  int  $id
     * @return \Illuminate\Http\Response
     */
    public function edit($id)
    {
        $ad = Ad::all()->find($id);
```

```php
        return response()->json($ad);
    }

    /**
     * Update the specified resource in storage.
     *
     * @param  \Illuminate\Http\Request  $request
     * @param  int  $id
     * @return \Illuminate\Http\Response
     */
    public function update(AdUpdateRequest $request, $id)
    {
        $ad = Ad::all()->find($id);
        DB::beginTransaction();

        if($request->has('image')){
            $imgName = $request->file('image')->getClientOriginalName();
            $exploded = explode('.',$imgName);
            $newImgName = time().'-'.$exploded[0].'.'.$request->file('image')->guessExtension();
            $request->image->move(public_path('assets/img'),$newImgName);
            $ad->update([
                'link'=>$request->link,
                'text'=>$request->text,
                'image'=>$newImgName,
                'active'=>$request->activity
            ]);
        }else{
            $ad->update([
                'link'=>$request->link,
                'text'=>$request->text,
                'active'=>$request->activity
            ]);
        }
        DB::commit();
        return response()->json(['data'=>'ok']);
    }

    /**
     * Remove the specified resource from storage.
     *
     * @param  int  $id
     * @return \Illuminate\Http\Response
     */
    public function destroy($id)
    {
        $ad = Ad::all()->find($id);
        $ad->delete();
        return response()->json(['msg'=>'Deleted']);
    }
}
```

4.1.1.3 AdminController.php

```php
<?php

namespace App\Http\Controllers;

use App\Models\Ad;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Storage;
```

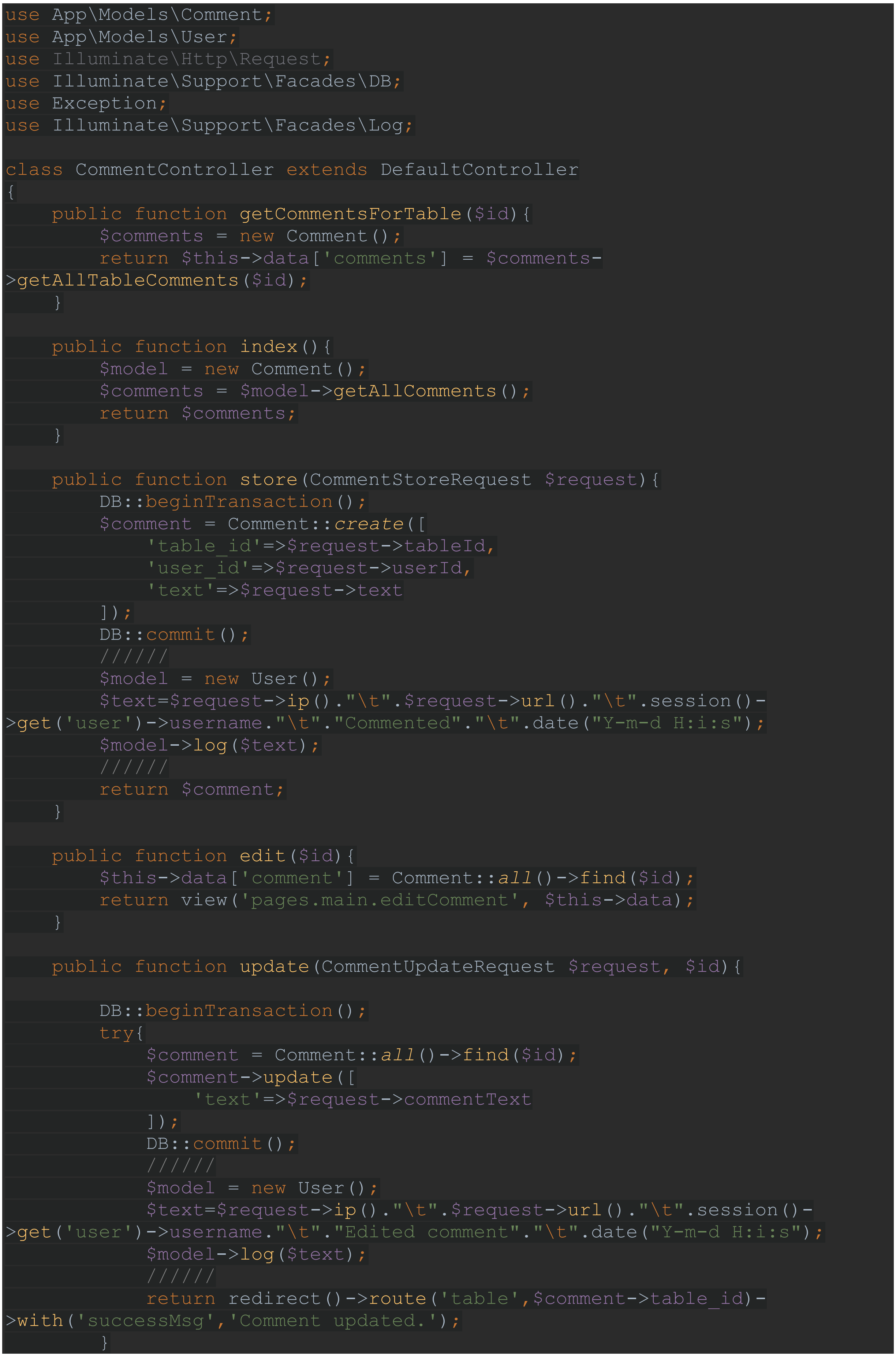
```php
class AdminController extends DefaultController
{
    public function index(){
        $this->data['test'] = Ad::where('active','1')->get();
        return view('pages.main.admin',$this->data);
    }

    public function readLog(Request $request){

        $filter = null;
        $send = null;
        if($request->has('filter')){
            $filter = $request->filter;
        }

        if(Storage::disk('local')->exists('log.txt')){
            $data = Storage::get('log.txt');
            $send = explode("\r\n",$data);
            if($filter){
                $filtered=[];
                foreach($send as $item){
                    $items = explode("\t",$item);
                    $date = explode(" ",$items[4]);
                    if($filter==$date[0]){
                        $filtered[]=$item;
                    }
                }
                $send=$filtered;
            }
        }
        return response()->json(['data'=>$send]);
    }

//    public function createUser(){
//        return view('pages.main.createUser',$this->data);
//    }

}
```

4.1.1.4 AuthorController.php

```php
<?php

namespace App\Http\Controllers;

class AuthorController extends DefaultController
{
    public function index(){
        return view('pages.main.author',$this->data);
    }
}
```

4.1.1.5 CommentController.php

```php
<?php

namespace App\Http\Controllers;

use App\Http\Requests\CommentStoreRequest;
use App\Http\Requests\CommentUpdateRequest;
```



```
use App\Models\Comment;
use App\Models\User;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;
use Exception;
use Illuminate\Support\Facades\Log;

class CommentController extends DefaultController
{
    public function getCommentsForTable($id){
        $comments = new Comment();
        return $this->data['comments'] = $comments-
>getAllTableComments($id);
    }

    public function index(){
        $model = new Comment();
        $comments = $model->getAllComments();
        return $comments;
    }

    public function store(CommentStoreRequest $request){
        DB::beginTransaction();
        $comment = Comment::create([
            'table_id'=>$request->tableId,
            'user_id'=>$request->userId,
            'text'=>$request->text
        ]);
        DB::commit();
        //////
        $model = new User();
        $text=$request->ip()."\t".$request->url()."\t".session()-
>get('user')->username."\t"."Commented"."\t".date("Y-m-d H:i:s");
        $model->log($text);
        //////
        return $comment;
    }

    public function edit($id){
        $this->data['comment'] = Comment::all()->find($id);
        return view('pages.main.editComment', $this->data);
    }

    public function update(CommentUpdateRequest $request, $id){

        DB::beginTransaction();
        try{
            $comment = Comment::all()->find($id);
            $comment->update([
                'text'=>$request->commentText
            ]);
            DB::commit();
            //////
            $model = new User();
            $text=$request->ip()."\t".$request->url()."\t".session()-
>get('user')->username."\t"."Edited comment"."\t".date("Y-m-d H:i:s");
            $model->log($text);
            //////
            return redirect()->route('table',$comment->table_id)-
>with('successMsg','Comment updated.');
        }
        catch (Exception $ex){
            DB::rollBack();
            Log::error($ex->getMessage());
            return redirect()->route('comment.edit',$id)-
>with('errorMsg','Something went wrong.');
        }

    }

    public function destroy($id){
        $comment = Comment::all()->find($id);
        $result = $comment->delete();
        //////
        $model = new User();
        $text=\request()->ip()."\t".\request()->url()."\t".session()-
>get('user')->username."\t"."Deleted comment"."\t".date("Y-m-d H:i:s");
        $model->log($text);
        //////
        return $result;
    }
}
```

4.1.1.6 *ContactController.php*

```
<?php

namespace App\Http\Controllers;

use App\Http\Requests\CommentStoreRequest;
use App\Http\Requests\ContactStoreRequest;
use App\Models\Message;
use Illuminate\Support\Facades\DB;
use Illuminate\Support\Facades\Log;


class ContactController extends DefaultController
{
    public function index(){
        return view('pages.main.contact',$this->data);
    }

    public function store(ContactStoreRequest $request){
        DB::beginTransaction();
        $message = Message::create($request->all());
        DB::commit();
        return $message;
    }
}
```

4.1.1.7 *DefaultController.php*

```
<?php

namespace App\Http\Controllers;

use App\Models\Menu;

class DefaultController extends Controller
{
    public $data=[];
```

```php
    public function __construct(){
        $this->data['menu'] = Menu::all();
    }
}
```

4.1.1.8 GameSystemController.php

```php
<?php

namespace App\Http\Controllers;

use App\Http\Requests\GameSystemStoreRequest;
use App\Http\Requests\GameSystemUpdateRequest;
use App\Models\GameSystems;
use App\Models\Table;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;

class GameSystemController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $systems = GameSystems::all();
        return $systems;
    }

    /**
     * Show the form for creating a new resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        //
    }

    /**
     * Store a newly created resource in storage.
     *
     * @param  \Illuminate\Http\Request  $request
     * @return \Illuminate\Http\Response
     */
    public function store(GameSystemStoreRequest $request)
    {
        DB::beginTransaction();
        $system = GameSystems::create($request->all());
        DB::commit();
        return response()->json(['feedback'=>'Added successfully.']);
    }

    /**
     * Display the specified resource.
     *
     * @param  int  $id
     * @return \Illuminate\Http\Response
     */
    public function show($id)
    {
        //
    }

    /**
     * Show the form for editing the specified resource.
     *
     * @param  int  $id
     * @return \Illuminate\Http\Response
     */
    public function edit($id)
    {
        $system = GameSystems::all()->find($id);
        return $system;
    }

    /**
     * Update the specified resource in storage.
     *
     * @param  \Illuminate\Http\Request  $request
     * @param  int  $id
     * @return \Illuminate\Http\Response
     */
    public function update(GameSystemUpdateRequest $request, $id)
    {
        $system = GameSystems::all()->find($id);
        DB::beginTransaction();
        $system->update($request->all());
        DB::commit();
        return response()->json(['feedback'=>'Updated successfully.']);
    }

    /**
     * Remove the specified resource from storage.
     *
     * @param  int  $id
     * @return \Illuminate\Http\Response
     */
    public function destroy($id)
    {

    }
}
```

4.1.1.9 HomeController.php

```php
<?php

namespace App\Http\Controllers;

use App\Models\Social;
use Illuminate\Http\Request;

class HomeController extends DefaultController
{

    public function index(){
        $this->data['socials'] = Social::all();
        return view('pages.main.home', $this->data);
    }
    public function error(){
        return view('pages.main.error', $this->data);
    }
}
```

4.1.1.10 LoginController.php

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\User;
use App\Models\Role;
use App\Http\Requests\LoginRequest;
use Illuminate\Support\Facades\Log;
use Exception;

class LoginController extends DefaultController
{
    public function index(){

            return view('pages.main.login', $this->data);
    }
    public function store(LoginRequest $request){
        try{
            $roleId = Role::all()->where('name','=','admin')->first()->id;
            $user = User::with('roles')->where('email','=',$request-
>email)->first();
            if(!$user->password==md5($request->password)){
                return redirect()->route('login')-
>with('errorMsg','Password incorrect.');
            }
            else{
                session()->put('user',$user);
                //////
                $model = new User();
                $text=$request->ip()."\t".$request->url()."\t".$user-
>username."\t"."Logged in"."\t".date("Y-m-d H:i:s");
                $model->log($text);
                //////
                if($user->role_id==$roleId){
                    return redirect()->route('admin');
                }
                return redirect()->route('account',$user->id);
            }
        }catch(Exception $ex){
            Log::error($ex->getMessage());
            return redirect()->route('login')->with('errorMsg','Something
went wrong, try again.');
        }
    }
    public function logout(){
        //////
        $model = new User();
        $text=\request()->ip()."\t".\request()->url()."\t".session()-
>get('user')->username."\t"."Logged out"."\t".date("Y-m-d H:i:s");
        $model->log($text);
        //////
        session()->remove('user');
        return redirect()->route('home');
    }
}
```

4.1.1.11 MessageController.php

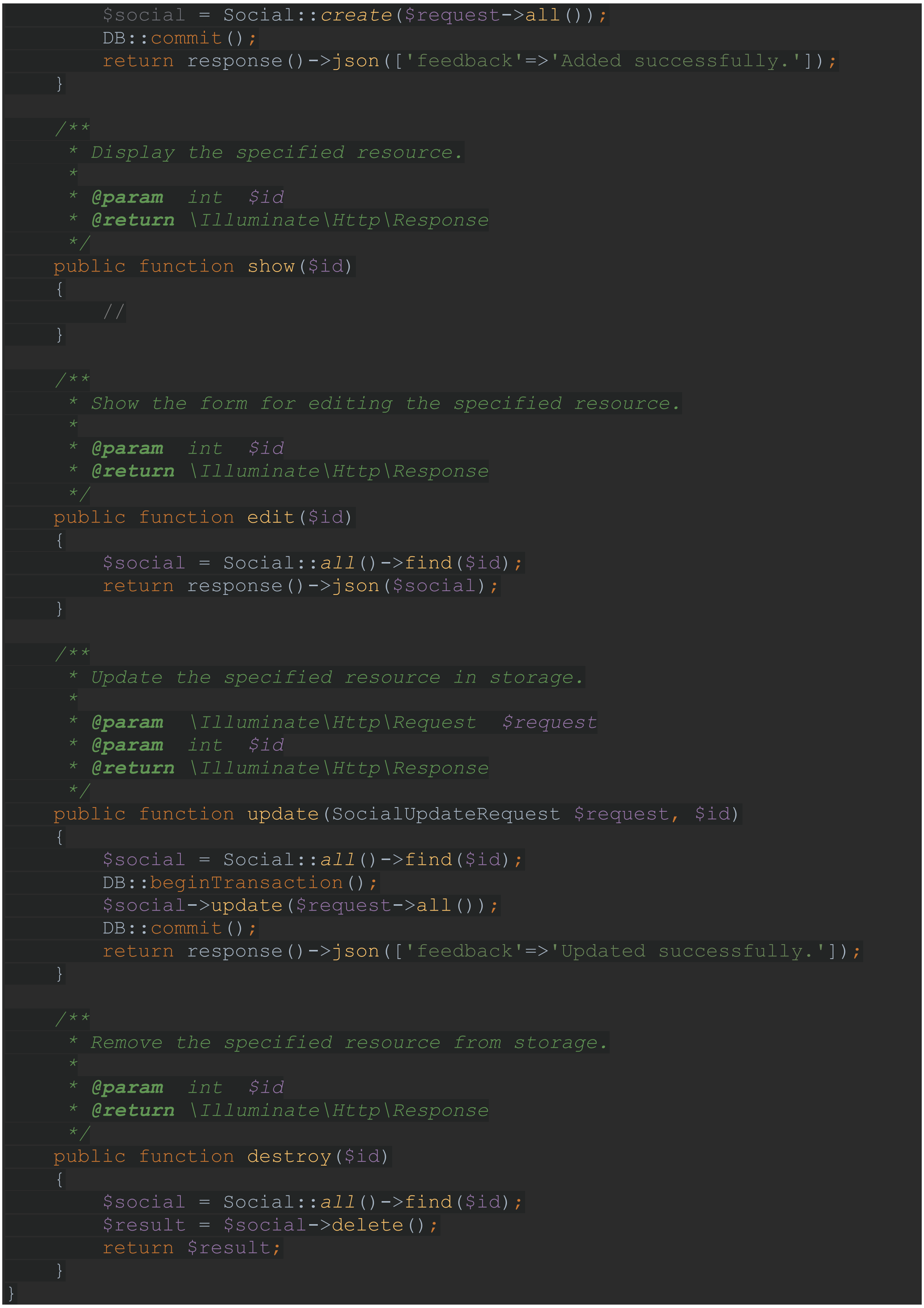
```
<?php

namespace App\Http\Controllers;

use App\Models\Message;
use Illuminate\Http\Request;

class MessageController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $data = Message::all();
        return $data;
    }


    public function destroy($id)
    {
        $message = Message::all()->find($id);
        $message->delete();
        return response()->json(['msg'=>'Deleted']);
    }
}
```

4.1.1.12 RegisterController.php

```
<?php

namespace App\Http\Controllers;
use App\Http\Requests\RegisterRequest;
use Illuminate\Support\Facades\DB;
use Exception;
use Illuminate\Support\Facades\Log;
use App\Models\User;
use App\Models\Role;

class RegisterController extends DefaultController
{
    public function index(){
        return view('pages.main.register',$this->data);
    }

    public function register(RegisterRequest $request){
        DB::beginTransaction();
        try{
            $role = Role::all()->where('name','=','user')->first()->id;
            $data = $request->all();
            $user = User::create([
                'username' => $data['username'],
                'email'=>$data['email'],
                'password'=>md5($data['password']),
                'image'=>'default.jpg',
                'alt'=> 'default image',
                'about'=>'',
                'role_id'=>$role
            ]);
            DB::commit();
            return redirect()->route('login');
        }catch(Exception $ex){
            DB::rollBack();
            Log::error($ex->getMessage());
            return redirect()->route('register')-
>with('errorMsg','Something went wrong, try again.');
        }
    }
}
```

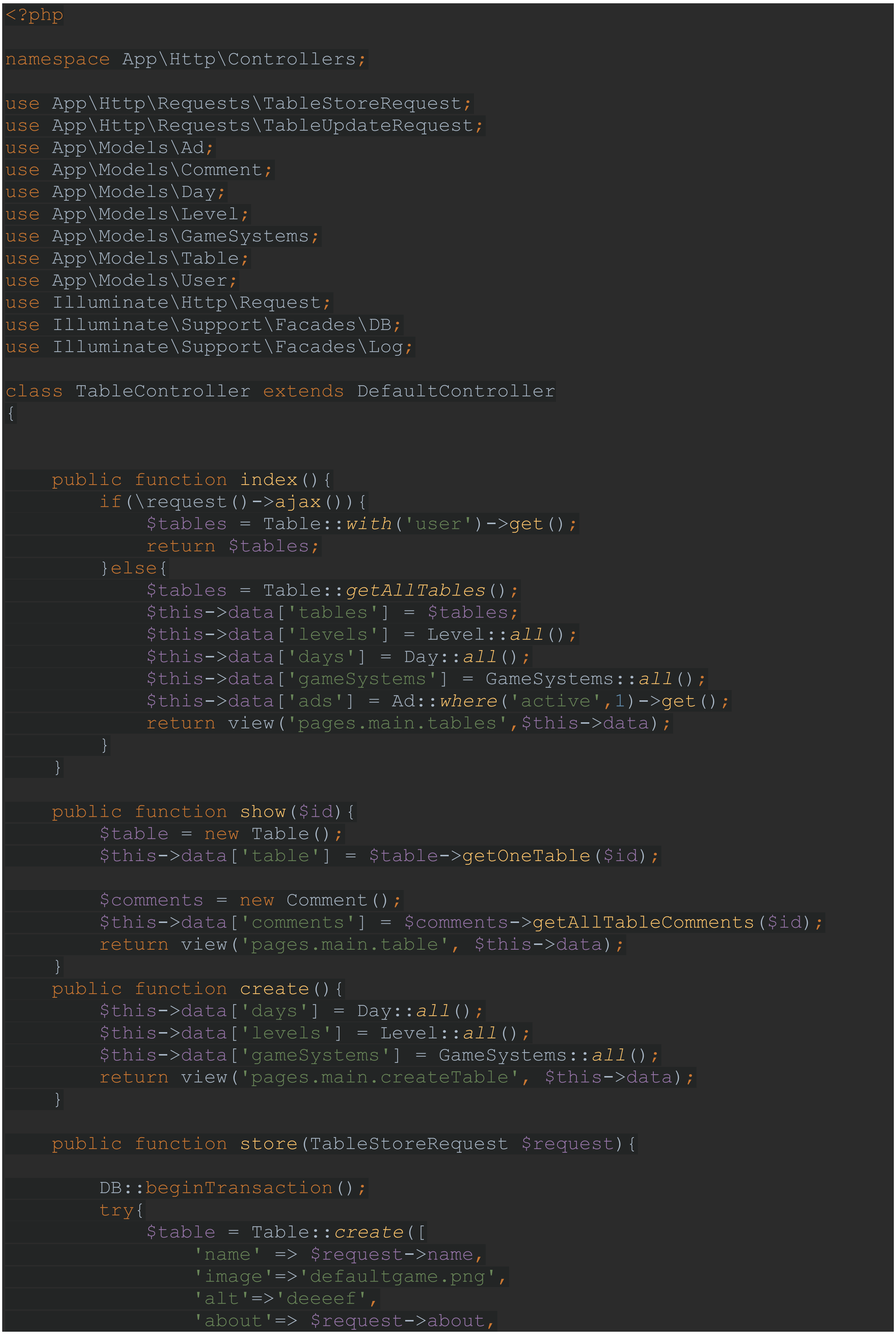
4.1.1.13 SocialController.php

```
<?php

namespace App\Http\Controllers;

use App\Http\Requests\SocialStoreRequest;
use App\Http\Requests\SocialUpdateRequest;
use App\Models\Social;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;

class SocialController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $socials = Social::all();
        return $socials;
    }

    /**
     * Show the form for creating a new resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        //
    }

    /**
     * Store a newly created resource in storage.
     *
     * @param  \Illuminate\Http\Request  $request
     * @return \Illuminate\Http\Response
     */
    public function store(SocialStoreRequest $request)
    {
        DB::beginTransaction();
```



```
        $social = Social::create($request->all());
        DB::commit();
        return response()->json(['feedback'=>'Added successfully.']);
    }

    /**
     * Display the specified resource.
     *
     * @param  int  $id
     * @return \Illuminate\Http\Response
     */
    public function show($id)
    {
        //
    }

    /**
     * Show the form for editing the specified resource.
     *
     * @param  int  $id
     * @return \Illuminate\Http\Response
     */
    public function edit($id)
    {
        $social = Social::all()->find($id);
        return response()->json($social);
    }

    /**
     * Update the specified resource in storage.
     *
     * @param  \Illuminate\Http\Request  $request
     * @param  int  $id
     * @return \Illuminate\Http\Response
     */
    public function update(SocialUpdateRequest $request, $id)
    {
        $social = Social::all()->find($id);
        DB::beginTransaction();
        $social->update($request->all());
        DB::commit();
        return response()->json(['feedback'=>'Updated successfully.']);
    }

    /**
     * Remove the specified resource from storage.
     *
     * @param  int  $id
     * @return \Illuminate\Http\Response
     */
    public function destroy($id)
    {
        $social = Social::all()->find($id);
        $result = $social->delete();
        return $result;
    }
}
```

4.1.1.14 TableController.php



```php
<?php

namespace App\Http\Controllers;

use App\Http\Requests\TableStoreRequest;
use App\Http\Requests\TableUpdateRequest;
use App\Models\Ad;
use App\Models\Comment;
use App\Models\Day;
use App\Models\Level;
use App\Models\GameSystems;
use App\Models\Table;
use App\Models\User;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;
use Illuminate\Support\Facades\Log;

class TableController extends DefaultController
{


    public function index(){
        if(\request()->ajax()){
            $tables = Table::with('user')->get();
            return $tables;
        }else{
            $tables = Table::getAllTables();
            $this->data['tables'] = $tables;
            $this->data['levels'] = Level::all();
            $this->data['days'] = Day::all();
            $this->data['gameSystems'] = GameSystems::all();
            $this->data['ads'] = Ad::where('active',1)->get();
            return view('pages.main.tables',$this->data);
        }
    }

    public function show($id){
        $table = new Table();
        $this->data['table'] = $table->getOneTable($id);

        $comments = new Comment();
        $this->data['comments'] = $comments->getAllTableComments($id);
        return view('pages.main.table', $this->data);
    }
    public function create(){
        $this->data['days'] = Day::all();
        $this->data['levels'] = Level::all();
        $this->data['gameSystems'] = GameSystems::all();
        return view('pages.main.createTable', $this->data);
    }

    public function store(TableStoreRequest $request){

        DB::beginTransaction();
        try{
            $table = Table::create([
                'name' => $request->name,
                'image'=>'defaultgame.png',
                'alt'=>'deeeef',
                'about'=> $request->about,
                'game_master'=> $request->gm,
                'date' => date('Y-m-d'),
                'user_id' => $request->user_id,
                'level_id' => $request->level,
                'game_system_id' => $request->gsys
            ]);
            $table->days()->attach($request->days);
            if($request->has('image')){
                $imgName = $request->file('image')-
>getClientOriginalName();
                $exploded = explode('.',$imgName);
                $newImgName = time().'-'.$exploded[0].'.'.$request-
>file('image')->guessExtension();
                $request->image-
>move(public_path('assets/img'),$newImgName);
                $table->image = $newImgName;
                $table->save();
            }
            ///////////
            $model = new User();
            $text=\request()->ip()."\t".\request()->url()."\t".session()-
>get('user')->username."\t"."Created table"."\t".date("Y-m-d H:i:s");
            $model->log($text);
            ///////////
            DB::commit();
            return redirect()->route('account',session()->get('user')->id)-
>with('successMsg', 'Table added.');
        }catch(\Exception $ex){
            DB::rollBack();
            Log::error($ex->getMessage());
            return redirect()->route('table.create')-
>with('errorMsg','Something went wrong.');
        }
    }

    public function edit($id){

        if(\request()->ajax()){
            $table = Table::all()->find($id);
            return $table;
        }else{
            $this->data['table'] = Table::with('days')->find($id);
            $this->data['days'] = Day::all();
            $this->data['levels'] = Level::all();
            $this->data['gameSystems'] = GameSystems::all();
            return view('pages.main.editTable', $this->data);
        }
    }

    public function update(Request $request, $id){
        if(\request()->ajax()){
            Table::validateAdminsUpdate($request);
            DB::beginTransaction();
            $table = Table::all()->find($id);
            $table->update($request->all());
            DB::commit();
            return response()->json(['feedback'=>'Updated successfully.']);
        }else{
            Table::validateUsersUpdate($request);
            DB::beginTransaction();
            try{
                $table = Table::all()->find($id);
                $table->update($request->except('image'));
                $table->days()->sync($request->days);
                if($request->has('image')){
                    $imgName = $request->file('image')-
>getClientOriginalName();
                    $exploded = explode('.',$imgName);
                    $newImgName = time().'-'.$exploded[0].'.'.$request-
>file('image')->guessExtension();
                    $request->image-
>move(public_path('assets/img'),$newImgName);
                    $table->image = $newImgName;
                    $table->save();
                }
                ///////////
                $model = new User();
                $text=\request()->ip()."\t".\request()-
>url()."\t".session()->get('user')->username."\t"."Updated
table"."\t".date("Y-m-d H:i:s");
                $model->log($text);
                ///////////
                DB::commit();
                return redirect()->route('table',$id)-
>with('successMsg','Table updated.');
            }catch(Exception $ex){
                DB::rollBack();
                Log::error($ex->getMessage());
                return redirect()->route('table.edit',$id)-
>with('errorMsg','Something went wrong.');
            }
        }
    }

    public function destroy($id){
        $table = Table::all()->find($id);
        $table->days()->detach();
        $table->delete();
        ///////////
        $model = new User();
        $text=\request()->ip()."\t".\request()->url()."\t".session()-
>get('user')->username."\t"."Deleted table"."\t".date("Y-m-d H:i:s");
        $model->log($text);
        ///////////
        if(\request()->ajax()){
            return response()->json(['feedback'=>'Deleted successfully.']);
        }else{
            return redirect()->route('account',session()->get('user')->id)-
>with('successMsg','Table deleted.');
        }
    }

//    public function getAllTables(){
//        return $this->data['tables'] = Table::getAllTables();
//    }


    public function filterTables(Request $request){
        $table = new Table();
        if($request->has('all')){
            $result = $table::getAllTables();
        }else{
            $result = $table->filterTables($request);
        }

        return response()->json($result);
    }
}
```

4.1.2 Middleware

4.1.2.1 CheckIsLogged.php

```
<?php

namespace App\Http\Middleware;

use Closure;
use Illuminate\Http\Request;

class CheckIsLogged
{
    /**
     * Handle an incoming request.
     *
     * @param  \Illuminate\Http\Request  $request
     * @param  \Closure(\Illuminate\Http\Request):
(\Illuminate\Http\Response|\Illuminate\Http\RedirectResponse)  $next
     * @return \Illuminate\Http\Response|\Illuminate\Http\RedirectResponse
     */
    public function handle(Request $request, Closure $next)
    {
        if($request->session()->has('user')){
            return $next($request);
        }

        return redirect()->route('error');
    }
}
```

4.1.2.2 CheckIsAdmin.php

```
<?php

namespace App\Http\Middleware;

use Closure;
use Illuminate\Http\Request;

class CheckIsAdmin
{
    /**
     * Handle an incoming request.
     *
     * @param  \Illuminate\Http\Request  $request
     * @param  \Closure(\Illuminate\Http\Request):
(\Illuminate\Http\Response|\Illuminate\Http\RedirectResponse)  $next
     * @return \Illuminate\Http\Response|\Illuminate\Http\RedirectResponse
     */
    public function handle(Request $request, Closure $next)
    {
        if($request->session()->has('user') && $request->session()-
>get('user')->role_id==1){
```

```
            return $next($request);
        }

        return redirect()->route('error');
    }
}
```

4.1.3 Requests

4.1.3.1 AdStoreRequest.php

```
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class AdStoreRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'text' => 'required',
            'link'=>'required',
            'image'=>'required|image|max:2000|mimes:jpg,png,jpeg'
        ];
    }
}
```

4.1.3.2 AdUpdateRequest.php

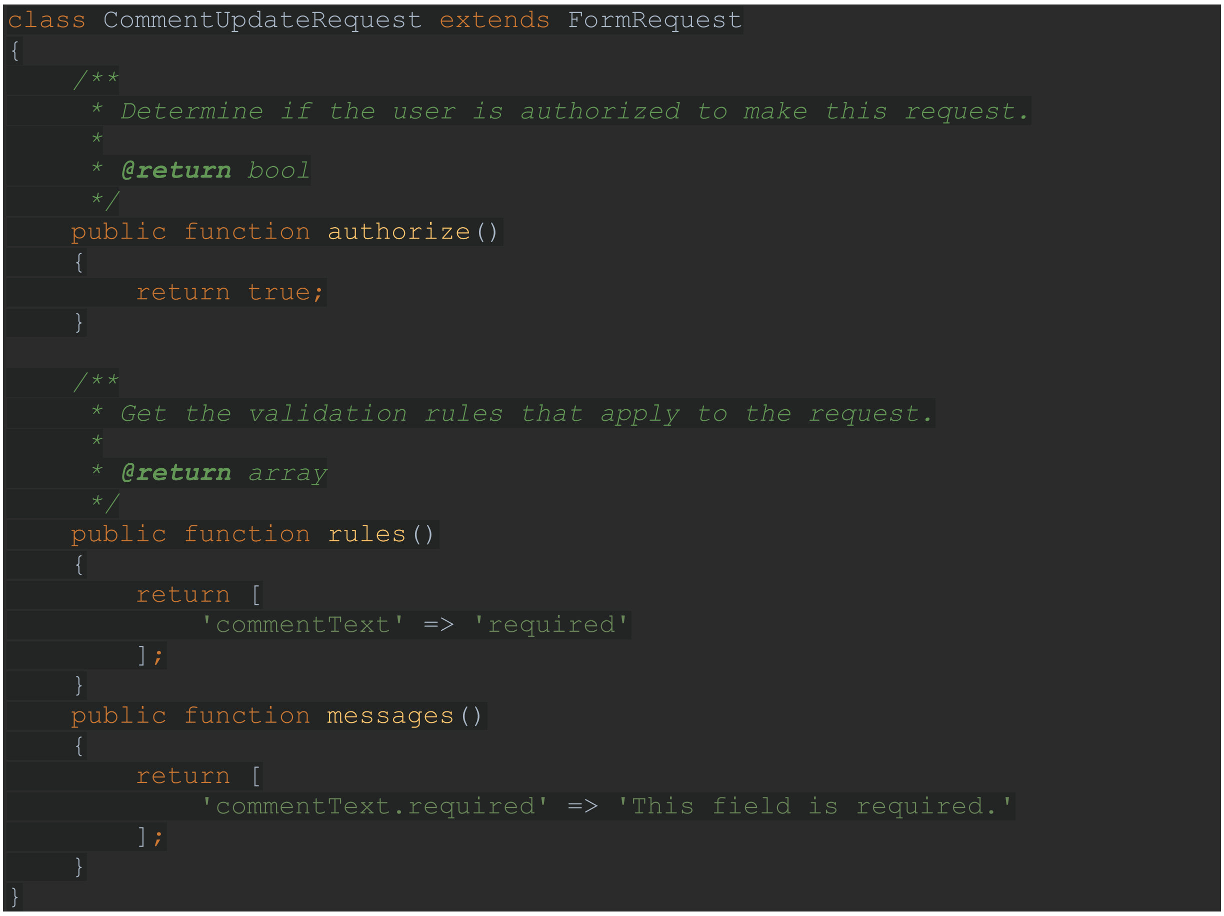
```
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class AdUpdateRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'text' => 'required',
            'link'=>'required',
            'image'=>'image|max:2000|mimes:jpg,png,jpeg'
        ];
    }
}
```

4.1.3.3 CommentStoreRequest.php

```
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class CommentStoreRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'text'=>'required',
            'userId'=>'required|exists:users,id',
            'tableId'=>'required|exists:tables,id'
        ];
    }
}
```

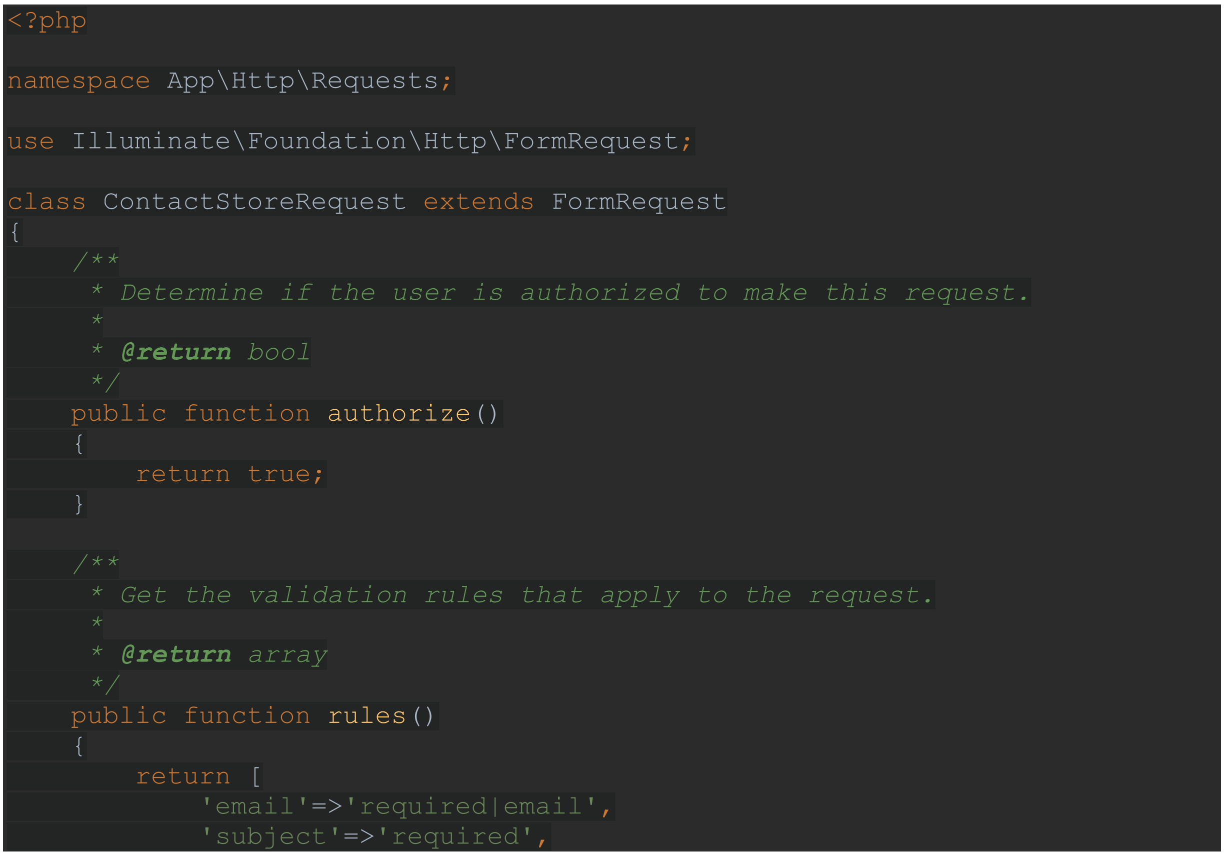
4.1.3.4 CommentUpdateRequest.php

```
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

```



```php
class CommentUpdateRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'commentText' => 'required'
        ];
    }
    public function messages()
    {
        return [
            'commentText.required' => 'This field is required.'
        ];
    }
}
```

4.1.3.5 ContactStoreRequest.php



```php
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class ContactStoreRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'email'=>'required|email',
            'subject'=>'required',
            'message'=>'required'
        ];
    }
}
```

4.1.3.6 GameSystemStoreRequest.php

```
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class GameSystemStoreRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'name'=> 'required|regex:/[A-Za-z\s\d\:?]{3,}+/'
        ];
    }
}
```

4.1.3.7 GameSystemUpdateRequest.php

```
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class GameSystemUpdateRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'name'=> 'required|regex:/[A-Za-z\s\d\:?]{3,}+/'
        ];
    }
}
```

4.1.3.8 LoginRequest.php

```
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class LoginRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'email' => 'required|email|max:255|exists:users,email',
            'password' => 'required|regex:/^(?=.*?[A-Z])(?=.*?[a-
z])(?=.*?[0-9])(?=.*?[#?!@$%^&*-]).{8,}$/'
        ];
    }

    public function messages()
    {
        return [
            'email.required' => 'Email is required.',
            'email.exists' => 'Email does not exist',
            'password.required' => 'Password is required.',
            'password.regex'=>'Password must be at least 8 characters,
contain at least 1 lowercase, at least 1 uppercase letter, special
character, and at least 1 digit.'
        ];
    }
}
```

4.1.3.9 RegisterRequest.php

```
<?php
```

```php
namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class RegisterRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'username' =>
'required|unique:users,username|alpha_num|min:4|max:20',
            'email'=>'required|email|max:255|unique:users,email',
            'password'=>'required|confirmed|regex:/^(?=.*?[A-Z])(?=.*?[a-
z])(?=.*?[0-9])(?=.*?[#?!@$%^&*-]).{8,}$/'
        ];
    }
    public function messages()
    {
        return [
            'username.required'=>'Username required.',
            'username.unique'=>'Username must be unique.',
            'username.aplhanumeric' => 'Username can contain only letters
and numbers.',
            'username.min'=>'Username cannot be shorter than 4
characters.',
            'username.max'=>'Username cannot be longer than 20
characters.',
            'email.required' => 'Email required.',
            'email.unique' => 'Email must be unique.',
            'password.required'=>'Password required.',
            'password.confirmed'=>'Passwords do not match.',
            'password.regex'=>'Password must be at least 8 characters,
contain at least 1 lowercase, at least 1 uppercase letter, special
character, and at least 1 digit.'
        ];
    }
}
```

4.1.3.10 SocialStoreRequest.php

```php
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;
```

```php
class SocialStoreRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'name' => 'required',
            'link' => 'required'
        ];
    }
}
```

4.1.3.11 SocialUpdateRequest.php

```php
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class SocialUpdateRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'name' => 'required',
            'link' => 'required'
        ];
    }
}
```

4.1.3.12 TableStoreRequest.php

```
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class TableStoreRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'name' => 'required',
            'image' => 'image|max:2000|mimes:jpg,png,jpeg',
            'days' => 'required|exists:days,id',
            'gm' => 'required',
            'gsys'=> 'required|exists:game_systems,id',
            'level' => 'required|exists:levels,id',
            'about'=>'required'
        ];
    }

    public function messages(){
        return [
            'name.required' => 'Name is required.',
            'image' => 'Uploaded file must be .jpg/.jpeg or .png',
            'image.image' => 'Uploaded file must be an image.',
            'image.max' => 'Uploaded file must not be larger than :max
kilobytes.',
            'days.exists' => 'The selected day does not exist.',
            'days.required' => 'Days required.',
            'gsys.required' => 'Game system is required.',
            'gsys.exists' => 'Game system is required.',
            'level.required' => 'Level is required.',
            'about.required' => 'Description is required.'
        ];
    }
}
```

4.1.3.13 TableUpdateRequest.php

```
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;
```

```
class TableUpdateRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'name' => 'required',
            'image' => 'image|max:2000|mimes:jpg,png,jpeg',
            'days' => 'required|exists:days,id',
            'gm' => 'required',
            'gsys'=> 'required|exists:game_systems,id',
            'level' => 'required|exists:levels,id'
        ];
    }

    public function messages(){
        return [
            'name.required' => 'Name is required.',
            'image' => 'Uploaded file must be .jpg/.jpeg or .png',
            'image.image' => 'Uploaded file must be an image.',
            'image.max' => 'Uploaded file must not be larger than :max
kilobytes.',
            'days.exists' => 'The selected day does not exist.',
            'days.required' => 'Days required.',
            'gsys.required' => 'Game system is required.',
            'gsys.exists' => 'Game system is required.',
            'level.required' => 'Level is required.'
        ];
    }
}
```

4.1.3.14 UserStoreRequest.php

```
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class UserStoreRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'username' =>
'required|unique:users,username|alpha_num|min:4|max:20',
            'email'=>'required|email|max:255|unique:users,email',
            'password'=>'required|regex:/^(?=.*?[A-Z])(?=.*?[a-z])(?=.*?[0-
9])(?=.*?[#?!@$%^&*-]).{8,}$/',
            'role'=>'required|exists:roles,id'
        ];
    }
    public function messages()
    {
        return [
            'username.required'=>'Username required.',
            'username.unique'=>'Username must be unique.',
            'username.aplhanumeric' => 'Username can contain only letters
and numbers.',
            'username.min'=>'Username cannot be shorter than 4
characters.',
            'username.max'=>'Username cannot be longer than 20
characters.',
            'email.required' => 'Email required.',
            'email.unique' => 'Email must be unique.',
            'password.required'=>'Password required.',
            'password.regex'=>'Password must be at least 8 characters,
contain at least 1 lowercase, at least 1 uppercase letter, special
character, and at least 1 digit.',
            'role.required' => 'Role required.',
            'role.exists' => 'Selected does not exist.',
        ];
    }
}
```

4.1.3.15 UserUpdateRequest.php

```
<?php

namespace App\Http\Requests;

use Illuminate\Foundation\Http\FormRequest;

class UserUpdateRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     *
     * @return bool
     */
    public function authorize()
    {
        return true;
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array
     */
    public function rules()
    {
        return [
            'image' => 'image|max:2000|mimes:jpg,png,jpeg',
            'gameroles' => 'required|exists:game_roles,id'
        ];
    }

    public function messages(){
        return [
            'required' => 'The :attribute field is required.',
            'image' => 'Uploaded file must be .jpg/.jpeg or .png',
            'image.image' => 'Uploaded file must be an image.',
            'image.max' => 'Uploaded file must not be larger than :max
kilobytes.',
            'gameroles.exists' => 'Selected role does not exist.'
        ];
    }
}
```

4.1.4 Mail

4.1.4.1 Mail.php

```
<?php

namespace App\Mail;

use Illuminate\Bus\Queueable;
use Illuminate\Contracts\Queue\ShouldQueue;
use Illuminate\Mail\Mailable;
use Illuminate\Queue\SerializesModels;

class Mail extends Mailable
{
    use Queueable, SerializesModels;

    /**
     * Create a new message instance.
     *
     * @return void
     */
    public function __construct($content)
    {
        $this->data = $content;
    }

    /**
     * Build the message.
     *
     * @return $this
     */
    public function build()
    {
        return $this->subject('Mail from Tablefinder.com')-
>view('pages.main.mail')->with('data',$this->data);
    }
}
```

4.1.5 Models

4.1.5.1 Ad.php

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Ad extends Model
{
    use HasFactory;

    protected $fillable=['text','link','image','active'];
}
```

4.1.5.2 Comment.php

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Support\Facades\DB;

class Comment extends Model
{
    use HasFactory;

    protected $fillable = ['table_id','user_id','text'];

    public function getAllTableComments($id){
        return DB::table('comments')
            ->join('users', 'comments.user_id','=','users.id')
            ->join('tables', 'comments.table_id','=','tables.id')
            ->where('comments.table_id','=',$id)
            ->select('comments.*','users.image as userimg','users.username
as username','tables.user_id as tabowner')
            ->get();
    }

    public function getUsersComments($id){
        return DB::table('comments')
            ->join('tables','comments.table_id','=','tables.id')
            ->where('comments.user_id','=',$id)
            ->select('comments.*','tables.name as tname')
            ->get();
    }

    public function getAllComments(){
        return DB::table('comments')
            ->join('users', 'comments.user_id','=','users.id')
            ->join('tables', 'comments.table_id','=','tables.id')
            ->select('comments.*','users.username as username','tables.name
as tablename')
            ->orderByDesc('comments.created_at')
            ->get();
    }

    public function tables(){
        return $this->belongsTo(Table::class);
    }
    public function users(){
        return $this->belongsTo(User::class);
    }
}
```

4.1.5.3 Day.php

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Day extends Model
{
    use HasFactory;

    public function tables(){
        return $this->belongsToMany(Table::class,'day_table');
    }
}
```

4.1.5.4 GameRoles.php

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class GameRoles extends Model
{
    use HasFactory;
    protected $table = 'game_roles';

    public function users(){
        return $this->belongsToMany(User::class,'users_game_roles');
    }
}
```

4.1.5.5 GameSystems.php

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
```

```php
class GameSystems extends Model
{
    use HasFactory;

    protected $fillable=['name'];
}
```

4.1.5.6 Level.php

```php
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Level extends Model
{
    use HasFactory;

    public function tables(){
        return $this->hasMany(Table::class);
    }
}
```

4.1.5.7 Menu.php

```php
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Menu extends Model
{
    use HasFactory;
}
```

4.1.5.8 Message.php

```php
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Message extends Model
{
    use HasFactory;

    protected $fillable = ['email','subject','message'];
}
```

4.1.5.9 Role.php

```php
<?php

namespace App\Models;
```

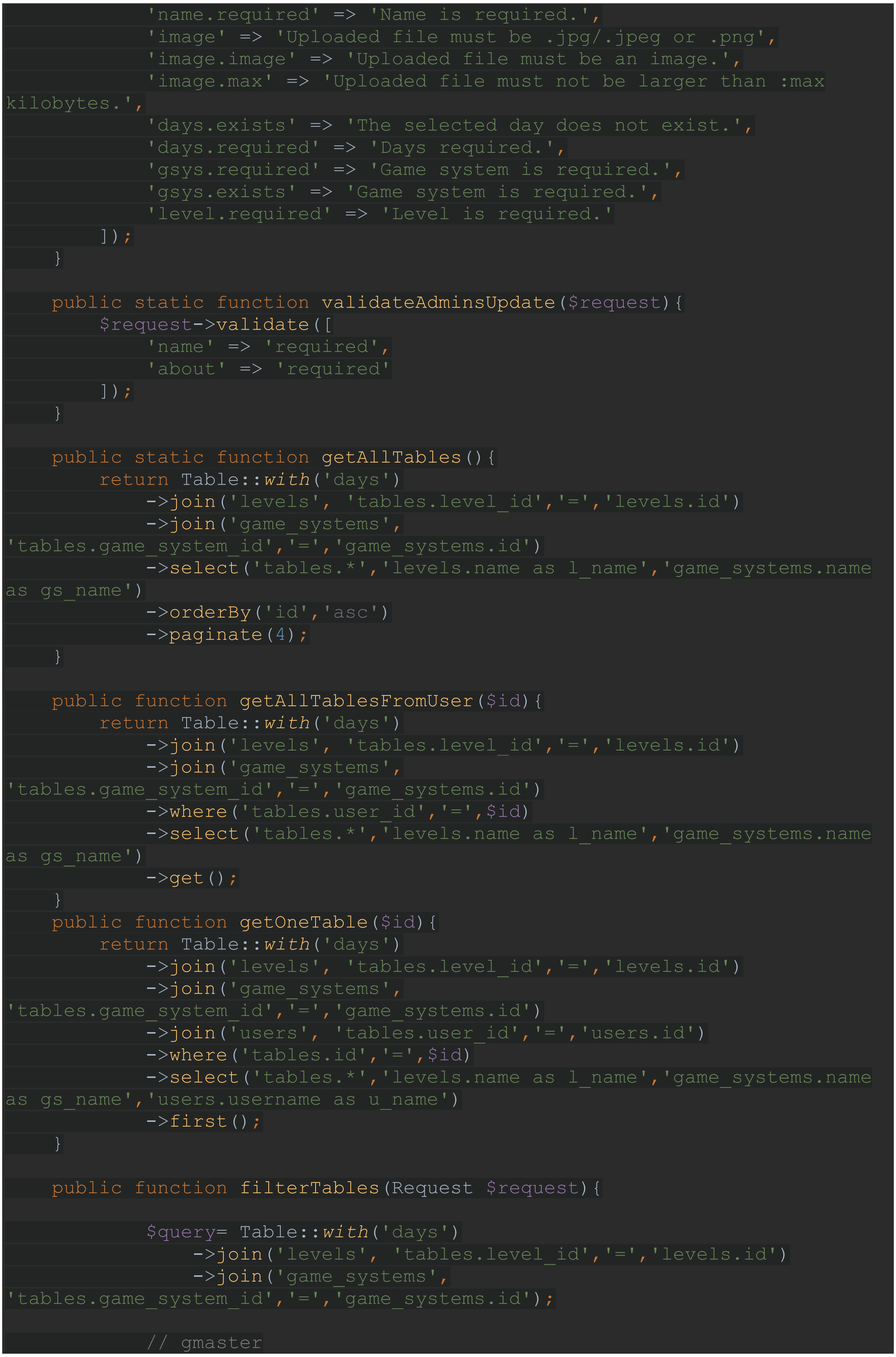
```php
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Role extends Model
{
    use HasFactory;
}
```

4.1.5.10 Social.php

```php
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Social extends Model
{
    use HasFactory;

    protected $fillable=['name','link'];
}
```

4.1.5.11 Table.php

```php
<?php

namespace App\Models;

use App\Http\Controllers\TableController;
use Illuminate\Http\Request;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Support\Facades\DB;

class Table extends Model
{
    use HasFactory;
    protected $fillable = [
        'name',
        'image',
        'alt',
        'about',
        'game_master',
        'level_id',
        'user_id',
        'game_system_id',
        'date'
    ];

    public static function validateUsersUpdate($request){
        $request->validate([
            'name' => 'required',
            'image' => 'image|max:2000|mimes:jpg,png,jpeg',
            'days' => 'required|exists:days,id',
            'gm' => 'required',
            'gsys'=> 'required|exists:game_systems,id',
            'level' => 'required|exists:levels,id'
        ],[
```



```
            'name.required' => 'Name is required.',
            'image' => 'Uploaded file must be .jpg/.jpeg or .png',
            'image.image' => 'Uploaded file must be an image.',
            'image.max' => 'Uploaded file must not be larger than :max
kilobytes.',
            'days.exists' => 'The selected day does not exist.',
            'days.required' => 'Days required.',
            'gsys.required' => 'Game system is required.',
            'gsys.exists' => 'Game system is required.',
            'level.required' => 'Level is required.'
        ]);
    }

    public static function validateAdminsUpdate($request){
        $request->validate([
            'name' => 'required',
            'about' => 'required'
        ]);
    }

    public static function getAllTables(){
        return Table::with('days')
            ->join('levels', 'tables.level_id','=','levels.id')
            ->join('game_systems',
'tables.game_system_id','=','game_systems.id')
            ->select('tables.*','levels.name as l_name','game_systems.name
as gs_name')
            ->orderBy('id','asc')
            ->paginate(4);
    }

    public function getAllTablesFromUser($id){
        return Table::with('days')
            ->join('levels', 'tables.level_id','=','levels.id')
            ->join('game_systems',
'tables.game_system_id','=','game_systems.id')
            ->where('tables.user_id','=',$id)
            ->select('tables.*','levels.name as l_name','game_systems.name
as gs_name')
            ->get();
    }
    public function getOneTable($id){
        return Table::with('days')
            ->join('levels', 'tables.level_id','=','levels.id')
            ->join('game_systems',
'tables.game_system_id','=','game_systems.id')
            ->join('users', 'tables.user_id','=','users.id')
            ->where('tables.id','=',$id)
            ->select('tables.*','levels.name as l_name','game_systems.name
as gs_name','users.username as u_name')
            ->first();
    }

    public function filterTables(Request $request){

            $query= Table::with('days')
                ->join('levels', 'tables.level_id','=','levels.id')
                ->join('game_systems',
'tables.game_system_id','=','game_systems.id');

            // gmaster
            if($request->has('gmaster') && $request->get('gmaster')!=''){
                $query = $query->where('tables.game_master', $request-
>gmaster);
            }
             // system
            if($request->has('gsys') && $request->get('gsys')!=0){
                $query = $query->where('tables.game_system_id',$request-
>gsys);
            }
            // level
            if($request->has('lvl')  && $request->get('lvl')!=0){
                $query = $query->where('tables.level_id',$request->lvl);
            }
            // name
            if($request->has('name')){
                $query = $query->where('tables.name', 'like', '%'.$request-
>name.'%');
            }
            // days
            if($request->has("days")){
                $query = $query->whereHas("days", function($d) use
($request){
                    $d->whereIn("days.id", $request->input("days"));
                 });
            }

            $query = $query->select('tables.*','levels.name as
l_name','game_systems.name as gs_name');

            if($request->has('sortCol') && $request->has('sortCol')){
                $query = $query->orderBy($request->sortCol,$request-
>sortVal);
            }
            else{
                $query = $query->orderBy('tables.id','asc');
            }


            $query = $query->paginate(4);


            return $query;
    }

    public function days(){
        return $this->belongsToMany(Day::class,'day_table');
    }

    public function user(){
        return $this->belongsTo(User::class);
    }

    public function level(){
        return $this->belongsTo(Level::class);
    }

    public function comments(){
        return $this->hasMany(Comment::class);
    }
}
```

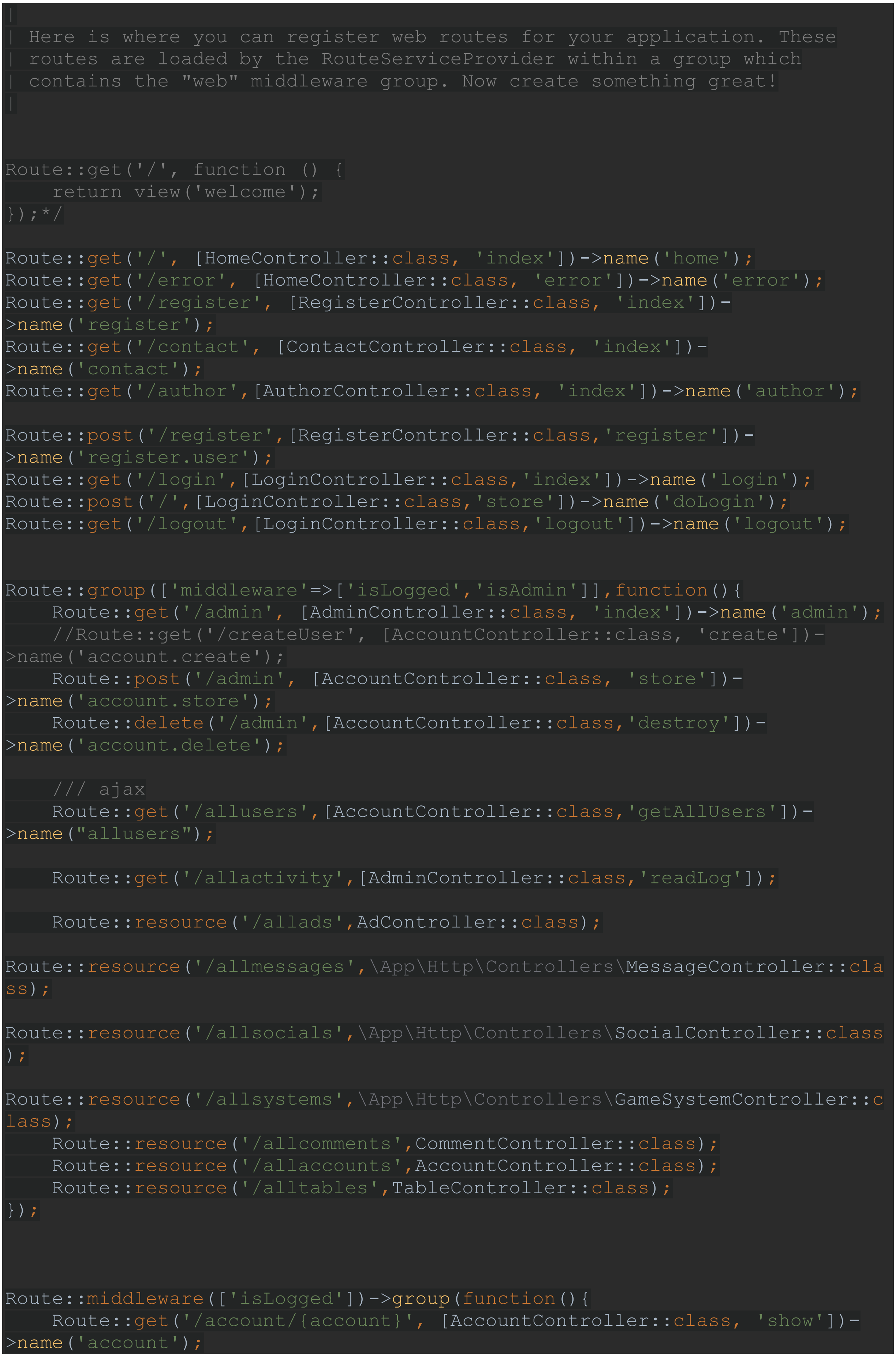
4.1.5.12 User.php

```php
<?php

namespace App\Models;

use Illuminate\Contracts\Auth\MustVerifyEmail;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Foundation\Auth\User as Authenticatable;
use Illuminate\Notifications\Notifiable;
use Illuminate\Support\Facades\Storage;
use Laravel\Sanctum\HasApiTokens;

class User extends Authenticatable
{
    use HasApiTokens, HasFactory, Notifiable;

    protected $connection = 'mysql';
    protected $table = 'users';

    /**
     * The attributes that are mass assignable.
     *
     * @var array<int, string>
     */

    protected $fillable = [
        'username',
        'email',
        'password',
        'image',
        'alt',
        'about',
        'role_id'
    ];

    /**
     * The attributes that should be hidden for serialization.
     *
     * @var array<int, string>
     */
    protected $hidden = [
        'password',
        'remember_token',
    ];

    /**
     * The attributes that should be cast.
     *
     * @var array<string, string>
     */

    protected $casts = [
        'email_verified_at' => 'datetime',
    ];

    public static function validateUsersUpdate($request){
        $request->validate([
            'image' => 'image|max:2000|mimes:jpg,png,jpeg',
            'gameroles' => 'required|exists:game_roles,id'
        ]);
    }
    public static function validateAdminsUpdate($request){
        $request->validate([
            'username' => 'required|alpha_num|min:4|max:20',
            'password'=>'regex:/^(?=.*?[A-Z])(?=.*?[a-z])(?=.*?[0-
9])(?=.*?[#?!@$%^&*-]).{8,}$/',
            'role_id'=>'required|exists:roles,id'
        ]);
    }

    public function log($data){
        if(Storage::disk("local")->exists("log.txt")){
            Storage::append("log.txt",$data);
        }
        else{
            Storage::disk("local")->put("log.txt",$data);
        }
    }

    public static function getAllUsers(){
        return User::with('roles')->get();
    }

    public function roles(){
        return $this->belongsTo(Role::class);
    }

    public function gameroles(){
        return $this->belongsToMany(GameRoles::class,'users_game_roles');
    }

    public function tables(){
        return $this->hasMany(Table::class);
    }

    public function comments(){
        return $this->hasMany(Comment::class);
    }
}
```

4.1.6 Routes

4.1.6.1 Web.php

```php
<?php

use Illuminate\Support\Facades\Route;
use App\Http\Controllers\HomeController;
use App\Http\Controllers\AuthorController;
use App\Http\Controllers\RegisterController;
use App\Http\Controllers\AccountController;
use App\Http\Controllers\AdminController;
use App\Http\Controllers\ContactController;
use App\Http\Controllers\TableController;
use App\Http\Controllers\LoginController;
use App\Http\Controllers\CommentController;
use App\Http\Controllers\AdController;
/*
|--------------------------------------------------------------------------
| Web Routes
|--------------------------------------------------------------------------
```



```
|
| Here is where you can register web routes for your application. These
| routes are loaded by the RouteServiceProvider within a group which
| contains the "web" middleware group. Now create something great!
|


Route::get('/', function () {
    return view('welcome');
});*/

Route::get('/', [HomeController::class, 'index'])->name('home');
Route::get('/error', [HomeController::class, 'error'])->name('error');
Route::get('/register', [RegisterController::class, 'index'])-
>name('register');
Route::get('/contact', [ContactController::class, 'index'])-
>name('contact');
Route::get('/author',[AuthorController::class, 'index'])->name('author');

Route::post('/register',[RegisterController::class,'register'])-
>name('register.user');
Route::get('/login',[LoginController::class,'index'])->name('login');
Route::post('/',[LoginController::class,'store'])->name('doLogin');
Route::get('/logout',[LoginController::class,'logout'])->name('logout');


Route::group(['middleware'=>['isLogged','isAdmin']],function(){
    Route::get('/admin', [AdminController::class, 'index'])->name('admin');
    //Route::get('/createUser', [AccountController::class, 'create'])-
>name('account.create');
    Route::post('/admin', [AccountController::class, 'store'])-
>name('account.store');
    Route::delete('/admin',[AccountController::class,'destroy'])-
>name('account.delete');

    /// ajax
    Route::get('/allusers',[AccountController::class,'getAllUsers'])-
>name("allusers");

    Route::get('/allactivity',[AdminController::class,'readLog']);

    Route::resource('/allads',AdController::class);

Route::resource('/allmessages',\App\Http\Controllers\MessageController::cla
ss);

Route::resource('/allsocials',\App\Http\Controllers\SocialController::class
);

Route::resource('/allsystems',\App\Http\Controllers\GameSystemController::c
lass);
    Route::resource('/allcomments',CommentController::class);
    Route::resource('/allaccounts',AccountController::class);
    Route::resource('/alltables',TableController::class);
});



Route::middleware(['isLogged'])->group(function(){
    Route::get('/account/{account}', [AccountController::class, 'show'])-
>name('account');
```

```
    Route::get('/account/{account}/edit', [AccountController::class,
'edit'])->name('account.edit');
    Route::put('/account/{account}', [AccountController::class, 'update'])-
>name('account.update');

    Route::get('/tables', [TableController::class, 'index'])-
>name('tables');
    Route::get('/tables/create', [TableController::class, 'create'])-
>name('table.create');
    Route::post('/tables', [TableController::class, 'store'])-
>name('table.store');

    Route::get('/table/{id}', [TableController::class, 'show'])-
>name('table');

    Route::get('/table/{id}/edit', [TableController::class, 'edit'])-
>name('table.edit');
    Route::put('/table/{id}', [TableController::class, 'update'])-
>name('table.update');
    Route::delete('/table/{id}', [TableController::class, 'destroy'])-
>name('table.destroy');



Route::get('/comments/{id}',[CommentController::class,'getCommentsForTable'
]);
    Route::post('/comments',[CommentController::class,'store'])-
>name('comment.store');
    Route::get('/comments/{id}/edit',[CommentController::class,'edit'])-
>name('comment.edit');
    Route::put('/comments/{id}',[CommentController::class,'update'])-
>name('comment.update');
    Route::delete('/comments/{id}',[CommentController::class,'destroy'])-
>name('comment.destroy');


////////// ajax
   // Route::get('/alltables',[TableController::class,'getAllTables'])-
>name("alltables");
    Route::get('/filtertables',[TableController::class,'filterTables'])-
>name("filtertables");
});


Route::post('/contact',[ContactController::class,'store'])-
>name('contact.store');

////////// mail send
Route::get('/contactMail',function (\App\Http\Requests\ContactStoreRequest
$request){
    $content=[
        'email' => $request->email,
        'subject' => $request->subject,
        'message'=>$request->message
    ];
    \Mail::to('katarina.kalanj.php2@gmail.com')->send(new
\App\Mail\Mail($content));

    return response(['feedback'=>'Message sent.']);
});
```

4.1.7 Views

4.1.7.1 Fixed.footer.blade.php

```
<div id="footer" class="bg-dark">
    <div class="d-flex flex-row justify-content-center">
        <div class="pt-3 text-white-50">
            <p>Table Finder 2022</p>
        </div>
    </div>
</div>
```

4.1.7.2 Fixed.head.blade.php

```
<head>
    <title>Table Finder @yield('title')</title>
    <meta charset="UTF-8">
    <link rel="icon" href="{{ asset('assets/img/tablefinder.ico') }}" />
    <meta name="csrf-token" content="{{ csrf_token() }}">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link href="{{ asset('assets/vendor/bootstrap/css/bootstrap.min.css')
}}" rel="stylesheet" />
    <link href="{{ asset('assets/css/style.css') }}" rel="stylesheet" />
    <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css" integrity="sha512-
9usAa10IRO0HhonpyAIVpjrylPvoDwiPUiKdWk5t3PyolY1cOd4DSE0Ga+ri4AuTroPR5aQvXU9
xC6qOPnzFeg==" crossorigin="anonymous" referrerpolicy="no-referrer" />
    <meta name="keywords" content="@yield('keywords')" />
    <meta name="description" content="@yield('description')" />
</head>
```

4.1.7.3 Fixed.navigation.blade.php

```
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
    <a class="navbar-brand" href="/"><i class="fa-solid fa-dice-d20"></i>
Table Finder</a>
    <button class="navbar-toggler" type="button" data-toggle="collapse"
data-target="#navbarSupportedContent" aria-
controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle
navigation">
        <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarSupportedContent">
        <ul class="navbar-nav ml-auto">
            <li class="nav-item active">
                <a class="nav-link" href="dokumentacija.pdf"
target="_blank">Dokumentacija <span class="sr-only">(current)</span></a>
            </li>
            @foreach($menu as $m)
                @if($m->visibility==0)
                    <li class="nav-item active">
                        <a class="nav-link" href="{{route($m-
>route)}}">{{$m->name}} <span class="sr-only"></span></a>
                    </li>
                @endif
                @if(!session()->has('user') && $m->visibility==1)
                        <li class="nav-item active">
                            <a class="nav-link" href="{{route($m-
>route)}}">{{$m->name}} <span class="sr-only"></span></a>
                        </li>
                    @endif
                @if(session()->has('user')  && $m->visibility==2)
                        @if($m->route=='account')
                            <li class="nav-item active">
                                <a class="nav-link" href="{{route($m-
>route,session()->get('user')->id)}}">{{$m->name}} <span class="sr-
only"></span></a>
                            </li>
                        @else
                            <li class="nav-item active">
                                <a class="nav-link" href="{{route($m-
>route)}}">{{$m->name}} <span class="sr-only"></span></a>
                            </li>
                        @endif
                    @endif
                @if(session()->has('user')  && (session()->get('user')-
>role_id==1)  && $m->visibility==3)
                        <li class="nav-item active">
                            <a class="nav-link" href="{{route($m-
>route)}}">{{$m->name}} <span class="sr-only"></span></a>
                        </li>
                    @endif
            @endforeach
            @if(session()->has('user'))
            <li class="nav-item active">
                <a class="nav-link" href="{{route('logout')}}">Logout <span
class="sr-only"></span></a>
            </li>
            @endif
        </ul>
    </div>
</nav>
```

4.1.7.4 Fixed.scripts.blade.php

```
<script src="{{ asset('assets/vendor/jquery/jquery.min.js') }}"></script>
<script src="{{ asset('assets/vendor/bootstrap/js/bootstrap.bundle.min.js')
}}"></script>
<script>
    const baseUrl ="{{url('/')}}"
    const page= "{{\Request::path()}}"
</script>
<script src="{{asset("assets/js/main.js")}}"
type="text/javascript"></script>
```

4.1.7.5 Layouts.layout.blade.php

```
<!DOCTYPE html>
<html lang="en">

@include('fixed.head')

<body>
@include('fixed.navigation')

@include('partials.success')
@include('partials.error')

@yield('content')

@include('fixed.footer')
```

```
@include('fixed.scripts')

</body>


</html>
```

4.1.7.6 Pages.main.account.blade.php

```
@extends('layouts.layout')

@section('title') - Account @endsection

@section('content')
    <div class="d-flex flex-column flex-md-row align-items-center align-
items-md-start justify-content-center">
        <div class="bg-dark card col-6 col-md-3 m-5 p-3 text-center d-flex
flex-column align-items-center justify-content-center profilePersonal
profilLeft">
            <div class="w-75">
                <img src="{{asset('assets/img/'.$user->image)}}"
class="img-fluid mb-2 profilSlika" alt="{{$user->alt}}">
                <p class="pb-2 infoBorder">
                    <span class="infoAcc text-uppercase">{{$user-
>username}}</span>
                </p>

                <p class="pb-3 infoBorder">
                    Date of Registration :<br> {{$user->created_at}}<br>
                </p>
                <p class="pb-2 infoBorder">
                    <span class="infoAcc text-uppercase">Roles :</span><br>
                    @foreach($user->gameroles as $role)
                        {{$role->name}} <br>
                    @endforeach
                </p>
                <p class="mb-0 ">
                    <span class="infoAcc text-uppercase">About :</span><br>
                    <div class="accAbout overflow-auto">
                    <p>{{$user->about}}</p>
                </div>
                </p>
            </div>
        </div>



        <div class="bg-dark card col-11 col-md-7 m-5 p-4 profilePersonal d-
flex flex-column align-items-center justify-content-center">
            <div class="w-75 d-flex align-items-between justify-content-
between mb-4">

                @if(session()->get('user')->id == $user->id)
                <div>
                    <a href="{{route('table.create')}}"><button
type="button" class="btn btn-outline-light btn-lg px-4" id="register"
name="register">New Table</button></a>
                </div>
                <div >
                    <a href="{{route('account.edit', $user->id)}}"><button
type="button" class="btn btn-outline-light btn-lg px-4" id="register"
name="register">Edit Account</button></a>
```

```
                </div>
                @endif

            </div>



            <div class="w-100  bg-dark card d-flex flex-column align-items-
center justify-content-center profilePersonal postBorder pb-4">
                <span class="text-left w-100 p-3 infoBorder"><h2>{{$user-
>username}}'s Tables</h2></span>

                <div class="col-12 flex-xl-row flex-column d-flex align-
items-center justify-content-around flex-wrap">
                    @foreach($tables as $table)
                        @include('partials.table',$table)
                    @endforeach
                </div>

            </div>

            <div class="w-100 mt-4 bg-dark card d-flex flex-column align-
items-center justify-content-center profilePersonal postBorder">
                <span class="text-left w-100 p-3 infoBorder"><h2>{{$user-
>username}}'s Comments</h2></span>
                <div class="w-100 d-flex flex-column align-items-center
justify-content-center p-3 mt-2 col-10">
                    <table class="table table-dark">
                        <thead>
                        <tr>
                            <th scope="col">Table</th>
                            <th scope="col">Comment</th>
                            <th scope="col">Date & Time</th>
                        </tr>
                        </thead>
                        <tbody>
                        @foreach($comments as $com)
                            <tr>
                                <th scope="row"><a
href="{{route('table',$com->table_id)}}">{{$com->tname}}</a></th>
                                <td>{{$com->text}}</td>
                                <td>{{$com->created_at}}</td>
                            </tr>
                        @endforeach

                        </tbody>
                    </table>
                </div>
            </div>
        </div>
    </div>
@endsection
```

4.1.7.7 Pages.main.admin.blade.php

```
@extends('layouts.layout')

@section('title') - Admin @endsection

@section('content')
    <div class="d-flex flex-column  align-items-center justify-content-
```

```
start min-vh-100" id="adminbckg">

        <div class="w-100 d-flex flex-column flex-lg-row justify-content-around align-items-center align-items-lg-start flex-wrap flex-shrink-0 p-3 text-white bg-dark" id="adminLeft">
            <div class=" text-center mb-2">
                    <h5 class="mb-2">Activity</h5>
                <a href="#" id="showAllActivity">Show All Activity</a>
            </div>
            <div class=" text-center mb-2">
                    <h5 class="mb-2">Users</h5>
                <a href="#" id="showAllUsers">Show All Users</a>  <br>
                <a href="#" id="createNewUser">Create New User</a>
            </div>
            <div class=" text-center mb-2">
                <h5 class="mb-2">Tables</h5>
                <a href="#" id="showAllTables">Show All Tables</a>
            </div>
            <div class=" text-center mb-2">
                <h5 class="mb-2">Comments</h5>
                <a href="#" id="showAllComments"> Show All Comments </a> <br>
            </div>
            <div class=" text-center mb-2">
                    <h5 class="mb-2">Game Systems</h5>
                <a href="#" id="showAllSystems"> Show All Systems </a><br>
                <a href="#" id="createNewSystem"> Create New System </a>
            </div>
            <div class=" text-center mb-2">
                    <h5 class="mb-2">Messages</h5>
                <a href="#" id="showAllMessages">Show All Messages</a> <br>
            </div>
            <div class=" text-center mb-2">
                <h5 class="mb-2">Ads</h5>
                <a href="#" id="showAllAds"> Show All Ads </a><br>
                <a href="#" id="createNewAdd"> Create New Ad </a>
            </div>
            <div class=" text-center mb-2">
                    <h5 class="mb-2">Socials</h5>
                <a href="#" id="showAllSocials"> Show All Socials </a><br>
                <a href="#" id="createNewSocial"> Add New Social Network </a><br>
            </div>
        </div>
        <div class="col-12 d-flex flex-column  justify-content-start align-items-center">

            <div class="col-10 text-center d-flex flex-column  justify-content-start align-items-center" id="adminContent">
                <div class="mt-4 col-3">
                    <img src="{{asset('assets/img/'.session()->get('user')->image)}}" class="img-fluid profilSlika"/>
                </div>
                <div class="m-4">
                    <h2>Welcome, {{session()->get('user')->username}}.</h2>

                </div>

            </div>

        </div>
```

```
    </div>
@endsection
```

4.1.7.8 Pages.main.author.blade.php

```
@extends('layouts.layout')

@section('title') - Author @endsection

@section('content')
    <section class="vh-100 login-bg">
        <div class="container  h-100">
            <div class="row d-flex justify-content-center align-items-
center h-100">
                <div class="col-12 col-md-8 col-lg-6 col-xl-5">
                    <div class="card bg-dark text-white" style="border-
radius: 1rem;">
                        <div class="card-body p-5 text-center">

                            <div class="mb-md-3 mt-md-4 ">
                                <div class="">
                                    <img src="assets/img/disc.jpg"
alt="slika123" class="img-fluid mb-2 slikaAuthor">
                                </div>
                                <div class="p-3 text-justify">
                                    <p>

                                    My name is Katarina Kalanj.<br>
                                    I'm currently a student, and this is
one of my projects.
                                    I've always loved being creative and
challenging myself.
                                    I'm passionate about making websites; a
team player who knows how to make your ideas a reality.<br>
                                    Click <span class="authUnder"><a
href="https://katarinakalanj.netlify.com/" target="_blank">here</a></span>
to see my portfolio.<br>
                                    Click <span class="authUnder"><a
href="models/wordDoc.php">here</a></span> to get a Word file with my info.
                                    </p>
                                </div>
                            </div>

                        </div>
                    </div>
                </div>
            </div>
        </div>
    </section>




    <div class="d-flex flex-column flex-md-row align-items-center align-
items-md-start justify-content-center">


    </div>
@endsection
```

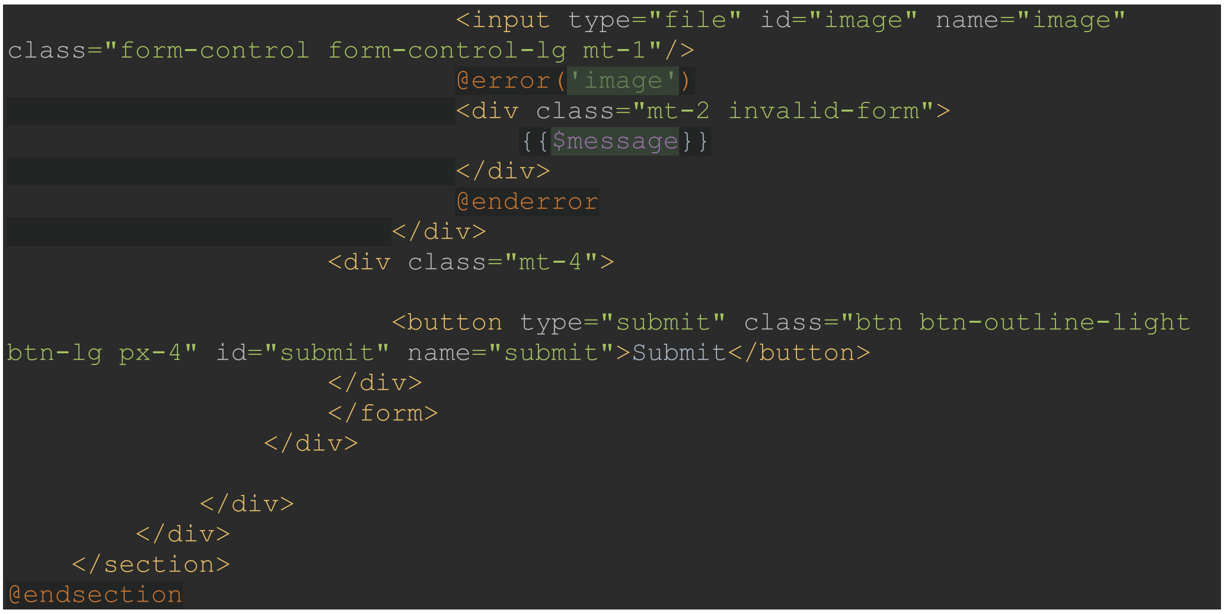
4.1.7.9 *Pages.main.contact.blade.php*

```
@extends('layouts.layout')
@section('title') - Contact Us @endsection

@section('content')
    <section class="vh-100 login-bg">
        <div class="container  h-100">
            <div class="row d-flex justify-content-center align-items-
center h-100">
                <div class="col-12 col-md-8 col-lg-6 col-xl-5">
                    <div class="card bg-dark text-white" style="border-
radius: 1rem;">
                        <div class="card-body p-5 text-center">

                            <div class="mb-md-3 mt-md-4 pb-3">

                                <div class="alert" role="alert"
id="contactFeedback">

                                </div>
                                <h2 class="fw-bold mb-2 text-
uppercase">Contact Us</h2>
                                <form>
                                <div class="form-outline form-white">
                                    <input type="email" id="email"
class="form-control form-control-lg mt-3"
placeholder="examplemail@mail.com"/>
                                    <label class="form-label mt-2"
for="email">Email</label>
                                    <div class="text-center mb-2 invalid-
form" id="emailError">
                                        Field filled incorrectly.
                                    </div>
                                </div>
                                <div class="form-outline form-white">
                                    <input type="text" id="subject"
class="form-control form-control-lg" />
                                    <label class="form-label mt-2"
for="text">Subject</label>
                                    <div class="text-center mb-2 invalid-
form" id="subjectError">
                                        Field cannot be empty.
                                    </div>
                                </div>
                                <div class="form-group">
                                    <textarea class="form-control contxt"
id="message" rows="3"></textarea>
                                    <label class="form-label mt-2"
for="message">Your Message</label>
                                    <div class="text-center mb-2 invalid-
form" id="messageError">
                                        Field cannot be empty.
                                    </div>
                                </div>
                                <button class="btn btn-outline-light btn-lg
px-5" id="submit" type="submit">Submit</button>
                                </form>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </section>
@endsection
```

4.1.7.10 Pages.main.createTable.blade.php

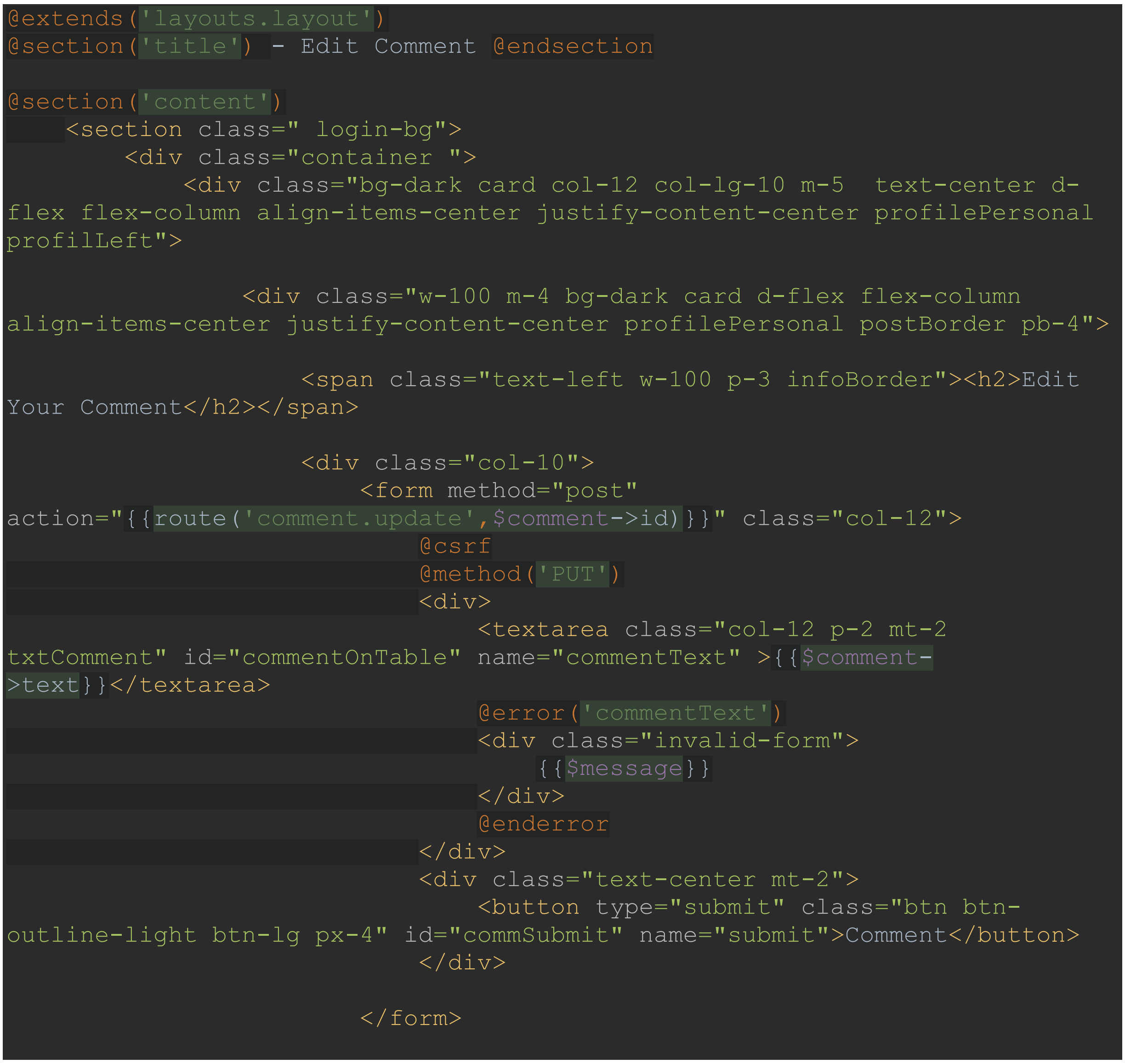
```
@extends('layouts.layout')

@section('title') - Create a New Table @endsection

@section('content')
    <section class=" login-bg">
        <div class="container ">
            <div class="bg-dark card col-12 col-lg-10 m-5  text-center d-flex flex-column align-items-center justify-content-center profilePersonal profilLeft">

                <div class="w-100 m-4 bg-dark card d-flex flex-column align-items-center justify-content-center profilePersonal postBorder pb-4">

                    <span class="text-left w-100 p-3 infoBorder"><h2>Create a New Table</h2></span>
                    <form class="col-10" method="post" action="{{route('table.store')}}" enctype="multipart/form-data">
                        @csrf
                        <input type="hidden" name="user_id" value="{{session()->get('user')->id}}">
                    <div class="form-outline form-white text-center mt-2">
                        <span class="infoAcc text-uppercase">Table Name :</span><br>
                        <input type="text" id="name" name="name" class="form-control form-control-lg mt-1" value="{{old('name')}}"/>
                        @error('name')
                        <div class="mt-2 invalid-form">
                            {{$message}}
                        </div>
                        @enderror
                    </div>
                    <span class="infoAcc text-uppercase mt-2">Availability :</span>
                    <p class="small mb-1 pb-lg-2 text-white-50">This is to help others know when you would like to have sessions.<br></p>
                    <div class=" form-outline form-white pl-5 mb-2 text-left">
                        @foreach($days as $d)
                            <input class="form-check-input" type="checkbox" name="days[]" value="{{$d->id}}" id="day{{$d->id}}">
                            <label class="form-check-label" for="day{{$d->id}}">
                                {{$d->name}}
                            </label><br>
                        @endforeach
                            @error('days')
                            <div class="mt-2 invalid-form">
                                {{$message}}
                            </div>
                            @enderror
                    </div>
                    <div class="form-outline  mt-2 form-white">
                        <span class="infoAcc text-uppercase ">Game Master
```

```
Required :</span><br>
                        <input type="radio" id="gm" name="gm" class="mt-1"
value="1" checked/> Yes
                        <input type="radio" id="gm" name="gm" class="ml-5
mt-1" value="0"/> No
                        @error('gm')
                        <div class="mt-2 invalid-form">
                            {{$message}}
                        </div>
                        @enderror
                    </div>
                    <span class="infoAcc text-uppercase mt-2">Game System
:</span>
                    <div class="form-outline form-white mb-2">
                        <select class="custom-select form-control mt-2"
id="gsys" name="gsys">
                            <option value="ASC-id" selected>Game
System...</option>
                            @foreach($gameSystems as $gs)
                                <option value="{{$gs->id}}">{{$gs-
>name}}</option>
                            @endforeach
                        </select>
                        @error('gsys')
                        <div class="mt-2 invalid-form">
                            {{$message}}
                        </div>
                        @enderror
                    </div>
                    <span class="infoAcc text-uppercase mt-2">Gaming
Expertise Level :</span>
                    <div class="form-outline  form-white ">
                        <select class="custom-select form-control mt-2 mb-
2" id="level" name="level">
                            <option value="ASC-id"
selected>Level...</option>
                            @foreach($levels as $lev)
                                <option value="{{$lev->id}}">{{$lev-
>name}}</option>
                            @endforeach
                        </select>
                        @error('level')
                        <div class="mt-2 invalid-form">
                            {{$message}}
                        </div>
                        @enderror
                    </div>
                    <span class="infoAcc text-uppercase">Table Description
:</span>
                    <div class="mb-2 form-group">
                        <textarea class="form-control mt-1 contxt"
name="about" id="about" rows="3">{{old('about')}}</textarea>
                        @error('about')
                        <div class="mt-2 invalid-form">
                            {{$message}}
                        </div>
                        @enderror
                    </div>
                        <div class="form-outline form-white text-center">
                            <span class="infoAcc text-uppercase">Image
:</span><br>
```



```
                            <input type="file" id="image" name="image"
class="form-control form-control-lg mt-1"/>
                            @error('image')
                            <div class="mt-2 invalid-form">
                                {{$message}}
                            </div>
                            @enderror
                        </div>
                    <div class="mt-4">

                        <button type="submit" class="btn btn-outline-light
btn-lg px-4" id="submit" name="submit">Submit</button>
                    </div>
                    </form>
                </div>

            </div>
        </div>
    </section>
@endsection
```

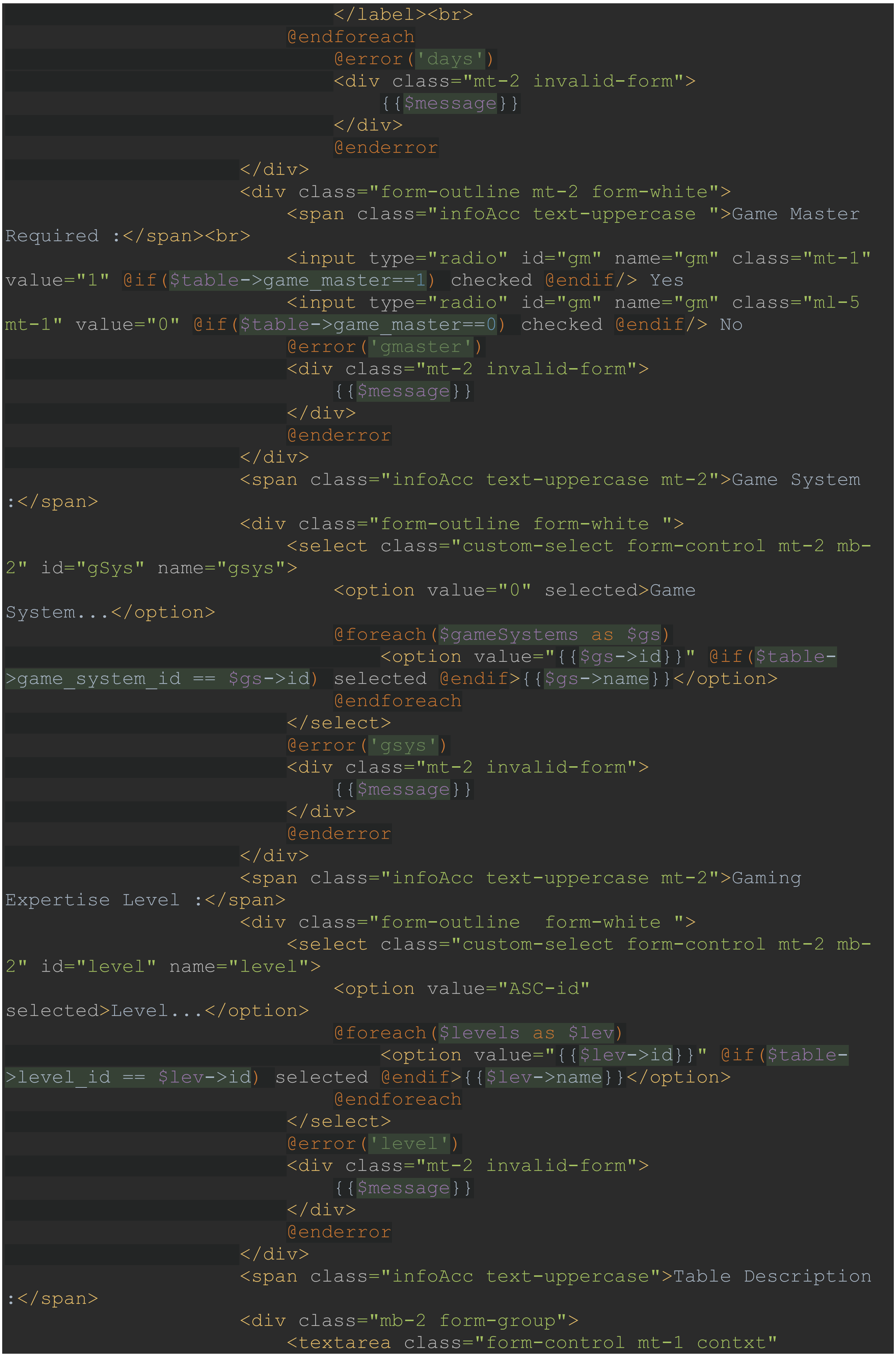
4.1.7.11 Pages.main.editComment.blade.php



```
@extends('layouts.layout')
@section('title') - Edit Comment @endsection

@section('content')
    <section class=" login-bg">
        <div class="container ">
            <div class="bg-dark card col-12 col-lg-10 m-5  text-center d-
flex flex-column align-items-center justify-content-center profilePersonal
profilLeft">

                <div class="w-100 m-4 bg-dark card d-flex flex-column
align-items-center justify-content-center profilePersonal postBorder pb-4">

                    <span class="text-left w-100 p-3 infoBorder"><h2>Edit
Your Comment</h2></span>

                    <div class="col-10">
                        <form method="post"
action="{{route('comment.update',$comment->id)}}" class="col-12">
                            @csrf
                            @method('PUT')
                            <div>
                                <textarea class="col-12 p-2 mt-2
txtComment" id="commentOnTable" name="commentText" >{{$comment-
>text}}</textarea>
                                @error('commentText')
                                <div class="invalid-form">
                                    {{$message}}
                                </div>
                                @enderror
                            </div>
                            <div class="text-center mt-2">
                                <button type="submit" class="btn btn-
outline-light btn-lg px-4" id="commSubmit" name="submit">Comment</button>
                            </div>

                        </form>
                    </div>


                </div>

            </div>
        </div>
    </section>
@endsection
```

4.1.7.12 Pages.main.editTable.blade.php

```
@extends('layouts.layout')

@section('title') - Update Your Table @endsection

@section('content')
    <section class=" login-bg">
        <div class="container ">
            <div class="bg-dark card col-12 col-lg-10 m-5  text-center d-flex flex-column align-items-center justify-content-center profilePersonal profilLeft">

                <div class="w-100 m-4 bg-dark card d-flex flex-column align-items-center justify-content-center profilePersonal postBorder pb-4">

                    <span class="text-left w-100 p-3 infoBorder"><h2>Edit Your Table</h2></span>
                    <form class="col-8" method="post" action="{{route('table.update',$table->id)}}" enctype="multipart/form-data">
                        @csrf
                        @method('PUT')
                    <div class="form-outline form-white text-center mt-2 mb-2">
                        <span class="infoAcc text-uppercase">Table Name :</span><br>
                        <input type="text" id="name" class="form-control form-control-lg mt-1" name="name" value="{{$table->name}}"/>
                        @error('name')
                        <div class="mt-2 invalid-form">
                            {{$message}}
                        </div>
                        @enderror
                    </div>
                    <span class="infoAcc text-uppercase mt-2">Availability :</span>
                    <p class="small mb-1 pb-lg-2 text-white-50">This is to help others know when you would like to have sessions.<br></p>
                    <div class=" form-outline form-white pl-5 mb-2 text-left">
                        @foreach($days as $d)
                            <input class="form-check-input" type="checkbox" name="days[]" value="{{$d->id}}" id="day{{$d->id}}"
                                       @if(isset($table) && in_array($d->id, $table->days()->pluck('days.id')->toArray()))
                            checked
                                @endif>
                            <label class="form-check-label" for="day{{$d->id}}">
                                {{$d->name}}
```

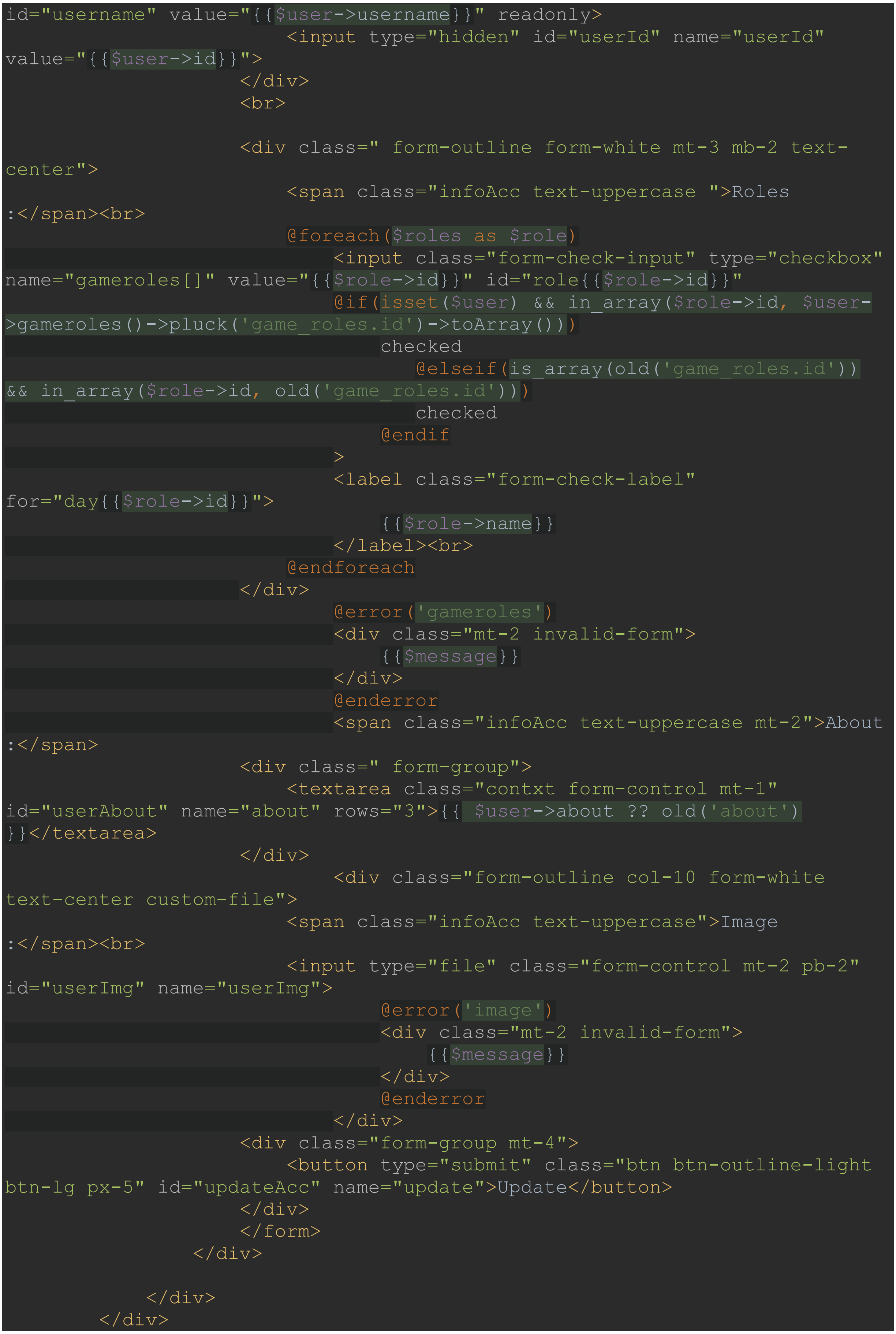


```
                            </label><br>
                        @endforeach
                            @error('days')
                            <div class="mt-2 invalid-form">
                                {{$message}}
                            </div>
                            @enderror
                    </div>
                    <div class="form-outline mt-2 form-white">
                        <span class="infoAcc text-uppercase ">Game Master
Required :</span><br>
                        <input type="radio" id="gm" name="gm" class="mt-1"
value="1" @if($table->game_master==1) checked @endif/> Yes
                        <input type="radio" id="gm" name="gm" class="ml-5
mt-1" value="0" @if($table->game_master==0) checked @endif/> No
                        @error('gmaster')
                        <div class="mt-2 invalid-form">
                            {{$message}}
                        </div>
                        @enderror
                    </div>
                    <span class="infoAcc text-uppercase mt-2">Game System
:</span>
                    <div class="form-outline form-white ">
                        <select class="custom-select form-control mt-2 mb-
2" id="gSys" name="gsys">
                            <option value="0" selected>Game
System...</option>
                            @foreach($gameSystems as $gs)
                                <option value="{{$gs->id}}" @if($table-
>game_system_id == $gs->id) selected @endif>{{$gs->name}}</option>
                            @endforeach
                        </select>
                        @error('gsys')
                        <div class="mt-2 invalid-form">
                            {{$message}}
                        </div>
                        @enderror
                    </div>
                    <span class="infoAcc text-uppercase mt-2">Gaming
Expertise Level :</span>
                    <div class="form-outline  form-white ">
                        <select class="custom-select form-control mt-2 mb-
2" id="level" name="level">
                            <option value="ASC-id"
selected>Level...</option>
                            @foreach($levels as $lev)
                                <option value="{{$lev->id}}" @if($table-
>level_id == $lev->id) selected @endif>{{$lev->name}}</option>
                            @endforeach
                        </select>
                        @error('level')
                        <div class="mt-2 invalid-form">
                            {{$message}}
                        </div>
                        @enderror
                    </div>
                    <span class="infoAcc text-uppercase">Table Description
:</span>
                    <div class="mb-2 form-group">
                        <textarea class="form-control mt-1 contxt"
```

```
name="about" id="exampleFormControlTextarea1" rows="3">{{$table-
>about}}</textarea>
                        @error('about')
                        <div class="mt-2 invalid-form">
                            {{$message}}
                        </div>
                        @enderror
                    </div>
                    <div class="form-outline form-white text-center">
                        <span class="infoAcc text-uppercase">Image
:</span><br>
                        <input type="file" id="image" name="image"
class="form-control form-control-lg mt-1"/>
                        @error('image')
                        <div class="mt-2 invalid-form">
                            {{$message}}
                        </div>
                    @enderror
                    </div>
                    <div class="mt-4">
                        <button type="submit" class="btn btn-outline-light
btn-lg px-4" name="submit">Update</button>
                    </div>
                    </form>
                </div>

            </div>
        </div>
    </section>
@endsection
```

4.1.7.13 Pages.main.editUser.blade.php

```
@extends('layouts.layout')

@section('title') - Edit Account @endsection

@section('content')
    <section class=" login-bg">
        <div class="container d-flex align-items-center justify-content-
center">
            <div class="bg-dark card col-12 col-lg-7 m-5  text-center d-
flex flex-column align-items-center justify-content-center profilePersonal
profilLeft">

                <div class="w-100 m-4 bg-dark card d-flex flex-column
align-items-center justify-content-center profilePersonal postBorder pb-4">
{{--{{dd($user)}}--}}

                    <span class="text-left w-100 p-3 infoBorder"><h2>Edit
Account</h2></span>
                        <form class="col-10" method="post"
action="{{route('account.update',$user->id)}}" enctype="multipart/form-
data">
                            @csrf
                            @method('PUT')
                    <div class="form-outline  form-white mt-2 text-center
custom-file">
                        <span class="infoAcc text-uppercase">Username
:</span><br>
                        <input type="text" class="form-control mt-2 pb-2"
```



```
id="username" value="{{$user->username}}" readonly>
                        <input type="hidden" id="userId" name="userId"
value="{{$user->id}}">
                    </div>
                    <br>

                    <div class=" form-outline form-white mt-3 mb-2 text-
center">
                        <span class="infoAcc text-uppercase ">Roles
:</span><br>
                        @foreach($roles as $role)
                            <input class="form-check-input" type="checkbox"
name="gameroles[]" value="{{$role->id}}" id="role{{$role->id}}"
                            @if(isset($user) && in_array($role->id, $user-
>gameroles()->pluck('game_roles.id')->toArray()))
                                checked
                                   @elseif(is_array(old('game_roles.id'))
&& in_array($role->id, old('game_roles.id')))
                                   checked
                                @endif
                            >
                            <label class="form-check-label"
for="day{{$role->id}}">
                                {{$role->name}}
                            </label><br>
                        @endforeach
                    </div>
                            @error('gameroles')
                            <div class="mt-2 invalid-form">
                                {{$message}}
                            </div>
                            @enderror
                            <span class="infoAcc text-uppercase mt-2">About
:</span>
                    <div class=" form-group">
                        <textarea class="contxt form-control mt-1"
id="userAbout" name="about" rows="3">{{ $user->about ?? old('about')
}}</textarea>
                    </div>
                            <div class="form-outline col-10 form-white
text-center custom-file">
                        <span class="infoAcc text-uppercase">Image
:</span><br>
                        <input type="file" class="form-control mt-2 pb-2"
id="userImg" name="userImg">
                                @error('image')
                                <div class="mt-2 invalid-form">
                                    {{$message}}
                                </div>
                                @enderror
                            </div>
                    <div class="form-group mt-4">
                        <button type="submit" class="btn btn-outline-light
btn-lg px-5" id="updateAcc" name="update">Update</button>
                    </div>
                    </form>
                </div>

            </div>
        </div>
```

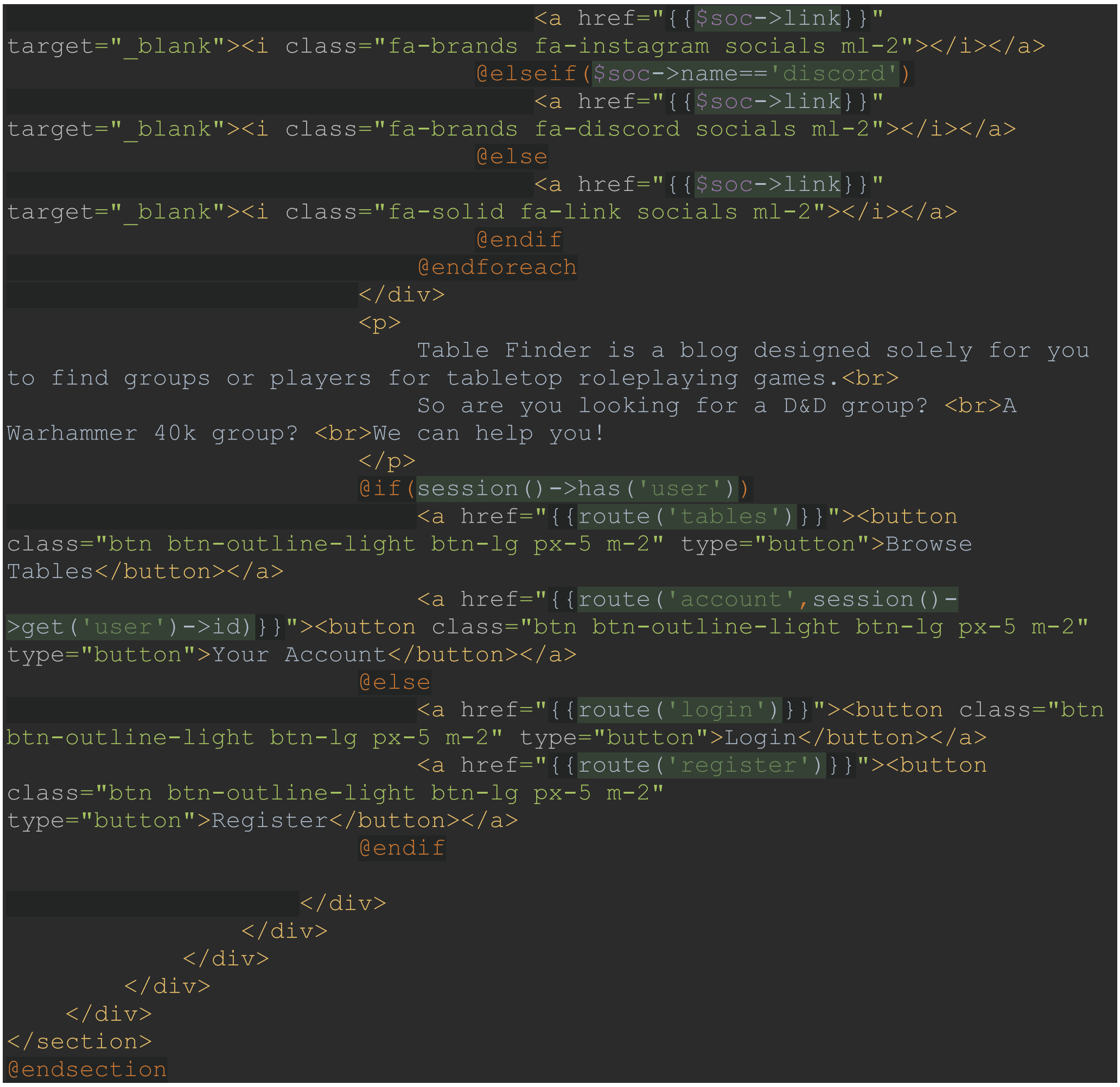
```
    </section>
@endsection
```

4.1.7.14 Pages.main.error.blade.php

```
@extends('layouts.layout')
@section('title') - 404 @endsection

@section('content')
    <section class="vh-100 login-bg">
        <div class="container  h-100">
            <div class="row d-flex justify-content-center align-items-
center h-100">
                <div class="col-12 col-md-8 col-lg-6 col-xl-5">
                    <div class="card bg-dark text-white" style="border-
radius: 1rem;">
                        <div class="card-body p-5 text-center">
                            <div class="mb-md-3 mt-md-4">
                                <h2 class="fw-bold mb-2 text-
uppercase">Error.</h2>
                                <a href="{{route('home')}}"><button
class="btn btn-outline-light btn-lg px-5 mt-2" type="submit">Return
Home</button></a>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </section>
@endsection
```

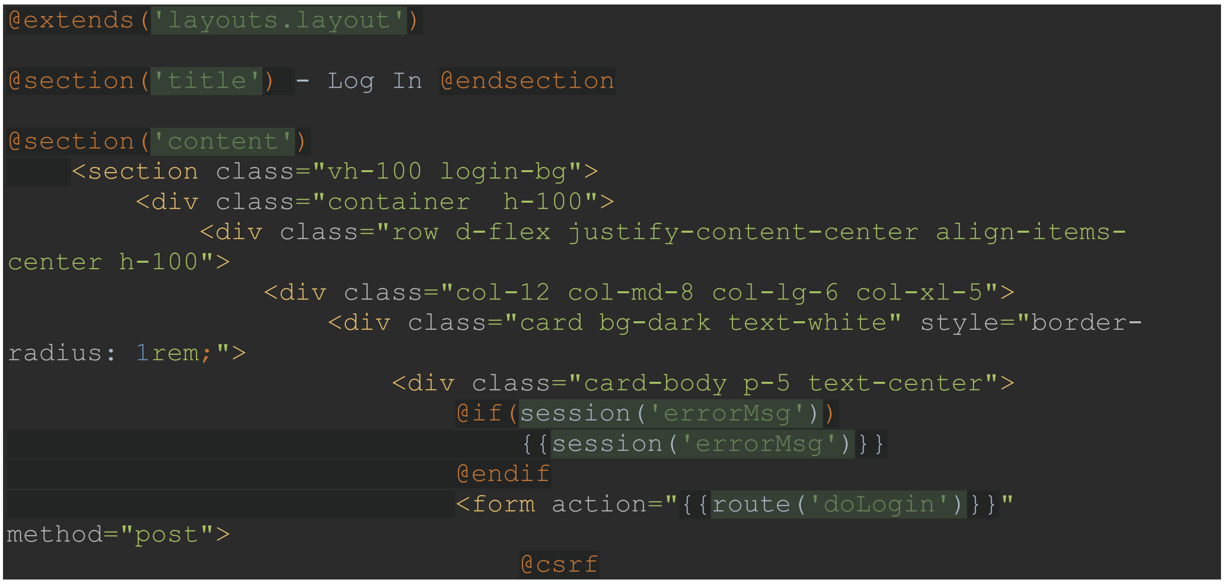
4.1.7.15 Pages.main.home.blade.php

```
@extends('layouts.layout')
@section('title') - Home @endsection

@section('content')
<section class="vh-100 login-bg">
    <div class="container  h-100">
        <div class="row d-flex justify-content-center align-items-center h-
100">
            <div class="col-12 col-md-10 m-5 ">
                <div class="card bg-dark text-white d-flex flex-row
justify-content-center align-items-center" style="border-radius: 1rem;">
                    <div class="card-body col-6 p-3 text-center">
                        <img src="{{asset('assets/img/home.jpg')}}"
class="img-fluid" id="homeImg">
                    </div>
                    <div class="col-6 text-center h-100">
                        <h1><i class="fa-solid fa-dice-d20"></i><br> Table
Finder</h1>
                        <div class="mb-2">
                            @foreach($socials as $soc)
                                @if($soc->name=='twitter')
                                    <a href="{{$soc->link}}"
target="_blank"><i class="fa-brands fa-twitter socials ml-2"></i></a>
                                @elseif($soc->name=='facebook')
                                    <a href="{{$soc->link}}"
target="_blank"><i class="fa-brands fa-facebook socials ml-2"></i></a>
                                @elseif($soc->name=='instagram')
```



```
                                    <a href="{{$soc->link}}"
target="_blank"><i class="fa-brands fa-instagram socials ml-2"></i></a>
                                @elseif($soc->name=='discord')
                                    <a href="{{$soc->link}}"
target="_blank"><i class="fa-brands fa-discord socials ml-2"></i></a>
                                @else
                                    <a href="{{$soc->link}}"
target="_blank"><i class="fa-solid fa-link socials ml-2"></i></a>
                                @endif
                            @endforeach
                        </div>
                        <p>
                            Table Finder is a blog designed solely for you
to find groups or players for tabletop roleplaying games.<br>
                            So are you looking for a D&D group? <br>A
Warhammer 40k group? <br>We can help you!
                        </p>
                        @if(session()->has('user'))
                            <a href="{{route('tables')}}"><button
class="btn btn-outline-light btn-lg px-5 m-2" type="button">Browse
Tables</button></a>
                            <a href="{{route('account',session()-
>get('user')->id)}}"><button class="btn btn-outline-light btn-lg px-5 m-2"
type="button">Your Account</button></a>
                        @else
                            <a href="{{route('login')}}"><button class="btn
btn-outline-light btn-lg px-5 m-2" type="button">Login</button></a>
                            <a href="{{route('register')}}"><button
class="btn btn-outline-light btn-lg px-5 m-2"
type="button">Register</button></a>
                        @endif

                    </div>
                </div>
            </div>
        </div>
    </div>
</section>
@endsection
```

4.1.7.16 Pages.main.login.blade.php



```
@extends('layouts.layout')

@section('title') - Log In @endsection

@section('content')
    <section class="vh-100 login-bg">
        <div class="container  h-100">
            <div class="row d-flex justify-content-center align-items-
center h-100">
                <div class="col-12 col-md-8 col-lg-6 col-xl-5">
                    <div class="card bg-dark text-white" style="border-
radius: 1rem;">
                        <div class="card-body p-5 text-center">
                            @if(session('errorMsg'))
                                {{session('errorMsg')}}
                            @endif
                            <form action="{{route('doLogin')}}"
method="post">
                                @csrf
                                <div class="mb-md-5 mt-md-4 pb-3">
                                    <h2 class="fw-bold mb-2 text-uppercase"><i class="fa-solid fa-dice-d20"></i> Login</h2>
                                    <div class="form-outline form-white mb-3">
                                        <input type="email" id="email" name="email" class="form-control form-control-lg mt-3" />
                                        <label class="form-label mt-2" for="email">Email</label>
                                        @error('email')
                                        <div class="invalid-form">
                                            {{$message}}
                                        </div>
                                        @enderror
                                    </div>
                                    <div class="form-outline form-white mb-4">
                                        <input type="password" id="password" name="password" class="form-control form-control-lg" />
                                        <label class="form-label mt-2" for="password">Password</label>
                                        @error('password')
                                        <div class="invalid-form">
                                            {{$message}}
                                        </div>
                                        @enderror
                                    </div>
                                    <button class="btn btn-outline-light btn-lg px-5" type="submit">Login</button>
                                </div>
                                <div>
                                    <p class="mb-0">Don't have an account? <a href="{{route('register')}}" class="text-white-50 fw-bold">Sign Up</a></p>
                                </div>
                            </form>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </section>
@endsection
```

4.1.7.17 Pages.main.mail.blade.php

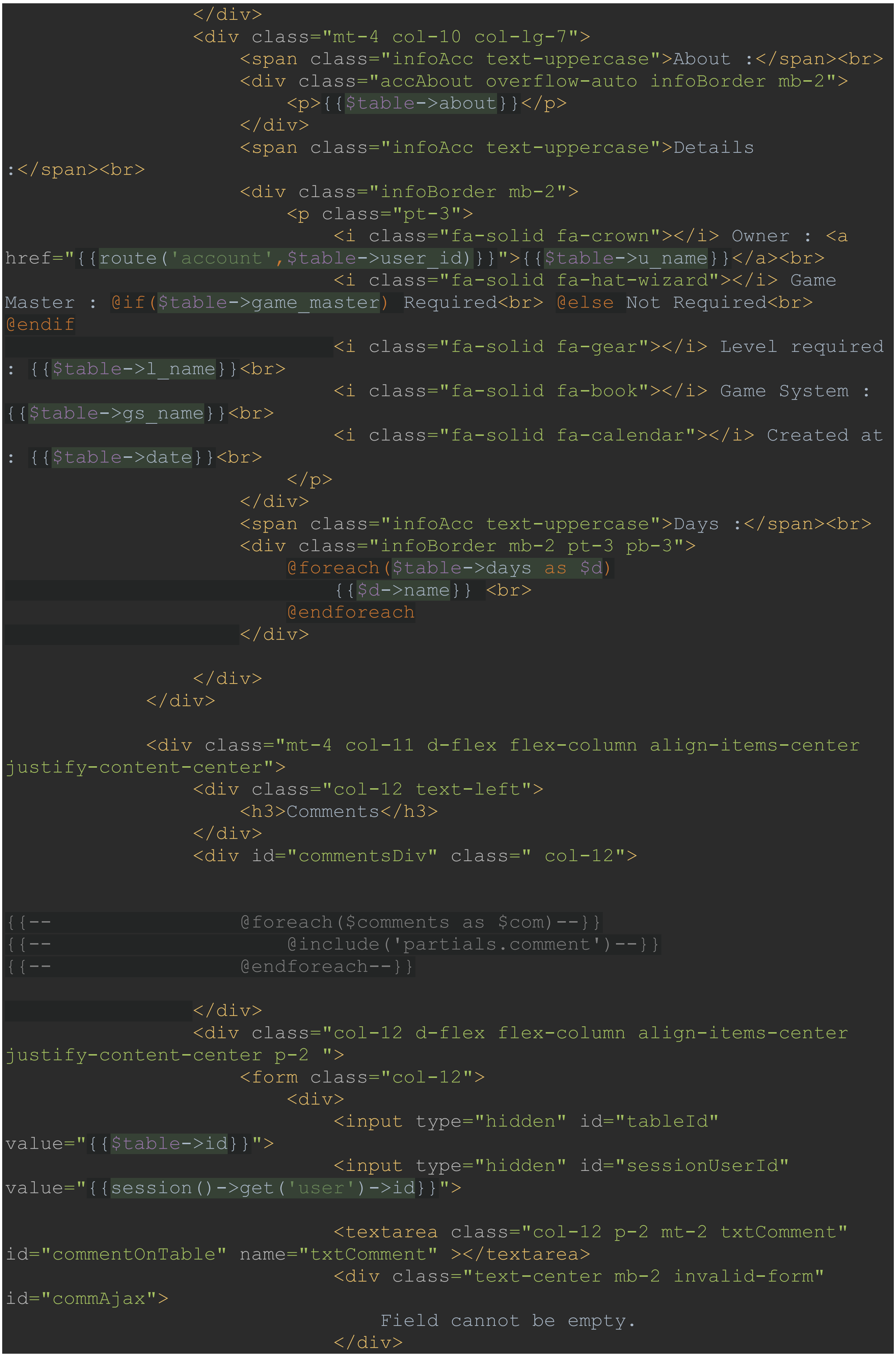
```
<!DOCTYPE html>
<html>
<head>
    <title>Tablefinder.com</title>
</head>
<body>
    <h4>From:</h4>
    <p>{{$data['email']}}</p>
    <h4>Subject:</h4>
    <p>{{$data['subject']}}</p>
    <h4>Message:</h4>
    <p>{{$data['message']}}</p>
</body>
</html>
```

4.1.7.18 Pages.main.register.blade.php

```
@extends('layouts.layout')

@section('title') - Sign Up @endsection

@section('content')
    <section class="vh-100 login-bg">
        <div class="container  h-100">
            <div class="row d-flex justify-content-center align-items-
center h-100">
                <div class="col-12 col-md-8 col-lg-6 col-xl-5">
                    <div class="card bg-dark text-white" style="border-
radius: 1rem;">
                        <div class="card-body p-5 text-center">

                            <div class="mb-md-3 mt-md-4">
                                <h2 class="fw-bold mb-3 text-uppercase"><i
class="fa-solid fa-dice-d20"></i> Sign up</h2>

                                <form action="{{route('register.user')}}"
method="post">
                                    @csrf
                                <div class="form-group mt-4">
                                    <input type="text" class="form-control
form-control-lg" id="username" name="username">
                                    <label for="username" class="form-label
mt-2 mb-0">Username</label><br>
                                    <p class=" small text-white-
50">Username can only contain letters and numbers, between 4 and 20
characters.</p>
                                    @error('username')
                                    <div class="invalid-form">
                                        {{$message}}
                                    </div>
                                    @enderror
                                </div>
                                <div class="form-group mt-2">
                                    <input type="text" class="form-control
form-control-lg" id="email" name="email">
                                    <label for="email" class="form-label
mt-2">Email</label><br>
                                    @error('email')
                                    <div class="invalid-form">
                                        {{$message}}
                                    </div>
                                    @enderror
                                </div>
                                <div class="form-group mt-2">
                                    <input type="password" class="form-
control form-control-lg" id="password" name="password">
                                    <label for="password" class="form-label
mt-2 mb-0">Password</label><br>
                                    <p class=" small text-white-
50">Password must be at least 8 characters, contain at least 1 lowercase,
at least 1 uppercase letter, special character, and at least 1 digit.</p>
                                    @error('password')
                                    <div class="invalid-form">
                                        {{$message}}
                                    </div>
                                    @enderror
                                </div>
                                <div class="form-group mt-2">
                                    <input type="password" class="form-
control form-control-lg" id="rePassword" name="password_confirmation">
                                    <label for="rePassword" class="form-
label mt-2"> Repeat Password</label><br>
                                </div>
                                <div class="form-group mt-4">
                                    <button type="submit" class="btn btn-
outline-light btn-lg px-5" id="register" name="register">Sign  Up</button>
                                </div>
                                </form>

                            </div>

                        </div>
                    </div>
                </div>
            </div>
        </div>
    </section>
@endsection
```

4.1.7.19 Pages.main.table.blade.php

```
@extends('layouts.layout')

@section('title') - {{$table->name}} @endsection

@section('content')
    <div class="container">
    <div class="bg-dark card  m-5 p-4 profilePersonal d-flex flex-column
align-items-center justify-content-center">

        <div class="w-100 bg-dark card d-flex flex-column align-items-
center justify-content-center profilePersonal postBorder pb-4">
            <span class="text-left w-100 p-3 infoBorder"><h2>{{$table-
>name}}
{{--{{dd($comments)}}--}}
                </h2></span>
            <div class="d-flex flex-column flex-lg-row pl-4 pr-4 align-
items-center justify-content-around">
                <div class="mt-4 col-10 col-lg-4">
                    <img src="{{asset('assets/img/'.$table->image)}}"
alt="{{$table->alt}}" class="mb-3 img-fluid postSlika">
                    <div class="text-center">
                    @if(session()->get('user')->id==$table->user_id)

                                <a href="{{route('table.edit',$table-
>id)}}">
                                    <i class="far fa-edit pr-2 pl-
2"></i>Edit Table
                                </a><br>
                        <form method="post"
action="{{route('table.destroy',$table->id)}}">
                            @csrf
                            @method('DELETE')
                            <button type="submit" class="mt-2 btn text-
light px-4"> <i class="far fa-trash-alt pr-2"></i>Delete Table</button>
                        </form>
                    @endif
                </div>
```



```
                </div>
                <div class="mt-4 col-10 col-lg-7">
                    <span class="infoAcc text-uppercase">About :</span><br>
                    <div class="accAbout overflow-auto infoBorder mb-2">
                        <p>{{$table->about}}</p>
                    </div>
                    <span class="infoAcc text-uppercase">Details
:</span><br>
                    <div class="infoBorder mb-2">
                        <p class="pt-3">
                            <i class="fa-solid fa-crown"></i> Owner : <a
href="{{route('account',$table->user_id)}}">{{$table->u_name}}</a><br>
                            <i class="fa-solid fa-hat-wizard"></i> Game
Master : @if($table->game_master) Required<br> @else Not Required<br>
@endif
                            <i class="fa-solid fa-gear"></i> Level required
: {{$table->l_name}}<br>
                            <i class="fa-solid fa-book"></i> Game System :
{{$table->gs_name}}<br>
                            <i class="fa-solid fa-calendar"></i> Created at
: {{$table->date}}<br>
                        </p>
                    </div>
                    <span class="infoAcc text-uppercase">Days :</span><br>
                    <div class="infoBorder mb-2 pt-3 pb-3">
                        @foreach($table->days as $d)
                            {{$d->name}} <br>
                        @endforeach
                    </div>

                </div>
            </div>

            <div class="mt-4 col-11 d-flex flex-column align-items-center
justify-content-center">
                <div class="col-12 text-left">
                    <h3>Comments</h3>
                </div>
                <div id="commentsDiv" class=" col-12">


{{--                    @foreach($comments as $com)--}}
{{--                        @include('partials.comment')--}}
{{--                    @endforeach--}}

                </div>
                <div class="col-12 d-flex flex-column align-items-center
justify-content-center p-2 ">
                    <form class="col-12">
                        <div>
                            <input type="hidden" id="tableId"
value="{{$table->id}}">
                            <input type="hidden" id="sessionUserId"
value="{{session()->get('user')->id}}">

                            <textarea class="col-12 p-2 mt-2 txtComment"
id="commentOnTable" name="txtComment" ></textarea>
                            <div class="text-center mb-2 invalid-form"
id="commAjax">
                                Field cannot be empty.
                            </div>
                        </div>
                        <div class="text-center mt-2">
                            <button type="button" class="btn btn-outline-
light btn-lg px-4" id="commSubmit" name="submit">Comment</button>
                        </div>

                    </form>
                </div>
            </div>




        </div>
    </div>
    </div>
@endsection
```

4.1.7.20 Pages.main.tables.blade.php

```
@extends('layouts.layout')

@section('title') - Find Tables @endsection

@section('content')
    <div class="d-flex flex-column flex-lg-row align-items-center align-
items-lg-start justify-content-center">
        <div class="order-2 order-lg-1 bg-dark card col-11 col-lg-7 m-5 p-4
profilePersonal d-flex flex-column align-items-center justify-content-
center">
            <div class="col-12 bg-dark card d-flex flex-column align-items-
center justify-content-center profilePersonal postBorder pb-4">
                <div id="allTables" class="col-12 flex-xl-row flex-column
d-flex align-items-center justify-content-around flex-wrap">
                </div>
                <div class="d-flex justify-content-center col-12 mt-4"
id="pagination">
                </div>
            </div>
        </div>
        <div class="order-1 order-lg-2 bg-dark card col-6 col-lg-3 m-5
text-center d-flex flex-column align-items-center justify-content-center
profilePersonal profilLeft">

            <div class="w-100 m-4 bg-dark card d-flex flex-column align-
items-center justify-content-center profilePersonal postBorder pb-4">

            <span class="text-left w-100 p-3 infoBorder"><h2>Tabletop
Search</h2></span>
                <form>
                    <input type="hidden" id="first" value="1">
                <div class="form-outline  form-white text-center mt-2">
                    <span class="infoAcc text-uppercase">Table :</span><br>
                    <input type="text" id="tableName" name="table"
class="form-control form-control-lg mt-1" />
                </div>
                <span class="infoAcc text-uppercase mt-2">Availability
:</span>
                <div class="pl-5 form-outline form-white  mb-2 text-left">
                        @foreach($days as $d)
                        <input class="form-check-input" type="checkbox"
```

```
class="days" name="days" value="{{$d->id}}" id="day{{$d->id}}">
                        <label class="form-check-label" for="day{{$d-
>id}}">
                            {{$d->name}}
                        </label><br>
                        @endforeach

                </div>

                <div class="form-outline  form-white text-center mt-2">
                    <span class="infoAcc text-uppercase">GAME MASTER
REQUIRED :</span><br>
                    <input type="radio" id="gm" name="gm" class="mt-1"
value="1"/> Yes
                    <input type="radio" id="gm" name="gm" class="ml-5 mt-1"
value="0"/> No
                </div>

                <div class="form-outline  form-white ">
                    <select class="custom-select form-control mt-3"
id="gSys" name="gSys">
                        <option value="0" selected>Game System...</option>
                        @foreach($gameSystems as $gs)
                        <option value="{{$gs->id}}">{{$gs->name}}</option>
                        @endforeach
                    </select>
                </div>

                <div class="form-outline form-white ">
                    <select class="custom-select form-control mt-3 mb-2"
id="ddSort" name="ddSort">
                        <option value="ASC-id" selected>Sort By...</option>
                        <option value="DESC-date">Newest</option>
                        <option value="ASC-date">Oldest</option>
                        <option value="ASC-name">A-Z</option>
                        <option value="DESC-name">Z-A</option>
                    </select>
                </div>
                <div class="form-outline mt-2  form-white ">
                    <select class="custom-select form-control mb-2"
id="lvl" name="ddSort">
                        <option value="0" selected>Level...</option>
                        @foreach($levels as $lev)
                            <option value="{{$lev->id}}">{{$lev-
>name}}</option>
                        @endforeach
                    </select>
                </div>
                <div class="mt-4">
                    <button type="button" class="btn btn-outline-light btn-
lg px-4" id="searchTables" name="submit">Search</button>
                    <button type="button" class="ml-3 btn btn-outline-light
btn-lg px-4" id="resetTables" name="reset">Reset</button>
                </div>
                </form>
            </div>

            <div id="carouselExampleControls" class="col-10 mb-4 carousel
slide" data-ride="carousel">
                <div class="carousel-inner">
                    @foreach($ads as $ad)
```

```
                        <div class="carousel-item @if($loop->index == 0)
active @endif">
                            <img class="d-block img-fluid carouselImages"
src="{{ asset('assets/img/' . $ad->image) }}" alt="">
                            <div class="carousel-caption d-none d-md-
block">
                                <a href="{{$ad->link}}"
target="_blank"><h5>{{$ad->text}}</h5></a>
                            </div>
                        </div>
                    @endforeach
                </div>
                <a class="carousel-control-prev"
href="#carouselExampleControls" role="button" data-slide="prev">
                    <span class="carousel-control-prev-icon" aria-
hidden="true"></span>
                    <span class="sr-only">Previous</span>
                </a>
                <a class="carousel-control-next"
href="#carouselExampleControls" role="button" data-slide="next">
                    <span class="carousel-control-next-icon" aria-
hidden="true"></span>
                    <span class="sr-only">Next</span>
                </a>
            </div>

        </div>

    </div>


@endsection
```

4.1.7.21 Partials.error.blade.php

```
@if(session()->has('errorMsg'))
    <div class="container mt-5 d-flex justify-content-center">
        <div class="alert alert-danger col-7" role="alert">
            {{ session('errorMsg')}}
        </div>
    </div>
@endif
```

4.1.7.22 Partials.success.blade.php

```
@if(session()->has('successMsg'))
    <div class="container mt-5 d-flex justify-content-center">
        <div class="alert alert-success col-7" role="alert">
            {{ session('successMsg')}}
        </div>
    </div>
@endif
```

4.1.7.23 Partials.table.blade.php

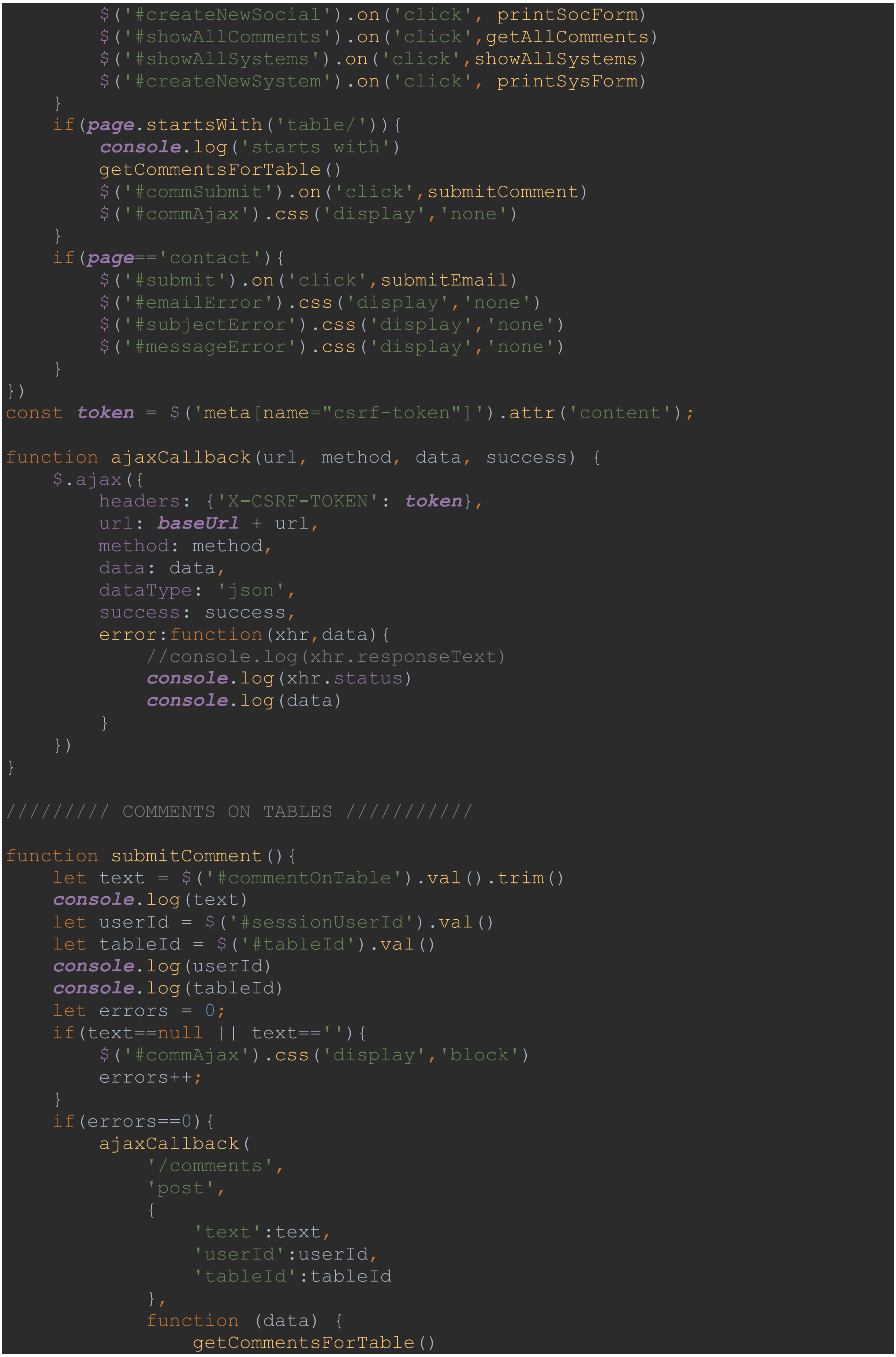
```
<div class="mt-4 mb-2 col-12 col-xl-5 p-3 testdiv d-flex flex-column
justify-content-between">
    <span class="text-left w-100 p-2 infoBorder"><h3>{{$table-
>name}}</h3></span>
    <div class="d-flex">
        <div class="col-6 d-flex justify-content-center align-items-
```

```
center">
            <img src="{{asset('assets/img/'.$table->image)}}"
alt="{{$table->alt}}" class="img-fluid pSlika">
        </div>
        <div class="col-6 p-0 d-flex justify-content-center align-items-
center">
            <p class="pt-3">
                <i class="fa-solid fa-hat-wizard"></i> Game Master : <br>
@if($table->game_master) Required<br> @else Not Required<br> @endif
                <i class="fa-solid fa-gear"></i> Level required :
<br>{{$table->l_name}}<br>
                <i class="fa-solid fa-calendar"></i> Availability : <br>
                @foreach($table->days as $day)
                    {{substr($day->name,0,3)}}
                @endforeach
                <br>
                <i class="fa-solid fa-book"></i> Game System :
<br>{{$table->gs_name}}<br>
            </p>
        </div>
    </div>

    <div class="text-right tableActions pt-3">
        <a href="{{route('table',$table->id)}}"><i class="fa-solid fa-eye
pr-2 pl-2"></i>see more</a>
        @if(session()->get('user')->id==$table->user_id)
            <a href="{{route('table.edit',$table->id)}}"><i class="far fa-
edit pr-2 pl-2"></i>edit</a>
            <form method="post" action="{{route('table.destroy',$table-
>id)}}">
                @csrf
                @method('DELETE')
                <button type="submit" class="p-0 btn text-light"><i
class="far fa-trash-alt pr-2 pl-2"></i>delete</button>
            </form>
        @endif
    </div>
</div>
```

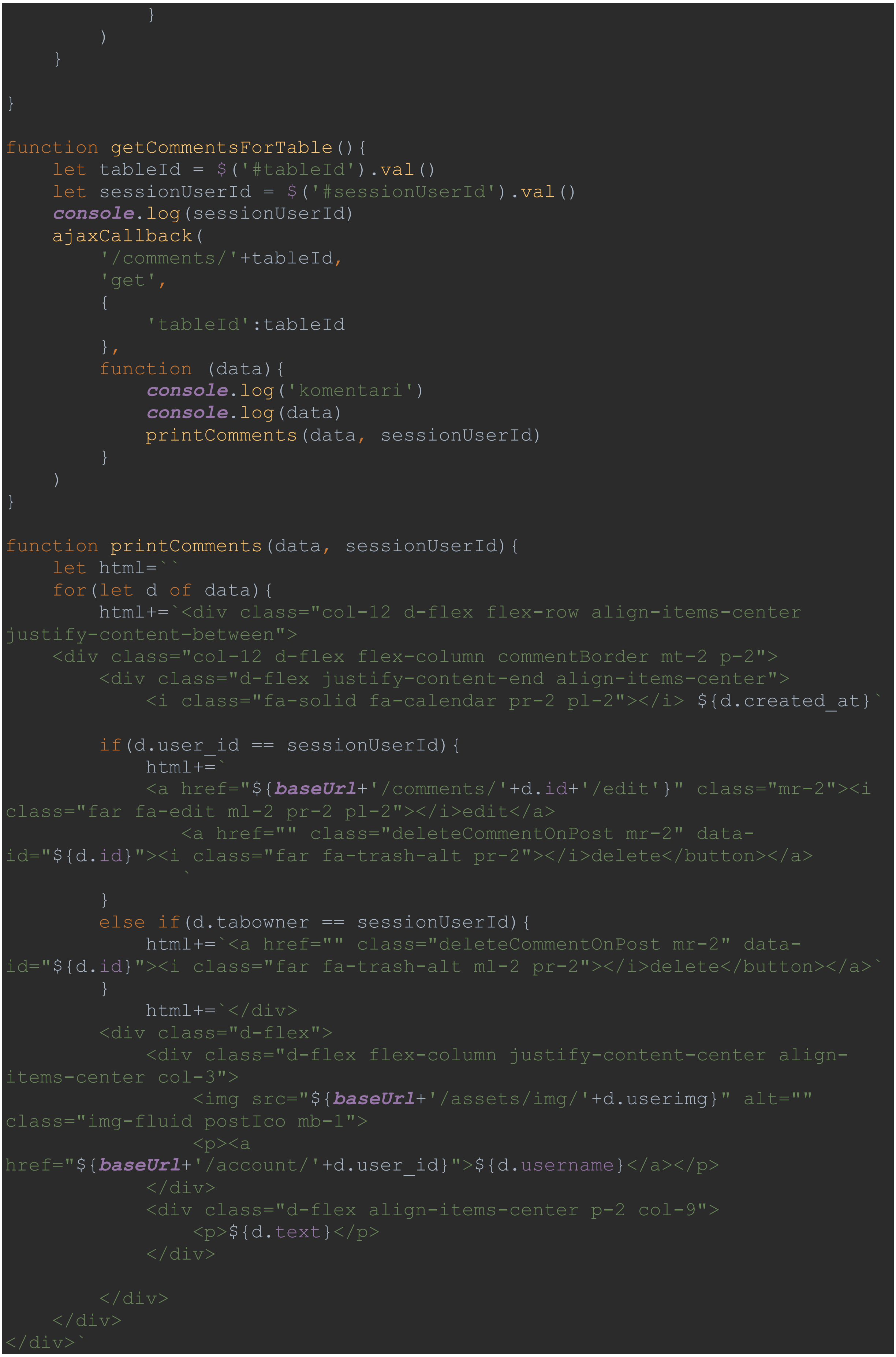
4.2 JAVASCRIPT

4.2.1 Main.js

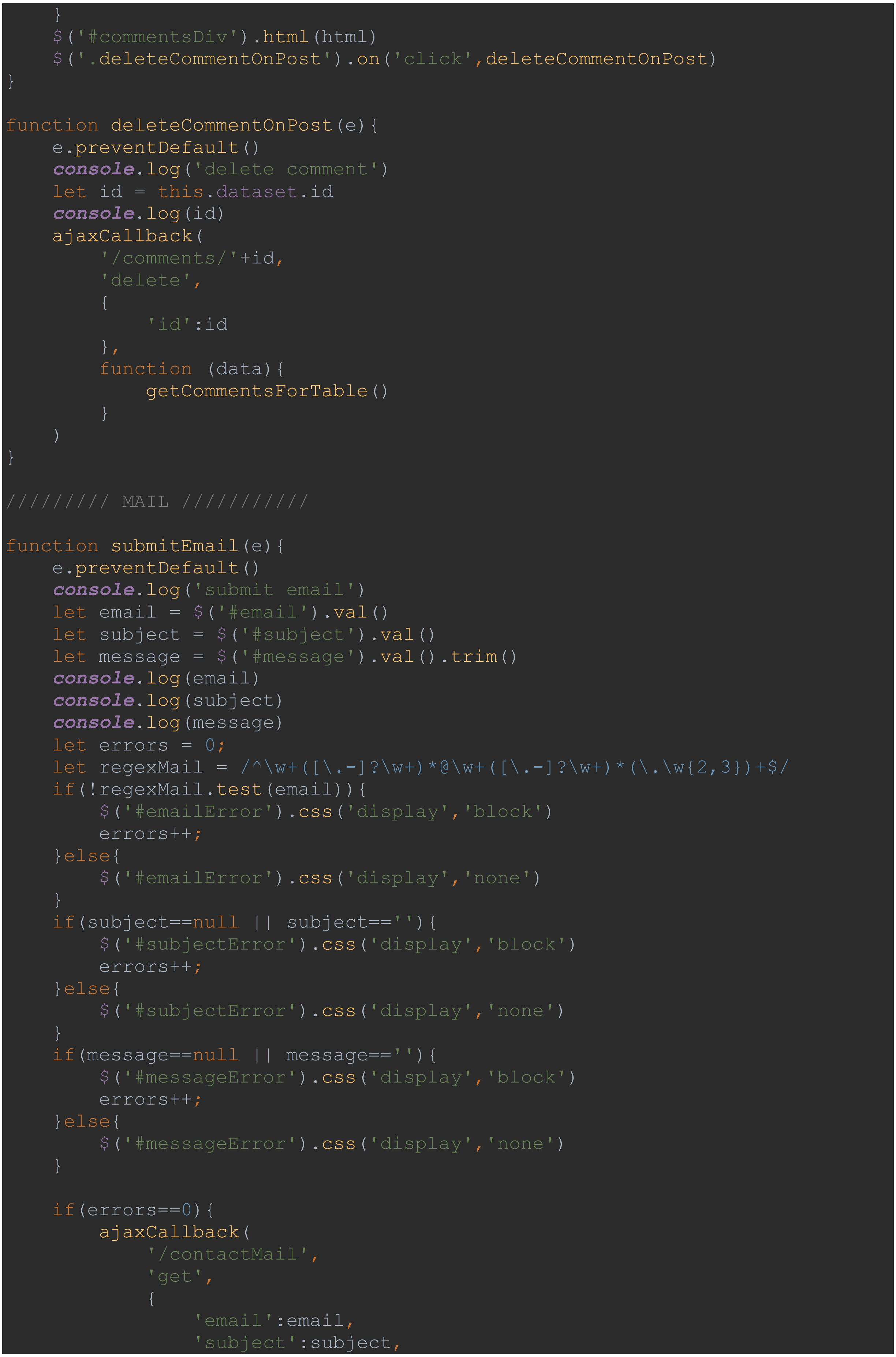
```
$(document).ready(function (){

    console.log(page)
    console.log(baseUrl)
    if(page=='tables'){
        getAllTables()
        $('#searchTables').on('click',filterTables)
        $('#resetTables').on('click',getAllTables)
    }
    if(page=='admin'){
        $('#showAllActivity').on('click',getAllActivity)
        $('#showAllUsers').on('click',getAllUsers)
        $('#createNewUser').on('click', userCreate)
        $('#showAllTables').on('click',getAllAdminTables)
        $('#showAllAds').on('click',showAllAds)
        $('#createNewAdd').on('click', printAdForm)
        $('#showAllMessages').on('click',getAllMessages)
        $('#showAllSocials').on('click',showAllSocials)
```



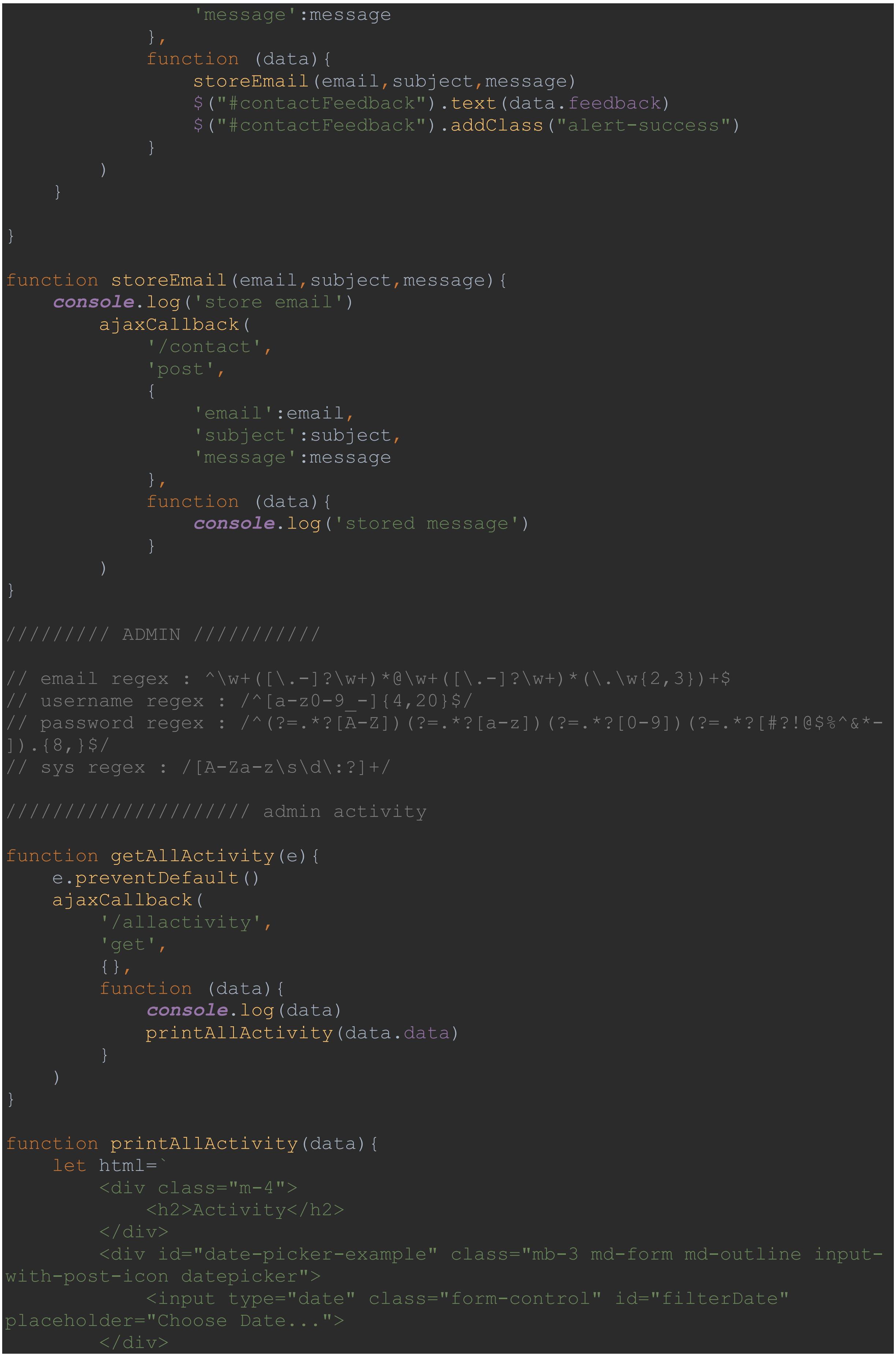
```
        $('#createNewSocial').on('click', printSocForm)
        $('#showAllComments').on('click',getAllComments)
        $('#showAllSystems').on('click',showAllSystems)
        $('#createNewSystem').on('click', printSysForm)
    }
    if(page.startsWith('table/')){
        console.log('starts with')
        getCommentsForTable()
        $('#commSubmit').on('click',submitComment)
        $('#commAjax').css('display','none')
    }
    if(page=='contact'){
        $('#submit').on('click',submitEmail)
        $('#emailError').css('display','none')
        $('#subjectError').css('display','none')
        $('#messageError').css('display','none')
    }
})
const token = $('meta[name="csrf-token"]').attr('content');

function ajaxCallback(url, method, data, success) {
    $.ajax({
        headers: {'X-CSRF-TOKEN': token},
        url: baseUrl + url,
        method: method,
        data: data,
        dataType: 'json',
        success: success,
        error:function(xhr,data){
            //console.log(xhr.responseText)
            console.log(xhr.status)
            console.log(data)
        }
    })
}

///////// COMMENTS ON TABLES ///////////

function submitComment(){
    let text = $('#commentOnTable').val().trim()
    console.log(text)
    let userId = $('#sessionUserId').val()
    let tableId = $('#tableId').val()
    console.log(userId)
    console.log(tableId)
    let errors = 0;
    if(text==null || text==''){
        $('#commAjax').css('display','block')
        errors++;
    }
    if(errors==0){
        ajaxCallback(
            '/comments',
            'post',
            {
                'text':text,
                'userId':userId,
                'tableId':tableId
            },
            function (data) {
                getCommentsForTable()
```



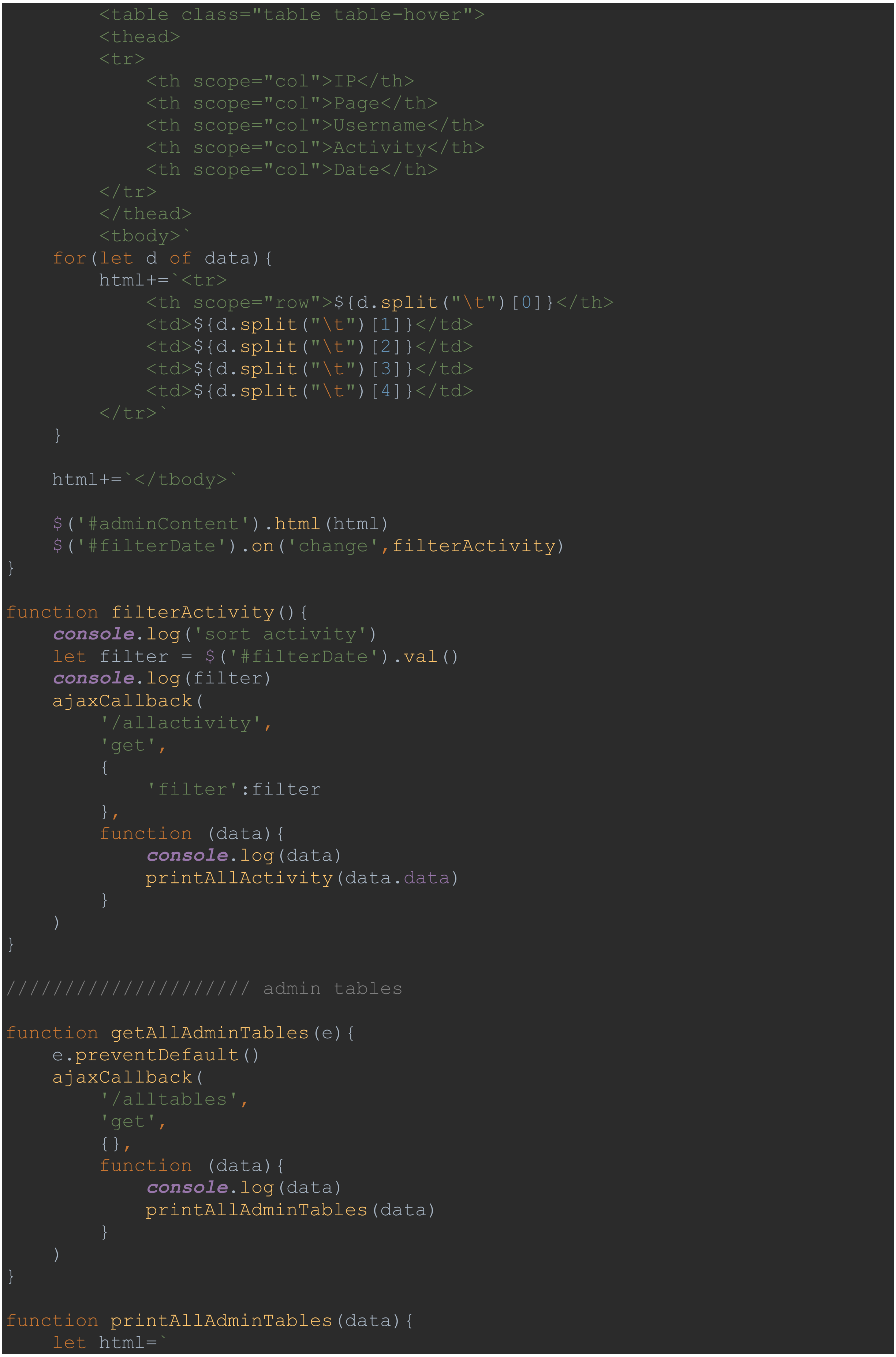
```
            }
        )
    }

}

function getCommentsForTable(){
    let tableId = $('#tableId').val()
    let sessionUserId = $('#sessionUserId').val()
    console.log(sessionUserId)
    ajaxCallback(
        '/comments/'+tableId,
        'get',
        {
            'tableId':tableId
        },
        function (data){
            console.log('komentari')
            console.log(data)
            printComments(data, sessionUserId)
        }
    )
}

function printComments(data, sessionUserId){
    let html=``
    for(let d of data){
        html+=`<div class="col-12 d-flex flex-row align-items-center
justify-content-between">
    <div class="col-12 d-flex flex-column commentBorder mt-2 p-2">
        <div class="d-flex justify-content-end align-items-center">
            <i class="fa-solid fa-calendar pr-2 pl-2"></i> ${d.created_at}`

        if(d.user_id == sessionUserId){
            html+=`
            <a href="${baseUrl+'/comments/'+d.id+'/edit'}" class="mr-2"><i
class="far fa-edit ml-2 pr-2 pl-2"></i>edit</a>
               <a href="" class="deleteCommentOnPost mr-2" data-
id="${d.id}"><i class="far fa-trash-alt pr-2"></i>delete</button></a>
               `
        }
        else if(d.tabowner == sessionUserId){
            html+=`<a href="" class="deleteCommentOnPost mr-2" data-
id="${d.id}"><i class="far fa-trash-alt ml-2 pr-2"></i>delete</button></a>`
        }
            html+=`</div>
        <div class="d-flex">
            <div class="d-flex flex-column justify-content-center align-
items-center col-3">
                <img src="${baseUrl+'/assets/img/'+d.userimg}" alt=""
class="img-fluid postIco mb-1">
                <p><a
href="${baseUrl+'/account/'+d.user_id}">${d.username}</a></p>
            </div>
            <div class="d-flex align-items-center p-2 col-9">
                <p>${d.text}</p>
            </div>

        </div>
    </div>
</div>`
```



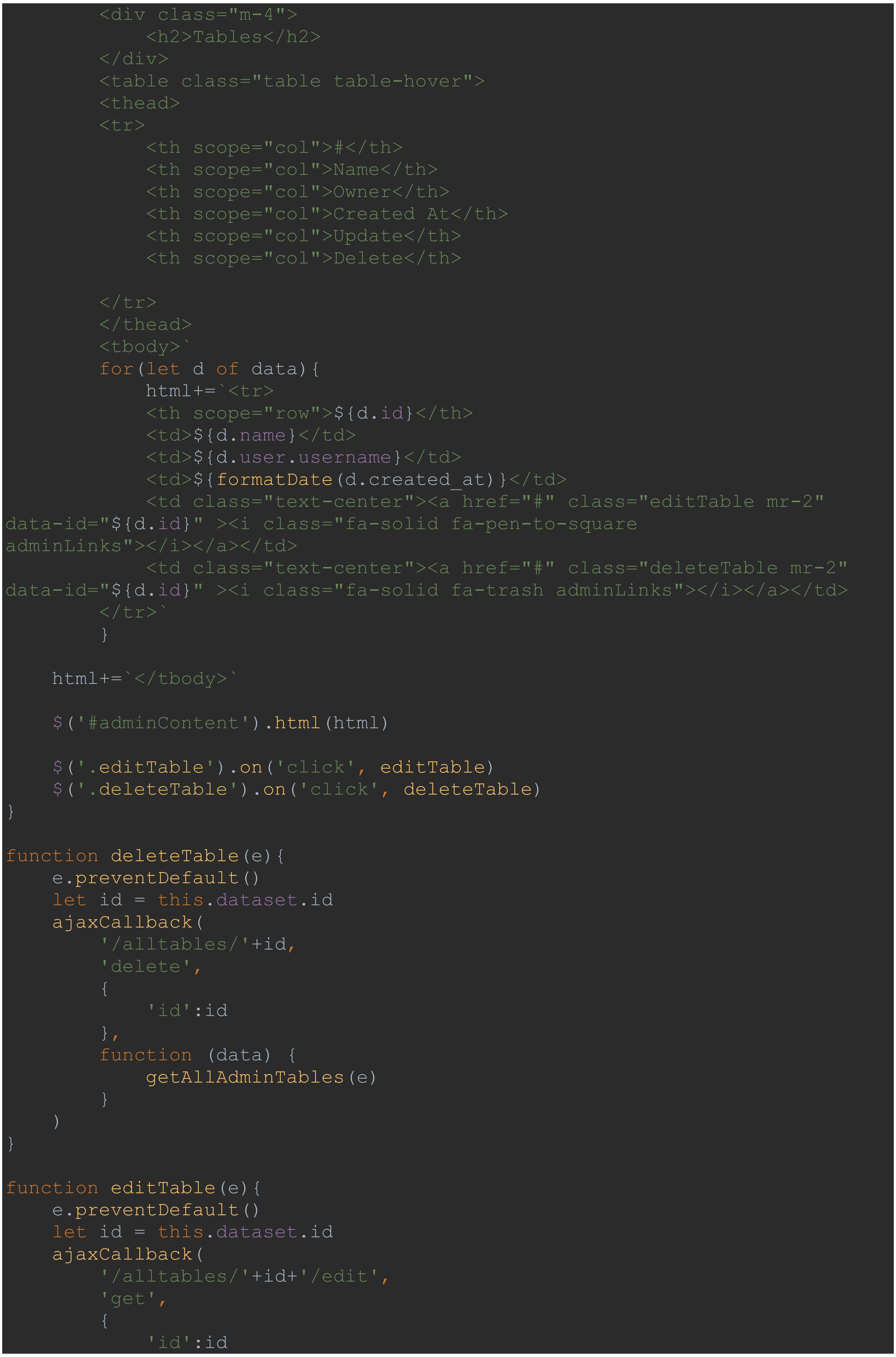
```
    }
    $('#commentsDiv').html(html)
    $('.deleteCommentOnPost').on('click',deleteCommentOnPost)
}

function deleteCommentOnPost(e){
    e.preventDefault()
    console.log('delete comment')
    let id = this.dataset.id
    console.log(id)
    ajaxCallback(
        '/comments/'+id,
        'delete',
        {
            'id':id
        },
        function (data){
            getCommentsForTable()
        }
    )
}

///////// MAIL ///////////

function submitEmail(e){
    e.preventDefault()
    console.log('submit email')
    let email = $('#email').val()
    let subject = $('#subject').val()
    let message = $('#message').val().trim()
    console.log(email)
    console.log(subject)
    console.log(message)
    let errors = 0;
    let regexMail = /^\w+([\.-]?\w+)*@\w+([\.-]?\w+)*(\.\w{2,3})+$/
    if(!regexMail.test(email)){
        $('#emailError').css('display','block')
        errors++;
    }else{
        $('#emailError').css('display','none')
    }
    if(subject==null || subject==''){
        $('#subjectError').css('display','block')
        errors++;
    }else{
        $('#subjectError').css('display','none')
    }
    if(message==null || message==''){
        $('#messageError').css('display','block')
        errors++;
    }else{
        $('#messageError').css('display','none')
    }

    if(errors==0){
        ajaxCallback(
            '/contactMail',
            'get',
            {
                'email':email,
                'subject':subject,
```



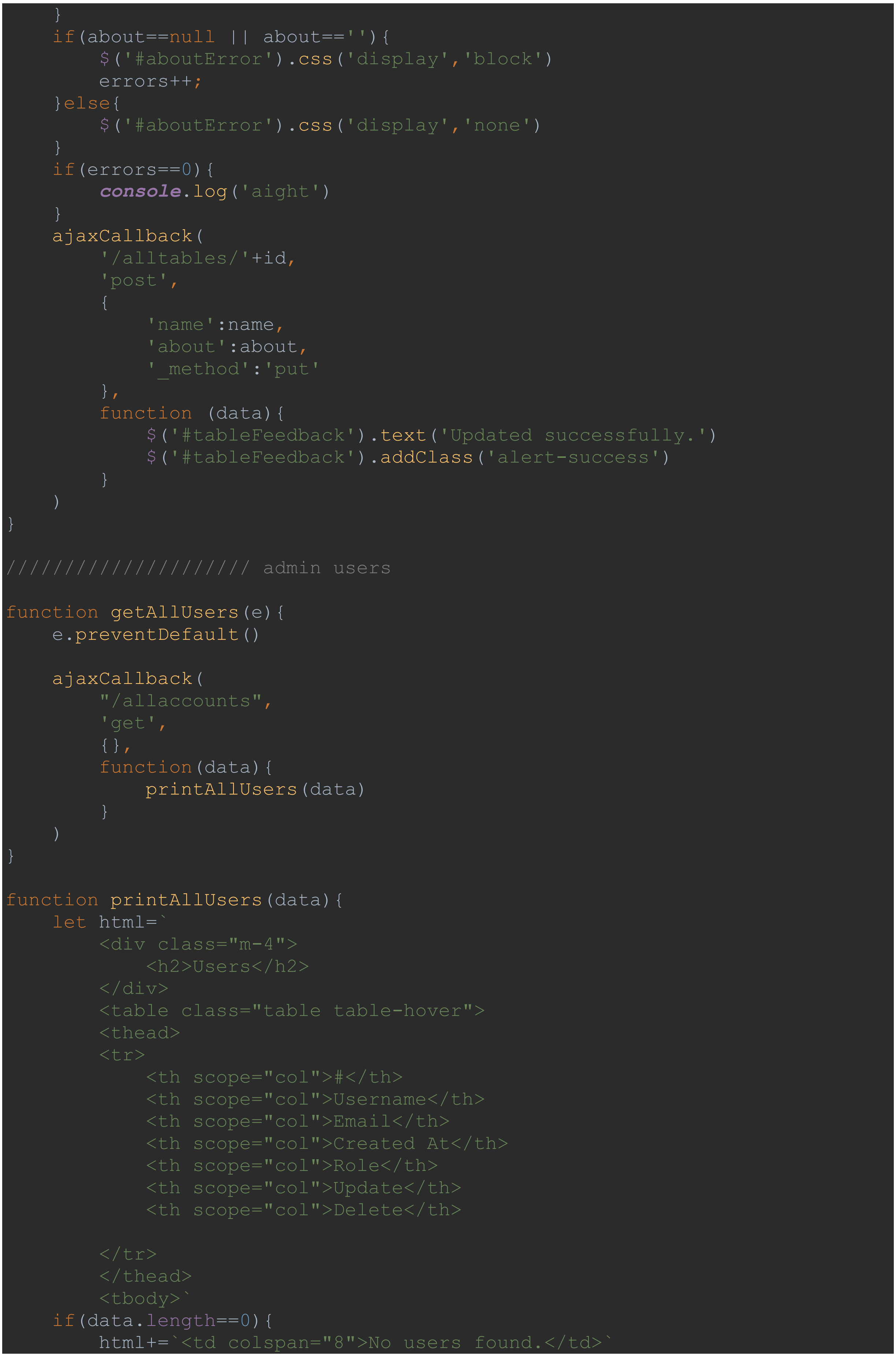
```
                'message':message
            },
            function (data){
                storeEmail(email,subject,message)
                $("#contactFeedback").text(data.feedback)
                $("#contactFeedback").addClass("alert-success")
            }
        )
    }

}

function storeEmail(email,subject,message){
    console.log('store email')
        ajaxCallback(
            '/contact',
            'post',
            {
                'email':email,
                'subject':subject,
                'message':message
            },
            function (data){
                console.log('stored message')
            }
        )
}

///////// ADMIN ///////////

// email regex : ^\w+([\.-]?\w+)*@\w+([\.-]?\w+)*(\.\w{2,3})+$
// username regex : /^[a-z0-9_-]{4,20}$/
// password regex : /^(?=.*?[A-Z])(?=.*?[a-z])(?=.*?[0-9])(?=.*?[#?!@$%^&*-
]).{8,}$/
// sys regex : /[A-Za-z\s\d\:?]+/

///////////////////// admin activity

function getAllActivity(e){
    e.preventDefault()
    ajaxCallback(
        '/allactivity',
        'get',
        {},
        function (data){
            console.log(data)
            printAllActivity(data.data)
        }
    )
}

function printAllActivity(data){
    let html=`
        <div class="m-4">
            <h2>Activity</h2>
        </div>
        <div id="date-picker-example" class="mb-3 md-form md-outline input-
with-post-icon datepicker">
            <input type="date" class="form-control" id="filterDate"
placeholder="Choose Date...">
        </div>
```



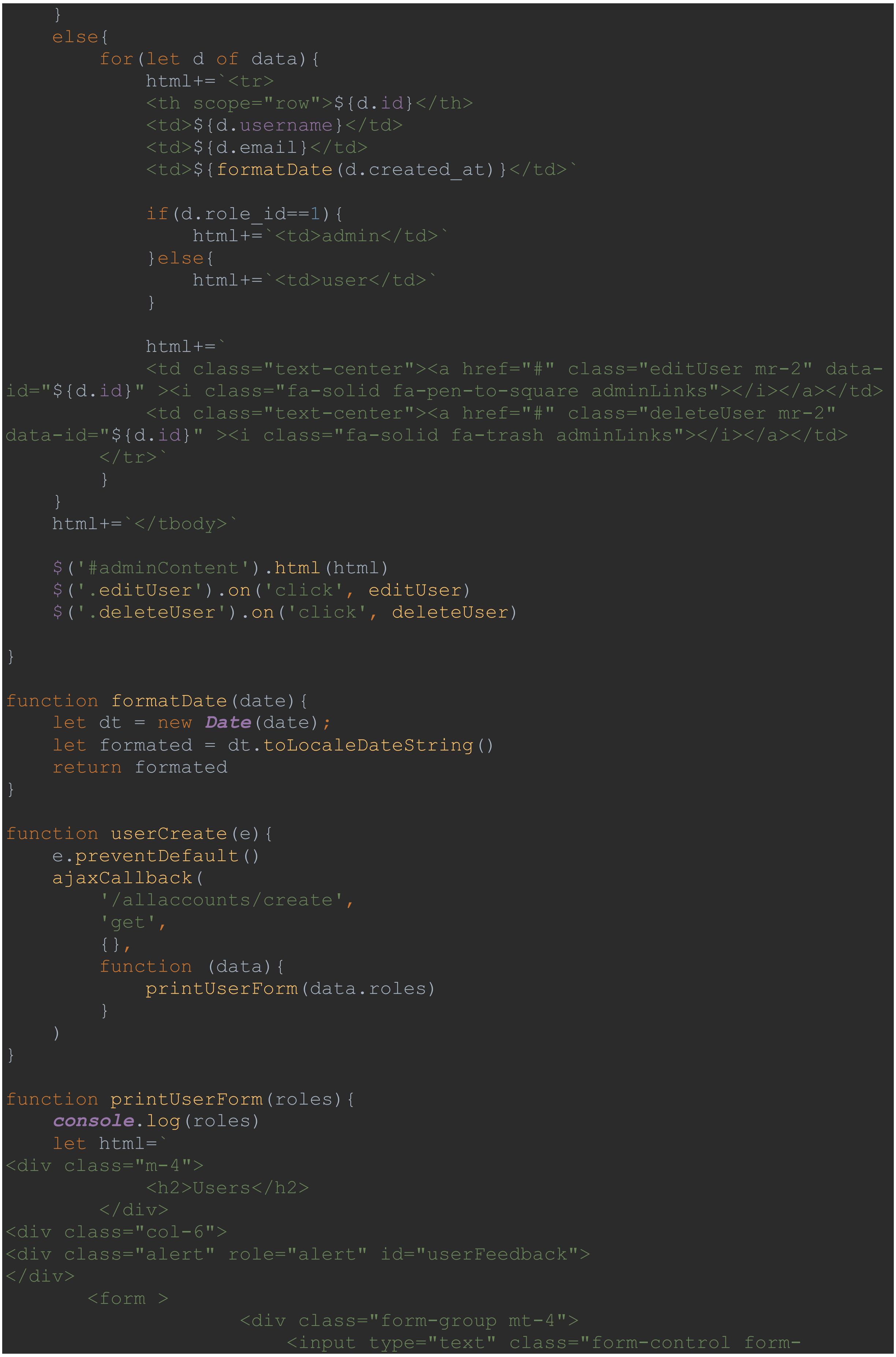
```
        <table class="table table-hover">
        <thead>
        <tr>
            <th scope="col">IP</th>
            <th scope="col">Page</th>
            <th scope="col">Username</th>
            <th scope="col">Activity</th>
            <th scope="col">Date</th>
        </tr>
        </thead>
        <tbody>`
    for(let d of data){
        html+=`<tr>
            <th scope="row">${d.split("\t")[0]}</th>
            <td>${d.split("\t")[1]}</td>
            <td>${d.split("\t")[2]}</td>
            <td>${d.split("\t")[3]}</td>
            <td>${d.split("\t")[4]}</td>
        </tr>`
    }

    html+=`</tbody>`

    $('#adminContent').html(html)
    $('#filterDate').on('change',filterActivity)
}

function filterActivity(){
    console.log('sort activity')
    let filter = $('#filterDate').val()
    console.log(filter)
    ajaxCallback(
        '/allactivity',
        'get',
        {
            'filter':filter
        },
        function (data){
            console.log(data)
            printAllActivity(data.data)
        }
    )
}

///////////////////// admin tables

function getAllAdminTables(e){
    e.preventDefault()
    ajaxCallback(
        '/alltables',
        'get',
        {},
        function (data){
            console.log(data)
            printAllAdminTables(data)
        }
    )
}

function printAllAdminTables(data){
    let html=`
```



```
        <div class="m-4">
            <h2>Tables</h2>
        </div>
        <table class="table table-hover">
        <thead>
        <tr>
            <th scope="col">#</th>
            <th scope="col">Name</th>
            <th scope="col">Owner</th>
            <th scope="col">Created At</th>
            <th scope="col">Update</th>
            <th scope="col">Delete</th>

        </tr>
        </thead>
        <tbody>`
        for(let d of data){
            html+=`<tr>
            <th scope="row">${d.id}</th>
            <td>${d.name}</td>
            <td>${d.user.username}</td>
            <td>${formatDate(d.created_at)}</td>
            <td class="text-center"><a href="#" class="editTable mr-2"
data-id="${d.id}" ><i class="fa-solid fa-pen-to-square
adminLinks"></i></a></td>
            <td class="text-center"><a href="#" class="deleteTable mr-2"
data-id="${d.id}" ><i class="fa-solid fa-trash adminLinks"></i></a></td>
        </tr>`
        }

    html+=`</tbody>`

    $('#adminContent').html(html)

    $('.editTable').on('click', editTable)
    $('.deleteTable').on('click', deleteTable)
}

function deleteTable(e){
    e.preventDefault()
    let id = this.dataset.id
    ajaxCallback(
        '/alltables/'+id,
        'delete',
        {
            'id':id
        },
        function (data) {
            getAllAdminTables(e)
        }
    )
}

function editTable(e){
    e.preventDefault()
    let id = this.dataset.id
    ajaxCallback(
        '/alltables/'+id+'/edit',
        'get',
        {
            'id':id
        },
        function (data){
            printEditTableForm(data)
        }
    )
}

function printEditTableForm(data){
    let html=`
<div class="m-4">
            <h2>Tables</h2>
        </div>
<div class="col-6">
<div class="alert" role="alert" id="tableFeedback">
                                </div>
                    <form enctype="multipart/form-data">
                        <div class="form-group mt-2">
                            <input type="text" class="form-control form-
control-lg" id="name" value="${data.name}">
                            <label for="email" class="form-label mt-
2">Name</label><br>
                            <div id="nameError" class="invalid-form">
                                Name required.
                            </div>
                        </div>
                        <div class="mb-2 form-group">
                        <textarea class="form-control mt-1 contxt"
name="about" id="about" rows="3">${data.about}</textarea>
                        <label for="email" class="form-label mt-
2">Description</label><br>
                        <div id="aboutError" class="mt-2 invalid-form">
                            Description required.
                        </div>
                    </div>
                        <div class="form-group mt-4">
                            <input type="hidden" id="tableId"
value="${data.id}">
                            <button type="submit" class="btn btn-outline-
dark btn-lg px-5" id="tableUpdateForm" name="">Update Table</button>
                        </div>
                    </form>
                </div>`
    $('#adminContent').html(html)
    $('#tableUpdateForm').on('click',updateTable)
    $('#nameError').css('display','none')
    $('#aboutError').css('display','none')

}

function updateTable(e){
    e.preventDefault()
    let id = $('#tableId').val()
    let name = $('#name').val()
    let about = $('#about').val()
    console.log(about)
    let errors=0
    if(name==null || name==''){
        $('#nameError').css('display','block')
        errors++;
    }else{
        $('#nameError').css('display','none')
```

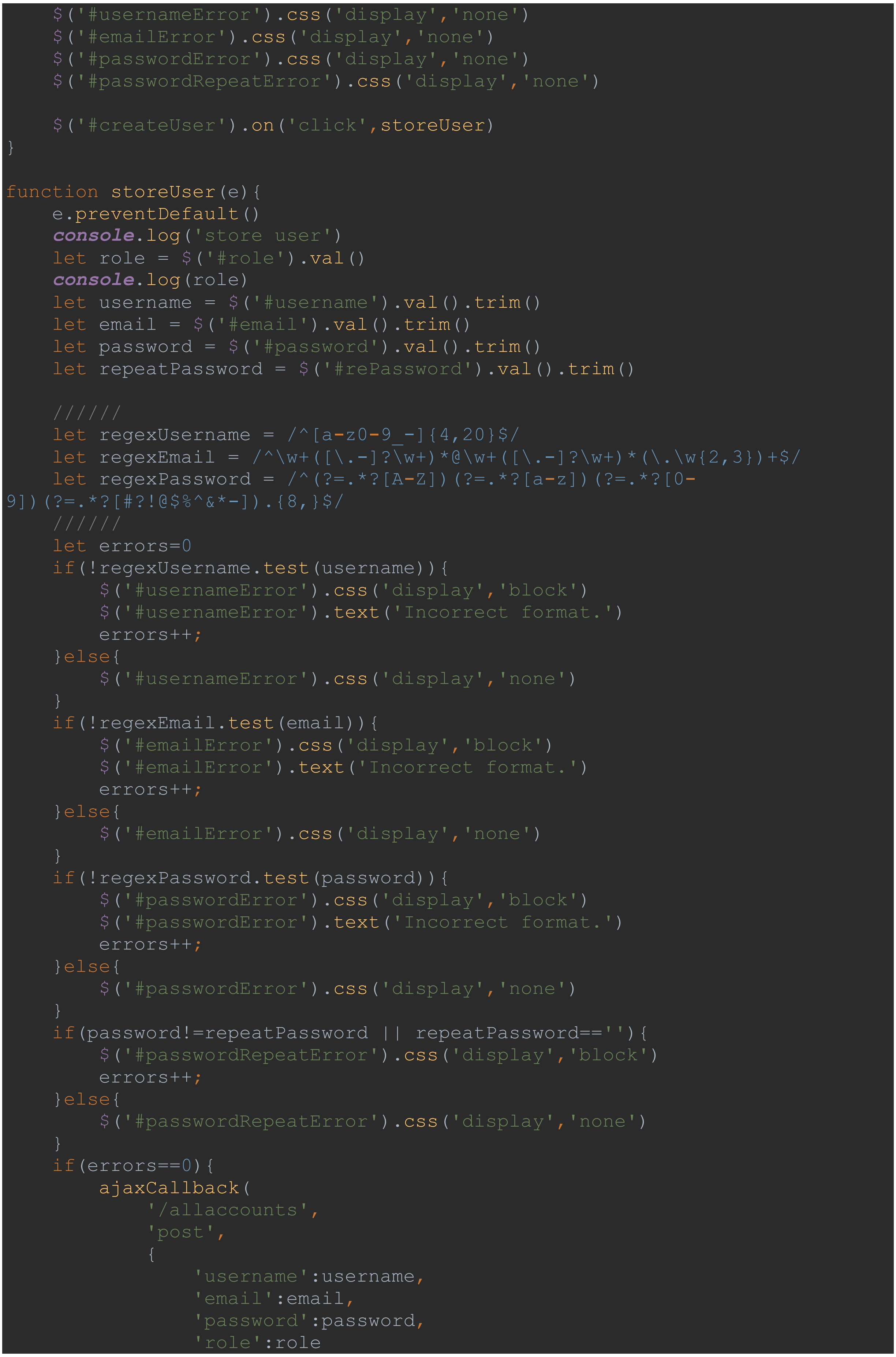


```
    }
    if(about==null || about==''){
        $('#aboutError').css('display','block')
        errors++;
    }else{
        $('#aboutError').css('display','none')
    }
    if(errors==0){
        console.log('aight')
    }
    ajaxCallback(
        '/alltables/'+id,
        'post',
        {
            'name':name,
            'about':about,
            '_method':'put'
        },
        function (data){
            $('#tableFeedback').text('Updated successfully.')
            $('#tableFeedback').addClass('alert-success')
        }
    )
}

///////////////////// admin users

function getAllUsers(e){
    e.preventDefault()

    ajaxCallback(
        "/allaccounts",
        'get',
        {},
        function(data){
            printAllUsers(data)
        }
    )
}

function printAllUsers(data){
    let html=`
        <div class="m-4">
            <h2>Users</h2>
        </div>
        <table class="table table-hover">
        <thead>
        <tr>
            <th scope="col">#</th>
            <th scope="col">Username</th>
            <th scope="col">Email</th>
            <th scope="col">Created At</th>
            <th scope="col">Role</th>
            <th scope="col">Update</th>
            <th scope="col">Delete</th>

        </tr>
        </thead>
        <tbody>`
    if(data.length==0){
        html+=`<td colspan="8">No users found.</td>`
```

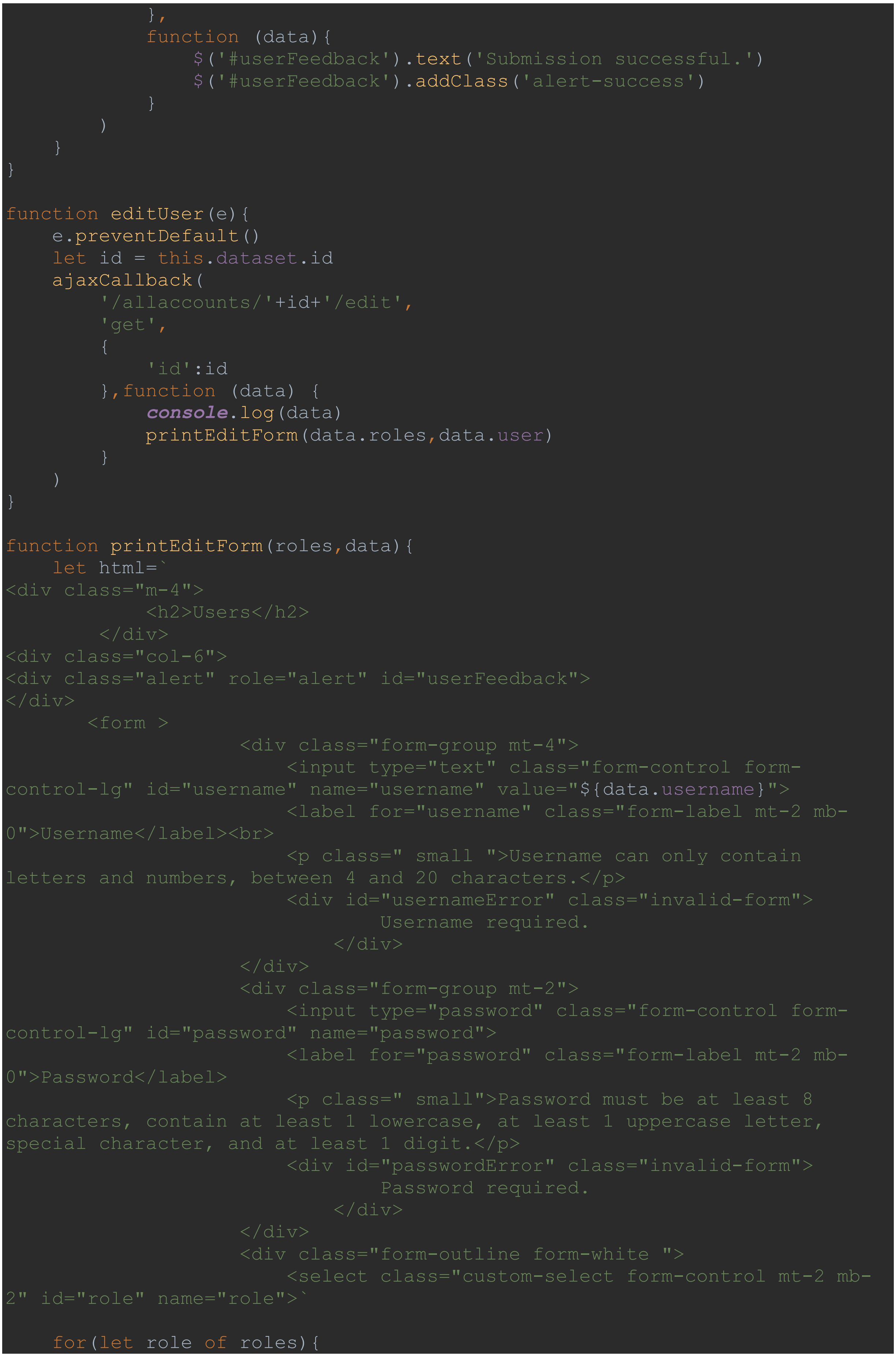


```
    }
    else{
        for(let d of data){
            html+=`<tr>
            <th scope="row">${d.id}</th>
            <td>${d.username}</td>
            <td>${d.email}</td>
            <td>${formatDate(d.created_at)}</td>`

            if(d.role_id==1){
                html+=`<td>admin</td>`
            }else{
                html+=`<td>user</td>`
            }

            html+=`
            <td class="text-center"><a href="#" class="editUser mr-2" data-
id="${d.id}" ><i class="fa-solid fa-pen-to-square adminLinks"></i></a></td>
            <td class="text-center"><a href="#" class="deleteUser mr-2"
data-id="${d.id}" ><i class="fa-solid fa-trash adminLinks"></i></a></td>
        </tr>`
        }
    }
    html+=`</tbody>`

    $('#adminContent').html(html)
    $('.editUser').on('click', editUser)
    $('.deleteUser').on('click', deleteUser)

}

function formatDate(date){
    let dt = new Date(date);
    let formated = dt.toLocaleDateString()
    return formated
}

function userCreate(e){
    e.preventDefault()
    ajaxCallback(
        '/allaccounts/create',
        'get',
        {},
        function (data){
            printUserForm(data.roles)
        }
    )
}

function printUserForm(roles){
    console.log(roles)
    let html=`
<div class="m-4">
            <h2>Users</h2>
        </div>
<div class="col-6">
<div class="alert" role="alert" id="userFeedback">
</div>
       <form >
                    <div class="form-group mt-4">
                        <input type="text" class="form-control form-
```

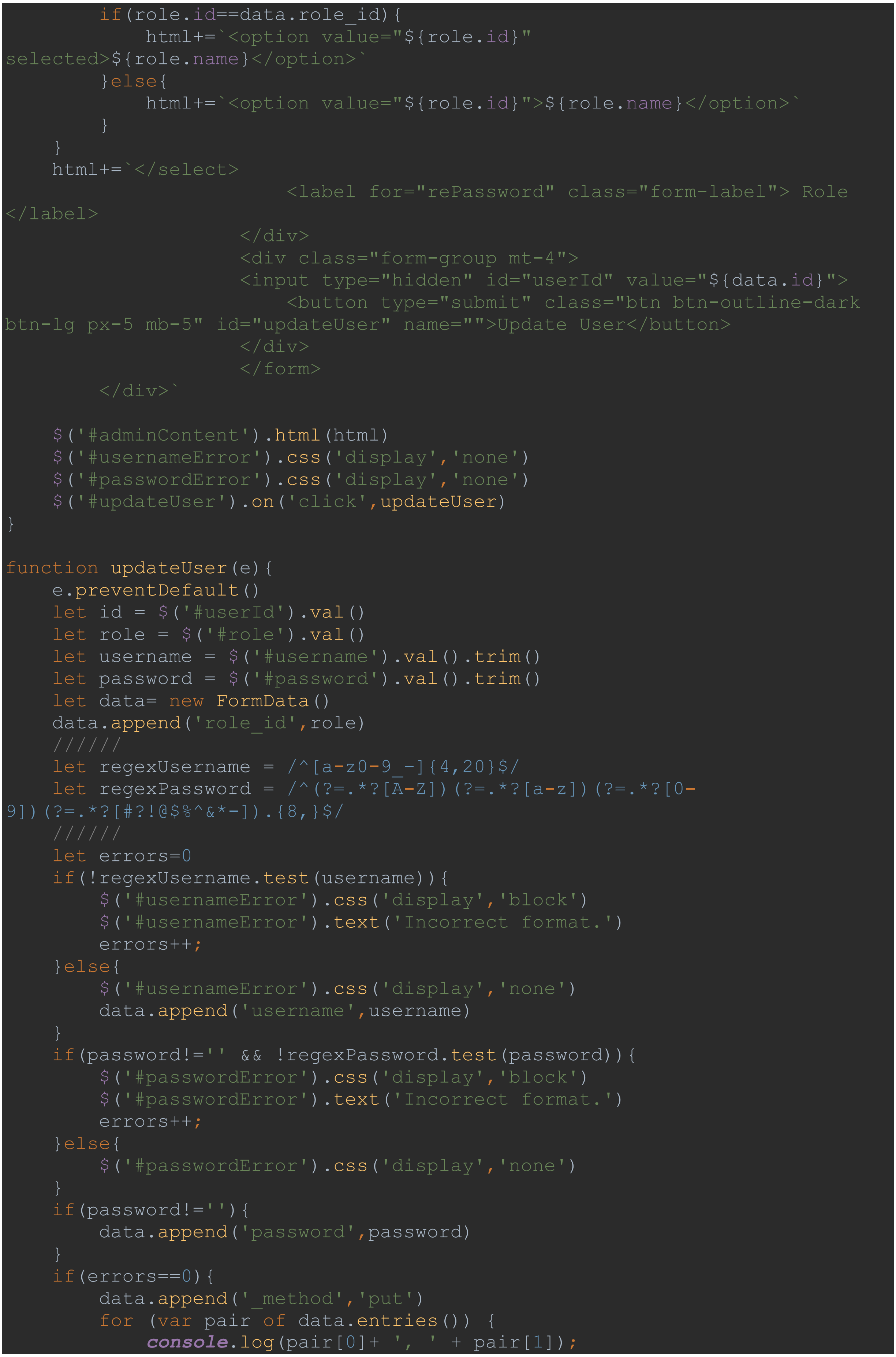
```
control-lg" id="username" name="username" value="">
                        <label for="username" class="form-label mt-2 mb-
0">Username</label><br>
                        <p class=" small ">Username can only contain
letters and numbers, between 4 and 20 characters.</p>
                        <div id="usernameError" class="invalid-form">
                                Username required.
                            </div>
                    </div>
                    <div class="form-group mt-2">
                        <input type="text" class="form-control form-
control-lg" id="email" name="email" value="">
                        <label for="email" class="form-label mt-
2">Email</label>
                        <div id="emailError" class="invalid-form">
                                Email required.
                            </div>
                    </div>
                    <div class="form-group mt-2">
                        <input type="password" class="form-control form-
control-lg" id="password" name="password">
                        <label for="password" class="form-label mt-2 mb-
0">Password</label>
                        <p class=" small">Password must be at least 8
characters, contain at least 1 lowercase, at least 1 uppercase letter,
special character, and at least 1 digit.</p>
                        <div id="passwordError" class="invalid-form">
                                Password required.
                            </div>
                    </div>
                    <div class="form-group mt-2">
                        <input type="password" class="form-control form-
control-lg" id="rePassword" name="password_confirmation">
                        <label for="rePassword" class="form-label mt-2">
Repeat Password</label>
                        <div id="passwordRepeatError" class="invalid-form">
                                Password doesn't match.
                            </div>
                    </div>
                    <div class="form-outline form-white ">
                        <select class="custom-select form-control mt-2 mb-
2" id="role" name="role">`

                        for(let role of roles){
                            html+=`
                                <option
value="${role.id}">${role.name}</option>
                            `
                        }
                        html+=`</select>
                        <label for="rePassword" class="form-label"> Role
</label>
                    </div>
                    <div class="form-group mt-4">
                        <button type="submit" class="btn btn-outline-dark
btn-lg px-5 mb-5" id="createUser" name="">Create User</button>
                    </div>
                    </form>
        </div>`

    $('#adminContent').html(html)
```



```
    $('#usernameError').css('display','none')
    $('#emailError').css('display','none')
    $('#passwordError').css('display','none')
    $('#passwordRepeatError').css('display','none')

    $('#createUser').on('click',storeUser)
}

function storeUser(e){
    e.preventDefault()
    console.log('store user')
    let role = $('#role').val()
    console.log(role)
    let username = $('#username').val().trim()
    let email = $('#email').val().trim()
    let password = $('#password').val().trim()
    let repeatPassword = $('#rePassword').val().trim()

    //////
    let regexUsername = /^[a-z0-9_-]{4,20}$/
    let regexEmail = /^\w+([\.-]?\w+)*@\w+([\.-]?\w+)*(\.\w{2,3})+$/
    let regexPassword = /^(?=.*?[A-Z])(?=.*?[a-z])(?=.*?[0-
9])(?=.*?[#?!@$%^&*-]).{8,}$/
    //////
    let errors=0
    if(!regexUsername.test(username)){
        $('#usernameError').css('display','block')
        $('#usernameError').text('Incorrect format.')
        errors++;
    }else{
        $('#usernameError').css('display','none')
    }
    if(!regexEmail.test(email)){
        $('#emailError').css('display','block')
        $('#emailError').text('Incorrect format.')
        errors++;
    }else{
        $('#emailError').css('display','none')
    }
    if(!regexPassword.test(password)){
        $('#passwordError').css('display','block')
        $('#passwordError').text('Incorrect format.')
        errors++;
    }else{
        $('#passwordError').css('display','none')
    }
    if(password!=repeatPassword || repeatPassword==''){
        $('#passwordRepeatError').css('display','block')
        errors++;
    }else{
        $('#passwordRepeatError').css('display','none')
    }
    if(errors==0){
        ajaxCallback(
            '/allaccounts',
            'post',
            {
                'username':username,
                'email':email,
                'password':password,
                'role':role
```



```
            },
            function (data){
                $('#userFeedback').text('Submission successful.')
                $('#userFeedback').addClass('alert-success')
            }
        )
    }
}

function editUser(e){
    e.preventDefault()
    let id = this.dataset.id
    ajaxCallback(
        '/allaccounts/'+id+'/edit',
        'get',
        {
            'id':id
        },function (data) {
            console.log(data)
            printEditForm(data.roles,data.user)
        }
    )
}

function printEditForm(roles,data){
    let html=`
<div class="m-4">
            <h2>Users</h2>
        </div>
<div class="col-6">
<div class="alert" role="alert" id="userFeedback">
</div>
       <form >
                    <div class="form-group mt-4">
                        <input type="text" class="form-control form-
control-lg" id="username" name="username" value="${data.username}">
                        <label for="username" class="form-label mt-2 mb-
0">Username</label><br>
                        <p class=" small ">Username can only contain
letters and numbers, between 4 and 20 characters.</p>
                        <div id="usernameError" class="invalid-form">
                                Username required.
                            </div>
                    </div>
                    <div class="form-group mt-2">
                        <input type="password" class="form-control form-
control-lg" id="password" name="password">
                        <label for="password" class="form-label mt-2 mb-
0">Password</label>
                        <p class=" small">Password must be at least 8
characters, contain at least 1 lowercase, at least 1 uppercase letter,
special character, and at least 1 digit.</p>
                        <div id="passwordError" class="invalid-form">
                                Password required.
                            </div>
                    </div>
                    <div class="form-outline form-white ">
                        <select class="custom-select form-control mt-2 mb-
2" id="role" name="role">`

    for(let role of roles){
```

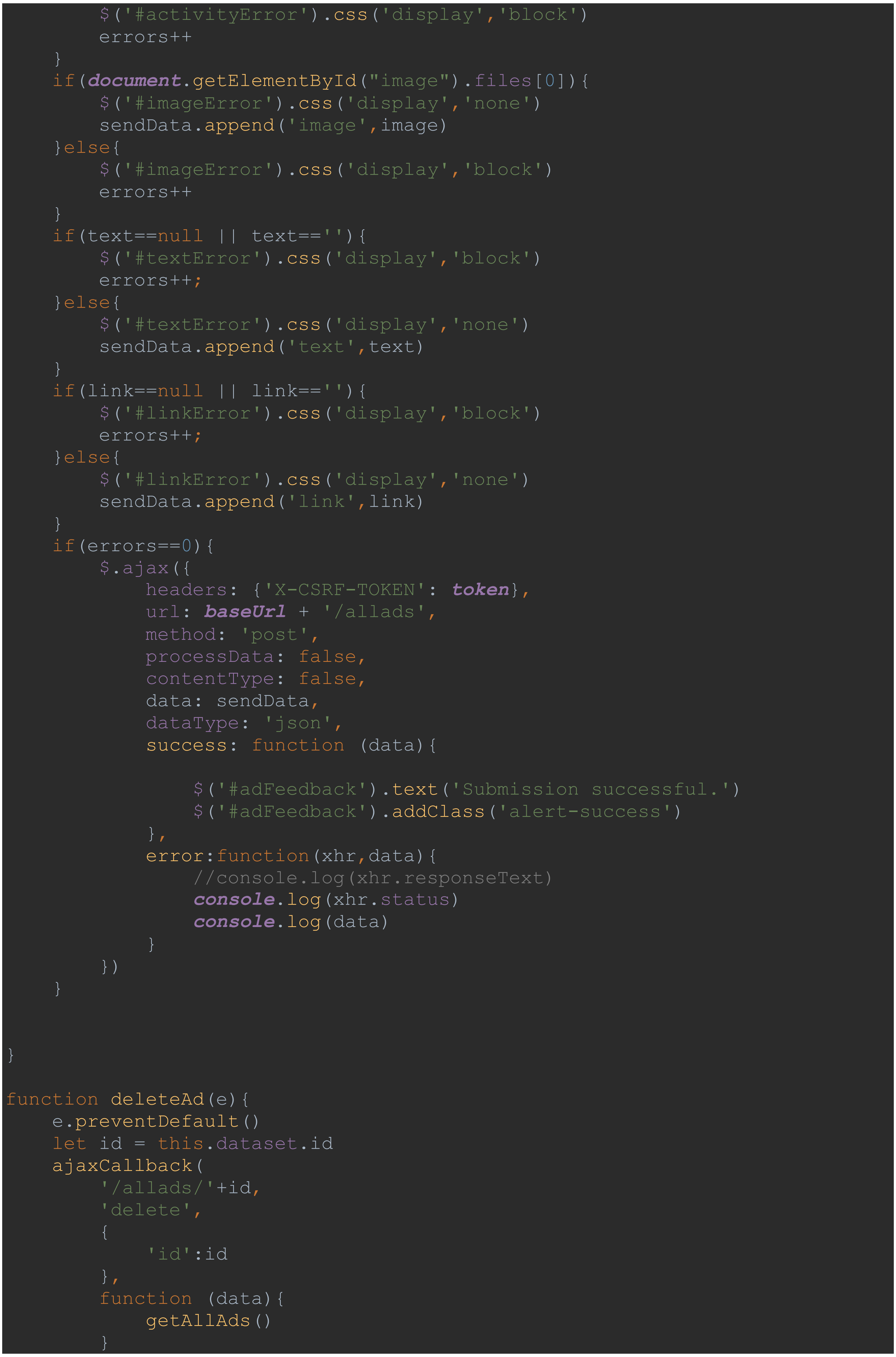


```
        if(role.id==data.role_id){
            html+=`<option value="${role.id}"
selected>${role.name}</option>`
        }else{
            html+=`<option value="${role.id}">${role.name}</option>`
        }
    }
    html+=`</select>
                        <label for="rePassword" class="form-label"> Role
</label>
                    </div>
                    <div class="form-group mt-4">
                    <input type="hidden" id="userId" value="${data.id}">
                        <button type="submit" class="btn btn-outline-dark
btn-lg px-5 mb-5" id="updateUser" name="">Update User</button>
                    </div>
                    </form>
        </div>`

    $('#adminContent').html(html)
    $('#usernameError').css('display','none')
    $('#passwordError').css('display','none')
    $('#updateUser').on('click',updateUser)
}

function updateUser(e){
    e.preventDefault()
    let id = $('#userId').val()
    let role = $('#role').val()
    let username = $('#username').val().trim()
    let password = $('#password').val().trim()
    let data= new FormData()
    data.append('role_id',role)
    //////
    let regexUsername = /^[a-z0-9_-]{4,20}$/
    let regexPassword = /^(?=.*?[A-Z])(?=.*?[a-z])(?=.*?[0-
9])(?=.*?[#?!@$%^&*-]).{8,}$/
    //////
    let errors=0
    if(!regexUsername.test(username)){
        $('#usernameError').css('display','block')
        $('#usernameError').text('Incorrect format.')
        errors++;
    }else{
        $('#usernameError').css('display','none')
        data.append('username',username)
    }
    if(password!='' && !regexPassword.test(password)){
        $('#passwordError').css('display','block')
        $('#passwordError').text('Incorrect format.')
        errors++;
    }else{
        $('#passwordError').css('display','none')
    }
    if(password!=''){
        data.append('password',password)
    }
    if(errors==0){
        data.append('_method','put')
        for (var pair of data.entries()) {
            console.log(pair[0]+ ', ' + pair[1]);
        }
        $.ajax({
            headers: {'X-CSRF-TOKEN': token},
            url: baseUrl + '/allaccounts/'+id,
            method: 'post',
            processData: false,
            contentType: false,
            data: data,
            dataType: 'json',
            success: function (data){
                $('#userFeedback').text('Update successful.')
                $('#userFeedback').addClass('alert-success')
            },error:function(xhr,data){
                //console.log(xhr.responseText)
                console.log(xhr.status)
                console.log(data)
            }
        })
    }
}

function deleteUser(e){
    e.preventDefault();
    let id = this.dataset.id
    ajaxCallback(
        '/allaccounts/'+id,
        'delete',
        {
            'id':id
        },
        function (data){
            getAllUsers(e)
        }
    )
}
///////////////////// admin ads

function showAllAds(e){
    e.preventDefault()
    getAllAds()
}

function getAllAds(){

    ajaxCallback(
        "/allads",
        'get',
        {},
        function(data){
            printAds(data)
        }
    )
}

function printAds(data){
    let html=`<div class="m-4">
            <h2>Ads</h2>
        </div>
        <table class="table table-hover">
        <thead>
        <tr>
```

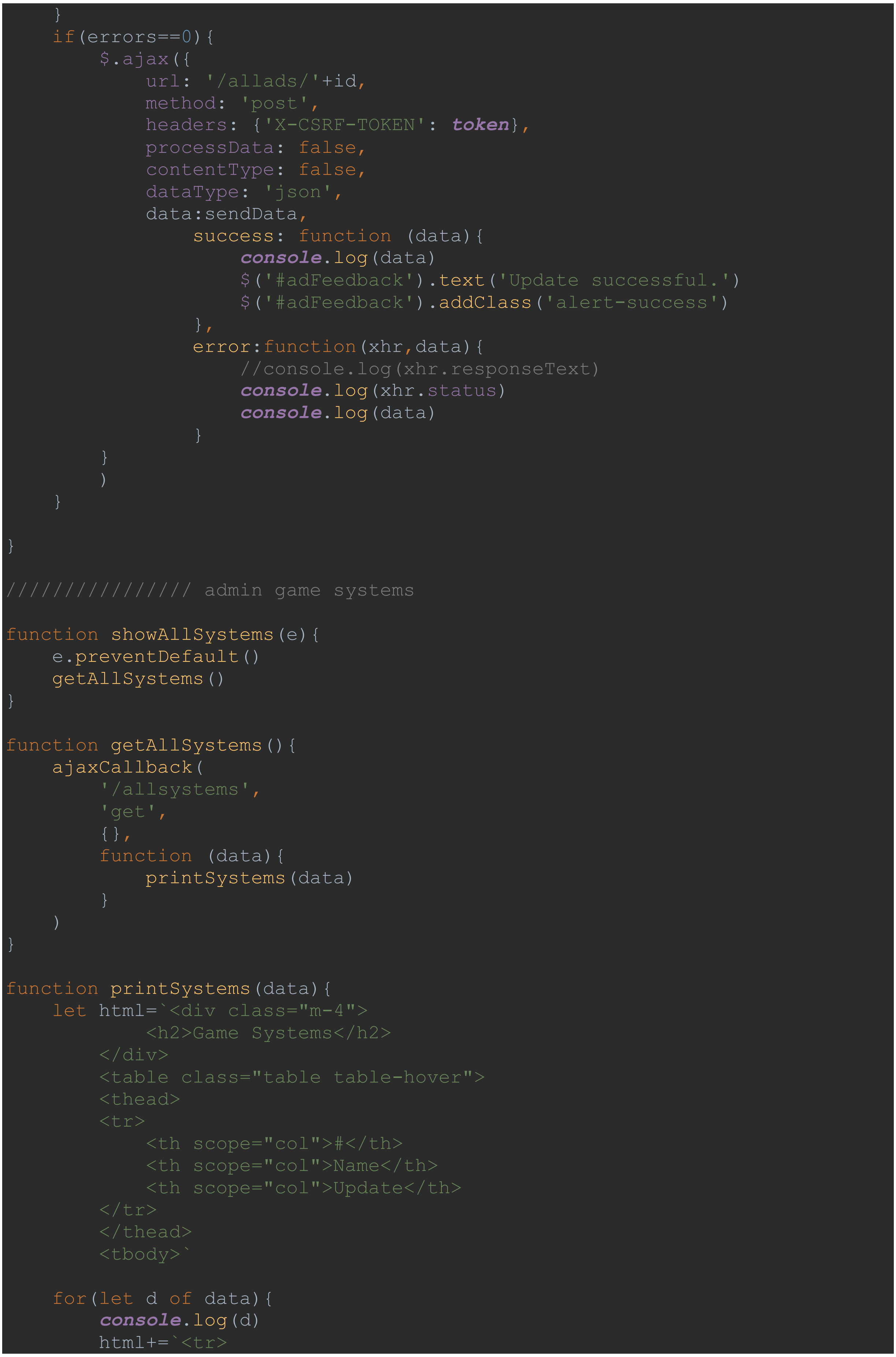
```
            <th scope="col">#</th>
            <th scope="col">Link</th>
            <th scope="col">Text</th>
            <th scope="col">Image</th>
            <th scope="col">Active</th>
            <th scope="col">Update</th>
            <th scope="col">Delete</th>

        </tr>
        </thead>
        <tbody>`

        for(let d of data){
            console.log(d)
            html+=`<tr>
            <th scope="row">${d.id}</th>
            <td>${d.link}</td>
            <td>${d.text}</td>
            <td><img src="${baseUrl+'/assets/img/'+d.image}"
class="carouselImages adminTableImg"></td>`
            if(d.active){
                html+=`<td>Yes</td>`
            }else{
                html+=`<td>No</td>`
            }

           html+= `<td class="text-center"><a href="#" class="editAd mr-2"
data-id="${d.id}" ><i class="fa-solid fa-pen-to-square
adminLinks"></i></a></td>
            <td class="text-center"><a href="#" class="deleteAd mr-2" data-
id="${d.id}" ><i class="fa-solid fa-trash adminLinks"></i></a></td>
        </tr>`
        }

    html+=`</tbody>`

    $('#adminContent').html(html)
    $('.deleteAd').click(deleteAd)
    $('.editAd').click(editAd)

}

function printAdForm(e, data=null){
    console.log(data)
    let html=`
<div class="m-4">
            <h2>Ads</h2>
        </div>
<div class="col-6">
<div class="alert" role="alert" id="adFeedback">
                                </div>
                    <form enctype="multipart/form-data">
                        <div class="form-group mt-2">
                            <input type="text" class="form-control form-
control-lg" id="text"`

        if(data!=null){
            html+=`value="${data.text}"`
        }

        html+= ` name="text">
```

```
                            <label for="email" class="form-label mt-
2">Text</label><br>
                            <div id="textError" class="invalid-form">
                                Text required.
                            </div>
                        </div>
                        <div class="form-group mt-2">
                            <input type="text" class="form-control form-
control-lg" id="link" `
        if(data!=null){
            html+=`value="${data.link}"`
        }

            html+= `name="link">
                            <label for="link" class="form-label mt-
2">Link</label><br>
                            <div id="linkError" class="invalid-form">
                                Link required.
                            </div>
                        </div>
                        <div>
                            <input type="file" class="form-control mt-2 pb-
2" id="image" name="userImg">
                            <label for="image" class="form-label mt-
2">Image</label><br>
                            <div id="imageError" class="invalid-form">
                                Image required.
                            </div>
                        </div>

                        <div>
                            Active:<br>`
                if(data!=null && data.active == 1){
                    html+=`
                    <input class="form-check-input" type="radio"
name="activity" value="1" checked>
                                <label class="form-check-label"
for="inlineRadio2">Yes</label><br>
                                <input class="form-check-input"
type="radio" name="activity" value="0">
                                <label class="form-check-label"
for="inlineRadio1">No</label><br>
                    `
                }
                else if(data!=null && data.active == 0){
                    html+=`
                    <input class="form-check-input" type="radio"
name="activity" value="1" >
                                <label class="form-check-label"
for="inlineRadio2">Yes</label><br>
                                <input class="form-check-input"
type="radio" name="activity" value="0" checked>
                                <label class="form-check-label"
for="inlineRadio1">No</label><br>
                    `
                }
                else{
                    html+=`
                    <input class="form-check-input" type="radio"
name="activity" value="1" >
                                <label class="form-check-label"
```

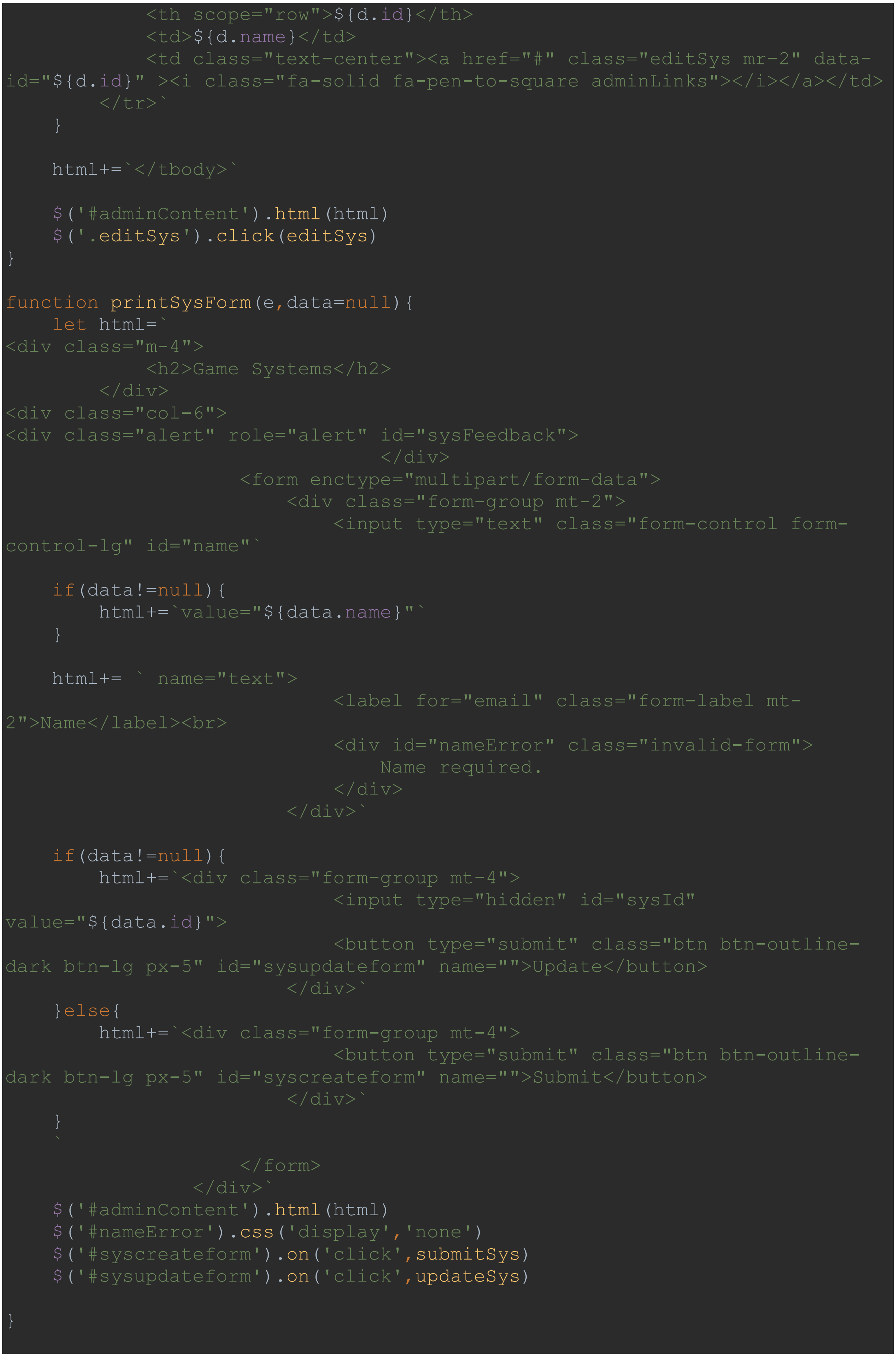
```
for="inlineRadio2">Yes</label><br>
                                <input class="form-check-input"
type="radio" name="activity" value="0">
                                <label class="form-check-label"
for="inlineRadio1">No</label><br>
                    `
                }

                    html+=`
                                <div id="activityError" class="invalid-
form">
                                    Activity required.
                                </div>
                        </div>`

                        if(data!=null){
                            html+=`<div class="form-group mt-4">
                            <input type="hidden" id="adId"
value="${data.id}">
                            <button type="submit" class="btn btn-outline-
dark btn-lg px-5" id="adupdateform" name="">Update</button>
                        </div>`
                        }else{
                            html+=`<div class="form-group mt-4">
                            <button type="submit" class="btn btn-outline-
dark btn-lg px-5" id="adcreateform" name="">Submit</button>
                        </div>`
                        }
                        `
                    </form>
                </div>`
    $('#adminContent').html(html)
    $('#textError').css('display','none')
    $('#linkError').css('display','none')
    $('#imageError').css('display','none')
    $('#activityError').css('display','none')

    $('#adcreateform').on('click',submitAd)
    $('#adupdateform').on('click',updateAd)

}

function submitAd(e){
    e.preventDefault()
    console.log('create ad')
    let image = document.getElementById("image").files[0]
    console.log(image)
    let text = $('#text').val().trim()
    let link = $('#link').val().trim()
    console.log(text)
    console.log(link)
    let activeChb=``
    let sendData = new FormData()
    let errors = 0
    if(document.querySelector("input[name=activity]:checked")){
        let activeChb =
document.querySelector("input[name=activity]:checked").value
        $('#activityError').css('display','none')
        console.log(activeChb)
        sendData.append('activity',activeChb)
    }else{
```



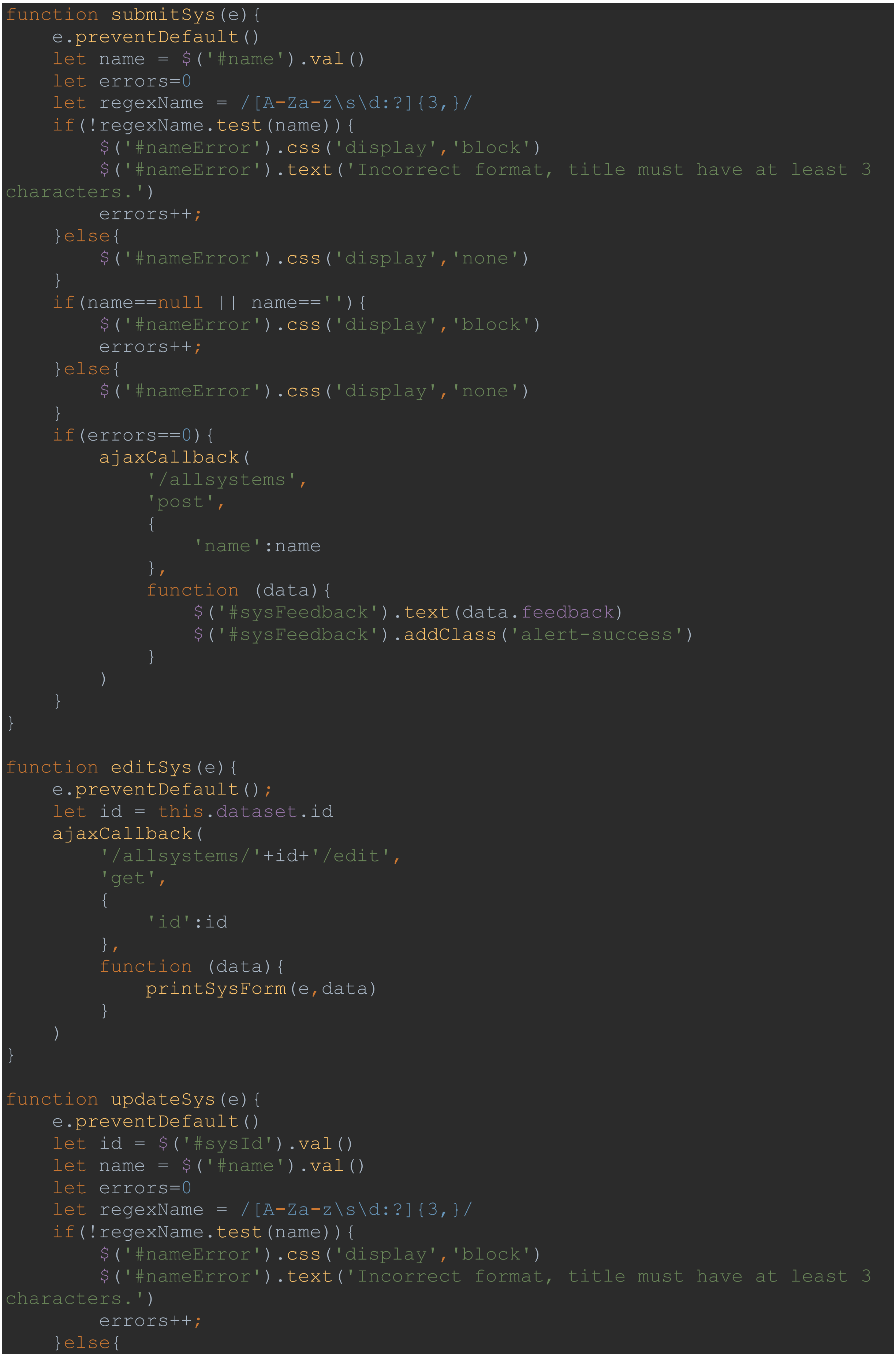
```
        $('#activityError').css('display','block')
        errors++
    }
    if(document.getElementById("image").files[0]){
        $('#imageError').css('display','none')
        sendData.append('image',image)
    }else{
        $('#imageError').css('display','block')
        errors++
    }
    if(text==null || text==''){
        $('#textError').css('display','block')
        errors++;
    }else{
        $('#textError').css('display','none')
        sendData.append('text',text)
    }
    if(link==null || link==''){
        $('#linkError').css('display','block')
        errors++;
    }else{
        $('#linkError').css('display','none')
        sendData.append('link',link)
    }
    if(errors==0){
        $.ajax({
            headers: {'X-CSRF-TOKEN': token},
            url: baseUrl + '/allads',
            method: 'post',
            processData: false,
            contentType: false,
            data: sendData,
            dataType: 'json',
            success: function (data){

                $('#adFeedback').text('Submission successful.')
                $('#adFeedback').addClass('alert-success')
            },
            error:function(xhr,data){
                //console.log(xhr.responseText)
                console.log(xhr.status)
                console.log(data)
            }
        })
    }


}

function deleteAd(e){
    e.preventDefault()
    let id = this.dataset.id
    ajaxCallback(
        '/allads/'+id,
        'delete',
        {
            'id':id
        },
        function (data){
            getAllAds()
        }
    )
}

function editAd(e){
    e.preventDefault()
    let id = this.dataset.id
    ajaxCallback(
        '/allads/'+id+'/edit',
        'get',
        {
            'id':id
        },
        function (data){
            printAdForm(e,data)
            console.log(data)
        }
    )
}

function updateAd(e){
    e.preventDefault()
    console.log('update')
    let image = document.getElementById("image").files[0]
    let text = $('#text').val().trim()
    let link = $('#link').val().trim()
    let id = $('#adId').val()
    let sendData = new FormData()
    let errors = 0
    if(document.querySelector("input[name=activity]:checked")){
        let activeChb =
document.querySelector("input[name=activity]:checked").value
        $('#activityError').css('display','none')
        console.log(activeChb)
        sendData.append('activity',activeChb)
    }else{
        $('#activityError').css('display','block')
        errors++
    }
    if(document.getElementById("image").files[0]){
        $('#imageError').css('display','none')
        sendData.append('image',image)
    }
    if(text==null || text==''){
        $('#textError').css('display','block')
        errors++;
    }else{
        $('#textError').css('display','none')
        sendData.append('text',text)
    }
    if(link==null || link==''){
        $('#linkError').css('display','block')
        errors++;
    }else{
        $('#linkError').css('display','none')
        sendData.append('link',link)
    }

    sendData.append('_method','put')
    console.log(sendData.values())
    for (var pair of sendData.entries()) {
        console.log(pair[0]+ ' = ' + pair[1]);
```



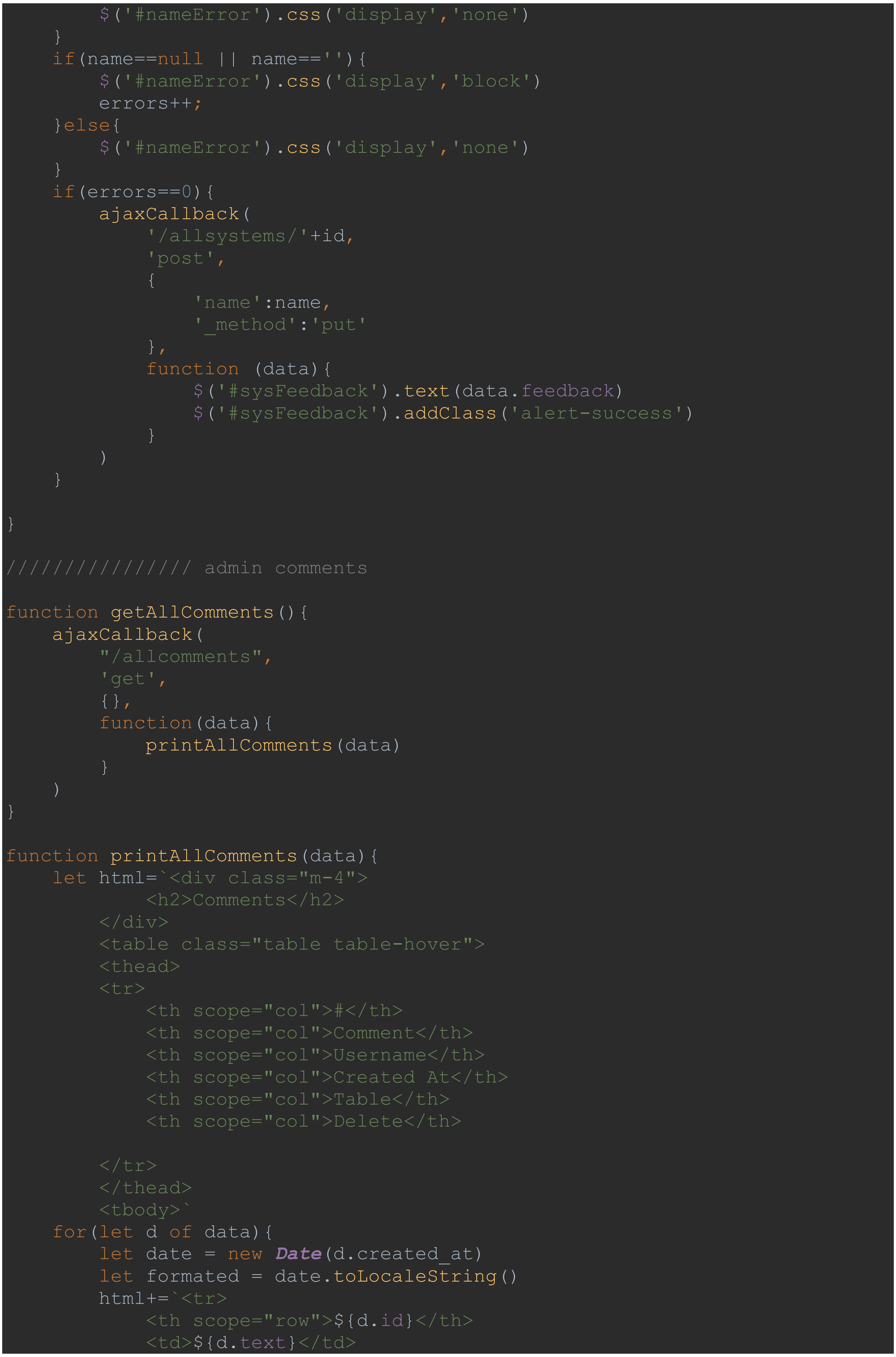
```
    }
    if(errors==0){
        $.ajax({
            url: '/allads/'+id,
            method: 'post',
            headers: {'X-CSRF-TOKEN': token},
            processData: false,
            contentType: false,
            dataType: 'json',
            data:sendData,
                success: function (data){
                    console.log(data)
                    $('#adFeedback').text('Update successful.')
                    $('#adFeedback').addClass('alert-success')
                },
                error:function(xhr,data){
                    //console.log(xhr.responseText)
                    console.log(xhr.status)
                    console.log(data)
                }
        }
        )
    }

}

//////////////// admin game systems

function showAllSystems(e){
    e.preventDefault()
    getAllSystems()
}

function getAllSystems(){
    ajaxCallback(
        '/allsystems',
        'get',
        {},
        function (data){
            printSystems(data)
        }
    )
}

function printSystems(data){
    let html=`<div class="m-4">
            <h2>Game Systems</h2>
        </div>
        <table class="table table-hover">
        <thead>
        <tr>
            <th scope="col">#</th>
            <th scope="col">Name</th>
            <th scope="col">Update</th>
        </tr>
        </thead>
        <tbody>`

    for(let d of data){
        console.log(d)
        html+=`<tr>
```



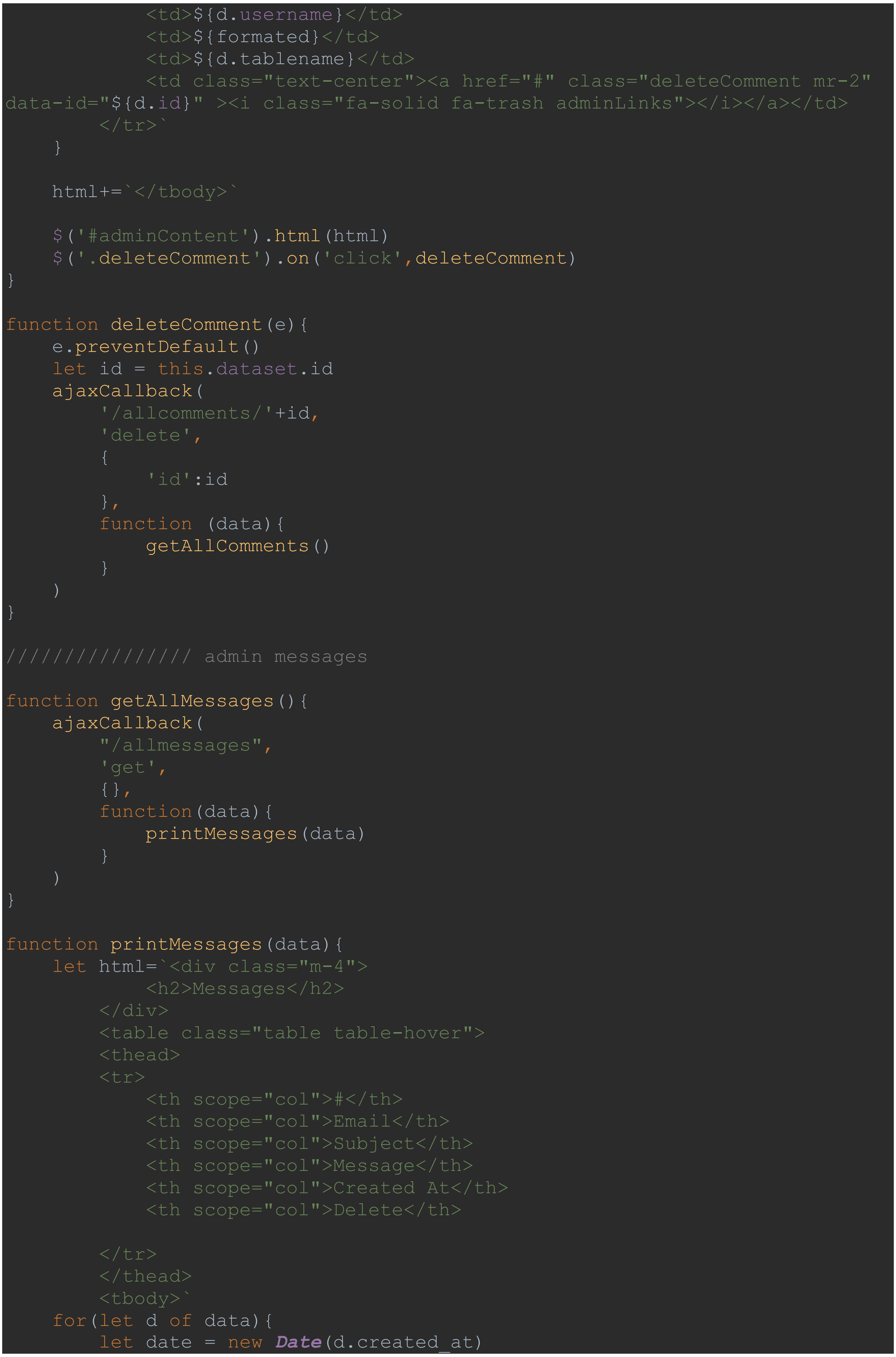
```
            <th scope="row">${d.id}</th>
            <td>${d.name}</td>
            <td class="text-center"><a href="#" class="editSys mr-2" data-
id="${d.id}" ><i class="fa-solid fa-pen-to-square adminLinks"></i></a></td>
        </tr>`
    }

    html+=`</tbody>`

    $('#adminContent').html(html)
    $('.editSys').click(editSys)
}

function printSysForm(e,data=null){
    let html=`
<div class="m-4">
            <h2>Game Systems</h2>
        </div>
<div class="col-6">
<div class="alert" role="alert" id="sysFeedback">
                                </div>
                    <form enctype="multipart/form-data">
                        <div class="form-group mt-2">
                            <input type="text" class="form-control form-
control-lg" id="name"`

    if(data!=null){
        html+=`value="${data.name}"`
    }

    html+= ` name="text">
                            <label for="email" class="form-label mt-
2">Name</label><br>
                            <div id="nameError" class="invalid-form">
                                Name required.
                            </div>
                        </div>`

    if(data!=null){
        html+=`<div class="form-group mt-4">
                            <input type="hidden" id="sysId"
value="${data.id}">
                            <button type="submit" class="btn btn-outline-
dark btn-lg px-5" id="sysupdateform" name="">Update</button>
                        </div>`
    }else{
        html+=`<div class="form-group mt-4">
                            <button type="submit" class="btn btn-outline-
dark btn-lg px-5" id="syscreateform" name="">Submit</button>
                        </div>`
    }
    `
                    </form>
                </div>`
    $('#adminContent').html(html)
    $('#nameError').css('display','none')
    $('#syscreateform').on('click',submitSys)
    $('#sysupdateform').on('click',updateSys)

}
```



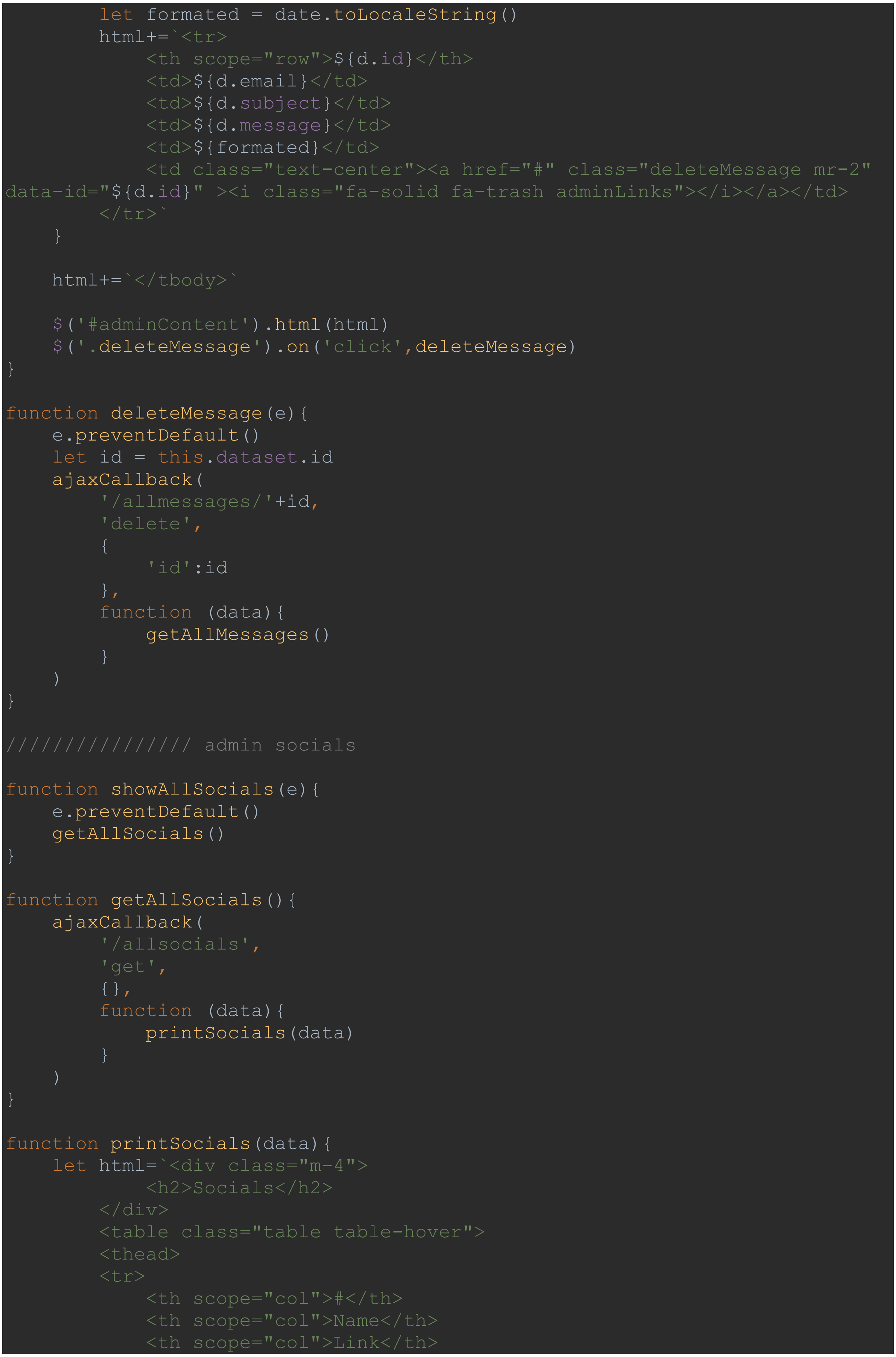
```
function submitSys(e){
    e.preventDefault()
    let name = $('#name').val()
    let errors=0
    let regexName = /[A-Za-z\s\d:?]{3,}/
    if(!regexName.test(name)){
        $('#nameError').css('display','block')
        $('#nameError').text('Incorrect format, title must have at least 3 
characters.')
        errors++;
    }else{
        $('#nameError').css('display','none')
    }
    if(name==null || name==''){
        $('#nameError').css('display','block')
        errors++;
    }else{
        $('#nameError').css('display','none')
    }
    if(errors==0){
        ajaxCallback(
            '/allsystems',
            'post',
            {
                'name':name
            },
            function (data){
                $('#sysFeedback').text(data.feedback)
                $('#sysFeedback').addClass('alert-success')
            }
        )
    }
}

function editSys(e){
    e.preventDefault();
    let id = this.dataset.id
    ajaxCallback(
        '/allsystems/'+id+'/edit',
        'get',
        {
            'id':id
        },
        function (data){
            printSysForm(e,data)
        }
    )
}

function updateSys(e){
    e.preventDefault()
    let id = $('#sysId').val()
    let name = $('#name').val()
    let errors=0
    let regexName = /[A-Za-z\s\d:?]{3,}/
    if(!regexName.test(name)){
        $('#nameError').css('display','block')
        $('#nameError').text('Incorrect format, title must have at least 3 
characters.')
        errors++;
    }else{
```



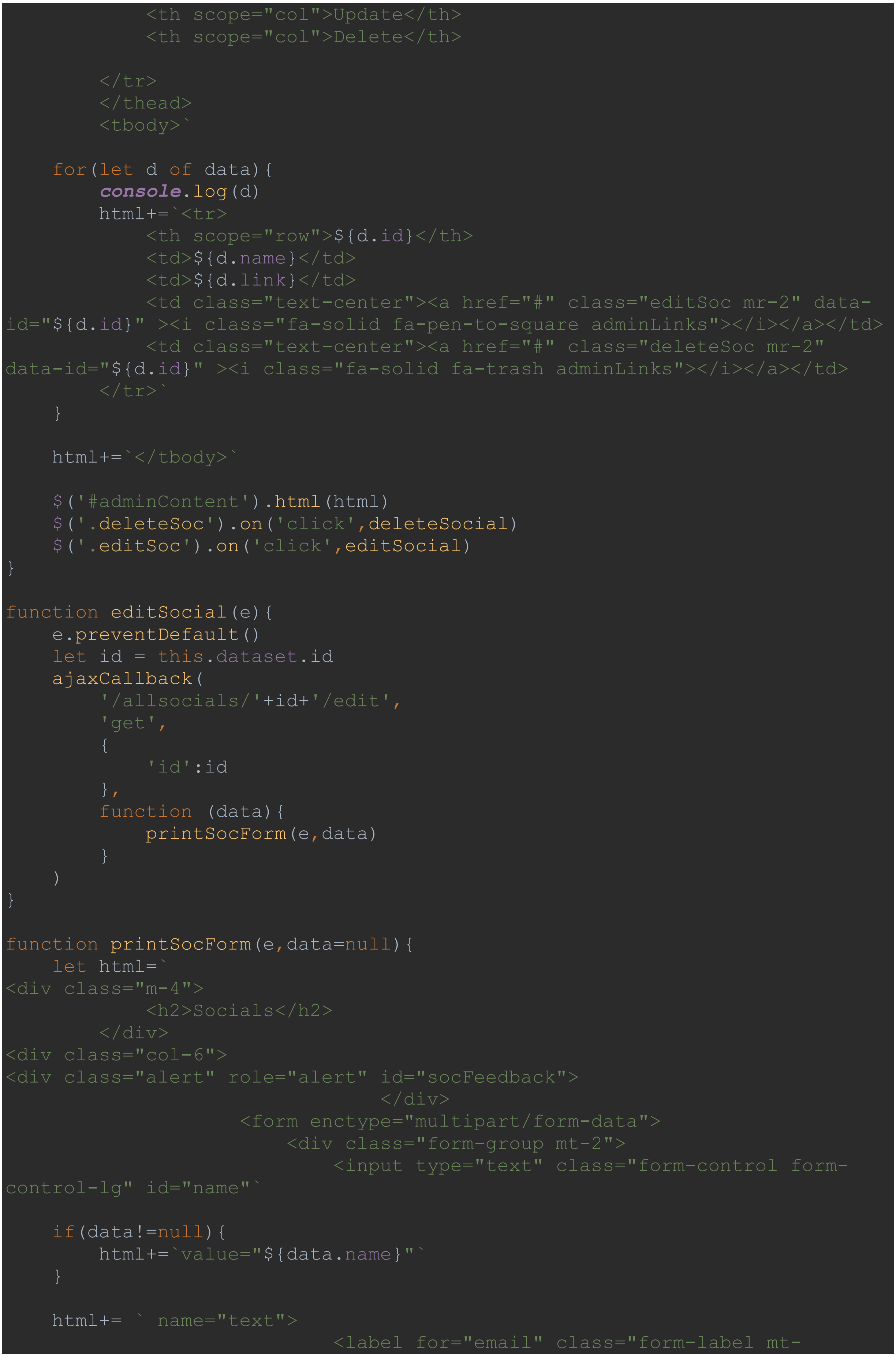
```
        $('#nameError').css('display','none')
    }
    if(name==null || name==''){
        $('#nameError').css('display','block')
        errors++;
    }else{
        $('#nameError').css('display','none')
    }
    if(errors==0){
        ajaxCallback(
            '/allsystems/'+id,
            'post',
            {
                'name':name,
                '_method':'put'
            },
            function  (data){
                $('#sysFeedback').text(data.feedback)
                $('#sysFeedback').addClass('alert-success')
            }
        )
    }

}

//////////////// admin comments

function getAllComments(){
    ajaxCallback(
        "/allcomments",
        'get',
        {},
        function(data){
            printAllComments(data)
        }
    )
}

function printAllComments(data){
    let html=`<div class="m-4">
            <h2>Comments</h2>
        </div>
        <table class="table table-hover">
        <thead>
        <tr>
            <th scope="col">#</th>
            <th scope="col">Comment</th>
            <th scope="col">Username</th>
            <th scope="col">Created At</th>
            <th scope="col">Table</th>
            <th scope="col">Delete</th>

        </tr>
        </thead>
        <tbody>`
    for(let d of data){
        let date = new Date(d.created_at)
        let formated = date.toLocaleString()
        html+=`<tr>
            <th scope="row">${d.id}</th>
            <td>${d.text}</td>
```



```
            <td>${d.username}</td>
            <td>${formated}</td>
            <td>${d.tablename}</td>
            <td class="text-center"><a href="#" class="deleteComment mr-2"
data-id="${d.id}" ><i class="fa-solid fa-trash adminLinks"></i></a></td>
        </tr>`
    }

    html+=`</tbody>`

    $('#adminContent').html(html)
    $('.deleteComment').on('click',deleteComment)
}

function deleteComment(e){
    e.preventDefault()
    let id = this.dataset.id
    ajaxCallback(
        '/allcomments/'+id,
        'delete',
        {
            'id':id
        },
        function  (data){
            getAllComments()
        }
    )
}

//////////////// admin messages

function getAllMessages(){
    ajaxCallback(
        "/allmessages",
        'get',
        {},
        function(data){
            printMessages(data)
        }
    )
}

function printMessages(data){
    let html=`<div class="m-4">
            <h2>Messages</h2>
        </div>
        <table class="table table-hover">
        <thead>
        <tr>
            <th scope="col">#</th>
            <th scope="col">Email</th>
            <th scope="col">Subject</th>
            <th scope="col">Message</th>
            <th scope="col">Created At</th>
            <th scope="col">Delete</th>

        </tr>
        </thead>
        <tbody>`
    for(let d of data){
        let date = new Date(d.created_at)
```

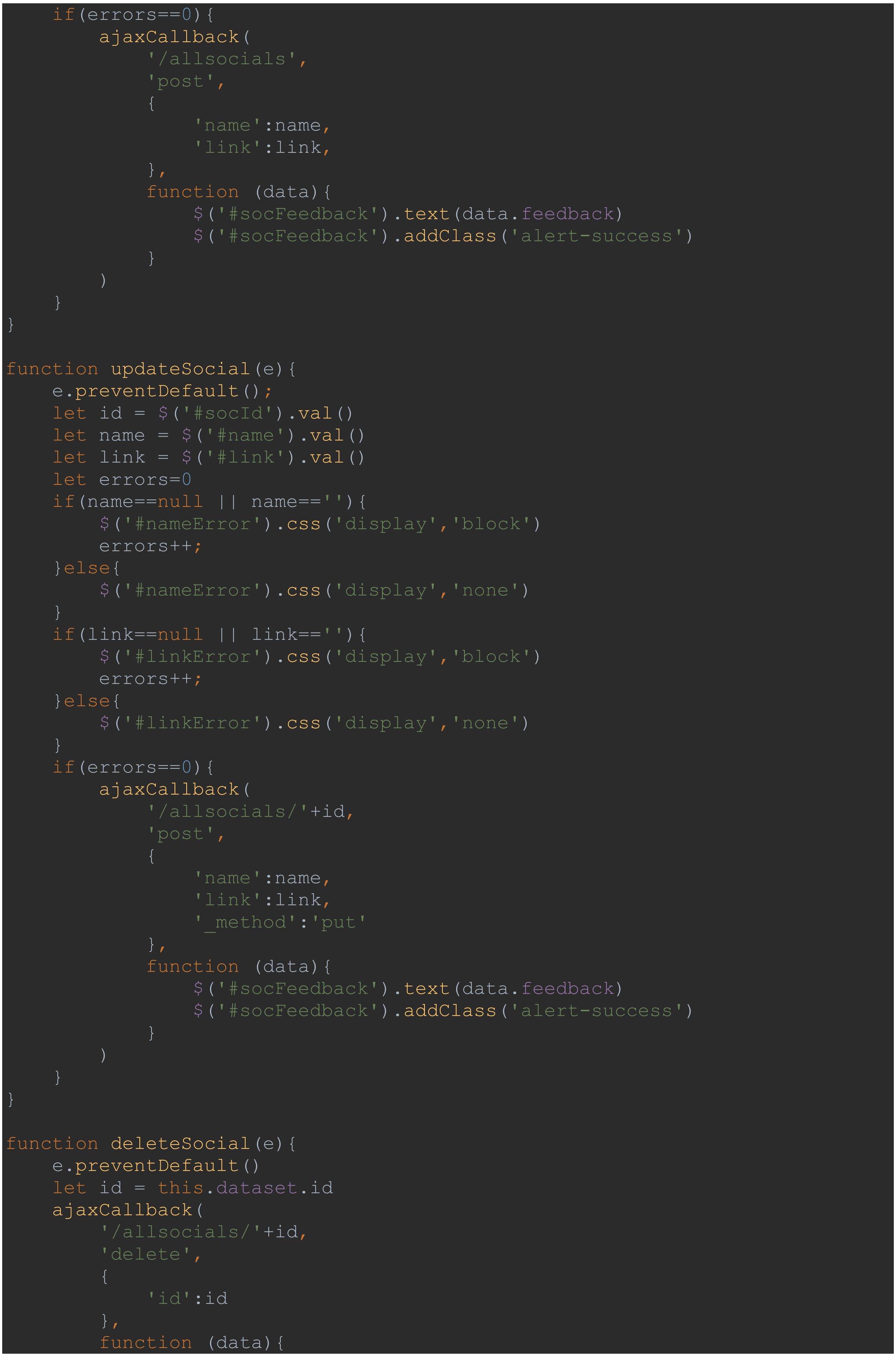


```
        let formated = date.toLocaleString()
        html+=`<tr>
            <th scope="row">${d.id}</th>
            <td>${d.email}</td>
            <td>${d.subject}</td>
            <td>${d.message}</td>
            <td>${formated}</td>
            <td class="text-center"><a href="#" class="deleteMessage mr-2"
data-id="${d.id}" ><i class="fa-solid fa-trash adminLinks"></i></a></td>
        </tr>`
    }

    html+=`</tbody>`

    $('#adminContent').html(html)
    $('.deleteMessage').on('click',deleteMessage)
}

function deleteMessage(e){
    e.preventDefault()
    let id = this.dataset.id
    ajaxCallback(
        '/allmessages/'+id,
        'delete',
        {
            'id':id
        },
        function (data){
            getAllMessages()
        }
    )
}

//////////////// admin socials

function showAllSocials(e){
    e.preventDefault()
    getAllSocials()
}

function getAllSocials(){
    ajaxCallback(
        '/allsocials',
        'get',
        {},
        function (data){
            printSocials(data)
        }
    )
}

function printSocials(data){
    let html=`<div class="m-4">
            <h2>Socials</h2>
        </div>
        <table class="table table-hover">
        <thead>
        <tr>
            <th scope="col">#</th>
            <th scope="col">Name</th>
            <th scope="col">Link</th>
```

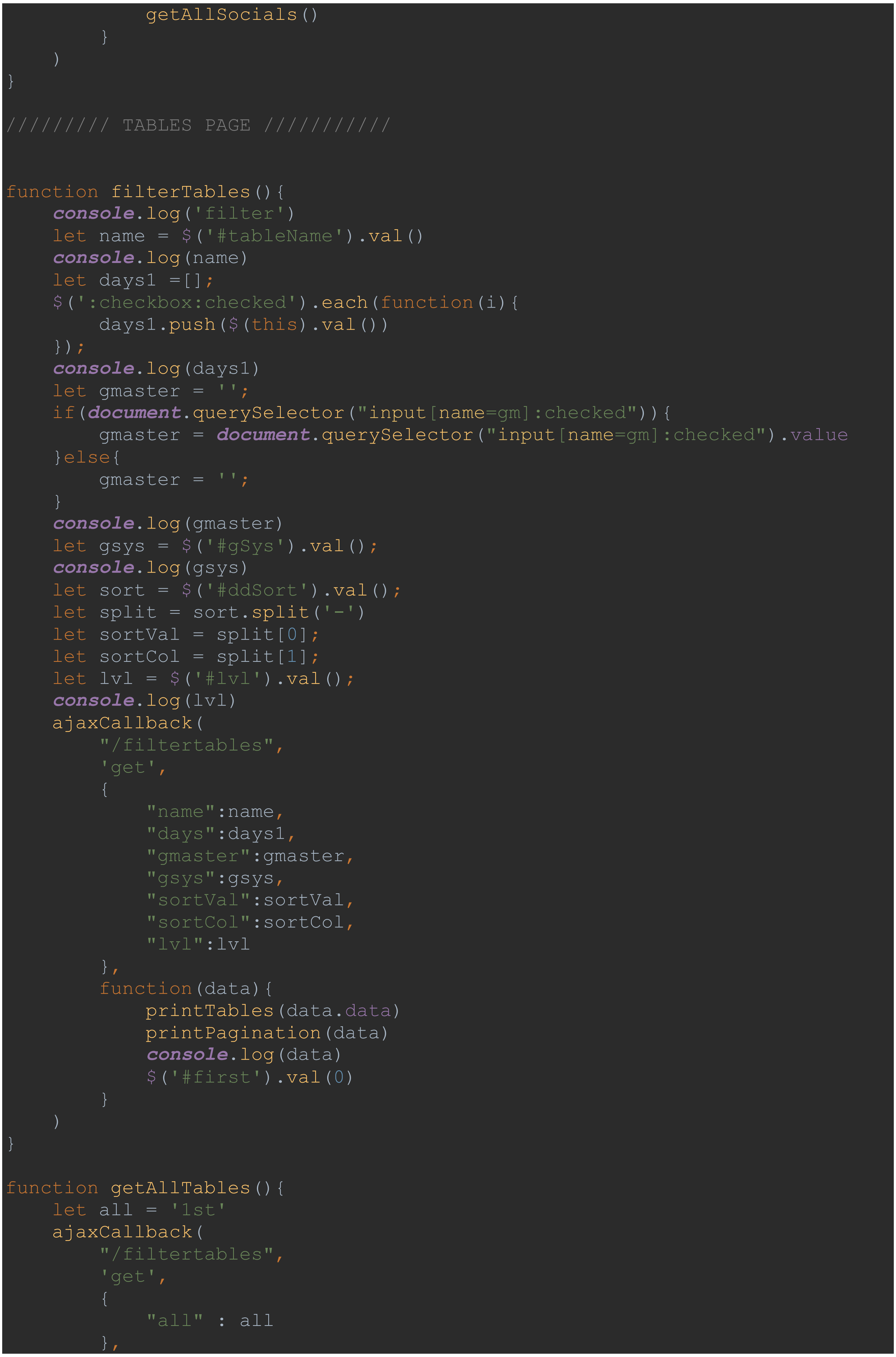


```
            <th scope="col">Update</th>
            <th scope="col">Delete</th>

        </tr>
        </thead>
        <tbody>`

    for(let d of data){
        console.log(d)
        html+=`<tr>
            <th scope="row">${d.id}</th>
            <td>${d.name}</td>
            <td>${d.link}</td>
            <td class="text-center"><a href="#" class="editSoc mr-2" data-
id="${d.id}" ><i class="fa-solid fa-pen-to-square adminLinks"></i></a></td>
            <td class="text-center"><a href="#" class="deleteSoc mr-2"
data-id="${d.id}" ><i class="fa-solid fa-trash adminLinks"></i></a></td>
        </tr>`
    }

    html+=`</tbody>`

    $('#adminContent').html(html)
    $('.deleteSoc').on('click',deleteSocial)
    $('.editSoc').on('click',editSocial)
}

function editSocial(e){
    e.preventDefault()
    let id = this.dataset.id
    ajaxCallback(
        '/allsocials/'+id+'/edit',
        'get',
        {
            'id':id
        },
        function (data){
            printSocForm(e,data)
        }
    )
}

function printSocForm(e,data=null){
    let html=`
<div class="m-4">
            <h2>Socials</h2>
        </div>
<div class="col-6">
<div class="alert" role="alert" id="socFeedback">
                                </div>
                    <form enctype="multipart/form-data">
                        <div class="form-group mt-2">
                            <input type="text" class="form-control form-
control-lg" id="name"`

    if(data!=null){
        html+=`value="${data.name}"`
    }

    html+= ` name="text">
                            <label for="email" class="form-label mt-
```

```
2">Name</label><br>
                            <div id="nameError" class="invalid-form">
                                Name required.
                            </div>
                        </div>
                        <div class="form-group mt-2">
                            <input type="text" class="form-control form-
control-lg" id="link" `
    if(data!=null){
        html+=`value="${data.link}"`
    }

    html+= `name="link">
                            <label for="link" class="form-label mt-
2">Link</label><br>
                            <div id="linkError" class="invalid-form">
                                Link required.
                            </div>
                        </div>`

    if(data!=null){
        html+=`<div class="form-group mt-4">
                            <input type="hidden" id="socId"
value="${data.id}">
                            <button type="submit" class="btn btn-outline-
dark btn-lg px-5" id="socupdateform" name="">Update</button>
                        </div>`
    }else{
        html+=`<div class="form-group mt-4">
                            <button type="submit" class="btn btn-outline-
dark btn-lg px-5" id="soccreateform" name="">Submit</button>
                        </div>`
    }
    `
                    </form>
                </div>`
    $('#adminContent').html(html)
    $('#nameError').css('display','none')
    $('#linkError').css('display','none')
    $('#soccreateform').on('click',submitSocial)
    $('#socupdateform').on('click',updateSocial)

}

function submitSocial(e){
    e.preventDefault();
    let name = $('#name').val()
    let link = $('#link').val()
    let errors=0
    if(name==null || name==''){
        $('#nameError').css('display','block')
        errors++;
    }else{
        $('#nameError').css('display','none')
    }
    if(link==null || link==''){
        $('#linkError').css('display','block')
        errors++;
    }else{
        $('#linkError').css('display','none')
    }
```



```
    if(errors==0){
        ajaxCallback(
            '/allsocials',
            'post',
            {
                'name':name,
                'link':link,
            },
            function (data){
                $('#socFeedback').text(data.feedback)
                $('#socFeedback').addClass('alert-success')
            }
        )
    }
}

function updateSocial(e){
    e.preventDefault();
    let id = $('#socId').val()
    let name = $('#name').val()
    let link = $('#link').val()
    let errors=0
    if(name==null || name==''){
        $('#nameError').css('display','block')
        errors++;
    }else{
        $('#nameError').css('display','none')
    }
    if(link==null || link==''){
        $('#linkError').css('display','block')
        errors++;
    }else{
        $('#linkError').css('display','none')
    }
    if(errors==0){
        ajaxCallback(
            '/allsocials/'+id,
            'post',
            {
                'name':name,
                'link':link,
                '_method':'put'
            },
            function (data){
                $('#socFeedback').text(data.feedback)
                $('#socFeedback').addClass('alert-success')
            }
        )
    }
}

function deleteSocial(e){
    e.preventDefault()
    let id = this.dataset.id
    ajaxCallback(
        '/allsocials/'+id,
        'delete',
        {
            'id':id
        },
        function (data){
```



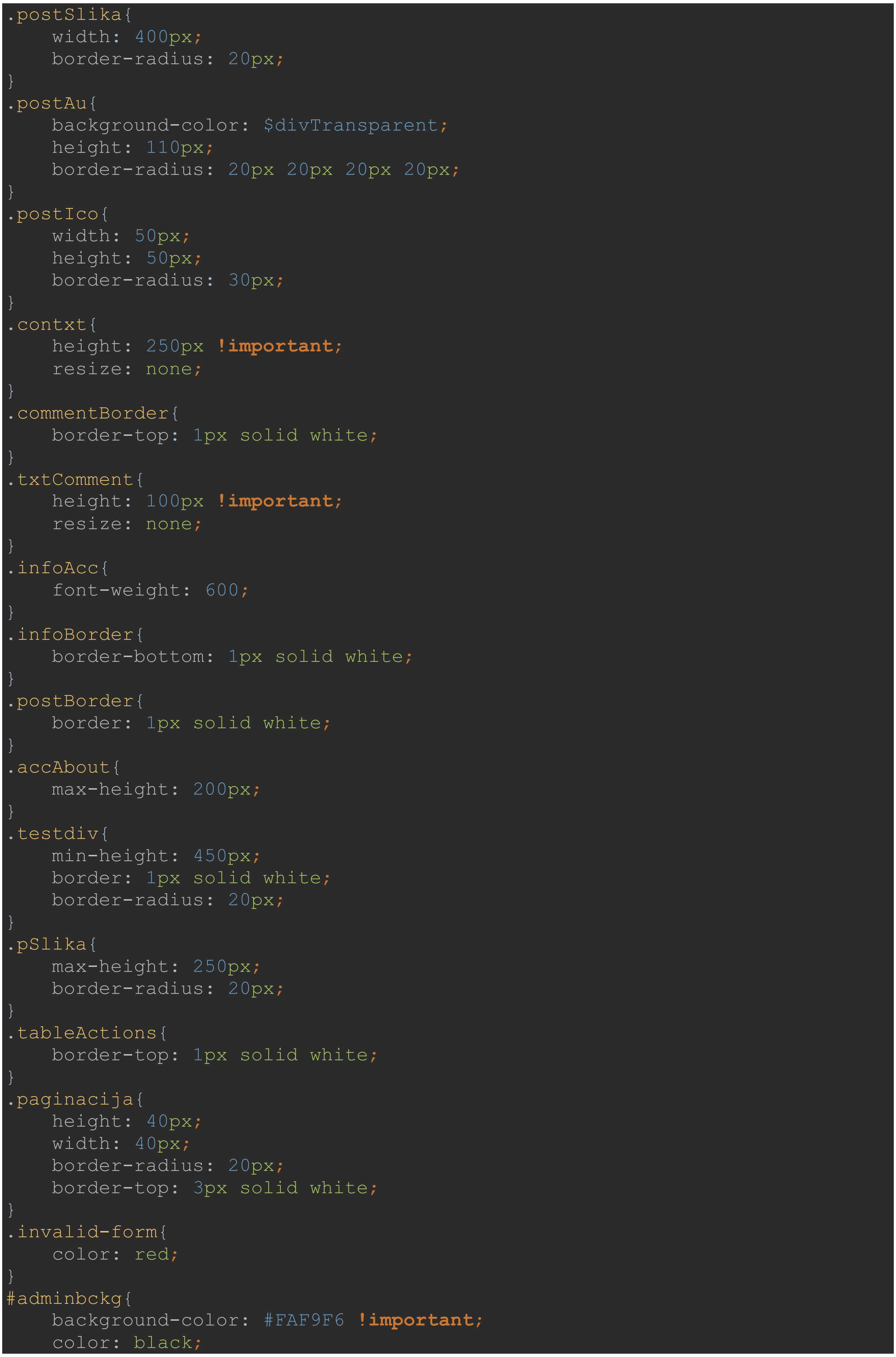
```
            getAllSocials()
        }
    )
}

///////// TABLES PAGE ///////////


function filterTables(){
    console.log('filter')
    let name = $('#tableName').val()
    console.log(name)
    let days1 =[];
    $(':checkbox:checked').each(function(i){
        days1.push($(this).val())
    });
    console.log(days1)
    let gmaster = '';
    if(document.querySelector("input[name=gm]:checked")){
        gmaster = document.querySelector("input[name=gm]:checked").value
    }else{
        gmaster = '';
    }
    console.log(gmaster)
    let gsys = $('#gSys').val();
    console.log(gsys)
    let sort = $('#ddSort').val();
    let split = sort.split('-')
    let sortVal = split[0];
    let sortCol = split[1];
    let lvl = $('#lvl').val();
    console.log(lvl)
    ajaxCallback(
        "/filtertables",
        'get',
        {
            "name":name,
            "days":days1,
            "gmaster":gmaster,
            "gsys":gsys,
            "sortVal":sortVal,
            "sortCol":sortCol,
            "lvl":lvl
        },
        function(data){
            printTables(data.data)
            printPagination(data)
            console.log(data)
            $('#first').val(0)
        }
    )
}

function getAllTables(){
    let all = '1st'
    ajaxCallback(
        "/filtertables",
        'get',
        {
            "all" : all
        },
        function(data){
            printTables(data.data)
            printPagination(data)
            $('#first').val(1)
        }
    )
}

function printPagination(data){
    let html =`<ul class="d-flex flex-row justify-content-center text-
center col-12 mb-0">`
    for(let i=1;i<=data.last_page;i++){
        html+=`<li class="paginacija m-2"><a href="" class="paginationNum"
data-id="${i}">${i}</a></li>`
    }
    html+=`</ul>`
    $('#pagination').html(html)
    $('.paginationNum').click(pagination)
}

function pagination(e){
    e.preventDefault()
    let page=this.dataset.id;
    console.log('PAGE - '+page)
    let all = $('#first').val()
    if(all==1){
        let all = '1st'
        ajaxCallback(
            "/filtertables",
            'get',
            {
                "all" : all,
                "page" : page
            },
            function(data){
                printTables(data.data)
                printPagination(data)
                $('#first').val(1)
            }
        )
    }
    else{
        let name = $('#tableName').val()
        let days1 =[];
        $(':checkbox:checked').each(function(i){
            days1.push($(this).val())
        });
        let gmaster = '';
        if(document.querySelector("input[name=gm]:checked")){
            gmaster =
document.querySelector("input[name=gm]:checked").value
        }else{
            gmaster = '';
        }
        let gsys = $('#gSys').val();
        let sort = $('#ddSort').val();
        let split = sort.split('-')
        let sortVal = split[0];
        let sortCol = split[1];
        let lvl = $('#lvl').val();
```

```
        ajaxCallback(
            "/filtertables",
            'get',
            {
                "name":name,
                "days":days1,
                "gmaster":gmaster,
                "gsys":gsys,
                "sortVal":sortVal,
                "sortCol":sortCol,
                "lvl":lvl,
                "page":page
            },
            function(data){ printTables(data.data)
                printPagination(data)
                console.log(data)
            }
        )
    }
}

function printTables(tables){
    let html=''
    if(tables.length==0){
        html+=`<div class="m-5"><h1>No tables match your
criteria.</h1></div>`
    }
    else{
        for(let table of tables){
            html+=`<div class="mt-4 mb-2 col-12 col-xl-5 p-3 testdiv d-flex
flex-column justify-content-between">
    <span class="text-left w-100 p-2 infoBorder"><h3>${table.name}
</h3></span>
    <div class="d-flex">
        <div class="col-6 d-flex justify-content-center align-items-
center">
            <img src="${baseUrl+'/assets/img/'+table.image}"
alt="${table.alt}" class="img-fluid pSlika">
        </div>
        <div class="col-6 p-0 d-flex justify-content-center align-items-
center">
            <p class="pt-3">
                <i class="fa-solid fa-hat-wizard"></i> Game Master : <br>`
            if(table.game_master){
                html+=`Required <br>`
            }
            else{
                html+=`Not Required <br>`
            }
            html+=`<i class="fa-solid fa-gear"></i> Level required :
${table.l_name}<br>
                <i class="fa-solid fa-calendar"></i> Availability : <br>`
            for(let day of table.days){
                html+=`${day.name.substr(0,3)} `
            }
            html+= `<br><i class="fa-solid fa-book"></i> Game System :
${table.gs_name}<br><br>
            </p>
        </div>
            </div>
            <div class="text-right tableActions pt-3">
                <a href="table/${table.id}"><i class="fa-solid fa-eye pr-2
pl-2"></i>see more</a>
            </div>
            </div>`
        }
    }
   $('#allTables').html(html)
}
```

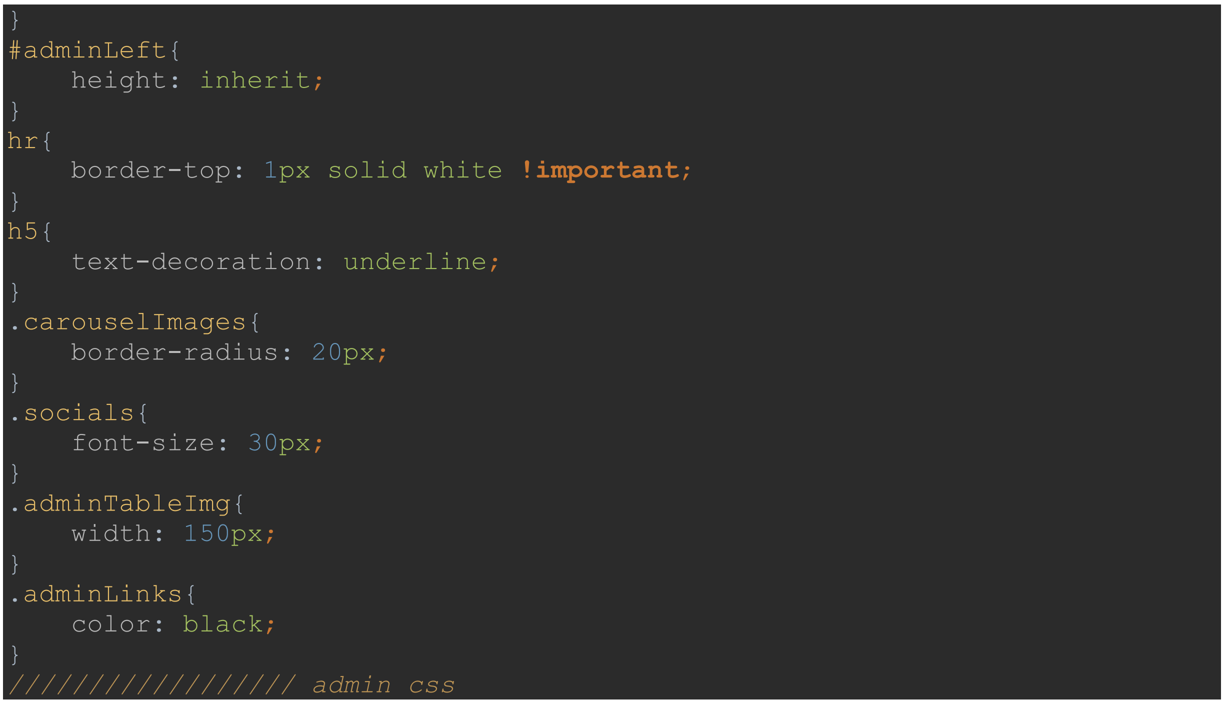
4.3 SCSS

4.3.1 Style.scss

```
html{
    scroll-behavior: smooth;
}
body {
    background: #6a11cb;
    background: -webkit-linear-gradient(to right, rgba(106, 17, 203, 1),
rgba(37, 117, 252, 1));
    background: linear-gradient(to right, rgba(106, 17, 203, 1), rgba(37,
117, 252, 1));
    color: white;
}
$divTransparent :rgba(0, 5, 0, 0.7);

ul{
    list-style: none;
}
th{
    vertical-align: middle;
}
a{
    color: white;
&:hover{
     color: white;
 }
}
#homeImg{
    border-radius: 20px;
}
.profileAuthor{
    height: 300px;
    background-color: $divTransparent;
    border-radius: 20px;
}
.slikaAuthor{
    border-radius: 20px;
    height: 250px;
}
.feedbackBad{
    display: none;
}
.profilSlika{
    border-radius: 20px;
    max-height: 180px;
}
.profilePersonal{
    border-radius: 20px;
}
```



```
.postSlika{
    width: 400px;
    border-radius: 20px;
}
.postAu{
    background-color: $divTransparent;
    height: 110px;
    border-radius: 20px 20px 20px 20px;
}
.postIco{
    width: 50px;
    height: 50px;
    border-radius: 30px;
}
.contxt{
    height: 250px !important;
    resize: none;
}
.commentBorder{
    border-top: 1px solid white;
}
.txtComment{
    height: 100px !important;
    resize: none;
}
.infoAcc{
    font-weight: 600;
}
.infoBorder{
    border-bottom: 1px solid white;
}
.postBorder{
    border: 1px solid white;
}
.accAbout{
    max-height: 200px;
}
.testdiv{
    min-height: 450px;
    border: 1px solid white;
    border-radius: 20px;
}
.pSlika{
    max-height: 250px;
    border-radius: 20px;
}
.tableActions{
    border-top: 1px solid white;
}
.paginacija{
    height: 40px;
    width: 40px;
    border-radius: 20px;
    border-top: 3px solid white;
}
.invalid-form{
    color: red;
}
#adminbckg{
    background-color: #FAF9F6 !important;
    color: black;
```



```
}
#adminLeft{
    height: inherit;
}
hr{
    border-top: 1px solid white !important;
}
h5{
    text-decoration: underline;
}
.carouselImages{
    border-radius: 20px;
}
.socials{
    font-size: 30px;
}
.adminTableImg{
    width: 150px;
}
.adminLinks{
    color: black;
}
////////////////// admin css
```