

Nama : Aida Hanifa

NIM : 2602078764

LB01

Final Assignment Web Programming

Routes

```
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use Illuminate\Support\Facades\Auth;
5 use App\Http\Controllers\PaymentController;
6 use App\Http\Controllers\ProfileController;
7 use App\Http\Controllers\HomeController;
8 use App\Http\Controllers\WishlistController;
9 use App\Http\Controllers\ChatController;
10 use App\Http\Controllers\AvatarController;
11
12 Auth::routes();
13
14 Route::get('/', [HomeController::class, 'index'])->name('home');
15
16 // fitur pembayaran
17 Route::get('/payment', [PaymentController::class, 'index'])->name('payment.index');
18 Route::post('/payment/store', [PaymentController::class, 'store'])->name('payment.store');
19 Route::post('/payment/add-to-wallet', [PaymentController::class, 'addToWallet'])->name('payment.addToWallet');
20 Route::get('/payment/success', [PaymentController::class, 'success'])->name('payment.success');
```

Code di atas merupakan route address untuk mengakses berbagai fitur. Route ini mencakup berbagai halaman dan fungsi, namun dengan syarat harus ter autentikasi. Penjelasannya:

- **Route '/'**: Mengakses halaman utama (home) menggunakan HomeController.
- **Route '/payment'**: Mengakses halaman pembayaran.

```
Route::middleware(['auth'])->group(function () { void
    // Profil
    Route::get('/profile', [ProfileController::class, 'show'])->name('profile.show');
    Route::post('/profile/update', [ProfileController::class, 'update'])->name('profile.update');
    Route::post('/profile/update-avatar', [ProfileController::class, 'updateAvatar'])->name('profile.update.avatar');
    Route::delete('/profile/remove/{friendId}', [ProfileController::class, 'remove'])->name('profile.remove');

    // Chat
    Route::get('/chat/{friendId}', [ChatController::class, 'show'])->name('chat.show');
    Route::post('/chat/{friendId}/send', [ChatController::class, 'send'])->name('chat.send');
    Route::get('/chat/start/{friendId}', [WishlistController::class, 'startChat'])->name('chat.start');

    // Notif
    Route::get('/notif', [WishlistController::class, 'showNotifications'])->name('notif');

    // Friends
    Route::post('/wishlist/{user}', [WishlistController::class, 'add'])->name('wishlist.add');
    Route::post('/wishlist/add/{userId}', [WishlistController::class, 'add'])->name('wishlist.add');
    Route::patch('/wishlist/remove/{wishlistId}', [WishlistController::class, 'removeNotification'])->name('wishlist.remove');

    // Avatar
    Route::get('/avatars', [AvatarController::class, 'index'])->name('avatars.index');
    Route::post('/avatars/buy', [AvatarController::class, 'buy'])->name('avatars.buy');
});

// Localization
Route::get('/set-locale/{locale}', function ($locale): mixed|RedirectResponse {
    if (in_array($locale, ['en', 'id'])) {
        session(['locale' => $locale]);
    }
    return redirect()->back();
})->name('set-locale');
```

Code diatas merupakan route address untuk mengakses website pages yang perlu login terlebih dahulu. Dengan middleware auth untuk memastikan hanya pengguna yang sudah login yang dapat mengakses fitur pentignya

Rute yang Memerlukan Autentikasi

Beberapa rute yang dikelompokkan middleware Auth:

- **Profil Pengguna:**
 - GET /profile: Menampilkan halaman profil pengguna.
 - POST /profile/update: Memperbarui data profil.
 - POST /profile/update-avatar: Mengganti avatar pengguna.
 - DELETE /profile/remove/{friendId}: Menghapus teman dari daftar teman.
- **Percakapan dan Chat:**
 - GET /chat/{friendId}: Menampilkan percakapan dengan teman tertentu.
 - POST /chat/{friendId}/send: Mengirim pesan kepada teman tertentu.
 - GET /chat/start/{friendId}: Memulai chat dengan teman.
- **Notifikasi:** /notif: Menampilka notifikasi.
- **Friends:**
 - POST /wishlist/{user} atau POST /wishlist/add/{userId}: Menambahkan pengguna ke wishlist.
 - PATCH /wishlist/remove/{wishlistId}: Meng Ok notifikasi pertemanan tertentu dan dihapuskan dari notif.
- **Avatar:**
 - GET /avatars: Menampilkan daftar avatar yang tersedia.
 - POST /avatars/buy: Membeli avatar pada halaman avatar.

Mengatur Bahasa

- **Route '/set-locale/{locale}':** Mengatur bahasa aplikasi (English atau Indonesia).

Dengan struktur ini, aplikasi menjadi lebih terorganisir dan mudah untuk dikelola.

Database (Migration and Seeder/Faker)

Terdapat total 5 database yang saya gunakan, 2 database digunakan untuk challenges

Migration User:

```

public function up(): void
{
    Schema::table('users', function (Blueprint $table): void {
        $table->string('profile_photo')->nullable();
        $table->string('gender')->nullable();
        $table->string('instagram')->nullable();
        $table->string('mobile')->nullable();
        $table->json('hobbies')->nullable();
        $table->integer('age')->nullable();
        $table->enum('friendship_intent', ['casual_friendships', 'relationships', 'business_networking'])->nullable();
        $table->integer('registration_fee')->default(0);
        $table->decimal('wallet_balance', 10, 2)->default(0);
    });
}

```

Saya menambahkan column coin dan foreign_key avatar untuk melakukan challenge

```

public function up(): void
{
    Schema::table('users', function (Blueprint $table): void {
        $table->unsignedInteger('coins')->default(100);
        $table->foreignId('avatar_id')->nullable()->constrained(table: 'avatars')->onDelete('set null');
    });
}

```

Pada ini tidak ada seeder karena data didapatkan dari user yang melakukan registration

Table Users:

	id	name	email	email_verified_at	password	remember_token	created_at	updated_at	profile_photo	gender	instagram	mobile	hobbies	age	friendship_intent	registration_fee	wallet_balance	coins	avatar_id
<input type="checkbox"/>	1	id@pma	id@pma		\$2y123p7u050E2gALPC3uA0u7F0Y3G4g7u0Qd0P4g		2025-01-09 15:57:54	2025-01-09 04:24:50	173036089.jpg	Male	https://www.instagram.com/teawag/	0812145678	["Reading", "Writing"]	27	relationships	119902	800.00	0	3
<input type="checkbox"/>	2	id@pma	id@pma		\$2y123p7u050E2gALPC3uA0u7F0Y3G4g7u0Qd0P4g		2025-01-09 15:57:54	2025-01-08 16:02:21	173034541.jpg	Female	https://www.instagram.com/teawag/	0812145678	["Singing", "Cooking", "Writing"]	23	relationships	179495	1000.00	100	1011
<input type="checkbox"/>	3	id@pma	id@pma		\$2y123p7u050E2gALPC3uA0u7F0Y3G4g7u0Qd0P4g		2025-01-09 15:57:54	2025-01-08 16:02:21	1011	Female	https://www.instagram.com/teawag/	0812145678	["Cooking", "Singing", "Dancing"]	28	casual_friendships	123456	200.00	50	3
<input type="checkbox"/>	4	id@pma	id@pma		\$2y123p7u050E2gALPC3uA0u7F0Y3G4g7u0Qd0P4g		2025-01-09 15:57:54	2025-01-09 15:18:23	173043053.jpg	Male	https://www.instagram.com/teawag/	0812145678	["Dancing", "Reading", "Writing"]	29	business_networking	100127	0.00	100	1011

Migration Wishlist (Friends)

```

Schema::create('wishlists', function (Blueprint $table): void {
    $table->id();
    $table->foreignId('user_id')->constrained('users')->onDelete('cascade');
    $table->foreignId('wishlist_user_id')->constrained('users')->onDelete('cascade');
    $table->timestamps();
});

```

```

public function up(): void
{
    Schema::table('wishlists', function (Blueprint $table): void {
        $table->boolean('hidden')->default(false)->after('wishlist_user_id');
    });
}

```

Tambahan columns 'hidden' agar notif Ketika sudah diterima dan ditekan button 'Ok' maka akan hilang

Tabele Whislist:

	id	user_id	wishlist_user_id	hidden	created_at	updated_at
<input type="checkbox"/> Edit Copy Delete	7	1	2	1	2025-01-09 04:24:23	2025-01-09 04:24:27
<input type="checkbox"/> Edit Copy Delete	8	2	1	1	2025-01-09 04:24:30	2025-01-09 04:24:33
<input type="checkbox"/> Edit Copy Delete	9	1	4	1	2025-01-09 15:18:28	2025-01-09 15:18:35
<input type="checkbox"/> Edit Copy Delete	10	4	1	1	2025-01-09 15:18:40	2025-01-09 15:18:44

Pada table ini berfungsi sebagai pengiriman request friends

Chats Migration:

```
public function up(): void
{
    Schema::create('chats', function (Blueprint $table): void {
        $table->id();
        $table->foreignId('sender_id')->constrained('users')->onDelete('cascade');
        $table->foreignId('receiver_id')->constrained('users')->onDelete('cascade');
        $table->text('message');
        $table->boolean('read')->default(false);
        $table->timestamps();
    });
}
```

Chats Table:

<div><div><div>←</div><div>T</div><div>→</div></div></div>				▼	id	sender_id	receiver_id	message	read	created_at	updated_at		
<input type="checkbox"/>		Edit		Copy		Delete	1	2	1	Halo	1	2025-01-08 14:04:55	2025-01-08 14:05:08
<input type="checkbox"/>		Edit		Copy		Delete	2	1	2	Hei^^!!	1	2025-01-08 14:05:22	2025-01-09 02:31:45
<input type="checkbox"/>		Edit		Copy		Delete	3	3	1	halo	1	2025-01-08 16:37:40	2025-01-08 16:37:48
<input type="checkbox"/>		Edit		Copy		Delete	4	1	3	haii^^	1	2025-01-08 16:37:57	2025-01-08 16:39:14
<input type="checkbox"/>		Edit		Copy		Delete	5	1	2	test	1	2025-01-09 02:31:40	2025-01-09 02:31:45
<input type="checkbox"/>		Edit		Copy		Delete	6	2	1	halo, masuk	1	2025-01-09 02:31:55	2025-01-09 03:59:10
<input type="checkbox"/>		Edit		Copy		Delete	7	1	2	ok	1	2025-01-09 04:00:46	2025-01-09 04:23:38
<input type="checkbox"/>		Edit		Copy		Delete	8	2	1	test	1	2025-01-09 04:23:49	2025-01-09 04:23:58
<input type="checkbox"/>		Edit		Copy		Delete	9	1	2	aman	1	2025-01-09 04:24:02	2025-01-09 04:33:17
<input type="checkbox"/>		Edit		Copy		Delete	10	1	2	halo	1	2025-01-09 04:27:18	2025-01-09 04:33:17
<input type="checkbox"/>		Edit		Copy		Delete	11	2	1	oke, masukk	0	2025-01-09 04:33:27	2025-01-09 04:33:27
<input type="checkbox"/>		Edit		Copy		Delete	12	4	1	halo bro	1	2025-01-09 16:47:25	2025-01-09 16:47:28
<input type="checkbox"/>		Edit		Copy		Delete	13	1	4	oy, halo	0	2025-01-09 16:47:37	2025-01-09 16:47:37

Sender_id akan sama dengan user mengirim walau dengan teman mana pun begitupun receiver_id

[Challenges Buy Avatar]

Avatar Migrations:

```
public function up(): void
{
    Schema::create('avatars', function (Blueprint $table): void {
        $table->id();
        $table->string('image_url');
        $table->unsignedInteger('price');
        $table->timestamps();
    });
}
```

Avatar Table:

<div><div>←</div><div>→</div></div>		<div>▼</div>	id	image_url	price	created_at	updated_at
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		1	https://via.placeholder.com/150x150.png/007799?tex...	5000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		2	https://via.placeholder.com/150x150.png/00aaaa?tex...	100	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		3	https://via.placeholder.com/150x150.png/007755?tex...	50	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		4	https://via.placeholder.com/150x150.png/0088ff?tex...	50	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		5	https://via.placeholder.com/150x150.png/005522?tex...	10000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		6	https://via.placeholder.com/150x150.png/0022bb?tex...	50000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		7	https://via.placeholder.com/150x150.png/0066bb?tex...	1000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		8	https://via.placeholder.com/150x150.png/0099bb?tex...	5000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		9	https://via.placeholder.com/150x150.png/003366?tex...	50000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		10	https://via.placeholder.com/150x150.png/0033ff?tex...	5000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		11	https://via.placeholder.com/150x150.png/007799?tex...	1000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		12	https://via.placeholder.com/150x150.png/00dd77?tex...	50000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		13	https://via.placeholder.com/150x150.png/00aa88?tex...	500	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		14	https://via.placeholder.com/150x150.png/00ee77?tex...	50	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		15	https://via.placeholder.com/150x150.png/009911?tex...	500	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		16	https://via.placeholder.com/150x150.png/00cc88?tex...	100	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		17	https://via.placeholder.com/150x150.png/00ff00?tex...	5000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		18	https://via.placeholder.com/150x150.png/0033dd?tex...	100000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		19	https://via.placeholder.com/150x150.png/000066?tex...	50	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		20	https://via.placeholder.com/150x150.png/0077dd?tex...	50000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		21	https://via.placeholder.com/150x150.png/003399?tex...	10000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		22	https://via.placeholder.com/150x150.png/008844?tex...	5000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		23	https://via.placeholder.com/150x150.png/00bb11?tex...	100	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		24	https://via.placeholder.com/150x150.png/0066ee?tex...	10000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		25	https://via.placeholder.com/150x150.png/002233?tex...	50000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		26	https://via.placeholder.com/150x150.png/002200?tex...	50000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		27	https://via.placeholder.com/150x150.png/003355?tex...	50	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		28	https://via.placeholder.com/150x150.png/004444?tex...	500	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		29	https://via.placeholder.com/150x150.png/00ccbb?tex...	50	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		30	https://via.placeholder.com/150x150.png/00bb55?tex...	50	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		31	https://via.placeholder.com/150x150.png/00bb88?tex...	100000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		32	https://via.placeholder.com/150x150.png/00aa77?tex...	1000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		33	https://via.placeholder.com/150x150.png/005577?tex...	5000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		34	https://via.placeholder.com/150x150.png/00ccff?tex...	5000	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		35	https://via.placeholder.com/150x150.png/007700?tex...	500	2025-01-08 14:02:48	2025-01-08 14:02:48
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		36	https://via.placeholder.com/150x150.png/0033ff?tex...	500	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		37	https://via.placeholder.com/150x150.png/0022dd?tex...	500	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		38	https://via.placeholder.com/150x150.png/00ee33?tex...	100000	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		39	https://via.placeholder.com/150x150.png/00ff55?tex...	1000	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		40	https://via.placeholder.com/150x150.png/007788?tex...	50	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		41	https://via.placeholder.com/150x150.png/00bb11?tex...	10000	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		42	https://via.placeholder.com/150x150.png/009955?tex...	100	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		43	https://via.placeholder.com/150x150.png/008877?tex...	1000	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		44	https://via.placeholder.com/150x150.png/000077?tex...	100000	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		45	https://via.placeholder.com/150x150.png/00ff99?tex...	100000	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		46	https://via.placeholder.com/150x150.png/009922?tex...	1000	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		47	https://via.placeholder.com/150x150.png/00ff88?tex...	1000	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		48	https://via.placeholder.com/150x150.png/0088ff?tex...	5000	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		49	https://via.placeholder.com/150x150.png/00ee44?tex...	500	2025-01-08 14:02:49	2025-01-08 14:02:49
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		50	https://via.placeholder.com/150x150.png/009955?tex...	500	2025-01-08 14:02:49	2025-01-08 14:02:49

Seeder

```

public function run(): void
{
    $faker = Faker::create();
    foreach (range(start: 1, end: 50) as $index) {
        Avatar::create([
            'image_url' => $faker->imageUrl(width: 150, height: 150, category: 'avatar', randomize: true, word: 'Avatar' . $index),
            'price' => $faker->randomElement(array: [50, 100, 500, 1000, 5000, 10000, 50000, 100000]),
        ]);
    }
}

```

Pada Avatar menggunakan sedder faker untuk datanya sesuai ketentuan challenge

Transaction Migrations:

```

public function up(): void
{
    Schema::create('transactions', function (Blueprint $table): void {
        $table->id();
        $table->foreignId('user_id')->constrained('users')->onDelete('cascade');
        $table->foreignId('avatar_id')->nullable()->constrained(table: 'avatars')->onDelete('set null');
        $table->unsignedInteger('amount');
        $table->timestamps();
    });
}

```

Transaction Table:

<div><div><div>←</div><div>T</div><div>→</div></div><div></div></div>						id	user_id	avatar_id	amount	created_at	updated_at
<div><div><div></div></div><div><div><div></div></div><div><div></div></div></div><div><div>Edit</div><div>Copy</div><div>Delete</div></div></div>	1	1	3	50	2025-01-08 14:12:15	2025-01-08 14:12:15					
<div><div><div></div></div><div><div><div></div></div><div><div></div></div></div><div><div>Edit</div><div>Copy</div><div>Delete</div></div></div>	2	1	4	50	2025-01-08 14:13:19	2025-01-08 14:13:19					
<div><div><div></div></div><div><div><div></div></div><div><div></div></div></div><div><div>Edit</div><div>Copy</div><div>Delete</div></div></div>	3	3	3	50	2025-01-08 16:45:30	2025-01-08 16:45:30					

Ini adalah database yang menyimpan transaksi dari avatar yang dibeli oleh para user.

Model

Class User:


```

0 references | 0 overrides
public function reverseWishlist(): HasMany
{
    return $this->hasMany(Wishlist::class, 'wishlist_user_id', 'id');
}

0 references | 0 overrides
public function unreadNotificationsCount(): int
{
    return Wishlist::where('wishlist_user_id', $this->id)
        ->whereDoesntHave('reverseWishlist', function ($query): void {
            $query->where('user_id', $this->id);
        })
        ->count();
}

0 references | 0 overrides
public function transactions(): HasMany
{
    return $this->hasMany(Transaction::class);
}

0 references | 0 overrides
public function avatar(): BelongsTo
{
    return $this->belongsTo(Avatar::class, 'avatar_id');
}

```

User dapat memiliki banyak request teman, dan dapat melakukan banyak transaksi, Memiliki relasi ke avatar untuk dapat mengganti profile dengan avatar yang dibeli.

Class Wishlist:

```

0 references | 0 overrides
public function user(): BelongsTo
{
    return $this->belongsTo(User::class, 'user_id');
}

0 references | 0 overrides
public function reverseWishlist(): HasOne
{
    return $this->hasOne(Wishlist::class, 'wishlist_user_id', 'user_id');
}

```

Memiliki relasi ke user yang mengirim pesan, dan memiliki hubungan satu arah dengan request teman untuk mengecek hubungan mutual dalam berteman

Class Chat:

```

0 references | 0 overrides
public function sender(): BelongsTo
{
    return $this->belongsTo(User::class, 'sender_id');
}

0 references | 0 overrides
public function receiver(): BelongsTo
{
    return $this->belongsTo(User::class, 'receiver_id');
}

```

Memiliki relasi ke user yang mengirim pesan dan teman sebagai penerima.

Class Transactions:

```

0 references | 0 overrides
public function user(): BelongsTo
{
    return $this->belongsTo(User::class);
}

0 references | 0 overrides
public function avatar(): BelongsTo
{
    return $this->belongsTo(Avatar::class);
}

```

Memiliki relasi ke user dan avatar

Class Avatar:

0 references | 0 overrides

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

0 references | 0 overrides

```
public function Transaction(): BelongsTo  
{  
    return $this->belongsTo(Transaction::class);  
}
```

Memiliki relasi ke user dan Transaction.

Controller

Register Controller:

```

0 references | 0 overrides
public function __construct()
{
    $this->middleware('guest');
}

0 references | 0 overrides
public function register(Request $request): mixed|RedirectResponse
{
    $validator = Validator::make($request->all(), [
        'name' => 'required|string|max:255',
        'email' => 'required|string|email|max:255|unique:users',
        'password' => 'required|string|min:8|confirmed',
        'gender' => 'required|in:Male,Female',
        'hobbies' => ['required', 'string', 'min:3', function ($attribute, $value, $fail): void {
            $hobbies = explode(separator: ',', string: $value);
            if (count(value: $hobbies) < 3) {
                $fail('The hobbies must have at least 3 items.');
            }
        }],
        'instagram' => 'required|url|regex:/^https:\/\/www.instagram.com\/[a-zA-Z0-9._]+$/ ',
        'mobile' => 'required|digits:10',
        'age' => 'required|numeric|min:18|max:35',
        'friendship_intent' => 'required|in:casual_friendships,relationships,business_networking',
        'registration_fee' => 'required|numeric|min:100000|max:125000',
    ]);

    if ($validator->fails()) {
        return redirect()->route('register')
            ->withErrors($validator)
            ->withInput();
    }

    $user = User::create([
        'name' => $request->name,
        'email' => $request->email,
        'password' => bcrypt($request->password),
        'gender' => $request->gender,
        'hobbies' => json_encode(value: explode(separator: ',', string: $request->hobbies)),
        'instagram' => $request->instagram,
        'mobile' => $request->mobile,
        'age' => $request->age,
        'friendship_intent' => $request->friendship_intent,
        'registration_fee' => $request->registration_fee,
    ]);

    Session::put('registration_fee', $request->registration_fee);

    Auth::login($user); // Login otomatis

    return redirect()->route('payment.index');
}

```

menangani proses registrasi pengguna baru pada aplikasi. Controller ini berisi fungsi-fungsi yang mengatur validasi data, pembuatan akun pengguna yang akan berhasil Ketika syarat sudah ditentukan. Setelah data valid, pengguna akan diarahkan ke halaman pembayaran dengan route payment.index.

Payment Controller:

```

// references | 0 implements
class PaymentController extends Controller
{
    // Menampilkan halaman pembayaran
    1 reference | 0 overrides
    public function index(): View
    {
        return view('payment');
    }

    1 reference | 0 overrides
    public function store(Request $request): mixed | RedirectResponse | View
    {
        $registrationFee = session('registration_fee');

        $request->validate([
            'paid_amount' => 'required|numeric',
        ]);

        $paidAmount = $request->paid_amount;

        if ($paidAmount < $registrationFee) {
            $underpaidAmount = $registrationFee - $paidAmount;
            return redirect()->back()->withErrors(['paid_amount' => 'You are still underpaid by ' . number_format(num: $underpaidAmount, decimals: 2)]);
        } elseif ($paidAmount > $registrationFee) {
            $overpaidAmount = $paidAmount - $registrationFee;

            return view('payment.overpaid', [
                'overpaidAmount' => $overpaidAmount
            ]);
        }

        return redirect()->route('payment.success');
    }

    1 reference | 0 overrides
    public function addToWallet(Request $request): RedirectResponse
    {
        $overpaidAmount = $request->input('overpaid_amount');

        if ($request->has('add_to_wallet') && $request->add_to_wallet === 'yes') {
            // Konversi overpaidAmount ke koin dan tambahkan ke wallet
            $user = Auth::user();
            $coinAmount = $overpaidAmount * 100; // Pemisalan 1 uang = 100 koin
            $user->wallet_balance += $coinAmount;
            $user->save();

            return redirect()->route('payment.success')->with('message', 'Payment successful! Your overpayment has been added to your wallet as ' . $coinAmount . ' coins.');
        }

        return redirect()->route('payment.index')->withErrors(['paid_amount' => 'You overpaid by ' . number_format(num: $overpaidAmount, decimals: 2) . '. Please enter the correct amount.']);
    }

    1 reference | 0 overrides
    public function success(): View
    {
        return view('payment.success');
    }
}

```

Controller ini berfungsi untuk agar pembayaran pada register dapat dilakukan dan untuk perhitungan overpaid maupun underpaid

Home Controller:

```

0 references | 0 overrides
public function index(Request $request): View
{
    $authUserId = Auth::id();
    $search = $request->input('search');
    $gender = $request->input('gender');
    $hobby = $request->input('hobby');

    $users = User::where('id', '!=', $authUserId)
        ->when($search, function (Builder $query, $search): void {
            $query->whereRaw('LOWER(hobbies) LIKE ?', ['%' . strtolower(string: $search) . '%']);
        })
        ->when($gender, function (Builder $query, $gender): void {
            $query->where('gender', '=', $gender);
        })
        ->when($hobby, function (Builder $query, $hobby): void {
            $query->whereRaw('LOWER(hobbies) LIKE ?', ['%' . strtolower(string: $hobby) . '%']);
        })
        ->get();

    foreach ($users as $user) {
        // Ngecek udah saling berteman
        $isMutual = Wishlist::where('user_id', $authUserId)
            ->where('wishlist_user_id', $user->id)
            ->exists() &&
            Wishlist::where('user_id', $user->id)
            ->where('wishlist_user_id', $authUserId)
            ->exists();

        // Nentuin statusnya
        $user->isMutual = $isMutual;
        $user->isFollowing = !$isMutual && Wishlist::where('user_id', $authUserId)
            ->where('wishlist_user_id', $user->id)
            ->exists();
    }

    return view('home', compact(var_name: 'users', var_names: 'search', 'gender', 'hobby'));
}

0 references | 0 overrides
public function searchByFieldOfWork(Request $request): View
{
    $search = $request->input('search');

    $users = User::select('id', 'name', 'profile_picture', 'profession', 'hobbies')
        ->whereRaw('LOWER(hobbies) LIKE ?', ['%' . strtolower(string: $search) . '%'])
        ->get();

    return view('search.result', compact(var_name: 'users', var_names: 'search'));
}

0 references | 0 overrides
public function filterByGender(): View
{
    $users = User::select('id', 'name', 'gender', 'profile_picture', 'profession', 'hobbies')->get();

    return view('filter.gender', compact(var_name: 'users'));
}

```

menangani berbagai fungsi terkait halaman utama aplikasi, termasuk pencarian pengguna berdasarkan hobi, gender, atau bidang pekerjaan, serta filtering data untuk menampilkan hasil tertentu.

Wishlist Controller:

```

class WishlistController extends Controller
{
  2 references | 0 overrides
  public function add(Request $request, $userId): RedirectResponse
  {
    $authUser = Auth::user();
    $targetUser = User::findOrFail($userId);

    if ($authUser->id == $targetUser->id) {
      return redirect()->route('home')->with('message', 'You cannot send a friend request to yourself.');
```

```

1 reference | 0 overrides
public function removeNotification(Request $request, $wishlistId): RedirectResponse
{
    $wishlist = Wishlist::findOrFail($wishlistId);

    if ($wishlist->wishlist_user_id == Auth::id()) {
        $wishlist->hidden = true;
        $wishlist->save();
    }

    return redirect()->back()->with('message', 'Notification removed!');
}

```

```

1 reference | 0 overrides
public function startChat($friendId): View
{
    $authUser = Auth::user();
    $friend = User::findOrFail($friendId);

    Chat::where('sender_id', $friendId)
        ->where('receiver_id', $authUser->id)
        ->where('read', false)
        ->update(['read' => true]);

    $messages = Chat::where(function (Chat $query) use ($authUser, $friend): void {
        $query->where('sender_id', $authUser->id)
            ->where('receiver_id', $friend->id);
    })
        ->orWhere(function (Chat $query) use ($authUser, $friend): void {
            $query->where('sender_id', $friend->id)
                ->where('receiver_id', $authUser->id);
        })
        ->orderBy('created_at', 'asc')
        ->get();

    return view('chat', compact('var_name' => 'friend', 'var_names' => 'messages'));
}

```

Controller ini memungkinkan user untuk menambahkan teman ke wishlist mereka, mengelola notifikasi terkait permintaan pertemanan atau pesan, serta memulai percakapan dengan pengguna lain.

Chat Controller:

0 references | 0 overrides

```
public function index(): View
{
    $user = Auth::user();

    $conversations = Chat::with(['sender', 'receiver'])
        ->where('sender_id', $user->id)
        ->orWhere('receiver_id', $user->id)
        ->get()
        ->groupBy(function (Chat $chat) use ($user): mixed {
            return $chat->sender_id === $user->id ? $chat->receiver_id : $chat->sender_id;
        });

    return view('chat.index', compact('var_name' => 'conversations'));
}
```

0 references | 0 overrides

```
public function showChatDetail($receiverId): View
{
    $receiver = User::findOrFail($receiverId);
    $user = Auth::user();

    $messages = Chat::where(function (Chat $query) use ($user, $receiver): void {
        $query->where('sender_id', $user->id)
            ->where('receiver_id', $receiver->id);
    })
        ->orWhere(function (Chat $query) use ($user, $receiver): void {
            $query->where('sender_id', $receiver->id)
                ->where('receiver_id', $user->id);
        })
        ->orderBy('created_at', 'asc')
        ->get();

    // Mark messages as read
    Chat::where('receiver_id', $user->id)
        ->where('sender_id', $receiver->id)
        ->update(['read' => true]);

    return view('chat.detail', compact('var_name' => 'messages', 'var_names' => 'receiver'));
}
```



```

0 references | 0 overrides
public function sendMessage(Request $request): mixed|RedirectResponse
{
    Log::info('sendMessage called', $request->all());

    $request->validate([
        'message' => 'required|string',
        'receiver_id' => 'required|exists:user,id',
    ]);

    try {
        $chat = Chat::create([
            'sender_id' => Auth::id(),
            'receiver_id' => $request->receiver_id,
            'message' => $request->message,
        ]);

        Log::info('Message stored:', $chat->toArray());

        return redirect()->route('chat.detail', ['receiverId' => $request->receiver_id]);
    } catch (\Exception $e) {
        Log::error('Error storing message:', ['error' => $e->getMessage()]);
        return back()->with('error', 'Failed to send the message.');
```

```

1 reference | 0 overrides
public function send(Request $request, $friendId): RedirectResponse

$authUser = Auth::user();
$friend = User::findOrFail($friendId);

// Periksa hubungan mutual
$isMutual = Wishlist::where('user_id', $authUser->id)
    ->where('wishlist_user_id', $friend->id)
    ->exists() &&
    Wishlist::where('user_id', $friend->id)
    ->where('wishlist_user_id', $authUser->id)
    ->exists();

if (!$isMutual) {
    return redirect()->route('home')->with('error', 'You cannot send a message to this user.');
```

```

// Nyimpan pesan di db
Chat::create([
    'sender_id' => $authUser->id,
    'receiver_id' => $friend->id,
    'message' => $request->input('message'),
]);

return redirect()->route('chat.start', ['friendId' => $friend->id])->with('message', 'Message sent!');
```

Controller ini memungkinkan pengguna untuk melihat daftar percakapan mereka yang sudah dilakukan oleh user dan temmanya, melihat detail pesan dengan pengguna tertentu, serta mengirim pesan.

Profile Controller:

```

1 reference | 0 overrides
public function show(): RedirectResponse|View
{
    $authUser = Auth::user();
    $purchasedAvatars = $authUser->transactions()->with('avatar')->get()->pluck('avatar');

    if (!$authUser) {
        return redirect()->route('login')->with('error', 'You must be logged in to access this page.');
```

```

    }

    $mutualFriends = User::whereIn('id', function ($query) use ($authUser): void {
        $query->select('wishlist_user_id')
            ->from('wishlists')
            ->where('user_id', $authUser->id)
            ->whereIn('wishlist_user_id', function ($subQuery) use ($authUser): void {
                $subQuery->select('user_id')
                    ->from('wishlists')
                    ->where('wishlist_user_id', $authUser->id);
            });
    })->get();

    return view('profile.show', compact('authUser', 'mutualFriends', 'purchasedAvatars'));
}

1 reference | 0 overrides
public function update(Request $request): RedirectResponse
{
    $authUser = Auth::user();

    $request->validate([
        'profile_photo' => 'nullable|image|mimes:jpeg,png,jpg,gif|max:2048',
    ]);

    if ($request->hasFile('profile_photo')) {
        $file = $request->file('profile_photo');
        $filename = time() . '.' . $file->getClientOriginalExtension();
        $file->move(directory: public_path('up/profile_photos'), name: $filename);
        $authUser->profile_photo = $filename;
    }

    $authUser->save();

    return redirect()->route('profile.show')->with('success', 'Profile updated successfully.');
```

```

}

```

```

public function remove($friendId): RedirectResponse
{
    $authUserId = Auth::id();
    if (!$authUserId) {
        return redirect()->route('login')->with('error', 'You must be logged in to perform this action.');
```

Log::info('Deleting from wishlist where user_id=' . \$authUserId . ' and wishlist_user_id=' . \$friendId);

```

    Wishlist::where('user_id', $authUserId)->where('wishlist_user_id', $friendId)->delete();
    Wishlist::where('user_id', $friendId)->where('wishlist_user_id', $authUserId)->delete();

    return redirect()->route('profile.show')->with('message', 'Friend removed successfully.');
```

1 reference | 0 overrides

```

public function updateAvatar(Request $request): RedirectResponse
{
    $request->validate([
        'avatar_id' => 'required|exists:avatars,id',
    ]);

    $user = Auth::user();

    $user->avatar_id = $request->avatar_id;

    if ($request->has('replace_photo') && $request->replace_photo) {
        $user->profile_photo = null;
    }

    $user->save();

    return redirect()->route('profile.show')->with('success', 'Avatar updated successfully!');
```

Controller ini memungkinkan pengguna untuk melihat detail dari profil mereka, memperbarui informasi seperti foto profil atau avatar, serta mengelola hubungan pertemanan seperti menghapus ataupun melakukan chat.

Avatar Controller:

```

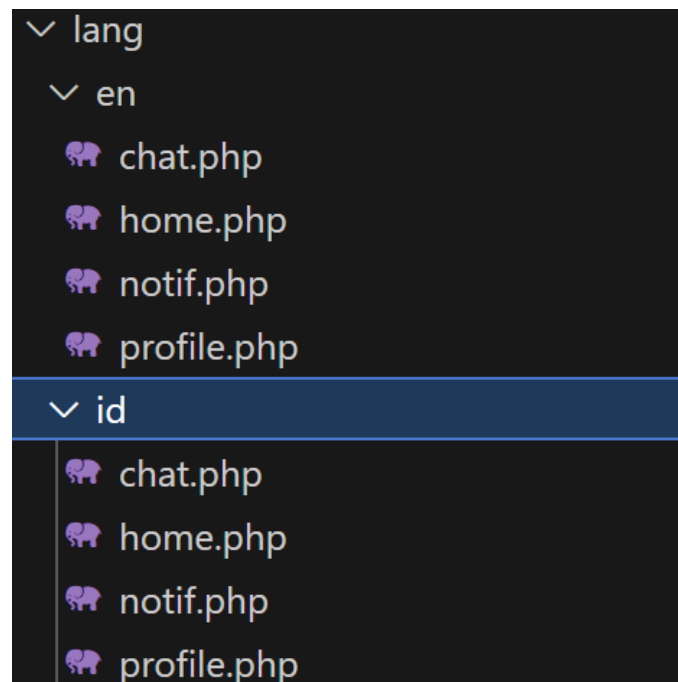
class AvatarController extends Controller
{
    1 reference | 0 overrides
    public function index(): View
    {
        $avatars = Avatar::all();
        return view('avatars.index', compact('avatars'));
    }

    1 reference | 0 overrides
    public function buy(Request $request): RedirectResponse
    {
        $avatar = Avatar::find($request->avatar_id);
        $user = Auth::user();

        if ($user->coins < $avatar->price) {
            return redirect()->back()->with('error', 'You do not have enough coins.');
```

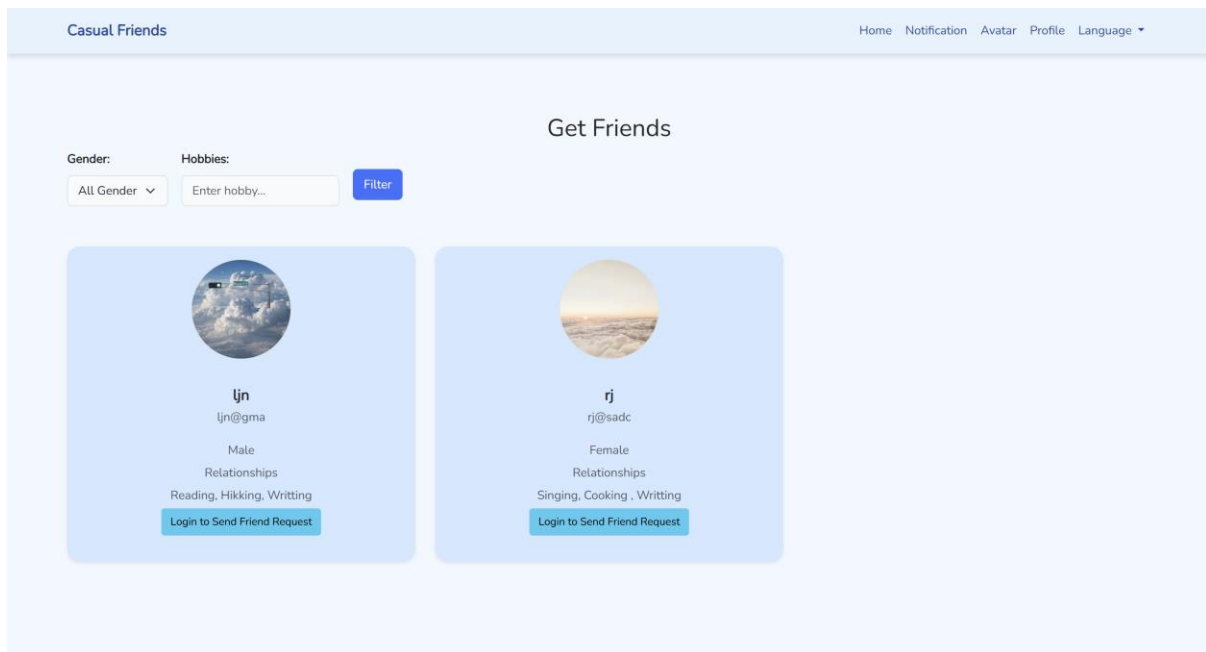
digunakan untuk mengelola fitur avatar dalam aplikasi, termasuk melihat daftar avatar yang tersedia dan membeli avatar baru.

Localization

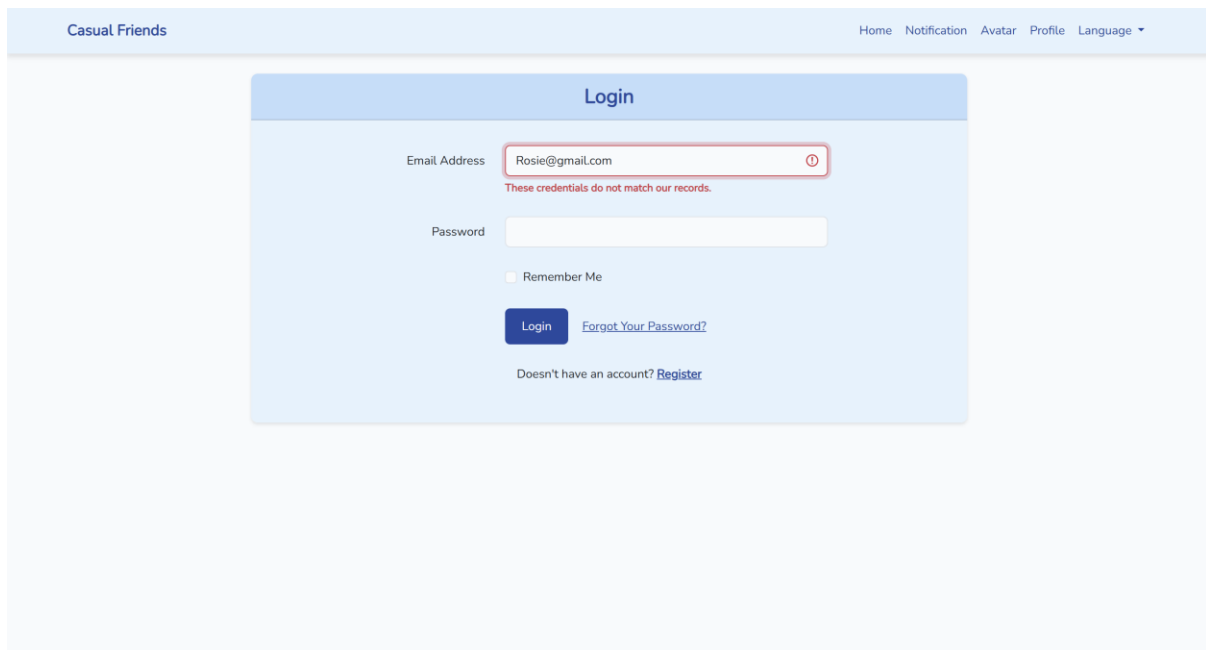


View

User Authentication



Home page akan terlihat begini jika belum login, semua option pada navbar tidak bisa diakses jika belum login. User tidak bisa mengirimkan request teman jika belum login.



Setelah diarahkan ke halaman login, user dapat login terlebih dahulu. Jika belum punya akun maka bisa menekan text register dibawah itu.

Selanjutnya akan diarahkan ke halaman register. User harus memasukan data sesuai ketentuan.
Setelahnya maka akan diarahkan ke halaman pembayaran untuk register

Payment

Jika kurang maka akan disebutkan underpaid

Jika berlebih maka akan dintanya apakah mau untuk di masukan ke e wallet atau masukan jumlah sesuai, jika no ma akan Kembali inut biaya nya dan akan di informasikan seperti dibawah ini

Casual Friends

Home Notification Avatar Profile Language ▾

Payment Page

Registration Fee
123143

Enter Amount Paid

You overpaid by 2.00. Please enter the correct amount.

Proceed to Payment

Namun jika memilih yes, maka akan lebihnya akan masuk kedalam e wallet

Casual Friends

Home Notification Avatar Profile Language ▾

Payment Successful

Your payment has been successfully processed. Thank you for registering!

Back to Home

Halaman ini akan muncul keteki pembayaran sudah berhasil. Dapat muncul dari persetujuan biaya berlebih yang akan masuk ke e wallet ataupun pembayaran yang pas.

Fitur Request Friends

Casual Friends

Home Notification Avatar Profile Language ▾

Get Friends

Gender: All Gender ▾ Hobbies: Enter hobby... Filter

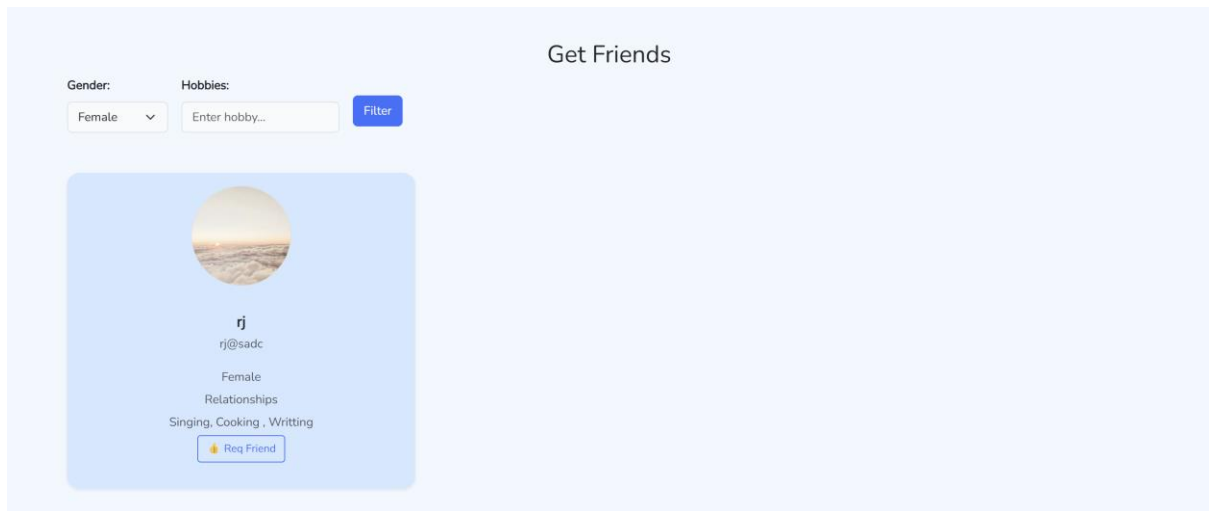
ljn
ljn@gma
Male
Relationships
Reading, Hiking, Writing
Req Friend

rj
rj@sadc
Female
Relationships
Singing, Cooking, Writing
Req Friend

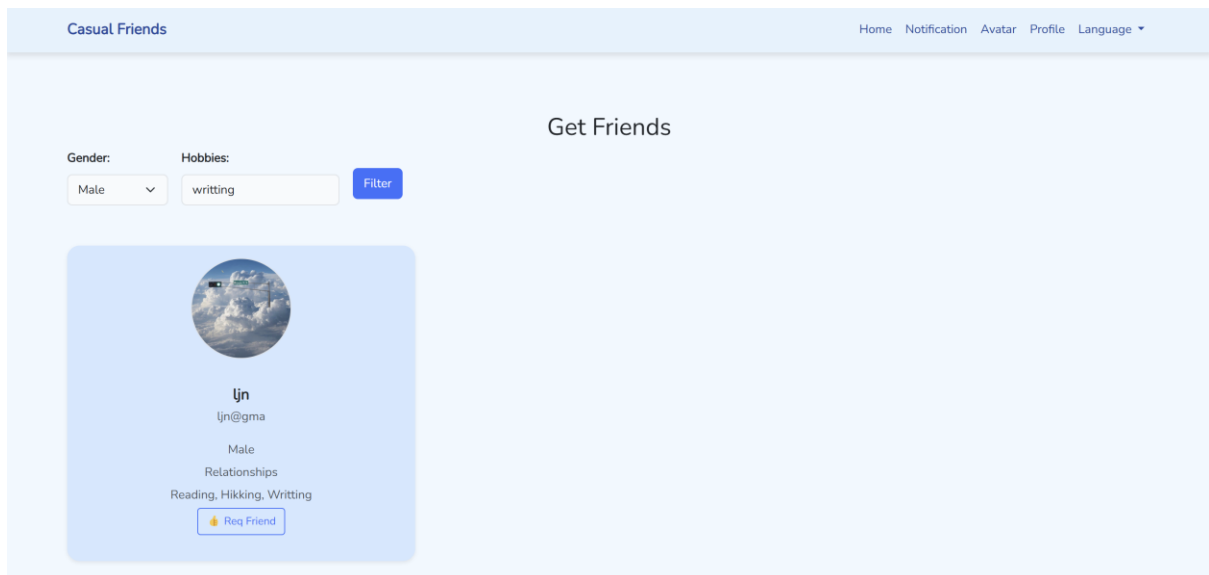
Setelah login maka fitur pada homepage sudah bisa di akses, kita dapat mengirim permintaan teman.

Filter/Searching

Filter Gender

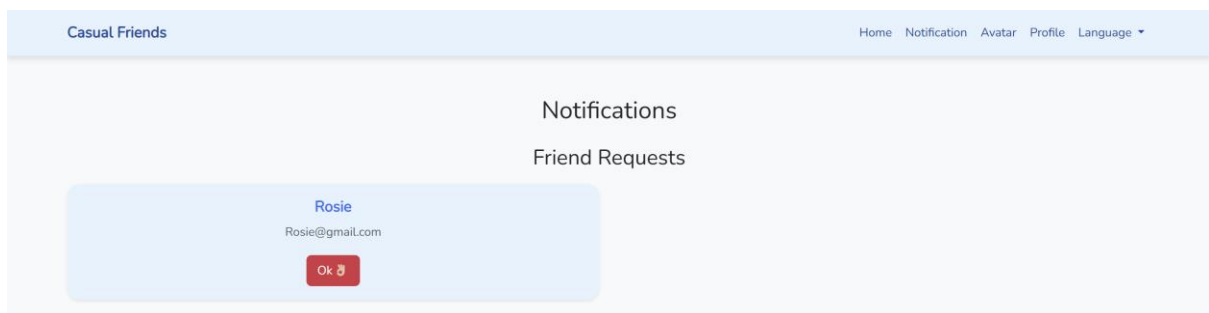


Searching 'writing' dengan gender 'Male'



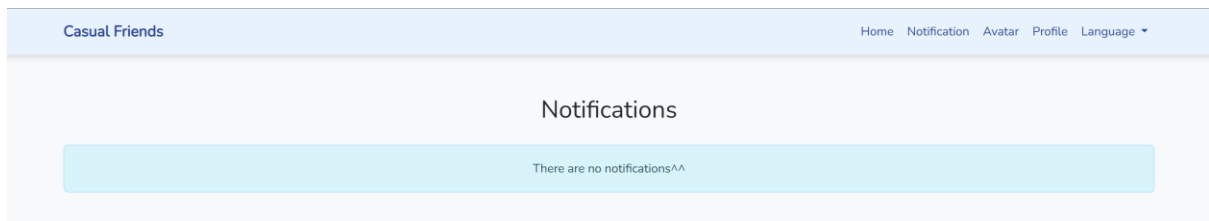


Status akan berubah menjadi requested Ketika friends request dikirimkan. Dan pada akun penerima akan ada notifikasi request friends

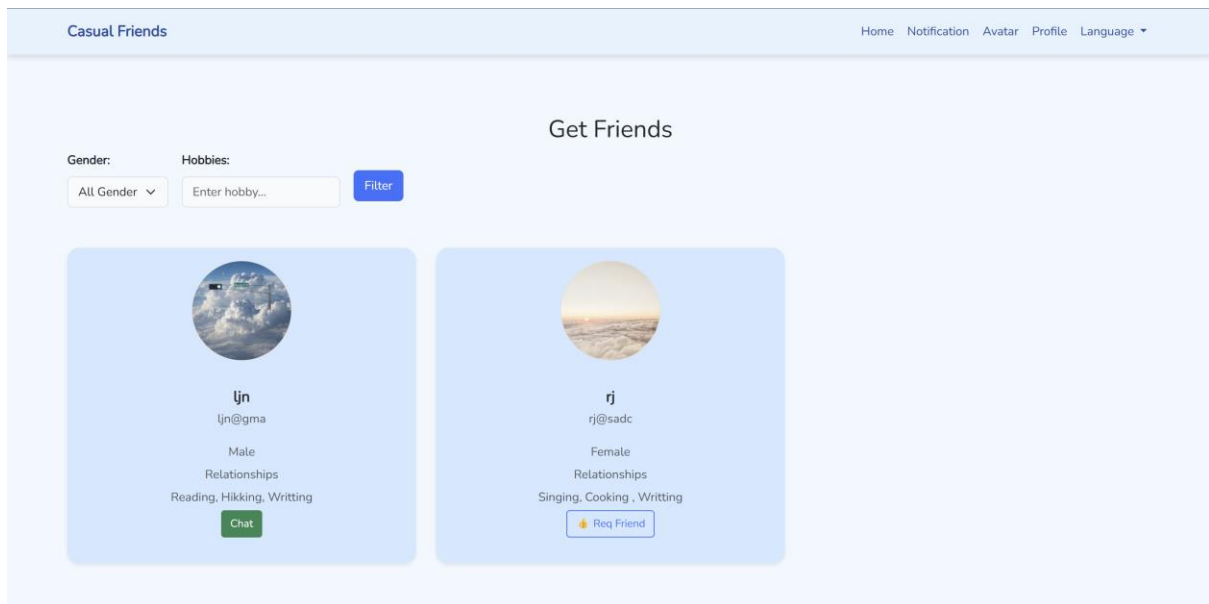


Namun ini hanya notifikasi, setelah menekan 'ok', penerima tetap harus menekan reg friends thumbs up pada home page agar dapat berteman

Notification

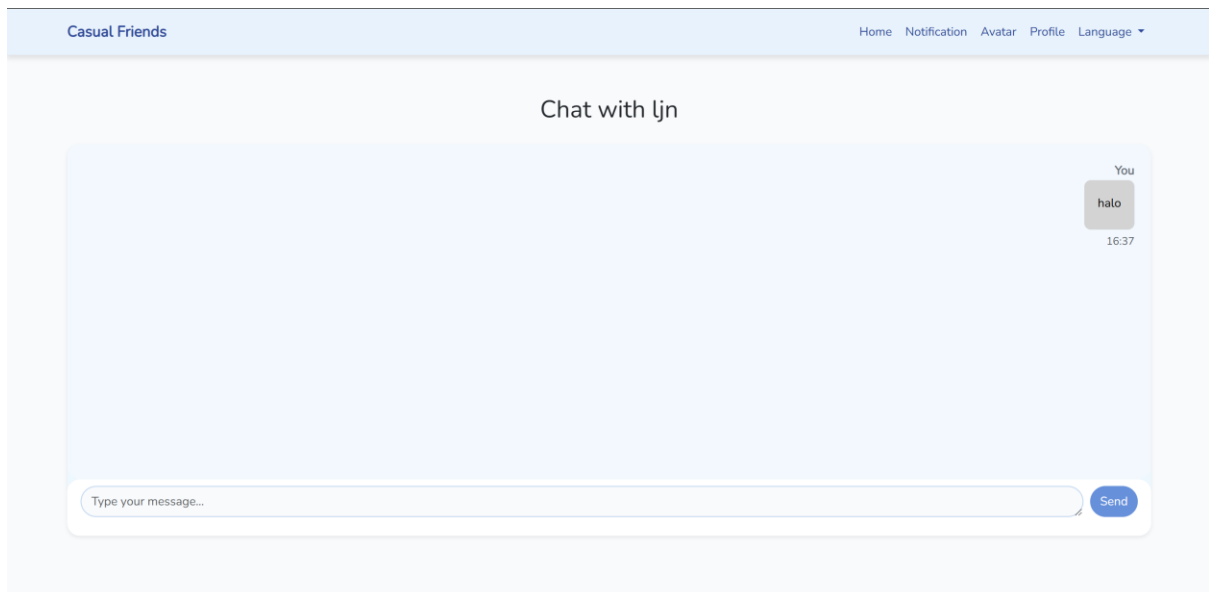


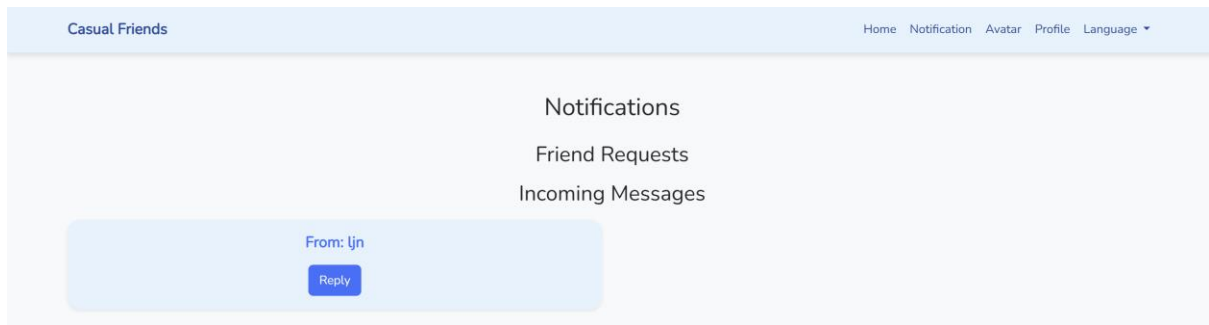
Notifikasi menjadi kosong setelah menekan button ok penerima juga menekan button thumbs maka sudah berteman dan bisa melakukan chat



memulai chat dan begini jadinya

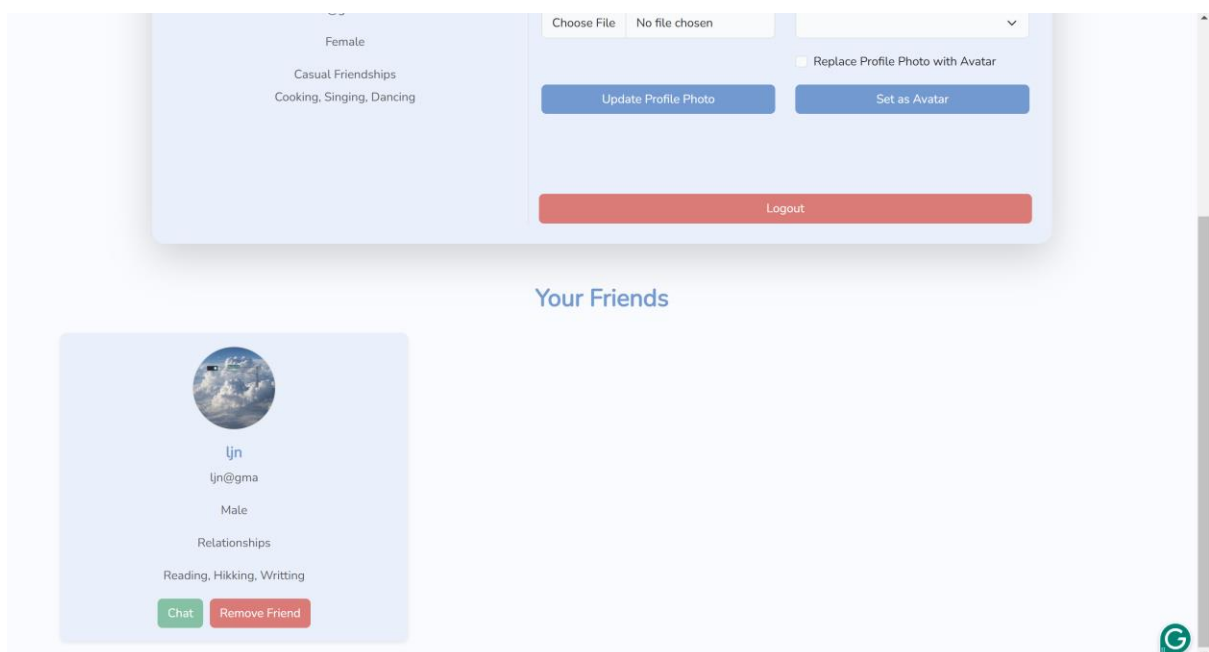
Messaging



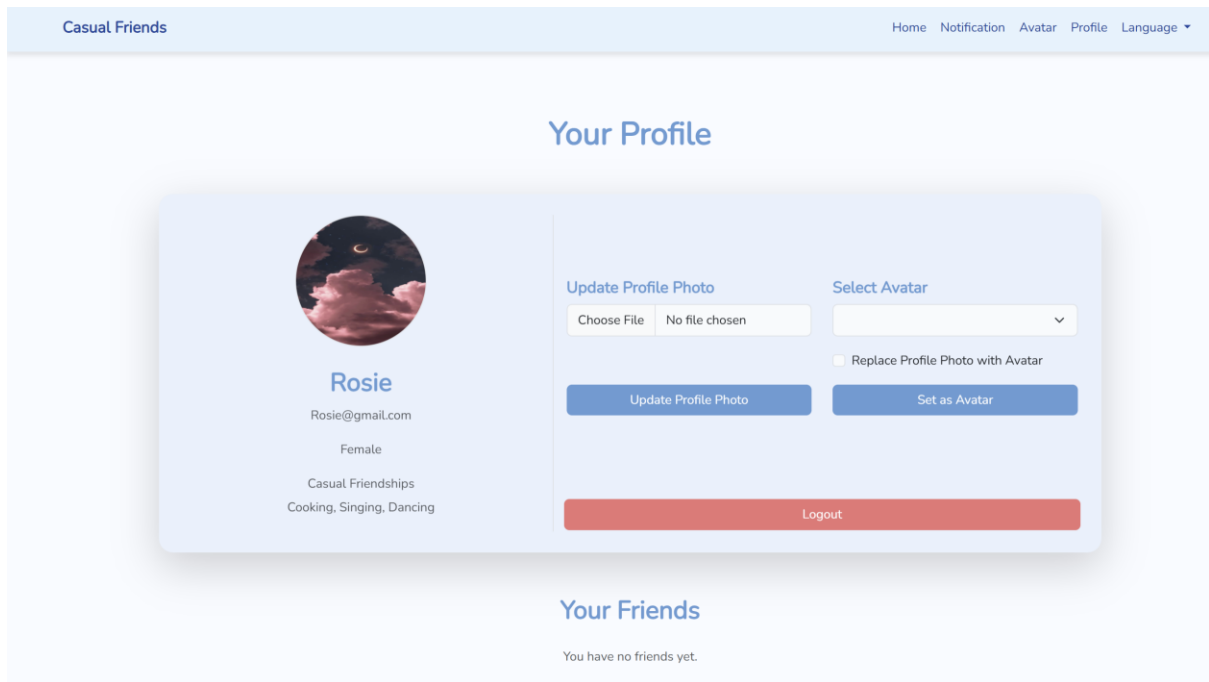


Munculah notifikasi pesan yang mana setelah kita membaca pesannya maka notifikasi akan hilang

Friends



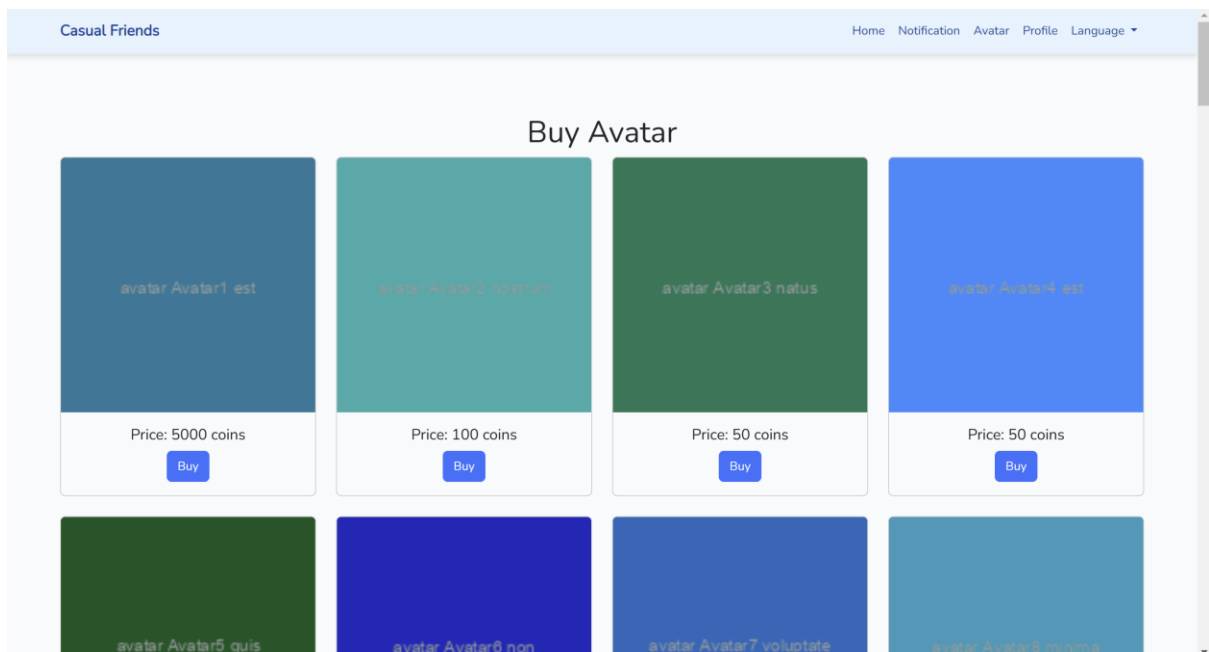
Muncul sudah nama teman yang berteman dengan kita pada profile page nya, kita dapat menghapus pertemenan juga



Ketika dihapus maka teman akan hilang. Kita juga dapat mengupdate photo profile

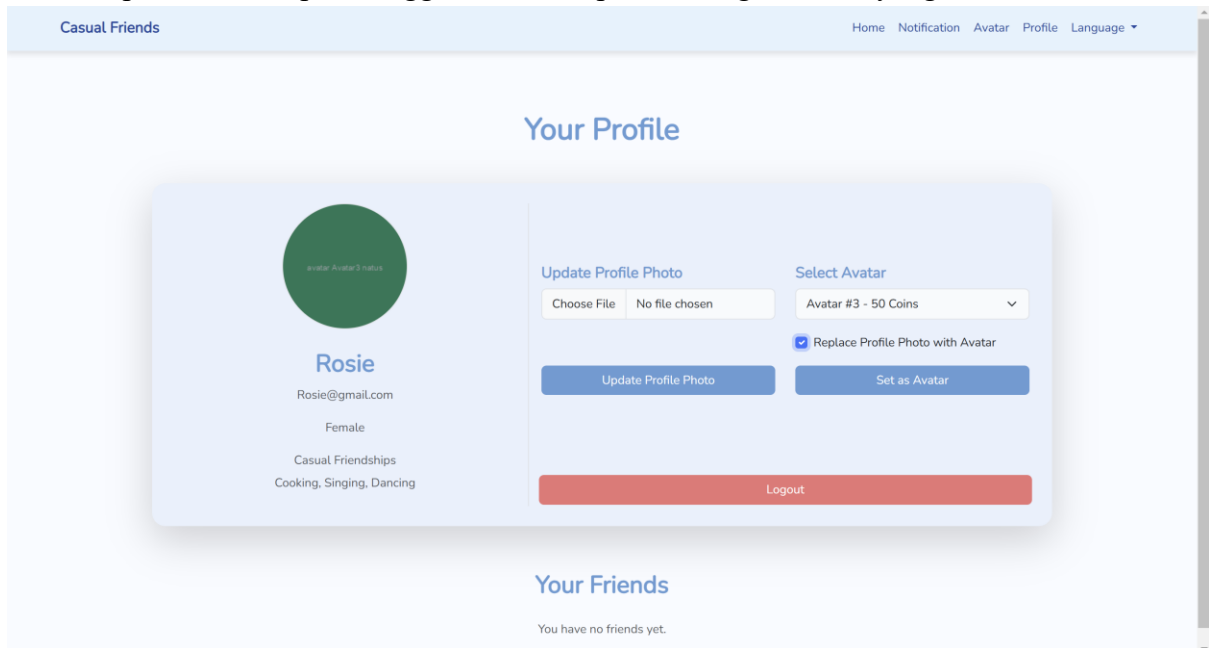
Fitur Tambahan

Membeli Avatar



Terdapat Halaman avatar Dimana merupakan tempat untuk kita membeli avatar dengan coin yang didapatkan saat awal mendaftar, yaitu 100 coin. Setelah membeli kita akan diarahkan ke

halaman profile dan dapat menggantikan foto profile dengan avatar yang telah kita beli



Dengan begitu maka foto profile kita menjadi avatar yang kita beli