



Higher Technological Institute in Beni Suef



Computer Systems and Networks Technology Department

POWER GYM

Submitted By Ra3d Team ⚡:

Mohammed Sayed ©

Hussien Anwar

Abdelrahman Mohammed

Nourhan Walid

Rawan Yousry

Youssef Abo Alhassan

Abdelrahman Hassan

Amira Ashraf

Nesreen Ahmed

Karim Thabt

Abdelrahman Boghdady

Supervised By:

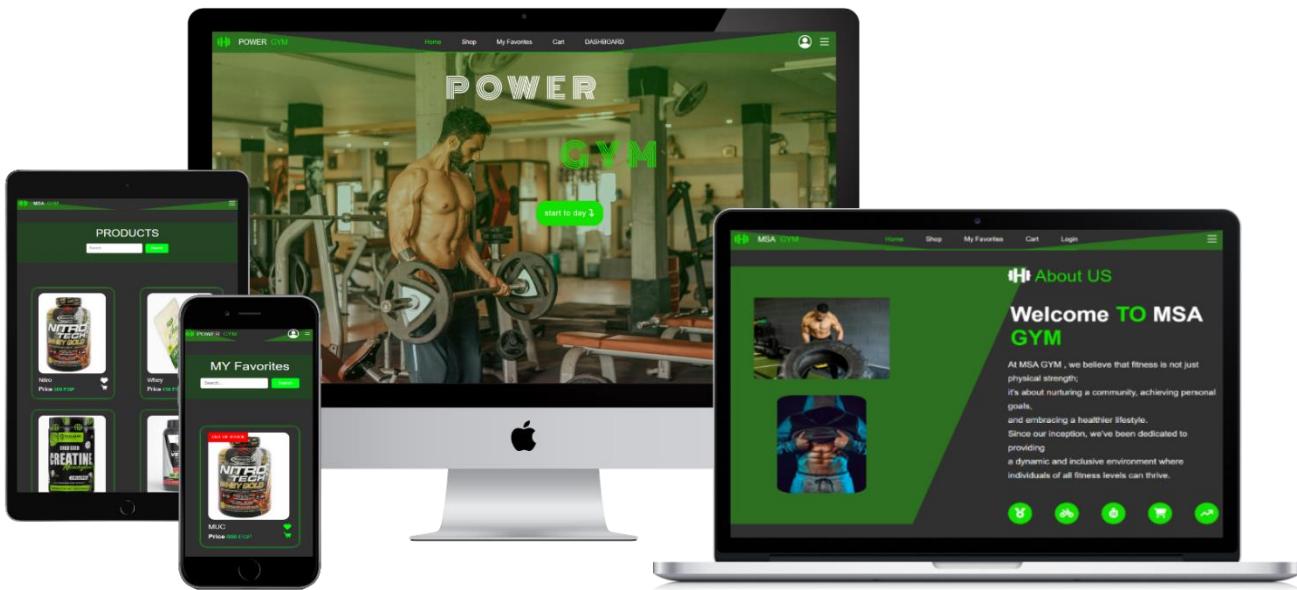
Dr. Hagar Mohamed

Assistant By:

Eng. Ahmed Hany



POWER GYM



[2024]

Content List:

Chapter 1: Introduction.	1
1,1. Website introduction.	
1,2. Languages.	
1,3. Objectives.	
1,4. Similar sites.	
Chapter 2: front end website.	14
2,1. Abstract about front end of the site.	
2,2. What are some popular genres in the Power Gym.	
2,3. Some ways Power Gym can promote their events.	
Chapter 3: future improvement.	21
3,1. Future improvement.	
3,2. Screen Shot {code & design}.	
3,3. Manual user.	
Chapter 4: back-end website.	42
4,1. Introduction back end.	
4,2. Web pages in uses.	
4,3. Screen Shot {code}.	
Chapter 5: database & conclusion.	58
5,1. Database & uses in site.	
5,2. Advantages & disadvantages.	
5,3. Conclusion of the project.	

Chapter 1

Introduction

Introduction:

Hello and welcome to the esteemed professors of. This topic is the result of extensive research and persistent effort. It has been meticulously developed with a significant investment of both time and energy, reflecting the dedication and hard work behind it. I sincerely hope for success in presenting this project and aim to cover all its aspects comprehensively during this presentation.

We will now proceed to review the key points of the first chapter.

1.1. Website introduction.

In today's rapidly advancing technological world, websites have become an essential part of the strategies of various organizations, including gyms. This graduation project aims to develop a comprehensive website for "Power Gym" that aims to enhance customer experience and facilitate access to the diverse services offered by the gym.

"Power Gym" is a modern fitness center that provides a wide range of activities and sports services designed to meet the needs of all members, from beginners to professional athletes. Through this website, we aim to create an interactive platform that makes it easier for members to book training sessions, get to know professional trainers, and explore special monthly offers and pricing.

This book provides a detailed explanation of the different sections that make up the website, starting from the main section to the pricing information and special monthly training packages. Below, we will explore each section in detail and how it contributes to achieving the desired objectives

1,2. Languages.

In the ever-evolving world of web development, a variety of languages and frameworks are utilized to create dynamic, responsive, and efficient websites. This article delves into the core technologies that form the backbone of modern web development: HTML, CSS, JavaScript, Bootstrap, PHP, Laravel, and MySQL. Understanding these tools is essential for anyone aspiring to build robust web applications.

HTML: The Structure of Web Pages

Hypertext Markup Language (HTML) is the standard language used to create the structure of web pages. It serves as the skeleton of a website, defining elements such as headings, paragraphs, links, images, and other content. HTML uses a system of tags to enclose content and provide semantic meaning to the text.

HTML5, the latest version of HTML, introduces new elements and attributes that provide greater flexibility and functionality. Features such as the `<header>`, `<footer>`, `<article>`, and `<section>` tags help structure content more meaningfully, enhancing both accessibility and SEO.

CSS: Styling the Web

Cascading Style Sheets (CSS) is the language used to style and layout web pages. While HTML provides the structure, CSS is responsible for the visual presentation. It controls the colors, fonts, spacing, layout, and overall look and feel of a website.

CSS3, the latest evolution of CSS, brings powerful new features such as transitions, animations, and flexible box layouts (Flexbox), allowing developers to create responsive and visually appealing designs. Media queries, another key feature of CSS3, enable websites to adapt their layout to different screen sizes and devices, a practice known as responsive design.

JavaScript: Adding Interactivity

JavaScript (JS) is a powerful scripting language that enables interactivity and dynamic content on web pages. Unlike HTML and CSS, which are static, JavaScript allows developers to create dynamic content that can respond to user actions, such as form submissions, button clicks, and real-time updates.

JavaScript can be used for a variety of tasks, from simple form validation to complex animations and web applications. With the advent of ES6 (ECMAScript 2015) and beyond, JavaScript has seen significant improvements, including new syntax, features like classes and modules, and better performance.

Bootstrap: Streamlining Responsive Design

Bootstrap is a popular front-end framework that simplifies the development of responsive and mobile-first websites. It provides a collection of CSS and JavaScript components, such as navigation bars, modals, and carousels, that can be easily integrated into web projects.

Bootstrap's grid system is one of its standout features, offering a flexible and responsive layout structure that adjusts to different screen sizes. By using pre-designed components and utilities, developers can quickly create consistent and visually appealing interfaces without writing extensive custom CSS.

PHP: Server-Side Scripting

PHP (Hypertext Preprocessor) is a widely-used open-source scripting language suited for web development. It runs on the server-side, meaning the code is executed on the server before being sent to the client's browser. PHP is particularly known for its efficiency in creating dynamic web pages and its ease of integration with databases.

One of the strengths of PHP is its extensive ecosystem of frameworks and libraries, which streamline development and reduce the amount of boilerplate code required. It is also highly compatible with a variety of databases, including MySQL, making it a versatile choice for web developers.

Laravel: The PHP Framework

Laravel is a powerful PHP framework designed to make web development more efficient and enjoyable. It follows the Model-View-Controller (MVC) architectural pattern, promoting a clean separation of concerns and making code more maintainable and scalable.

Laravel comes with a host of features out of the box, such as routing, authentication, and templating, which simplify common web development tasks. Its elegant syntax and comprehensive documentation have made it a favorite among PHP developers. Laravel also includes Eloquent ORM, a powerful tool for interacting with databases in an object-oriented manner.

MySQL: Managing Data

MySQL is a relational database management system (RDBMS) that is widely used in web development to store and manage data. It uses Structured Query Language (SQL) to perform database operations such as querying, inserting, updating, and deleting data.

MySQL is known for its performance, reliability, and ease of use. It can handle large amounts of data and is used in many high-traffic websites and applications. The combination of PHP and MySQL is a popular choice for building dynamic and data-driven websites due to their compatibility and ease of integration.

Conclusion

The development of the "Power Gym" website showcases the powerful synergy of various web technologies to create a seamless and engaging user experience. HTML and CSS lay the structural and stylistic foundations, ensuring the site is both functional and visually appealing. JavaScript adds interactivity, making the site dynamic and responsive to user inputs, while Bootstrap simplifies the design process and ensures that the website looks great on all devices.

On the server side, PHP and the Laravel framework provide a robust and scalable backend, handling everything from user authentication to database interactions with efficiency and security. MySQL serves as the backbone for data storage, ensuring that all user information, bookings, and gym data are managed effectively.

Together, these technologies create a comprehensive and efficient platform for "Power Gym," enabling it to offer its members a superior online experience. The website not only facilitates easy access to gym schedules, trainer information, and pricing plans but also strengthens the connection between the gym and its members, fostering a community centered around fitness and wellness.

As we move forward, the "Power Gym" website will continue to evolve, leveraging the latest advancements in web development to meet the ever-changing needs of its users. This project exemplifies the potential of combining various web technologies to create a powerful digital solution that enhances both user satisfaction and operational efficiency.

The successful implementation of this website is a testament to the dedication and hard work invested in mastering these technologies. It also highlights the transformative impact that a well-designed web application can have on a business, setting a new standard for digital interaction in the fitness industry. With this foundation in place, "Power Gym" is well positioned to thrive in the digital age, offering its members a convenient, interactive, and engaging online experience.

1,3. Objectives.

The primary objective of the "Power Gym" website project is to create an engaging and user-friendly platform that enhances the overall experience for gym members and potential clients. By leveraging a combination of modern web technologies, we aim to achieve the following specific goals:

Enhanced User Experience:

Create an intuitive and responsive design that provides a seamless experience across all devices.

Efficient Content Management:

Enable easy updates and management of content, including class schedules, trainer profiles, and pricing plans.

Secure User Authentication and Data Management:

Ensure secure login, user authentication, and safe handling of personal data.

Scalable and Maintainable Architecture:

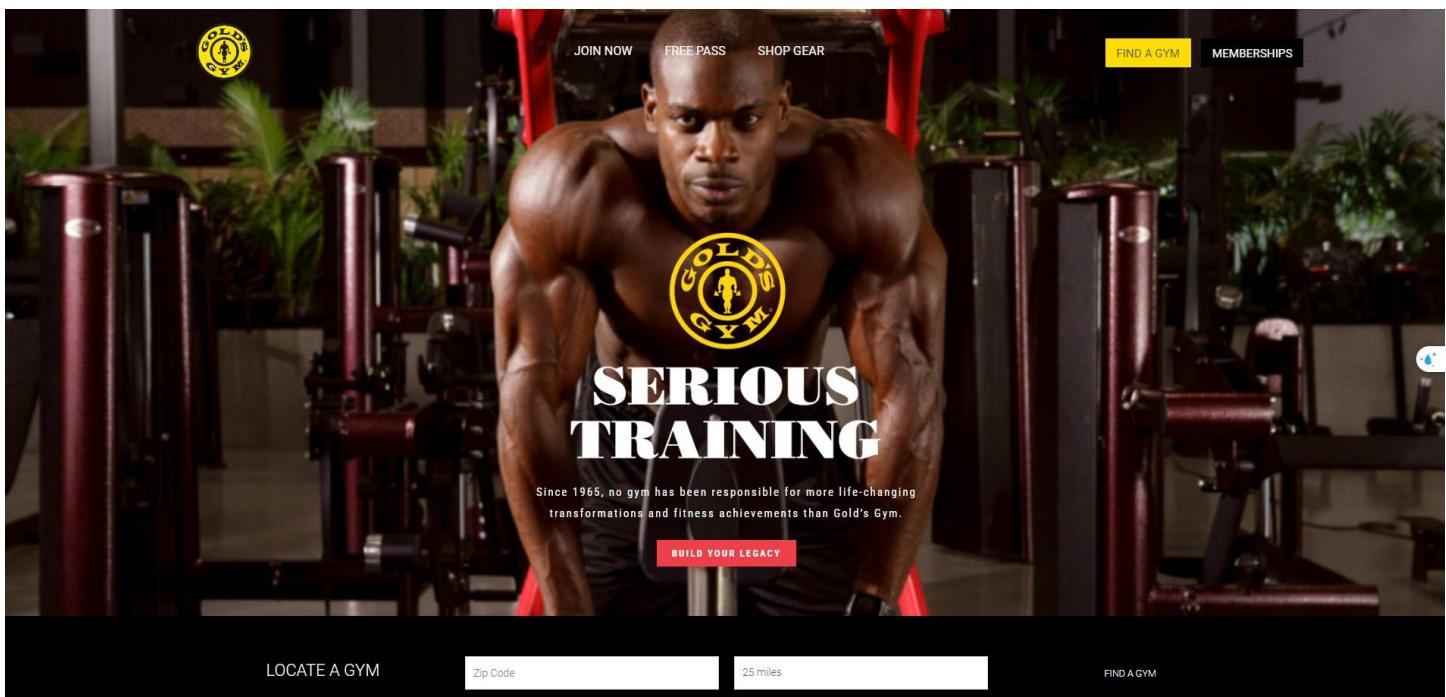
Develop a scalable architecture that can handle growing user numbers and new features in the future.

Improved Communication and Community Building:

Facilitate better communication between the gym and its members through contact forms, newsletters, and event announcements.

1.4. Similar sites.

1-Gold's Gym



Overview:

Gold's Gym is one of the most iconic fitness brands globally, known for its extensive facilities, comprehensive fitness programs, and a strong community focus. Established in 1965, Gold's Gym has grown to become a leader in the fitness industry, offering a wide range of services designed to help members achieve their fitness goals.

Features:

Online Class Schedules: Members can easily view and book classes online, ensuring they can plan their workouts around their schedules.

Membership Information: Detailed information about various membership plans, including pricing, benefits, and promotional offers.

Personal Training Options: Access to professional personal trainers who provide customized workout plans and guidance.

2-Planet Fitness

The screenshot shows the Planet Fitness website homepage. At the top, there's a navigation bar with links for "Find a Club", "About Planet Fitness", "My Account", "English", "Search", and a yellow "Join Us" button. The main banner features a purple background with a man on a stationary bike giving a thumbs up and a woman on a treadmill smiling. The text "JOIN THE JUDGEMENT FREE ZONE®" and "WITH MEMBERSHIPS STARTING AS LOW AS \$10 A MONTH" is prominently displayed. Below the banner, there's a search bar and a "Join Us" button. The section below is titled "MEMBERSHIPS" and describes two types: "PF BLACK CARD®" (Best Value) and "Classic".

MEMBERSHIPS

We offer the PF Black Card® Membership and Classic Membership. Both get you access to **The Judgement Free Zone®**, and tons of cardio and strength equipment.

PF BLACK CARD® **Best Value**
Starting at

Classic
Starting at

Overview:

Planet Fitness is known for its welcoming and non-intimidating environment, branded as the "Judgment Free Zone." With affordable membership plans and a focus on creating a comfortable workout atmosphere, Planet Fitness appeals to a broad audience, including beginners and casual gym-goers.

Features:

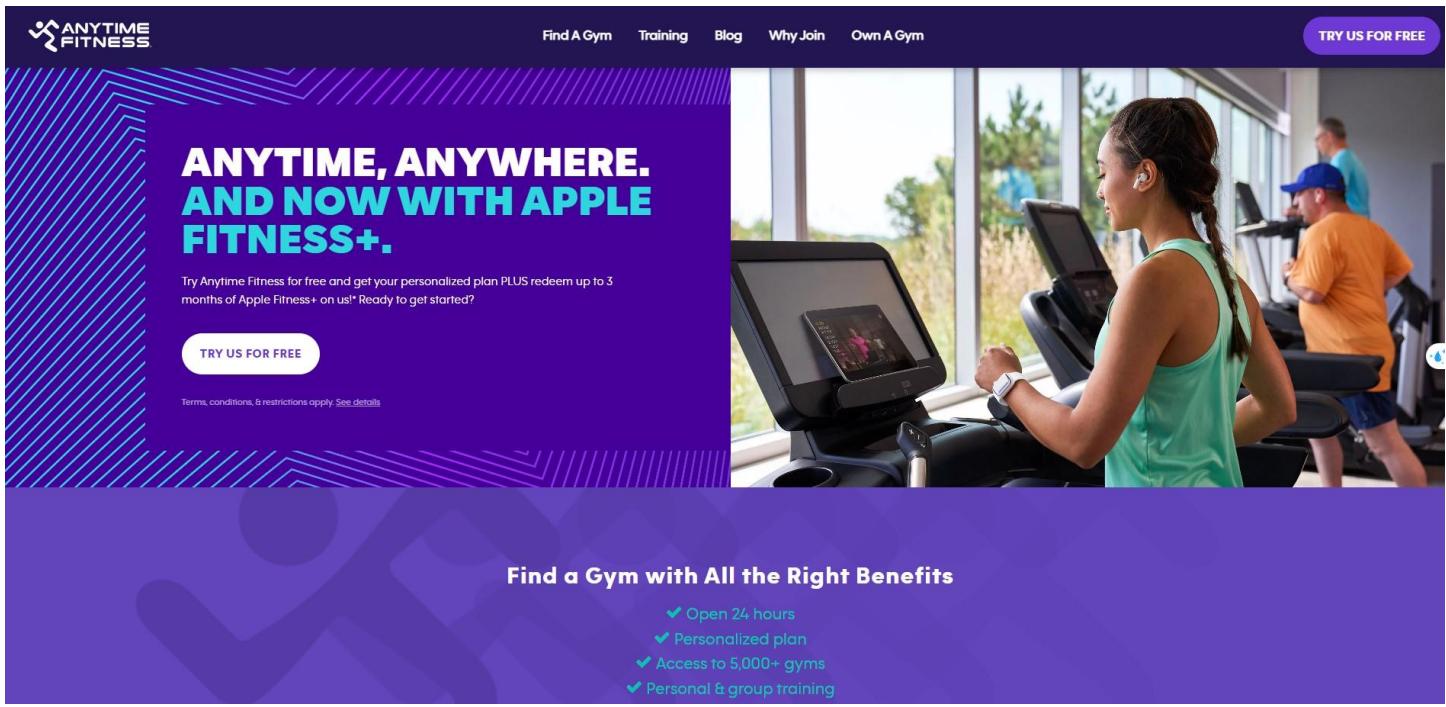
Membership Deals: Regular promotions and deals to attract new members and retain existing ones.

Virtual Tours: Online tours of gym facilities, helping potential members get a feel for the environment before joining.

Workout Videos: A library of workout videos that members can follow from home or at the gym.

Member Login: Secure login for members to manage their accounts, view billing information, and track their workout history.

3- Anytime Fitness:

The image shows the homepage of the Anytime Fitness website. At the top, there's a dark header with the "ANYTIME FITNESS" logo, navigation links for "Find A Gym", "Training", "Blog", "Why Join", and "Own A Gym", and a prominent "TRY US FOR FREE" button. Below the header is a large purple banner with the text "ANYTIME, ANYWHERE. AND NOW WITH APPLE FITNESS+." in white and blue. It also includes a small description about a free trial and a "TRY US FOR FREE" button. To the right of the banner is a photograph of a woman running on a treadmill in a gym. Below the banner, there's a section titled "Find a Gym with All the Right Benefits" followed by a list of four benefits with checkmarks.

Overview:

Anytime Fitness is known for its 24/7 access to fitness facilities, making it a popular choice for individuals with varying schedules. With over 4,000 locations worldwide, Anytime Fitness offers a consistent and reliable gym experience wherever members are located

Features:

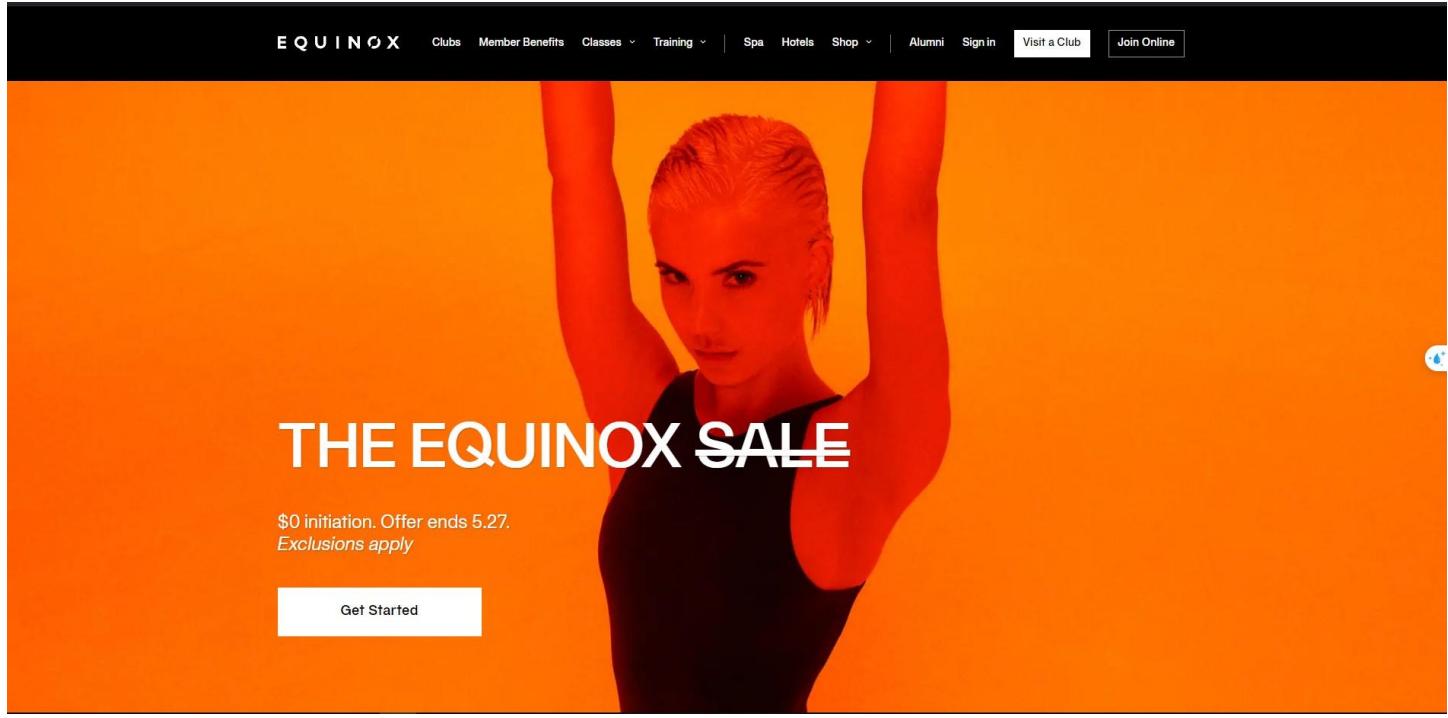
Gym Locator: An easy-to-use tool to find the nearest Anytime Fitness location.

Membership Plans: Detailed information about different membership options and pricing.

Personal Training: Access to certified personal trainers for customized fitness plans.

Fitness Tips: A blog and resource center with fitness tips, workout routines, and health advice.

4- Equinox:



Overview:

Equinox is a high-end fitness brand known for its luxurious facilities, personalized services, and innovative fitness programs. Catering to a premium clientele, Equinox focuses on delivering a holistic wellness experience that combines fitness, spa services, and lifestyle amenities.

Features:

Luxury Facilities: State-of-the-art gyms with top-notch equipment and amenities, including spas and juice bars.

Personal Training: Access to highly qualified personal trainers who offer customized workout plans and one-on-one sessions.

Group Fitness Classes: A wide range of classes, from yoga and Pilates to high-intensity interval training (HIIT).

Digital Platform: An advanced digital platform offering on-demand workouts, wellness advice, and virtual classes.

Chapter 2

Front-End

Website

2,1. Abstract about front end of the site.

The front end of a Power Gym website serves as the gateway for visitors to explore the gym's offerings and services. Designed to engage and inform, this section welcomes visitors with captivating visuals and intuitive navigation, enticing them to discover more about what the gym has to offer.

Similar to a well-curated storefront, the front end of the Power Gym website features prominently displayed sections dedicated to key aspects of the gym experience. Here, visitors can explore profiles of experienced coaches, showcasing their expertise and specialties. Engaging content highlights the unique training methodologies and personalized approach that set Power Gym apart.

As visitors delve deeper into the website, they encounter dedicated sections for exploring membership packages tailored to their fitness goals and preferences. Interactive features allow users to compare packages, view pricing options, and easily sign up for membership online. The layout is designed for clarity and ease of use, ensuring that visitors can quickly find the information they need to make informed decisions about their fitness journey.

In addition to access to coaches and membership packages, the front end of the Power Gym website provides convenient avenues for visitors to get in touch. The contact section features user-friendly forms and contact information, inviting visitors to reach out with inquiries or schedule consultations with coaches.

2.2. What are some popular genres in the book store.

In the context of the Power Gym website, where users explore various aspects of fitness and gym services, the "genres" would refer to categories or sections that users frequently visit or interact with. Let's categorize these sections as follows:

1. **Coaches:** Users can explore profiles of experienced coaches, view their specialties, and learn about their training methodologies.
2. **Packages:** This section showcases different membership packages tailored to users' fitness goals and preferences, allowing them to compare options and choose the one that best suits their needs.
3. **Contact:** Users can find contact information and user-friendly forms to inquire about gym services, schedule consultations with coaches, or address any inquiries or concerns.
4. **Shop:** Here, users can browse and purchase gym-related products, equipment, apparel, supplements, and accessories directly through the website.

5. **Favorites:** Users have the option to save their favorite coaches, membership packages, or products for easy access and reference in future visits.
6. **Cart:** This section displays items that users have added to their cart while shopping on the website, allowing them to review and manage their selections before completing their purchases.
7. **Login/Register:** These sections facilitate user authentication and account creation. Users can log in to access personalized content, view their order history, or manage their account settings. Alternatively, new users can register for an account to unlock additional features and benefits.

These sections cater to different user needs and preferences, providing a comprehensive and user-friendly experience for individuals interested in exploring fitness options, connecting with coaches, purchasing gym products, or accessing personalized services.

2.2. Some ways book stores can promote their events.

Power Gym can employ various strategies to effectively promote their events and engage their target audience. Here are some ways they can do so:

- 1. Social Media Marketing:** Utilize platforms like Facebook, Instagram, Twitter, and LinkedIn to create event pages, share event details, post engaging content, and run targeted advertising campaigns. Encourage followers to RSVP, share the event with their networks, and use event-specific hashtags to generate buzz and increase visibility.
- 2. Email Campaigns:** Send out targeted email campaigns to existing members and subscribers, highlighting upcoming events, workshops, classes, or special promotions. Personalize the content based on the recipient's interests and preferences, and include compelling visuals, event details, and calls to action to encourage attendance.
- 3. Website Promotion:** Feature upcoming events prominently on the Power Gym website's homepage or events page. Provide detailed information about each event, including date, time, location, description, and registration instructions. Incorporate eye-catching visuals, testimonials, and past event highlights to capture visitors' attention and entice them to learn more.

4. In-Person Promotion: Display posters, banners, or flyers in high-traffic areas within the gym to inform members about upcoming events. Train staff members to actively promote events to gym-goers during their visits, and encourage them to share event details with friends, family, or colleagues who might be interested.

5. Collaborations and Partnerships: Partner with local businesses, influencers, fitness professionals, or community organizations to co-host events, cross-promote each other's services, and expand the reach of promotional efforts. Collaborators can help spread the word about events to their respective networks, increasing exposure and attendance.

6. Online Advertising: Consider investing in paid advertising channels such as Google Ads, Facebook Ads, or Instagram Ads to target specific demographics, interests, or geographic locations. Use compelling ad creatives, persuasive copy, and clear calls to action to drive traffic to event registration pages or landing pages.

7. Referral Programs: Incentivize existing members to refer friends, family, or colleagues to attend events at Power Gym. Offer rewards, discounts, or exclusive perks for successful referrals or for bringing a certain number of guests to events. Word-of-mouth referrals can be a powerful driver of event attendance and brand awareness.

8. Community Engagement: Engage with the local community by hosting free fitness classes, workshops, or wellness events open to the public. Collaborate with local schools, businesses, or community centers to reach a broader audience and foster positive relationships with potential members.

By leveraging a combination of these promotional strategies, **Power Gym** can effectively raise awareness, drive attendance, and create memorable experiences for participants at their events.

Chapter 3

Future

Improvement

3.1. Future Improvement.

For Power Gym, continual improvement is key to staying competitive and meeting the evolving needs of its members. Here are some future improvement areas that Power Gym could consider:

- 1. Enhanced Personalization:** Invest in technology and data analytics to better understand members' preferences, fitness goals, and workout habits. Utilize this information to tailor personalized recommendations, workout plans, and class schedules for individual members, enhancing their overall experience and satisfaction.

- 2. Virtual Training and Classes:** In response to changing consumer preferences and lifestyle demands, consider expanding virtual training options and offering online fitness classes. Develop a user-friendly platform or mobile app that allows members to access live or on-demand workouts, receive virtual coaching, and track their progress from anywhere, at any time.

3. Advanced Equipment and Facilities: Regularly update and upgrade gym equipment and facilities to incorporate the latest advancements in fitness technology and design. Introduce innovative equipment, such as smart fitness machines, wearable devices, or immersive workout experiences, to provide members with cutting-edge training options and keep them engaged.

4. Nutritional Guidance and Support: Recognizing the importance of nutrition in achieving fitness goals, offer members access to nutritional counseling, meal planning services, and healthy eating resources. Partner with nutritionists, dietitians, or wellness experts to provide personalized guidance and support, helping members make informed choices and maintain a balanced lifestyle.

5. Community Building Initiatives: Foster a strong sense of community and belonging among members by organizing social events, fitness challenges, and group activities. Create online forums, social media groups, or virtual communities where members can connect, share experiences.

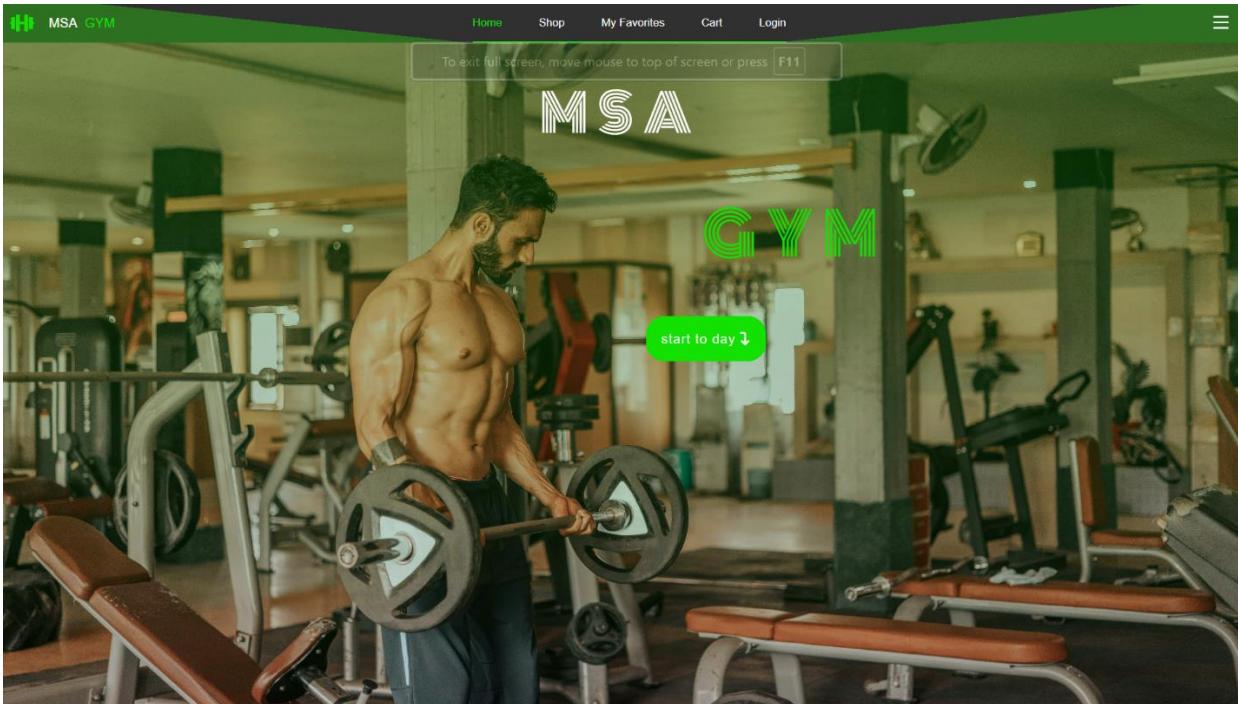
6. Wellness Programs and Services: Expand beyond traditional fitness offerings to encompass holistic wellness programs and services that address mental, emotional, and spiritual well-being. Introduce mindfulness classes, meditation sessions, stress management workshops, and other wellness initiatives to help members achieve balance and overall wellness.

7. Environmental Sustainability: Embrace sustainability initiatives and eco-friendly practices to reduce the gym's environmental footprint and promote environmental stewardship. Implement energy-efficient technologies, reduce waste generation, and incorporate sustainable materials in facility design and operations. Educate members about the importance of sustainability and encourage eco-conscious behaviors within the gym community.

8. Continuous Feedback and Improvement: Establish channels for gathering feedback from members on their experiences, preferences, and suggestions for improvement. Regularly solicit input through surveys, focus groups, or feedback forms, and use this information to identify areas for enhancement and prioritize future improvement efforts.

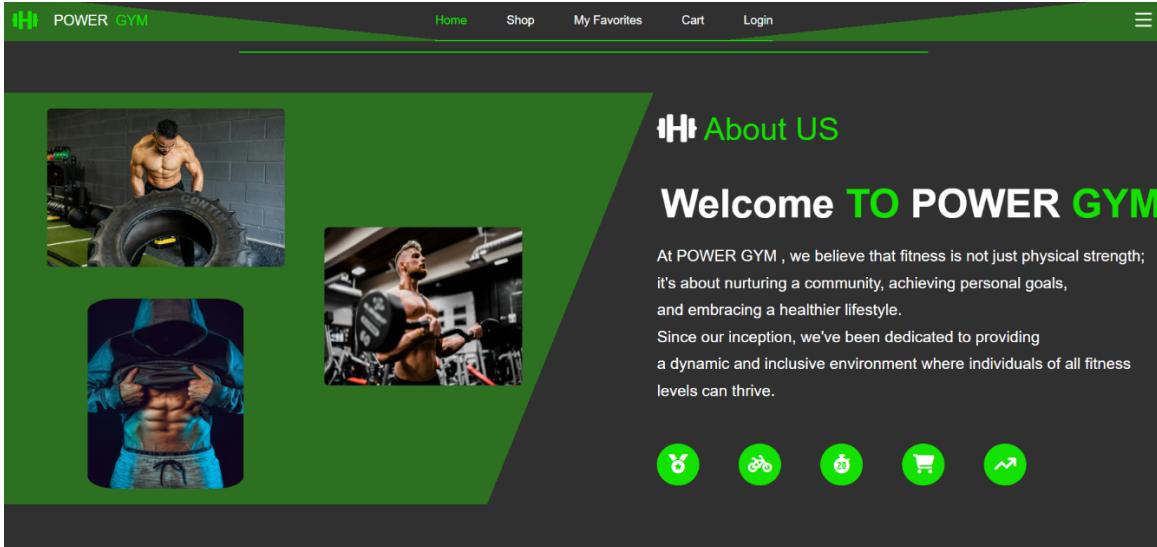
3,2. Screen Shot {Code & Design}.

Welcome Section



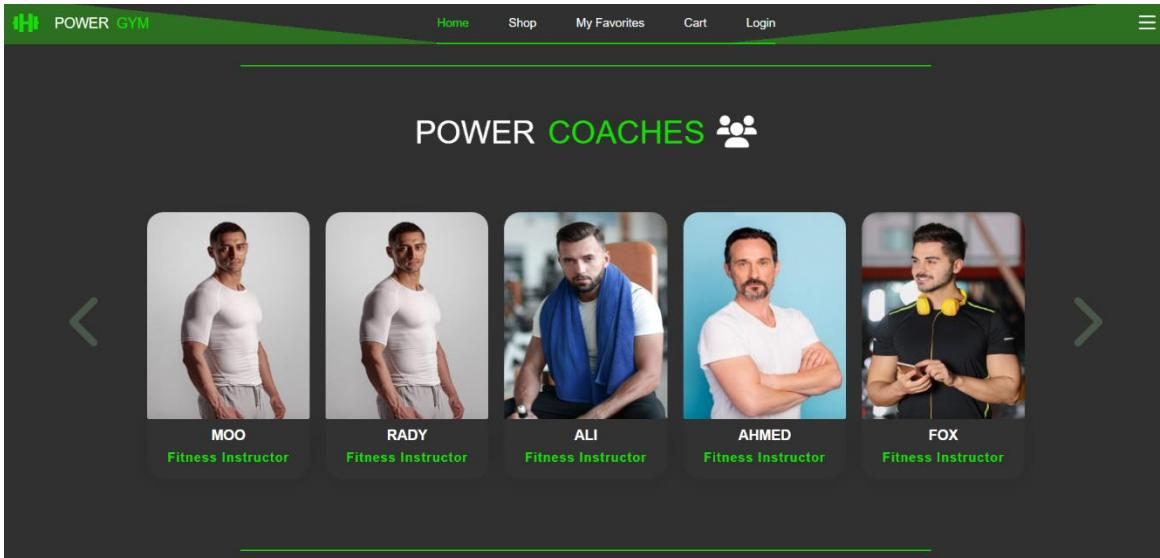
```
•••
<!-- Welcome -->
<div class="container" >
  <div class="start">
    <h1 class="monoton-regular">{{\$LogoName->firstname}}</h1>
    <h2 class="monoton-regular">{{\$LogoName->secondname}}</h2>
    <a href="#start">Start To Day <i class="fa-solid fa-arrow-turn-down" style="color: #ffffff;"></i></a>
  </div>
</div>
```

About Us Section



```
      . . .
<!-- About Us Section--&gt;
&lt;section&gt;
    &lt;div class="about"&gt;
        &lt;div class="one"&gt;
            &lt;div class="oo"&gt;
                &lt;img src="{{ asset('Files/images/AboutUs/138f43c3-93f6-4653-aa78-5b7c4c549971.jpg') }}" style="width: 300px; height: 200px; margin: 20px; border-radius: 5px;"&gt;
                &lt;img src="{{ asset('Files/images/AboutUs/7e430418-6cd8-4dcd-919c-290dcabb31ff.jpg') }}" class="three"&gt;
            &lt;/div&gt;
            &lt;div class="ooo"&gt;
                &lt;img src="{{ asset('Files/images/AboutUs/7e4e0b2d-cba2-461b-aba7-2e3204ba614f.jpg') }}" class="two"&gt;
            &lt;/div&gt;
        &lt;/div&gt;
        &lt;div class="text"&gt;
            &lt;h3 class="bebas-neue-regular"&gt;&lt;i class="fa-solid fa-dumbbell" style="color: rgb(255, 255, 255);&gt;&lt;/i&gt; About US &lt;/h3&gt;
            &lt;br&gt;
            &lt;br&gt;
            &lt;h1 class="bebas-neue-regular"&gt;Welcome &lt;span&gt;TO&lt;/span&gt; {{ $LogoName-&gt;firstname }} &lt;span&gt;{{ $LogoName-&gt;secondname }}&lt;/span&gt;&lt;/h1&gt;
            &lt;p&gt;At {{ $LogoName-&gt;firstname }} {{ $LogoName-&gt;secondname }}, we believe that fitness is not just physical strength; it's about nurturing a community, achieving personal goals, and embracing a healthier lifestyle. Since our inception, we've been dedicated to providing a dynamic and inclusive environment where individuals of all fitness levels can thrive.&lt;/p&gt;
        &lt;center class="about_icons"&gt;
            &lt;div class="circle"&gt;
                &lt;i class="fa-solid fa-medal fa-xl"&gt;&lt;/i&gt;
            &lt;/div&gt;
            &lt;div class="circle"&gt;
                &lt;i class="fa-solid fa-bicycle fa-xl"&gt;&lt;/i&gt;
            &lt;/div&gt;
        &lt;/center&gt;
    &lt;/div&gt;
&lt;/section&gt;</pre>
```

Coaches Section

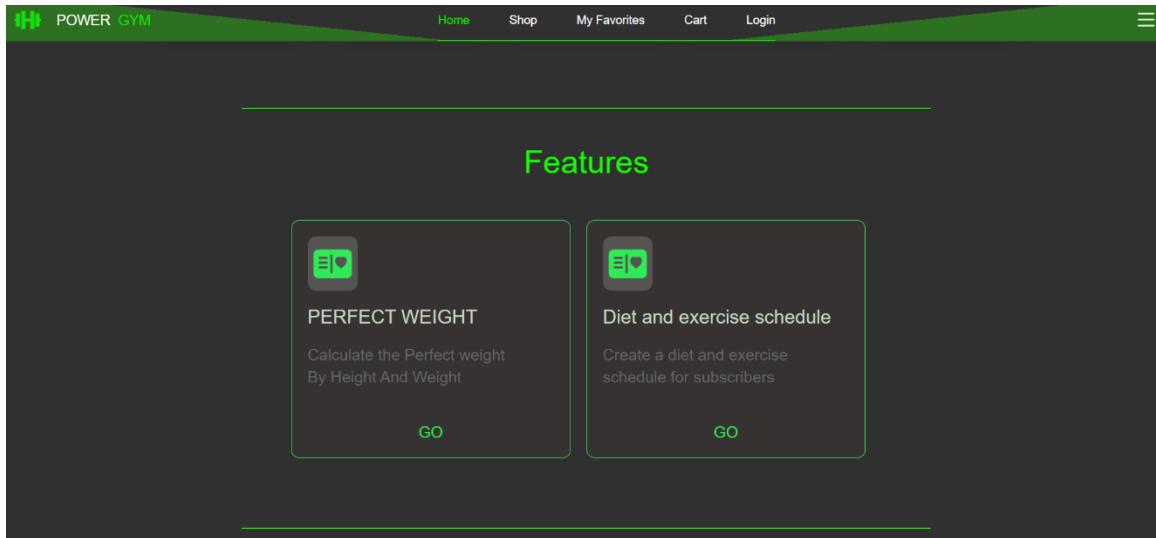


```
● ● ●

@if (isset($coaches) && count($coaches) > 0)
@foreach ($coaches as $coach)
@if (isset($coach->image->path))
<br id="coach">
<hr>
<br>
<!-- coaches Section-->
<div class="custom-container">
<div id="coaches_body">
<section class="sl-container">
<div class="sli-title">
<h1 class="bebas-neue-regular">power <span class="bebas-neue-regular">coaches</span><i style="margin-left: 10px;" class="fa-solid fa-users"></i></h1>
</div>
<div class="sli-content">
<div class="sli-con-ul">
<ul class="sli-list" style="justify-content: center;">
@foreach ($coaches as $coache)
@if (isset($coache->image->path))
<li class="item-card">
<div class="cont-card">
<div class="cont-img">
 }})
<div class="cont-title">
<h4>{{ $coache->name }}</h4>
<p>Fitness Instructor</p>
</div>
</div>
</li>
@endif
@endforeach
</ul>
</div>
<div class="cor-rev-next">
<span id="next" class="next">
<i class="fa-duotone fa-chevron-right"></i>
</span>
<span id="prev" class="rever">
<i class="fa-duotone fa-chevron-left"></i>
</span>

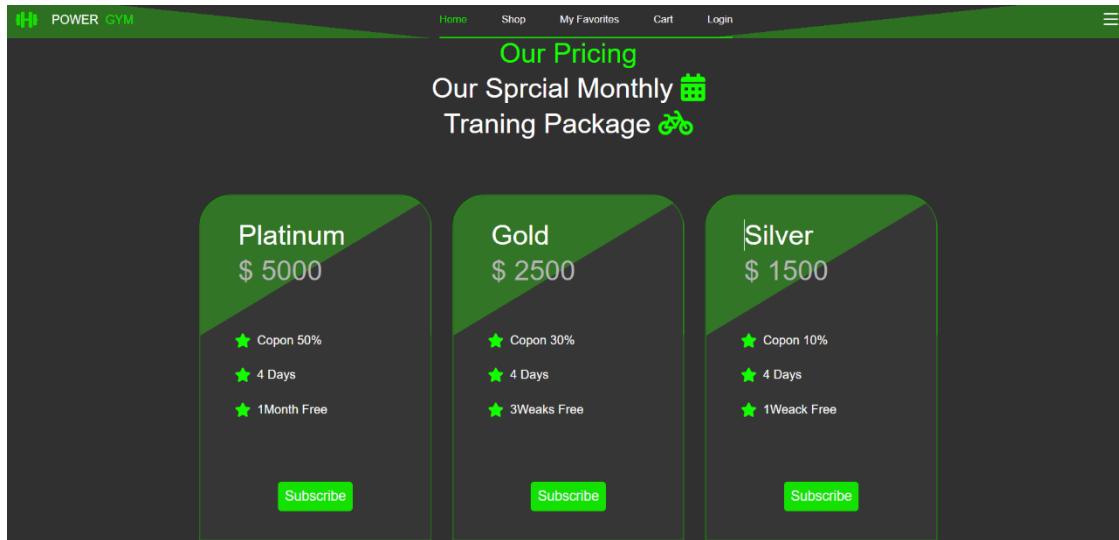
```

Features Section



```
!<-- Features Section-->
<div>
    <div class="h3">
        <h3 class="bebas-neue-regular" style="color: #13ff00;">Features</h3>
    </div>
    <div class="container3">
        <div class="section2">
            <i id="i" class="bi bi-postcard-heart-fill"></i>
            <h2>PERFECT WEIGHT</h2>
            <p>
                Calculate the Perfect weight <br>
                By Height And Weight
            </p>
            <center>
                <div id="a">
                    <a href="{{route('perfectweight')}}">GO</a>
                </div>
            </center>
        </div>
        <div class="section2">
            <i id="i" class="bi bi-postcard-heart-fill"></i>
            <h2>Diet and exercise schedule</h2>
            <p>
                Create a diet and exercise <br> schedule for subscribers
            </p>
            <center>
                <div id="a">
                    <a href="{{route('dietTrain')}}">GO</a>
                </div>
            </center>
        </div>
    </div>
</div>
```

Packages Section



```
① ② ③

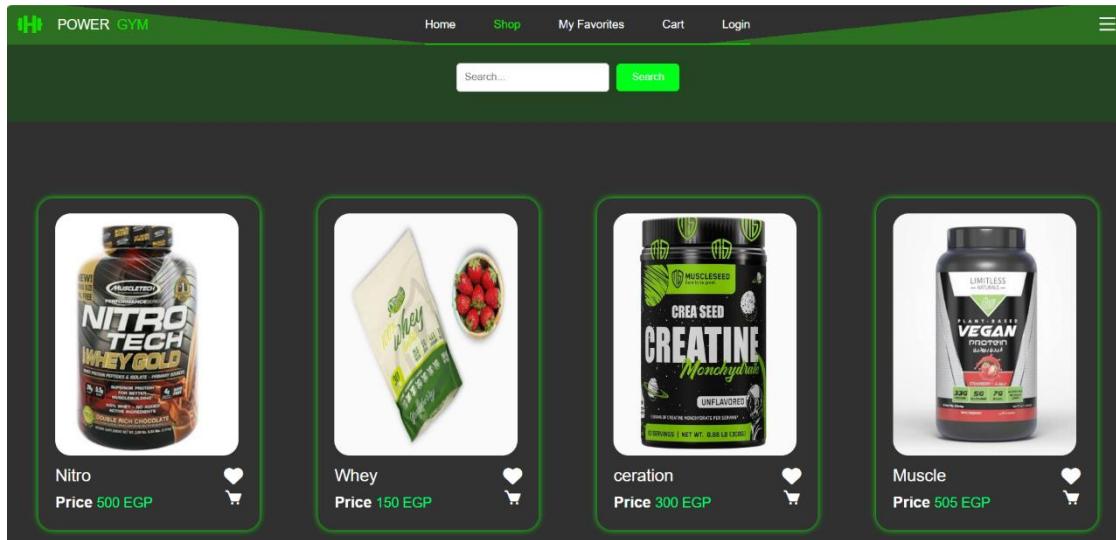
@if (count($packages) > 0)
<hr>
<br>
<!-- Subscribe Section -->
<section>
    <center>
        <h4 class="bebas-neue-regular" style="color: #13ff00;"> Our Pricing</h4>
        <h2 class="bebas-neue-regular"> Our Special Monthly <i class="fa-solid fa-calendar-days" style="color: #13ff00;"></i></h2>
        <h2 class="bebas-neue-regular">Traning Package <i class="fa-solid fa-bicycle" style="color: #13ff00;"></i></h2>
    </center>
    <div class="container2">
        <div class="card_section">
            @foreach ($packages as $package )
            <div class="card">
                <div class="card_head">
                    <h2 class="bebas-neue-regular">{{ $package->name }}</h2>
                    <h2 id="gray" class="bebas-neue-regular">$ {{ $package->price }}</h2>
                </div>
                <div class="card_body">
                    @if (isset($package->features))
                    @foreach ($package->features as $features )
                    <div class="features">
                        <i class="fa-solid fa-star fa-lg" style="color: #13ff00;"></i>
                        <h2 class="nanum-gothic-regular">{{ $features->feature }}</h2>
                    </div>
                    @endforeach
                    @endif
                </div>
                <div>
                    <form action="{{ route('paypal',['id'=>$package->id]) }}" method="post">
                        @csrf
                        <center>
                            <button type="submit" class="Subscribe">Subscribe</button>
                        </center>
                    </form>
                </div>
            </div>
            @endforeach
        </div>
        <br><br><br>
        @if (isset($allpackages) && count($allpackages)>3)
        <center>
            <a href="{{ route('Packages') }}" class="see_more">See More <i class="fa-solid fa-arrow-right"></i></a>
        </center>
        @endif
    </section>
@endif
```

Contact Section

The screenshot shows a website for "POWER GYM". At the top, there's a navigation bar with links for Home, Shop, My Favorites, Cart, and Login. Below the navigation is a green header bar with the text "Contact With Us". The main content area features a map of Cairo and surrounding areas, with a red marker indicating the gym's location. To the left of the map, there's a sidebar with icons and text for "OPEN DAYS", "OPEN TIME", "EMAIL", and "Call". To the right of the map is a contact form with fields for "Your Name", "Your Email", and a large text area for "Message", followed by a "Send Message" button.

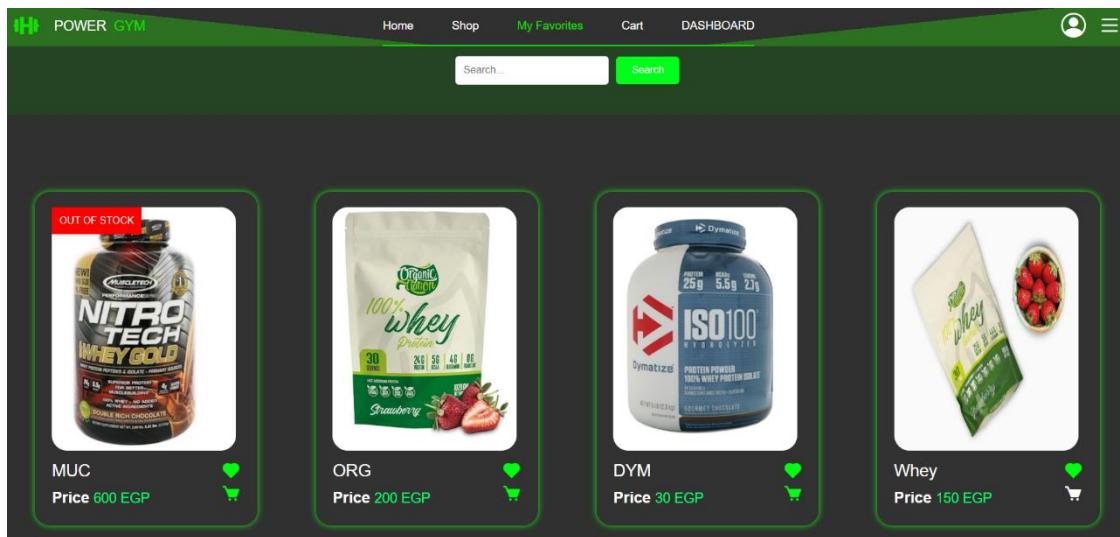
```
<!--Contact Us Section-->
<section id="con_section">
    <center>
        <h1 style="color:#13ff00;" class="bebas-neue-regular"><i class="bi bi-envelope-check"></i>Contact With Us</h1>
    </center>
    <div class="marg">
        <iframe src="{{ $contact->location }}" frameborder="0" allowfullscreen loading="lazy" referrerPolicy="no-referrer-when-downgrade"></iframe>
        <div class="half">
            <div class="con_card">
                @if ($contact->daysopen != null)
                    <div class="inf">
                        <div><i class="bi bi-geo-alt"></i></div>
                        <div class="start_s">
                            <h4>OPEN DAYS:</h4>
                            <p>{{ $contact->daysopen }}</p>
                        </div>
                    </div>
                @endif
                @if ($contact->timeopen != null)
                    <div class="inf">
                        <div><i class="bi bi-geo-alt"></i></div>
                        <div class="start_s">
                            <h4>OPEN TIME:</h4>
                            <p>{{ $contact->timeopen }}</p>
                        </div>
                    </div>
                @endif
                @if ($contact->email != null)
                    <div class="inf">
                        <div><i class="bi bi-envelope"></i></div>
                        <div class="start_s">
                            <h4>EMAIL:</h4>
                            <p>{{ $contact->email }}</p>
                        </div>
                    </div>
                @endif
                @if ($contact->phone != null)
                    <div class="inf">
                        <div><i class="bi bi-telephone"></i></div>
                        <div class="start_s">
                            <h4>Call:</h4>
                            <p>{{ $contact->phone }}</p>
                        </div>
                    </div>
                @endif
            </div>
            <form class="card2" action="https://api.web3forms.com/submit" method="POST">
                <input type="hidden" name="access_key" value="b3ad8481-ec5d-4180-b5c8-"
            </form>
        </div>
    </div>
```

Shop



```
<!--Contact Us Section-->
<section id="con_section">
    <center>
        <h1 style="color:#13ff00;" class="bebas-neue-regular"><i class="bi bi-envelope-check"></i>Contact With Us</h1>
    </center>
    <div class="marg">
        <iframe src="{{ $contact->location }}" frameborder="0" allowfullscreen loading="lazy" referrerPolicy="no-referrer-when-downgrade"></iframe>
        <div class="half">
            <div class="con_card">
                @if ($contact->daysopen != null)
                    <div class="inf">
                        <div><i class="bi bi-geo-alt"></i></div>
                        <div class="start_s">
                            <h4>OPEN DAYS:</h4>
                            <p>{{ $contact->daysopen }}</p>
                        </div>
                    </div>
                @endif
                @if ($contact->timeopen != null)
                    <div class="inf">
                        <div><i class="bi bi-geo-alt"></i></div>
                        <div class="start_s">
                            <h4>OPEN TIME:</h4>
                            <p>{{ $contact->timeopen }}</p>
                        </div>
                    </div>
                @endif
                @if ($contact->email != null)
                    <div class="inf">
                        <div><i class="bi bi-envelope"></i></div>
                        <div class="start_s">
                            <h4>EMAIL:</h4>
                            <p>{{ $contact->email }}</p>
                        </div>
                    </div>
                @endif
                @if ($contact->phone != null)
                    <div class="inf">
                        <div><i class="bi bi-telephone"></i></div>
                        <div class="start_s">
                            <h4>Call:</h4>
                            <p>0{{ $contact->phone }}</p>
                        </div>
                    </div>
                @endif
            </div>
        </div>
        <form class="card2" action="https://api.web3forms.com/submit" method="POST">
            <input type="hidden" name="access_key" value="b3ad8481-ec5d-4180-b5c8-"
        </form>
    </div>
</section>
```

Favorites



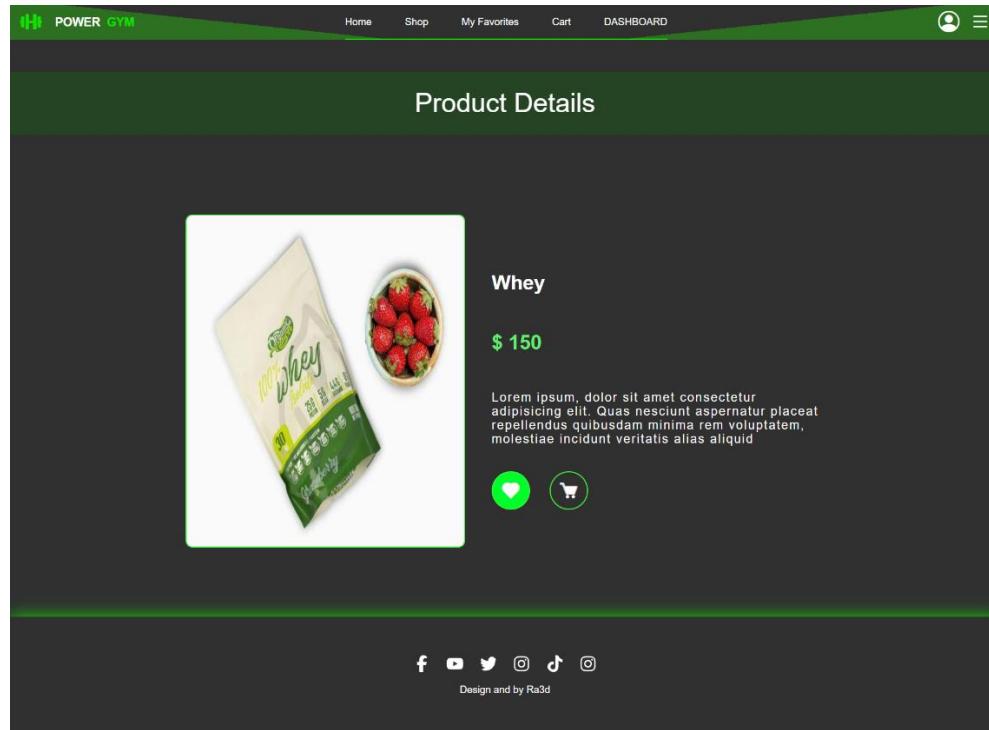
```
    font-size: 25px
}

#out {
    position: absolute;
    color: white;
    background-color: red;
    padding: 10px;
    font-size: 13px;
}
</style>
@endsection
@include('layouts.sidebar.sidebar')
<div style="height: 100px;"></div>
<div class="all">
    <center style="padding: 19px;
background-color: #00800040;">
        <h1 class="bebas-neue-regular">MY Favorites </h1>
        <br>
        <form id="form" action="{{ route('usershop') }}" method="GET">
            <input value="{{ request() -> input('query') }}" id="input" type="text" name="query"
placeholder="Search..."/>
            <button id="button" type="submit">Search</button>
        </form>
    </center>
    <div class="shop">
        @foreach ($favorites as $favorite)
            <div class="img_continer">
                @if ($favorite->product->quantity == 0)
                    <h1 id="out">OUT OF STOCK</h1>
                @endif
                

                <div class="icon">
                    <livewire:favorites :productId="$favorite->product->id" />
                    <livewire:cart :productId="$favorite->product->id" />
                </div>

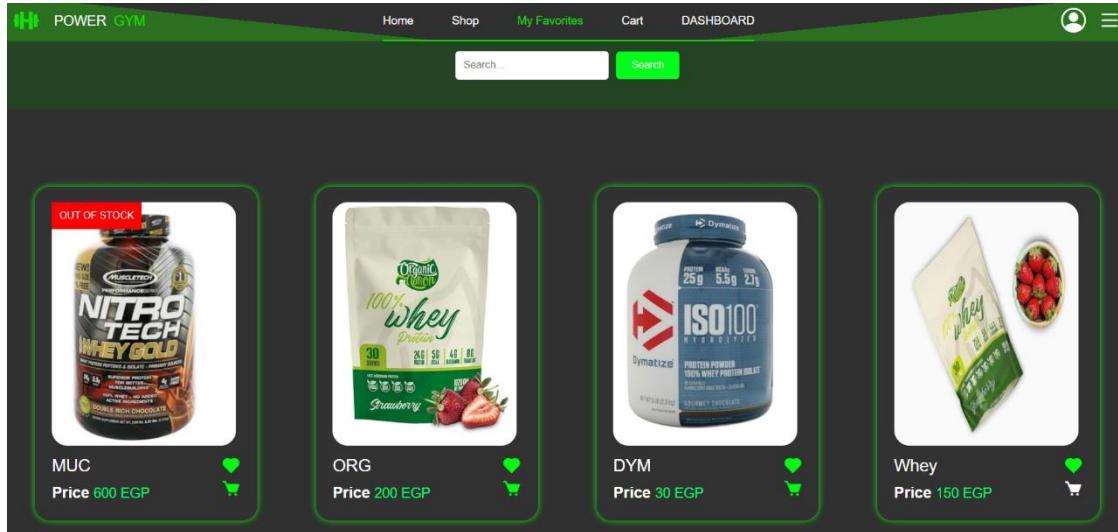
                <div class="text">
                    <h2>{{ $favorite->product->name }}</h2>
                </div>
                <div class="text">
                    @if (!isset($favorite->product->newprice))
                        <h4>Old Price <del><span> {{ $favorite->product->price }} EGP</span></del></h4>
                    @else
                        <h4>New Price <span> {{ $favorite->product->newprice }} EGP</span></h4>
                    @endif
                </div>
            </div>
        @endforeach
    </div>
    @section('script')
        <script>
            $(document).ready(function() {
                $('#input').on('input', function() {
                    var query = $(this).val();
                    if (query === '') {
                        $('#out').css('display', 'none');
                    } else {
                        $('#out').css('display', 'block');
                    }
                });
            });
        </script>
    @endsection
</div>
```

Product Details



```
<x-app-layout>
    @section('style')
        <link rel="stylesheet" href="{{asset('css/productdetails.css')}}">
        <link rel="stylesheet" href="{{asset('css/all.css')}}">
        <link href='https://unpkg.com/boxicons@2.1.4/css/boxicons.min.css' rel='stylesheet'>
    @endsection
    @include('layouts.sidebar.sidebar')
    <div style="height:100px;"></div>
    <center style="padding: 19px; background-color: #00800040;">
        <h1 class="bebas-neue-regular">Product Details</h1>
    </center>
    <div class="content">
        <div class="img_content">
            @if ($product->quantity == 0)
                <h1 id="out">OUT OF STOCK</h1>
            @endif
            
            <h2>{{ $product->name }}</h2>
            <h2>$ <span>{{ $product->price }}</span></h2>
            <p>
                {!! nl2br(e($product->description)) !!}
            </p>
        <div id="butons" style="display: flex; gap:30px">
            <livewire:favorites :productId="$product->id" />
            <livewire:cart :productId="$product->id" />
        </div>
    @section('script')
    @endsection
</x-app-layout>
```

Cart



```

        font-size: 25px
    }

    #out {
        position: absolute;
        color: white;
        background-color: red;
        padding: 10px;
        font-size: 13px;
    }

```

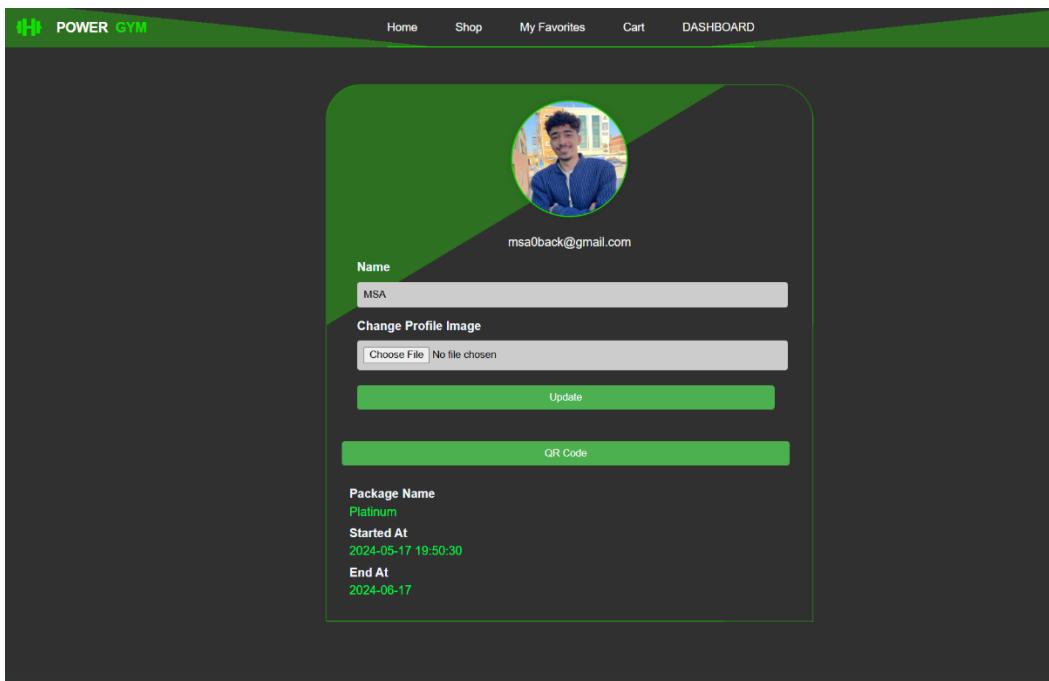
@endsection
@include('layouts.sidebar.sidebar')
<div style="height: 100px;"></div>
<div class="all">
<center style="padding: 19px;
background-color: #00800040;">
<h1 class="bebas-neue-regular">MY Favorites </h1>

<form id="form" action="{{ route('usershop') }}" method="GET">
<input value="{{ request()->input('query') }}" id="input" type="text" name="query"
placeholder="Search...">
<button id="button" type="submit">Search</button>
</form>
</center>
<div class="shop">
@foreach (\$favorites as \$favorite)
<div class="img_continer">
@if (\$favorite->product->quantity == 0)
<h1 id="out">OUT OF STOCK</h1>
@endif

<div class="icon">
<livewire:favorites :productId="\$favorite->product->id" />
<livewire:cart :productId="\$favorite->product->id" />
</div>
<div class="text">
<h2>{{ \$favorite->product->name }}</h2>
</div>
<div class="text">
@if (isset(\$favorite->product->newprice))
<h4>Old Price {{ \$favorite->product->price }} EGP</h4>
<h4>New Price {{ \$favorite->product->newprice }} EGP</h4>
@else
<h4> Price {{ \$favorite->product->price }} EGP</h4>
@endif
</div>
@endforeach
</div>
</div>

@section('script')

Profile

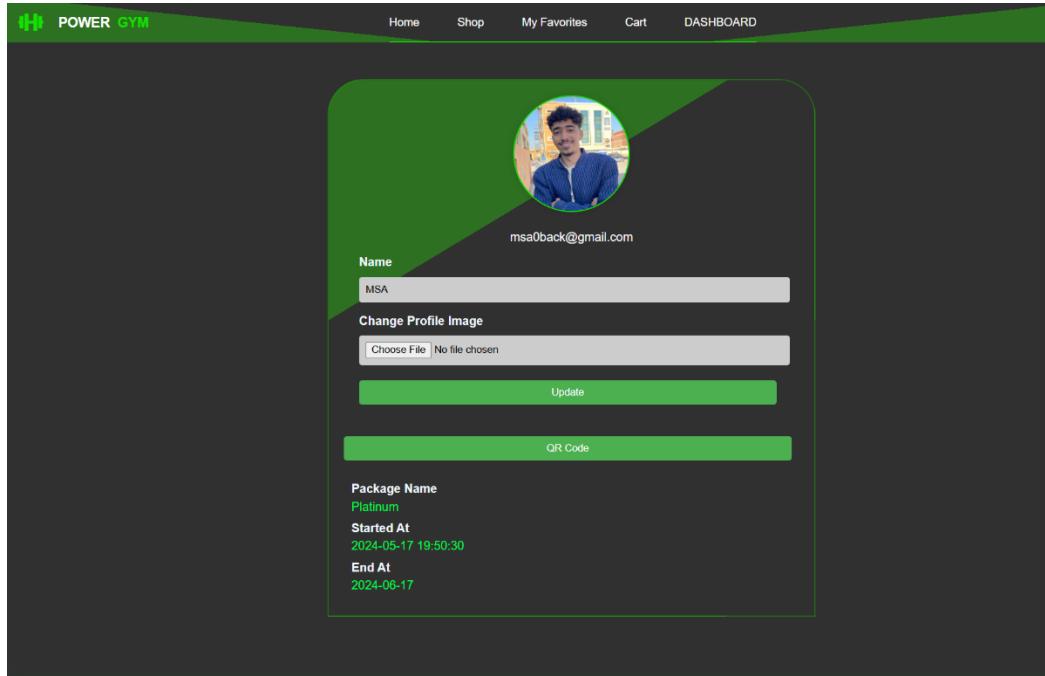


```
<center>
    {{ auth()->user()->email }}
</center>
</div>
<div style="color: rgb(255, 255, 255)" class="info-item">
<center>
    {{ auth()->user()->phone }}
</center>
</div>
<form method="post" action="{{ route('profile.update') }}" class="mt-6 space-y-6">
    @csrf
    @method('patch')
    @if (session('status'))
        <div id="statusAlert" class="alert alert-success" role="alert">
            {{ session('status') }}
        </div>
    @endif
    <div class="profile-info">
        <div class="info-item">
            <span>Name</span>
            <input value="{{ auth()->user()->name }}" name="name" type="text" id="name">
            <input-error style="color: red" :messages="$errors->get('name')"/>
        </div>
        <div class="info-item">
            <span>Change Profile Image</span>
            <input type="file" id="new-profile-img" name="path" accept="image/*">
            <input-error style="color: red" :messages="$errors->get('path')"/>
        </div>
        <button type="submit">Update</button>
    </div>
</form>
<br>
<br>
<button onclick="toggleQRCode()"> QR Code</button>
<br>
<div class="info-item2" id="qrCodeDiv" style="display:none;">
    <br>
    <div id="qrcode">
        <center>
            {{ $qrCode }}
        </center>
    </div>
</div>

<br>
@if (auth()->user()->subscriber)
<div style="margin: 10px;color:rgb(0, 255, 72); " class="info-item">
    <span>Package Name</span>
    {{ auth()->user()->subscriber->package->name }}
</div>
<div style="margin: 10px;color:rgb(0, 255, 72); " class="info-item">
    <span>Started At</span>
    {{ auth()->user()->subscriber->created_at }}
</div>
<div style="margin: 10px;color:rgb(0, 255, 72); " class="info-item">
    <span>End At</span>
    {{ auth()->user()->subscriber->end_at }}
</div>
@endif
{{-- <div class="info-item">
    <button >Update Password</button>
</div> --}}

```

Perfect Weight



```
<center>
    {{ auth()->user()->email }}
</center>
</div>
<div style="color: rgb(255, 255, 255)" class="info-item">
    <center>
        {{ auth()->user()->phone }}
    </center>
</div>
<form method="post" action="{{ route('profile.update') }}" class="mt-6 space-y-6">
    @csrf
    @method('patch')
    @if (session('status'))
        <div id="statusAlert" class="alert alert-success" role="alert">
            {{ session('status') }}
        </div>
    @endif
    <div class="profile-info">
        <div class="info-item">
            <span>Name</span>
            <input value="{{ auth()->user()->name }}" name="name" type="text" id="name">
            <x-input-error style="color: red" :messages="$errors->get('name')"/>
        </div>
        <div class="info-item">
            <span>Change Profile Image</span>
            <input type="file" id="new-profile-img" name="path" accept="image/*">
            <x-input-error style="color: red" :messages="$errors->get('path')"/>
        </div>
        <button type="submit">Update</button>
    </div>
</form>
<br>
<br>
<button onclick="toggleQRCode()"> QR Code</button>
<br>
<div class="info-item2" id="qrCodeDiv" style="display:none;">
    <br>
    <div id="qrcode">
        <center>
            {{ $qrCode }}
        </center>
    </div>
</div>
<br>
@if (auth()->user()->subscriber)
<div style="margin: 10px;color:rgb(0, 255, 72); " class="info-item">
    <span>Package Name</span>
    {{ auth()->user()->subscriber->package->name }}
</div>
<div style="margin: 10px;color:rgb(0, 255, 72); " class="info-item">
    <span>Started At</span>
    {{ auth()->user()->subscriber->created_at }}
</div>
<div style="margin: 10px;color:rgb(0, 255, 72); " class="info-item">
    <span>End At</span>
    {{ auth()->user()->subscriber->end_at }}
</div>
@endif
{{-- <div class="info-item">
    <button >Update Password</button>
</div> --}}
```

3.3. Manual User.

1. Getting Started:

Login/Register:

- **Register:** If you are a new user, click on the "Register" button on the Login page. Fill in the required information, including your name, email address, password, and any other necessary details. Submit the form to create your account.
- ****Login**:** If you already have an account, click on the "Login" button. Enter your email address and password, then click "Submit" to access your account.

2. Exploring Coaches:

Coaches Section:

- Navigate to the "Coaches" section from the main menu.
- Browse through the list of available coaches, each with a profile detailing their expertise, experience, and specialties.
- Click on a coach's profile to view more information, including their training philosophy, certifications, and client testimonials.
- If interested, you can favorite a coach by clicking the "Favorite" button, allowing for quick access in the future.

3. Selecting Packages

Packages Section:

- Go to the "Packages" section to explore different membership options.
- Compare various packages based on duration, price, and included services.
- Click on a package to view detailed information about what it includes, such as access to classes, personal training sessions, and facility usage.
- Select a package that suits your needs and click "Sign Up" to proceed with the purchase. Follow the prompts to complete the payment process.

4. Contacting Power Gym

Contact Section:

- Access the "Contact" section to reach out to Power Gym for any inquiries or support.
- Fill out the contact form with your name, email address, and message. Click "Submit" to send your inquiry.
- You can also find direct contact details, such as phone numbers and email addresses, for more immediate assistance.

5.Shopping for Merchandise

Shop Section:

- Visit the "Shop" section to browse gym-related products, including apparel, supplements, equipment, and accessories.
- Click on an item to view more details, such as sizes, colors, and pricing.
- Add desired items to your cart by clicking the "Add to Cart" button.
- Review your cart by clicking on the "Cart" icon. Adjust quantities or remove items as needed before proceeding to checkout.

6. Managing Favorites

Favorites Section:

- Access the "Favorites" section to view all your favorited coaches, packages, and products.
- Use this section to quickly revisit items or services you are interested in.
- Manage your favorites by adding or removing items as your preferences change.

7. Using the Cart

Cart Section:

- Click on the "Cart" icon to view items you have added from the Shop.
- Review the items, check quantities, and ensure everything is correct.
- Click "Checkout" to proceed with the payment process. Enter your shipping and payment information as prompted to complete your purchase.

8. Account Management

Profile Management:

- Access your account settings by clicking on your profile icon or username.
- Update your personal information, such as your name, email address, and password.
- View your order history, track current orders, and manage your membership details.
- Adjust your notification preferences and privacy settings as needed.

9. Booking Classes and Sessions.

Class and Session Booking:

- Navigate to the "Classes" or "Sessions" section (if available) to view the schedule of upcoming classes and training sessions.
- Browse the calendar to find suitable dates and times.
- Click on a class or session to view more details and book your spot. Confirm your booking and receive a confirmation email with the details.

Conclusion

This manual covers the essential interactions you will have with the Power Gym website. By following these instructions, you can easily navigate through the platform, manage your membership, and take full advantage of the services and products offered. Enjoy your journey towards better fitness and health with Power Gym!

Chapter 4

Back-End

Website

4.1. Introduction back end.

Back-end development, also known as server-side development, refers to the creation and management of the technology and logic that powers the components behind the scenes of a web application. It involves everything that users don't see directly but is crucial for the functionality, performance, and scalability of a website or application. The back end includes server, database, APIs, and server-side applications. Here, we will explore the essentials of back-end development and delve into one of the popular frameworks, PHP Laravel.

The backend of a web application is crucial for its overall functionality, security, and scalability. Here are some key reasons why the backend is important and its advantages:

1. **Data Management**: The backend is responsible for managing the application's data, including storage, retrieval, and manipulation. It interacts with databases to store user information, content, and other relevant data securely. Effective data management ensures that the application operates smoothly and efficiently.
2. **Business Logic**: The backend contains the business logic of the application, which governs how data is processed, rules are enforced, and operations are performed. It implements algorithms, calculations, and workflows necessary for the application to perform its intended tasks accurately and consistently.
3. **Security**: Backend systems play a critical role in ensuring the security of the application and its data. They handle user authentication, authorization, and access control mechanisms to protect sensitive information from unauthorized access, manipulation, or theft. Additionally, backend developers implement security best practices to mitigate common threats such as SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF).

The Components of Back-End Development

1. Server:

- A server is a powerful computer that hosts the web application and serves it to users over the internet.
- It processes client requests, such as loading a web page or submitting a form, and sends appropriate responses.

2. Database:

- Database's store and manage data in a structured format, allowing for efficient retrieval, insertion, and updating of information.
- Common database systems include MySQL, PostgreSQL, MongoDB, and SQLite.

3. Server-Side Languages:

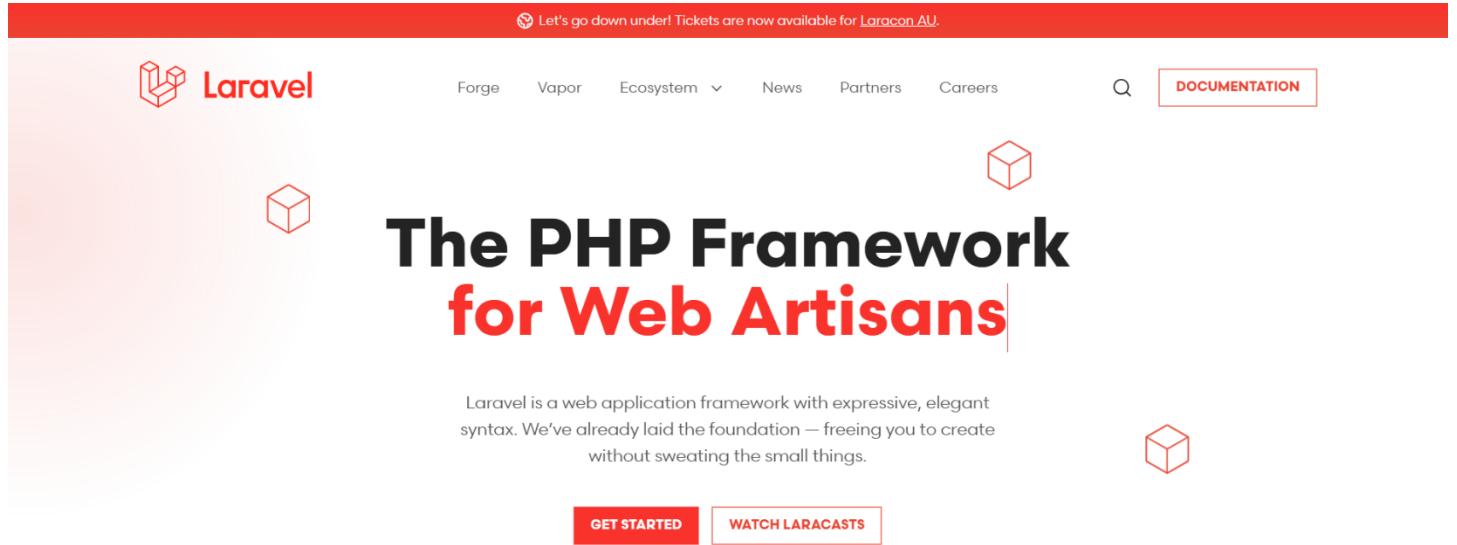
- These are programming languages used to write the logic that runs on the server. Popular choices include PHP, Python, Ruby, Java, and Node.js (JavaScript).

5. Server-Side Frameworks:

- Frameworks provide a structured environment for building and maintaining web applications. They offer libraries, tools, and best practices to streamline development. Examples include Express.js (Node.js), Django (Python), Ruby on Rails (Ruby), and Laravel (PHP).

We use PHP Laravel as the backend for Power Gym

Back-end development forms the backbone of any robust web application, handling data processing, business logic, and server management. PHP, with its simplicity and widespread use, remains a strong choice for back-end development. Laravel, with its rich feature set and elegant syntax, elevates PHP development, enabling developers to build powerful, scalable, and secure web applications efficiently. Whether you're a novice or an experienced developer, embracing Laravel can significantly enhance your productivity and code quality.



4.2. uses in web pages.

Admin pages are dedicated sections of a website accessible exclusively to registered administrators, granting them special permissions to manage the website's content and settings. These privileged areas typically include functionalities such as user management, content creation, editing, deletion, and configuration options crucial for maintaining and updating the website.

The screenshot shows a dark-themed admin dashboard for 'POWER GYM'. At the top, there's a navigation bar with 'Home' and 'DASHBOARD' tabs. On the far right is a user icon and a three-dot menu. The main area features a grid of cards with various statistics:

Category	Count	Icon	Action
USERS	6	User icon	SEE ALL
COACHES	5	Coach icon	SEE ALL
ADMIN ROLES	4	Role icon	SEE ALL
PACKAGES	4	Package icon	SEE ALL
LOGO & Name		Logo icon	SEE ALL
CONTACT US		Bag icon	SEE ALL
Equipments	0	Gear icon	SEE ALL
Subscribers	1	Subscription icon	SEE ALL
Products	9	Product icon	SEE ALL
Dite Train	9	Bag icon	SEE ALL

On the right side, there's a vertical sidebar menu with the following items:

- USERS
- COACHES
- SUBSCRIBERS
- PRODUCTS
- PACKAGES
- EQUIPMENTS
- LOGO & NAME
- CONTACT & INFO
- DITE

At the bottom, there are social media sharing icons (Facebook, YouTube, Twitter, Instagram, TikTok) and a credit line: 'Design and by Ra3d'.

In general, admin pages in the backend include several tasks, such as:

- Adding, editing, and deleting users, content, and pages.
- Adding, editing, and deleting Packages.
- Adding, editing, and deleting Products, content, and pages.
- Adding, editing, and deleting Logo & Name.
- Managing The Orders
- Managing permissions and controlling who can access certain content.

The screenshot shows a dark-themed admin dashboard for a gym management system. The top navigation bar includes a logo, the site name 'POWER GYM', and links for 'Home' and 'DASHBOARD'. A search bar with a magnifying glass icon is also present. On the right side, there is a sidebar with a user profile icon and a three-dot menu icon. The sidebar is organized into sections: USERS, COACHES, SUBSCRIBERS, PRODUCTS, PACKAGES (which is highlighted in green), EQUIPMENTS, and LOGO & NAME. The main content area is titled 'Packages' and shows a table with the following data:

#	Name	Price	Status	features	Actions
3	Platinum	5000	● Active	[Edit]	[Edit] [Delete]
6	Gold	2500	● Active	[Edit]	[Edit] [Delete]
7	Silver	1500	● Active	[Edit]	[Edit] [Delete]
8	asmin	500	● Active	[Edit]	[Edit] [Delete]

At the bottom of the page, there are social media sharing icons (Facebook, YouTube, Twitter, Instagram, TikTok, LinkedIn) and a small note: 'Design and by Ra3d'.

In **summary**, admin pages are essential for effective website management, enabling administrators to control content, users, settings, and security aspects efficiently.

4,3. Screen Shot {Code & Design}.

Dashboard



```
<?php

namespace App\Http\Controllers\Admin;

use App\Http\Controllers\Controller;
use App\Models\Equipment;
use App\Models\Package;
use App\Models\Products;
use App\Models\Role;
use App\Models\Subscribe;
use App\Models\User;
use Illuminate\Http\Request;

class dashboard extends Controller
{
    public function index(){
        $users = User::where('role', 'Coache')->get();
        $coachesCount = $users->count();
        $usersCount=User::count();
        $packagesCount = Package::count();
        $rolesCount=Role::count();
        $productsCount=Products::count();
        $EquipmentsCount=Equipment::count();
        $sub = Subscribe::where('active', 1)->get();
        $SubscribeCount=$sub->count();
        return view('dashboard', compact('coachesCount', 'usersCount',
        'packagesCount', 'rolesCount', 'productsCount', 'EquipmentsCount', 'SubscribeCount'));
    }
}
```

Users Dashboard

The screenshot shows a dark-themed dashboard for 'POWER GYM'. At the top, there's a navigation bar with 'Home' and 'DASHBOARD' buttons. On the left, a sidebar titled 'USERS' lists categories: COACHES, SUBSCRIBERS, PRODUCTS, PACKAGES, EQUIPMENTS, LOGO & NAME, CONTACT & INFO, DITE, and Log Out. The main area displays a table titled 'Total: 6' containing user information:

#	Image	Name	Email	Role	verify	Actions
1		MOO	suberadmin@gmail.com	Coache	●	
2		MSA	msa0back@gmail.com	SuberAdmin	●	
3		RADY	msa0back2@gmail.com	Coache	●	
4		ALI	msa@gmail.com	Coache	●	
5		Ahmed	ahm@gmail.com	Coache	●	
6		Fox	fox@gmail.com	Coache	●	

```
public function index(Request $request)
{
    $query = trim($request->input('query'));

    $users = User::where(function ($q) use ($query) {
        $q->where('name', 'LIKE', "%$query%");
        ->orWhere('email', 'LIKE', "%$query%");
    })
    ->orderByRaw("CASE WHEN name LIKE '$query%' THEN 1 ELSE 2 END")
    ->get();
    $roles = Role::get();

    return view('MainPages.AdminPages.users.users', compact('users', 'roles'));
}

public function delete($id){
    $user = user::find($id);
    // Check if user is trying to edit a superadmin role
    if ($user->role === 'SuberAdmin') {
        return redirect()->back()->with('success', 'You cannot delete a SuperAdmin.');
    }
    // Check if the product exists
    if (!$user) {
        return redirect()->route('AdminUsers')->with('error', 'user Not Found');
    }
    // Proceed with the deletion
    $user->delete();
    return redirect()->back()->with('success', 'Data Deleted successfully');
}
```

Packages Dashboard

The screenshot shows a dark-themed dashboard for 'POWER GYM'. The top navigation bar includes 'Home' and 'DASHBOARD'. On the right, a sidebar menu lists categories: USERS, COACHES, SUBSCRIBERS, PRODUCTS, PACKAGES (which is selected and highlighted in green), EQUIPMENTS, and LOGO & NAME. The main content area is titled 'Packages' and shows a list of four packages:

#	Name	Price	Status	features	Actions
3	Platinum	5000	Active	[Icon]	[Edit] [Delete]
6	Gold	2500	Active	[Icon]	[Edit] [Delete]
7	Silver	1500	Active	[Icon]	[Edit] [Delete]
8	asmin	500	Active	[Icon]	[Edit] [Delete]

Total: 4

```
public function store(request $request){
    $validatedData = $request->validate([
        'name' => 'required|string|max:25',
        'price' => 'required|numeric|regex:/^\d+(\.\d{1,2})?$/|max:999999.99',
    ]);

    $Package = new Package;
    $Package->name = $validatedData['name'];
    $Package->price = $validatedData['price'];
    $Package->save();
    return redirect()->route('AdminPackages')->with('success', 'Data stored successfully');
}

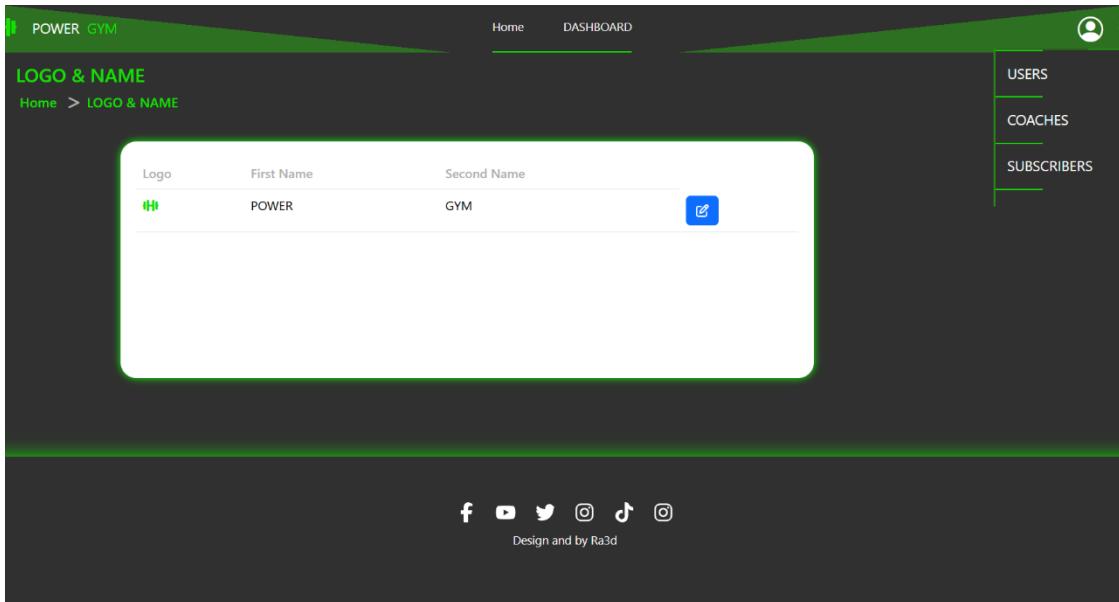
public function update(request $request,$id){
    $validatedData = $request->validate([
        'name' => 'required|string|max:25',
        'status'=>'required',
        'price' => 'required|numeric|regex:/^\d+(\.\d{1,2})?$/|max:999999.99',
    ]);

    $Package = Package::find($id);
    $Package->name = $validatedData['name'];
    $Package->status = $validatedData['status'];
    $Package->price = $validatedData['price'];
    $Package->save();
    return redirect()->route('AdminPackages')->with('success', 'Data Updated successfully');
}

public function delete($id){
    $package = package::find($id);
    // Check if the product exists
    if (!$package) {
        return redirect()->route('AdminPackages')->with('error', 'Package Not Found');
    }

    // Proceed with the deletion
    $package->delete();
    return redirect()->route('AdminPackages')->with('success', 'Data Deleted successfully');
}
```

Logo & Name Dashboard



```
public function index(Request $request)
{
    $ModelsLogoName = ModelsLogoName::first();
    return view('MainPages.AdminPages.logo&Name.logoName', compact('ModelsLogoName'));
}

public function update(request $request,$id){
    $LogoName = ModelsLogoName::find($id);

    $validatedData = $request->validate([
        'firstname' => 'required|string|max:25',
        'secondname' => 'max:25',
        'path' => 'image|mimes:jpeg,png,jpg,gif,svg|max:2048',
    ]);
    if ($request->file('path')) {
        $image = $request->file('path');
        $imageName = $image->getClientOriginalName();
        $path = $image->storeAs('Logo', $imageName, 'public');
    } else {
        $path = $LogoName->logopath;
    }
    $LogoName =ModelsLogoName::find($id);
    $LogoName->firstname = $validatedData['firstname'];
    $LogoName->secondname = $validatedData['secondname'];
    $LogoName->logopath = $path;
    $LogoName->save();
    return redirect()->back()->with('success', 'Data Updated successfully');
}
```

Subscribers Dashboard

The screenshot shows a dark-themed web application interface. At the top, there's a navigation bar with links for 'Home', 'Dashboard', 'Logout', and other menu items. On the left, a sidebar lists categories: 'USERS', 'COACHES' (which is currently selected), 'SUBSCRIBERS' (highlighted in green), 'PRODUCTS', and 'PACKAGES'. The main content area is titled 'Coaches' and shows a list with one item. The list includes columns for '#', 'Image', 'Pckage Name', 'User Name', 'Phone', 'Sub Date', 'End At', 'Active', and 'Actions'. The single item listed is '2' (id), 'Platinum' (Image), 'MSA' (User Name), '2024-05-17 19:50:30' (Sub Date), '2024-06-17' (End At), '1' (Active), and two small icons for 'Edit' and 'Delete' in the 'Actions' column. A search bar and a 'Total: 1' message are also present.

The screenshot shows a code editor window with a dark theme. The code is written in PHP and handles subscriber operations. It includes methods for indexing, renewing, and resetting subscribers. The indexing method uses a query builder to filter active or specific subscribers based on a search term, orders them by active status, and retrieves packages. The renew method updates the end date and active status of a subscriber. The reset method sets the end date and active status to null. The code uses Carbon for date parsing and saving.

```
public function index(Request $request)
{
    $query = $request->input('query');

    $subscribers = Subscribe::where(function ($q) use ($query) {
        $q->where('active', 'LIKE', "%$query%");
        ->orWhere('id', 'LIKE', "%$query%");
    })
    ->orderByRaw("CASE WHEN active LIKE '$query%' THEN 1 ELSE 2 END")->get();
    $packages = Package::get();

    return view('MainPages.AdminPages.subscribers.subscribe', compact('subscribers',
    'packages'));
}

public function renew(Request $request, $id)
{
    $subscribers = Subscribe::find($id);
    $subscribers->created_at = now();
    $subscribers->end_at = Carbon::parse($subscribers->end_at)->addMonth();
    $subscribers->active = 1;
    $subscribers->save();
    return redirect()->route('AdminSubscribers')->with('success', 'Data Updated
successfully');
}

public function reset(Request $request, $id)
{
    $subscribers = Subscribe::find($id);
    $subscribers->created_at = null;
    $subscribers->end_at = null;
    $subscribers->active = 0;
    $subscribers->save();
    return redirect()->route('AdminSubscribers')->with('success', 'Data Updated
successfully');
}
```

Products Dashboard

The screenshot shows a dark-themed dashboard for 'POWER GYM'. At the top, there's a navigation bar with 'Home' and 'DASHBOARD' tabs, a search bar, and a user profile icon. A sidebar on the right lists various sections: USERS, COACHES, SUBSCRIBERS, PRODUCTS (which is currently selected), PACKAGES, EQUIPMENTS, LOGO & NAME, CONTACT & INFO, DITE, and Log Out. The main content area is titled 'SHOP' and shows a table with four rows of product data:

#	Image	Name	New Price	Quantity	Description	Actions
1		Nitro	500	10	Quas nesciunt aspernatur placeat repellendus quibusdam minima rem voluptatem, molestiae incident veritatis alias aliquid	
2		Whey	150	5	Quas nesciunt aspernatur placeat repellendus quibusdam minima rem voluptatem, molestiae incident veritatis alias aliquid	
3		Ceration	300	10	Quas nesciunt aspernatur placeat repellendus quibusdam minima rem voluptatem, molestiae incident veritatis alias aliquid	
4		Muscle	505	50	Quas nesciunt aspernatur placeat repellendus	

```

public function usershop(Request $request)
{
    $query = $request->input('query');
    $products = ModelsShop::query();

    if (is_numeric($query)) {
        // Search within the price range
        $products->where('price', '>=', 0)
            ->where('price', '<=', $query);
    } else {
        // Search by name or exact price
        $products->where(function ($q) use ($query) {
            $q->where('name', 'LIKE', "%$query%")
                ->orWhere('price', 'LIKE', "%$query%");
        });
        // Order by name starting with the query string
        $products->orderByRaw("CASE WHEN name LIKE '$query%' THEN 1 ELSE 2 END");
    }
    $products = $products->get();
    return view('MainPages.shop', compact('products'));
}

public function store(request $request){
    $validatedData = $request->validate([
        'name' => 'required|string|max:25',
        'description' => 'required|string|max:255',
        'quantity' => 'required|numeric|regex:/^\d+(\.\d{1,2})?$/|max:999999.99',
        'path' => 'required|image|mimes:jpeg,png,jpg,gif,svg|max:2048', // Adjusted for images
        'price' => 'required|numeric|regex:/^\d+(\.\d{1,2})?$/|max:999999.99',
    ]);
    $image = $request->file('path');
    $imageName = $image->getClientOriginalName();
    $path = $image->storeAs('Equipments', $imageName, 'public');

    $product = new ModelsShop;
    $product->name = $validatedData['name'];
    $product->description = $validatedData['description'];
    $product->price = $validatedData['price'];
    $product->quantity = $validatedData['quantity'];
}

```

Contact Dashboard

The screenshot shows a dark-themed dashboard for 'POWER GYM'. At the top, there's a navigation bar with 'POWER GYM' logo, 'Home', and 'DASHBOARD' buttons. On the right, there's a sidebar with icons for 'USERS', 'COACHES', 'SUBSCRIBERS', 'PRODUCTS', 'PACKAGES', 'EQUIPMENTS', 'LOGO & NAME', and 'CONTACT & INFO'. The main content area is titled 'contact info' and shows the following data:

Key	Value
location	
phone	1110562097
email	Gym@gmail.com
whatsapp	1110562097
facebook link	mm
instgram link	lkl
X link	uyuh
open days	FROM Saturday To Monday
open time	From 9AM To 9PM
Actions	

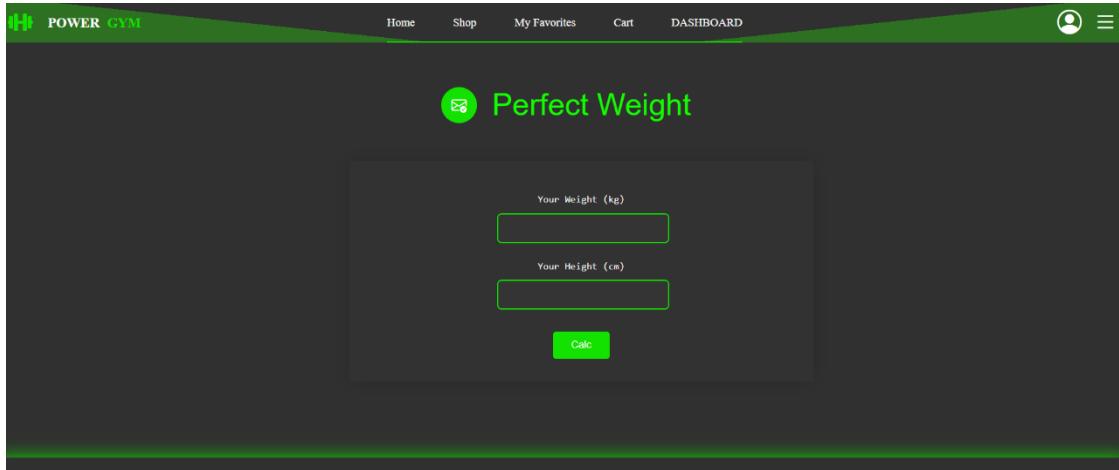
The screenshot shows a code editor with PHP code. The code defines two functions: 'index()' and 'update(request \$request, \$id)'. The 'index()' function retrieves the first contact from the database and returns it as a view. The 'update()' function finds the contact by ID, validates the input data, and updates the contact's attributes with the validated values. Finally, it saves the contact and redirects back with a success message.

```
public function index(){
    $contacts=contacts::first();
    return view('MainPages.AdminPages.contact.contact',compact('contacts'));
}

public function update(request $request,$id){
    $contact = contacts::find($id);

    $validatedData = $request->validate([
        'location' => 'required|string',
        'phone' => 'required|max:255',
        'whatsapp' => 'max:255',
        'facebooklink' => 'max:255',
        'instgramlink' => 'max:255',
        'xlink' => 'max:255',
        'daysopen' => 'max:255',
        'timeopen' => 'max:255',
    ]);
    $contact =contacts::find($id);
    $contact->location = $validatedData['location'];
    $contact->phone = $validatedData['phone'];
    $contact->whatsapp = $validatedData['whatsapp'];
    $contact->facebooklink = $validatedData['facebooklink'];
    $contact->instgramlink = $validatedData['instgramlink'];
    $contact->timeopen = $validatedData['timeopen'];
    $contact->daysopen = $validatedData['daysopen'];
    $contact->xlink = $validatedData['xlink'];
    $contact->save();
    return redirect()->back()->with('success', 'Data Updated successfully');
}
```

Perfect Weight



```
public function index()
{
    return view('MainPages.weight.perfectweight');
}
public function result(Request $request)
{
    // Get weight and height from the request
    $weight = $request->input('weight'); // in kg
    $height = $request->input('height'); // in cm

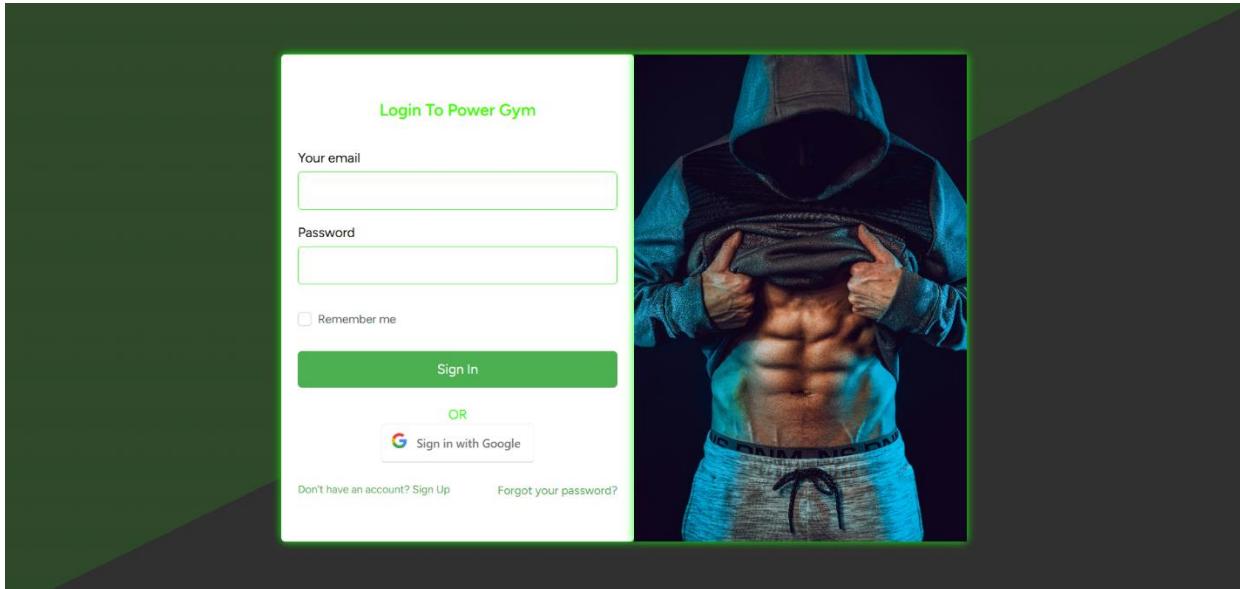
    // Convert height to meters
    $heightInMeters = $height / 100;

    // Calculate BMI
    $bmi = $weight / ($heightInMeters * $heightInMeters);

    // Determine the perfect weight range based on BMI
    $minPerfectWeight = 18.5 * ($heightInMeters * $heightInMeters);
    $maxPerfectWeight = 24.9 * ($heightInMeters * $heightInMeters);

    // Check if the weight is normal, underweight, or overweight
    $weightStatus = '';
    if ($weight < $minPerfectWeight) {
        $weightStatus = 'Underweight';
    } elseif ($weight > $maxPerfectWeight) {
        $weightStatus = 'Overweight';
    } else {
        $weightStatus = 'Normal';
    }
    // Return the BMI, perfect weight range, and weight status
}
```

Login

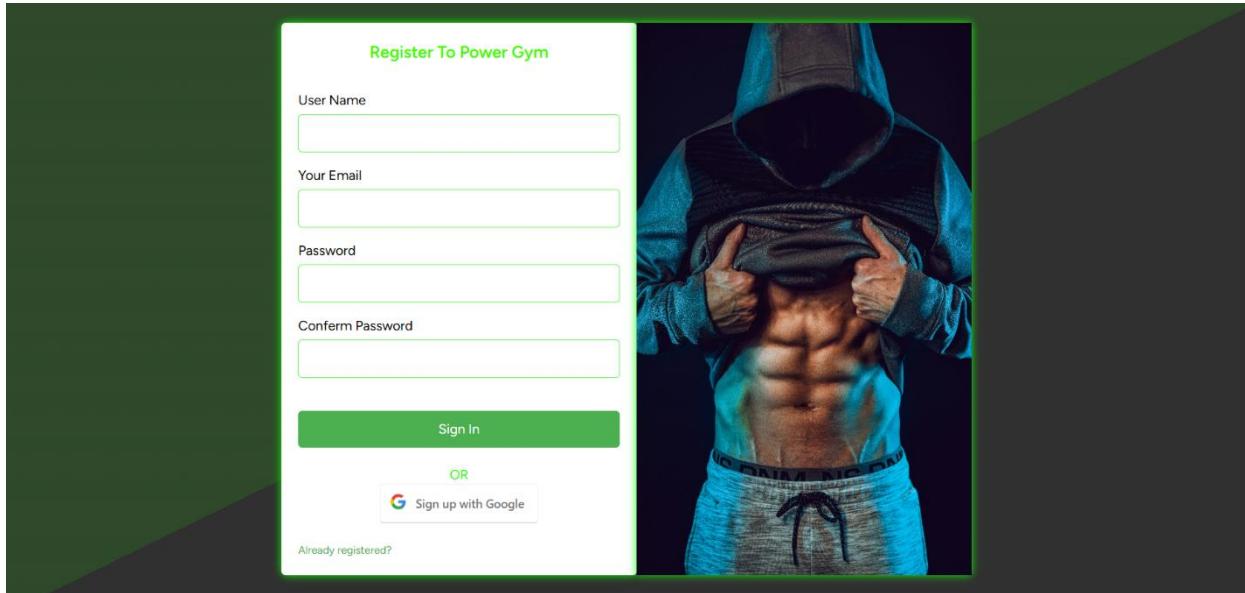


```
public function store(LoginRequest $request): RedirectResponse
{
    $request->authenticate();

    $request->session()->regenerate();

    // return redirect()->intended(route('dashboard', absolute: false))
    return redirect()->route('home');
}
```

Register



```
public function store(Request $request): RedirectResponse
{
    $request->validate([
        'name' => ['required', 'string', 'max:255'],
        'email' => ['required', 'string', 'lowercase', 'email', 'max:255',
        'unique:' . User::class],
        'password' => ['required', 'confirmed', Rules\Password::defaults()],
    ]);

    $user = User::create([
        'name' => $request->name,
        'email' => $request->email,
        'password' => Hash::make($request->password),
    ]);
    $social = Social::create([
        'social_type' => "form",
        'user_id'=>$user->id,
    ]);
    // Generate code for user
    $user->generateCode();
    $user->notify(new CodeVerify());
    event(new Registered($user));
    Auth::login($user);
    return redirect()->route('CodeVerify');
}
```

Chapter 5

Database &

Conclusion

