

Software Engineering Department

Ort Braude College

Capstone Project Phase B – 61998

**Social network for healthy life**

Project code: 23-2-D-11

**Supervisor:**

Dr. Zakharia Frenkel

**Authors:**

Reem Kadmany

[reem.kadmany@e.braude.ac.il](mailto:reem.kadmany@e.braude.ac.il)

Roaa Bader

[roaa.bader@e.braude.ac.il](mailto:reem.kadmany@e.braude.ac.il)

GitHub link: <https://github.com/roaab3/SocialnetworkForHealth>

**Abstract:**

Over time, online health communities and websites become more massive and full of unreliable information. As a result, finding trustworthy information can take a lot of time and effort. That's why we need a fast and reliable way to get trustworthy information from reliable sources.

Our goal is to enable users to share their personal experiences, rate, like, and discuss the efficiency and effectiveness of the data. In addition, the community will implement a profit-sharing model.

**Contents:**

Abstract …………………………………………………………………………………………. 2

1. Introduction …………………………………………………………………………………. 6
2. Background and related work ……………………………………………………………….. 7

2.1 Social impact of Community Generated Content ……………………...……………….... 7

2.2 Meritocracy ………………………………………………………………………………. 8

2.2.1 Problems caused by fake content in the CGC ……………………………………….. 8

2.2.2 Difficulties faced by the CGC due to limited content ……………………………….. 8

2.2.3 Ways to maintain CGC quality by paying moderators ………………………..…….. 8

2.2.4 Approaches through a merit-based algorithm ……………………………………….. 9

2.3 Meritocratic networks…………...……………………………. ……………………..…... 9

2.4 Meritocracy and social network for healthy life project …………………..……………. 10

3. Expected achievements …………………...….……………..……………………………….. 11

3.1 Technology challenge …………………...………………………………………………. 11

3.1.1 Performance ………………….………………………………………….…………. 11

3.1.2 User experience ………………….……………………………………………….... 11

3.1.3 Security ………………….…………………………………………………………. 11

3.2 Criteria of success ……………….……………………………………………………… 12

3.3 Unique features ….……………………………………………………………………… 12

4. Technology choices ……………...……….…………………………………………………. 13

4.1 Technologies for Back-End ……………………….…………………………………….. 14

4.1.1 NodeJS ……………………….………………………………………….…………. 14

4.1.2 Express NodeJS ……………….…………………………………………………… 15

4.1.4 MongoDB ……………………………………………………………….…………. 16

4.2. Technologies for Front-End …………………...….…..……………………………….. 17

4.2.1 ReactJS ………………………..………………………………………….…………... 17

5. Engineering Process ………………...…………………………………………….…………. 18

5.1 Define app purpose …………………..………………………………………….………. 18

5.2 Algorithmic challenge ………………...…...………...……………………….…………. 18

5.2.2 Awarding system ………………...…...………...…………………….…..……..…. 18

5.2.3 Asking and answering questions …………………...….…………...……..………….. 18

5.3 Package Diagram ………..………....………...…………….………………………....…. 19

5.4 Activity Diagram ……..…..…..…...…...………...…………….……………………..…. 20

5.4.1 Adding club …….………………...………...…………….……………………..……. 20

5.4.2 Adding comment….………………..………...…………….……………………..…... 20

5.4.3 Moderator verification …….……….....….......……...…….…………………………. 21

6. Implementation process ………………..…...………………….………………………....…. 22

7. Challenges ………………..…...………………………….…….………………………....…. 23

8. Testing plan ………..……..…...………………………….…….………………………....…. 24

8.1 Login screen ……………..…….………...…...……...……….……………………..…... 26

8.2 Sign-up screen …………...…….………...…...……...……….……………………..…... 27

8.3 Unlogged user screen ………….………...…...……...……….……………………..…... 28

8.4 Logged user screen …………….………...…...……...……….……………………..…... 29

8.5 Moderator screen ……...……….………...…...……...……….……………………..…... 30

9. Results and conclusion ………..……..…..……………….…….………………………....…. 31

10. Lessons that we learned ………………..……………….…….………………………....…. 32

11. User Guide ……………………………..……………….…….………………………....…. 33

11.1 Operational Process …………….….…...…...……...……….……………………..…... 33

11.1.1 Login …………………...…...….…...…...……...……….……………………..…... 33

11.1.2 Login ……………..………………....…...……....……….……………………..…... 33

11.1.3 Register ………………….….….…...…....……...……….……………………..…... 33

11.2 Unlogged user …………….…………...….....……...……….……………………..…... 34

11.2.1 Home screen ………………..….....……...…………….……………………...…... 34

11.2.2 Club screen ………………..….....……...…………….……………………...…….. 35

11.2.3 View About Club ……….…………..….....…………….……………………...…... 36

11.2.4 View Club - Publications ……………....……………………………………...…... 37

11.2.5 View club - Admin ………………..….....…..………….……………………...…... 38

11.2.6 Author screen …………….………..….....…..………….……...……………...…... 38

11.2.7 Author profile - Publications ………………..…....…….……………………...…... 39

11.2.8 Author profile - club ……………………..….....……………………………...…... 40

11.2.9 Author profile - profile …………………..….....……………………………...…... 40

11.3 Logged user …………….……...……...….....……...……….……………………..…... 41

11.3.1 Home screen ………………..….....……...……………..……………………...…... 41

11.3.2 Club screen ………………..….....……...…………….……………………...…….. 42

11.3.3 View Club - Publications ……………...……………….……………………...…... 43

11.3.4 Author screen ……………....……………...……..…….……………………...…... 43

11.3.5 View Author - Publications …………....……………….……………………...…... 45

11.3.6 View Profile…………...……..………....………...…….……………………...…... 45

11.3.7 View Profile - Publications…..………....……………...……………………...…... 46

11.3.8 View Profile - Clubs…………...….…....……………….……………………...…... 46

11.3.9 View Profile - Profile …………………....………………………..…………...…... 47

11.3.10 Edit Profile……..…...……..………....……………....……………………...……. 48

11.3.11 Change password …….…...……..………....……………….………………...…... 49

11.4 Admin ……………….….……...……...….....……...……….……………………..…... 49

11.4 Moderator …………...….……...……...….....……...……….……………………..…... 51

12. Maintenance Guide ……………………………..……………….……………………....…. 53

**1. Introduction:**

People who are receiving medical care and caregivers are using the internet for support more often. They rely on online groups, websites, and blogs to help them with their needs. Some online communication is replacing traditional doctor-patient communication, but it can provide helpful information and support that is not easily found anywhere else. More specifically, patients and caregivers are using online communities to search for health information.

An online health community (OHC) is an online chat group in which people can connect, share experiences, and seek/provide support. People who participate in these online chats usually want to find someone who faced a similar experience and learn from it.

There is not much agreement on the best way to create OHC, and there is a lot of research on its benefits. Even though there is limited research on its benefits, OHC still seems to be helpful for patients and caregivers in many situations. Such as providing valuable information about diseases and treatments.

One of the main reasons people use OHC is to share health information with untrained people.

In many cases, people don't make their best decisions when they share and trust information from online communities. However, using incorrect information can have harmful consequences and this can lead to serious negative effects.

To solve these problems, our project will use a meritocracy network where the reliability of information is based on the talent, effort, and achievements of the user. Furthermore, we will implement a share-profit model where users can gain profits based on their posts, and this will encourage experts to join our community.

* This project is a follow-up project to another project that failed to enter the market, and our purpose is to improve the previous project. Our project is different from the others in the GUI and functionality. In our project, the users can join small/big groups and this will improve the performance of the website. In addition, the previous project did not have a lot of activities that could be offered to the user, meanwhile, we expanded the activities of the user. Also, in our project, some experts will provide the reliability of the content.

**2. Background and related work:**

**2.1 Social Impact of Community-Generated Content:**

A growing body of evidence suggests that community-generated content (CGC) on platforms ranging from Yelp to Facebook has a large causal impact on economic and social outcomes ranging from restaurant decisions to voting behavior.

We look at how CGC influences the behavior and structure of markets and highlight potential areas for future research [1]. Recently, consumers are increasingly making purchasing decisions based on reviews authored by fellow community members rather than relying on professional reviewers.

Examples of online services that show how reviews affect your business:

Amazon:

According to research that was led by Bazaarvoice most U.S. shoppers (56%) identified customer reviews as the primary e-commerce feature they depended on to make informed purchase choices. [2]

Yelp:

By merging the ratings given to restaurants on Yelp with sales information for every restaurant that operated in Seattle over six years, according to the findings of Luca’s study in 2011 an independent restaurant experiences an almost 5% rise in sales with each additional star in their Yelp rating, while such as increase does not affect chain restaurants. This result supports the idea that customers already have more knowledge about chain restaurants, so they rely less on Yelp to learn about them.[1]

eBay:

One clear effect of this review is how the rating given by buyers affects the price at which a seller can sell their item.

This shows that when a seller on eBay receives negative feedback, they can expect to receive more negative feedback around 25% quicker than before according to a study conducted by Cabral and Hortaҫsu (2010). As we can see, the main concern in reviews is assessing how reliable each person is, rather than judging the quality of the product. [1]

**2.2 Meritocracy:** a reliable approach to make sure the quality of CGC is maintained.

**2.2.1 Problems caused by fake content in the CGC:**

One of the primary benefits of CGC is that it provides social proof. Social proof means that when people see others doing something, they are more likely to do the same, especially those who have used the product. According to [Stackla](https://www.businesswire.com/news/home/20190220005302/en/Stackla-Survey-Reveals-Disconnect-Content-Consumers-Marketers), 79 percent of people when they see content created by other users, it strongly affects their decisions.[3]

While CGC can have many benefits for businesses, it can also have negative impacts if not managed properly. The main issue in content production is maintaining control over content quality, and this issue has become increasingly important. Content quality is important because people rely on reviews and recommendations to make decisions regarding their time and money.

Yet the challenges are massive, They are a result of the advantages of CGC. The normal editorial solutions are not feasible, since the editors cannot go through hundreds of millions of reviews.

**2.2.2 Difficulties faced by the CGC due to limited content:**

A useful tactic that you can use to achieve good quality CGC is to identify the reviewer. Their name, where they are from, and if they are an employee. Offering this level of detail increases trust and helps to give confidence to the customers that the brand isn’t keeping secrets or spreading false information.

**2.2.3 Ways to maintain CGC quality by paying moderators:**

A way to ensure quality is to adopt a pre-moderation process where content is reviewed. This allows moderators to filter out any inappropriate or low-quality submissions.

Amazon revealed that they've invested more than $400 million to protect customers from review abuse and other types of misbehavior. They also stopped more than 13 million attempts to post fake reviews and took action against more than five million people trying to manipulate reviews[4].

**2.2.4 Approaches through a merit-based algorithm:**

* Verified Purchase

The common verified purchase feature shows potential buyers that the review is real because they have purchased the product. This label helps other customers assess the reliability of the review.

* Reviewer Sorting

When customers provide feedback, Amazon asks very specific questions about their buying experience and who they are. Reviewers are sorted into three categories to give other consumers a better idea of the review's true feelings.

* Personal profiles of the reviewers

This feature, which is inspired by the meritocratic network, allows users to check the reliability of a reviewer. By looking up their profile, the users can see their overall scores for helpful reviews and the other products they have reviewed.

**2.3 Meritocratic Networks:**

To handle complex content effectively, it is recommended to use meritocratic networks because they offer many advantages. These networks give priority to talent, effort, and achievements, making sure that reliable and trustworthy information becomes more visible and recognized.

We will give a summary of one algorithm and explain the methods and techniques used by the most important merit-based networks:

**Reddit**

Reddit is a platform that prioritizes the feedback of its users when it comes to what it shows first. It has a unique system, where users can like/dislike and this can help to identify what people are interested in, and what they don't. Using this information, Reddit can quickly go through a lot of posts to bring attention to content that other users have found valuable. [6]

Reddits' posts ranking: [7]

* When it comes to submission, the time of posing is very important. Usually, more recent stories are given a higher ranking compared to other ones.
* The count of the upvotes will not be considered in the ranking. For example, a story with 10 upvotes and a story with 50 upvotes will have the same ranking.
* Controversial stories that get similar amounts of upvotes and downvotes will get a low ranking compared to stories that mainly get upvotes.
* If a story has a similar number of upvotes and downvotes it will have a lower ranking compared to stories that mostly received upvotes.

**2.4 Meritocracy and Social Network for Healthy Life project:**

Individuals are rewarded based on their abilities, talent and achievements, and not based on their social or economic background[5]. At the same time, the huge success of Reddit suggests that meritocracy algorithms are a very powerful tool.

The availability of medical information on the Internet is relatively limited compared to many other domains of knowledge. For example, one major issue is that existing platforms do not offer a way to ask important questions and get reliable answers on time.

Our project aimed to make use of the most effective existing approaches and develop new ideas.

**3. Expected achievements:**

In our project we expect to create an online health community that offers reliable and trustworthy information, support, and motivation that enable individuals to take control of their well-being quickly, using recommendation technologies.

**3.1 Technology challenges:** The success of a web application relies on various elements that can be challenging. The ability to overcome these challenges determines whether the web platform achieves success or falls short.

We will present some of these elements:

**3.1.1 Performance:**

Web applications need to be fast-loading, responsive, and reliable to meet the needs of today's online audience. Poor performance can be a result of inefficient code, inefficient databases and much more.

For example, we will use MongoDB because one of its features is that it backs up old, and unused data and removes it, and by doing that we can optimize the performance of the database.

**3.1.2 User experience:**

This is the process of building products that are useful, easy, and enjoyable for people to use. Within an organization, the purpose of user experience design is to consider how a product, app, or website experience makes the user feel, how an interface looks, and how easy it is for the user to accomplish their goals.

In our project, we are going to use ReactJS as the front end since it uses DOM that efficiently updates and renders the user interface. This improves the performance of the application. As a result, we get a smoother and more responsive user experience.

**3.1.3 Security:**

In today’s modern area, It’s not surprising that websites are a top priority for hackers. Therefore, security should be considered in each stage of the software development life cycle. For example, we will encrypt the password using encryption techniques like hash.

**3.2 Criteria of success:**

* A final website where users can seek fast and trustworthy information about health topics.
* The website will let users offer and use services based on its functionalities.
* Fast performance and easy-to-use user interface.
* The ranking system will rank the users’ activities based on their achievements.

**3.3 Unique features:**

* Awarding system: enable ranking the activities of a user such as posts/comments based on their relevance to the topic.
* Asking and answering questions: this feature allows individuals to seek guidance or advice from the community regarding various health-related topics. Users can post their questions, and members can provide answers based on their experiences.
* Service history: within this feature, the users can easily reference their past activities, progress, and achievements.
* Online meetings: every user could start a live meeting using zoom application and allow other users to join the meeting by sharing the link as a post and get benefits from them by discussing useful topics.
* Moderator verification: the website has a moderator role that ensures all information is reliable, thereby ensuring the website provides trustworthy information.
* Image sharing: the user has the ability to upload images, and this feature adds depth to users' experiences and allows sharing experiences with pictures.
* Interaction tools: the user can add likes/comments to the posts. This enables users to engage with each other's posts, offering support, advice, and encouragement.

**4. Technology choices:**

When we talk about developing software webs or applications we need to consider two things: frontend and backend.

For the back-end, we chose several options including NodeJS, and MongoDB. And for the front-end, we chose reactJS.

In our project, we will use React to create the client browser, and then the client will send a request for data to the server. After that, The server will retrieve the data that was requested from the MongoDB database and send it back to the client and the pages will be re-rendered with the received data.

From that point, only the updated data on the server is sent back to the client, and then React re-renders the specific components of the page that require updating.

The diagram below shows the technologies that we chose in the front-end and back-end:

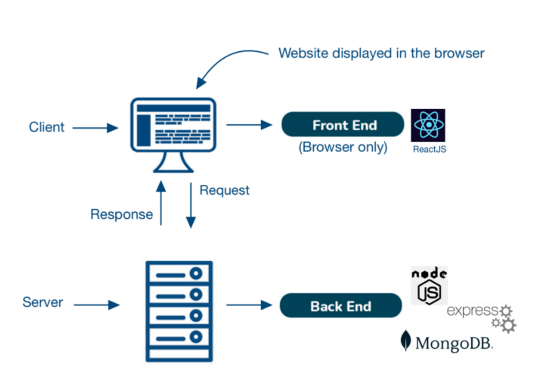
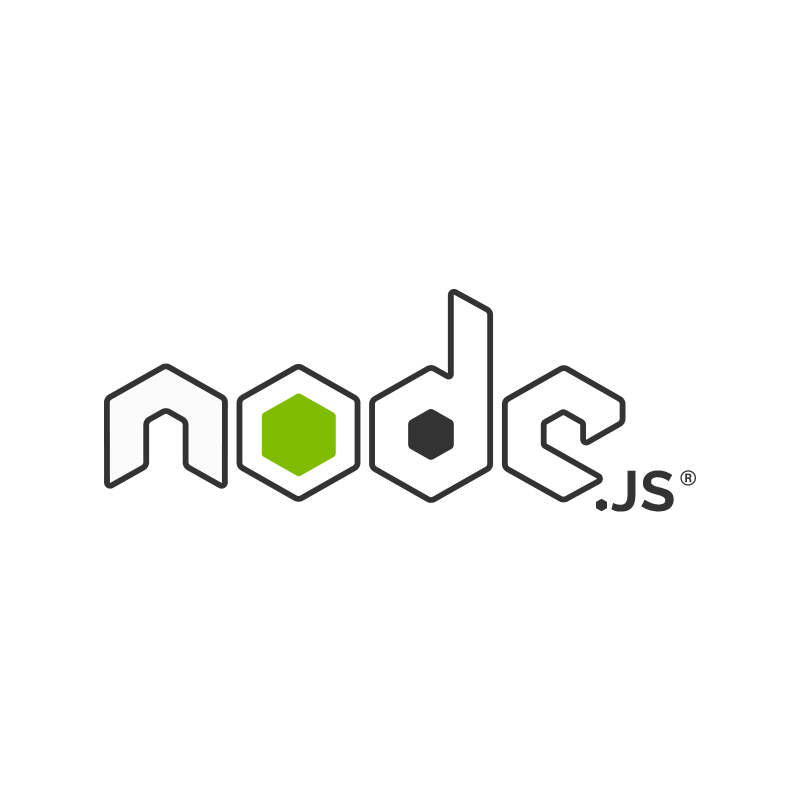


Figure 1

**4.1. Technologies for Back-End:**

**4.1.1 NodeJS**

****

Node.js is an open-source and cross-platform JavaScript runtime environment, it allows developers to write server-side code using javaScript. It runs the V8 JavaScript engine, the core of Google Chrome, outside of the browser. This allows Node.js to be very performant.

It provides a set of asynchronous I/O primitives in its standard library that prevent JavaScript code from blocking and generally. Moreover, A Node.js app runs in a single process, without creating a new thread for every request. [8]

Now that we’ve covered what NodeJS is, let’s take a closer look at some of its key benefits:

NodeJS benefits

1. High performance.

2. Easy to learn and quick to adapt.

3. Offers easy scalability.

4. It targets all major platforms.

5. Flexible

NodeJS drawbacks

1. Unable to process heavy computational tasks.

2. Unstable API.

3. Lacks a strong library support system.

In conclusion, NodeJS is an excellent option for developers who want to build highly scalable applications with JavaScript. Most importantly, it can be used for back-end servers and front-end.

**4.1.2 Express NodeJS**

****

Express is a small framework that sits on top of Node.js’s web server functionality to simplify its APIs and add helpful new features.

It is used to build a single page, multipage, and hybrid web application. It provides a powerful set of features to efficiently manage routes, requests, and views along with a beautiful boilerplate for your web applications.[11]

Moreover, several such platforms are unable to handle a higher level of requests, but with the help of Express.JS, you would be able to handle requests efficiently as it offers you the support of I/Q request handling.

Express benefits

1. Fast

2. Easy to learn.

3. Saves a lot of coding time.

4. Fast server-side development.

5. Makes it easier as it identifies the exact part where bugs are.

Express drawbacks

1. Error messages are challenging to understand.

2. Issues with the callbacks.

Finally, express is a flexible framework that helps developers create web applications quickly.

**4.1.4 MongoDB**

****

MongoDB is an open-source document-oriented NoSQL database, Data in MongoDB is stored in documents with key-value pairs instead of using tables and rows. This makes the data more flexible when compared to SQL databases. MongoDB is used to store a large scale of data while still performing rapidly. It also has a flexible data model that enables you to store unstructured data. [10]

Features:

MongoDB offers several features, including:

1. Flexible.

3. Provides high performance.

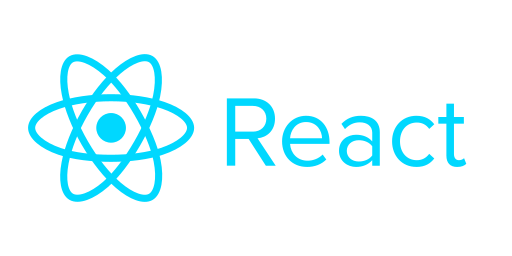
4. Stores files with all sizes easily without complicating the stack.

5. Easy to administer in the case of failures.

6. Integrates with a wide range of tools and platforms.

**4.2. Technologies for Front-End:**

**4.2.1 ReactJS**

****

React is an open-source, JavaScript library for developing user interfaces (UI) in web applications achieving faster results with less code. [13]

React allows us to create reusable components; each component is an individual piece of a final interface. As a result, ReactJS combines the speed of JavaScript.

The primary role of react in the application is to handle the view layer of that application and make sure they are displayed efficiently. [13]

**5. Engineering Process:**

**5.1 Define app purpose:** in the beginning, we defined the purpose of our website, and the problem that the website is going to solve: build an online health community where users can find reliable information.

**5.2 Algorithmic challenge:**

We faced several challenges, we will explain the task and our solution for the three main challenges.

**5.2.1 Awarding system:**

* The task: the system should encourage the users’ activities (adding posts/comments) to motivate them to stay active on the social network.
* Our solution: In the system, users have the opportunity to add posts and comments. Each user will receive bonus points based on their activities, such as adding comments or posts, contributing reliable content, answering questions, and more.

**5.2.3 Asking and answering questions:**

* The task: our task is to answer optimal and reliable answers for a wide range of questions as much as possible.
* Our solution: users can estimate the relevance of an answer on the website. Moderators will verify its reliability and award extra points accordingly.

**5.3 Package Diagram:**

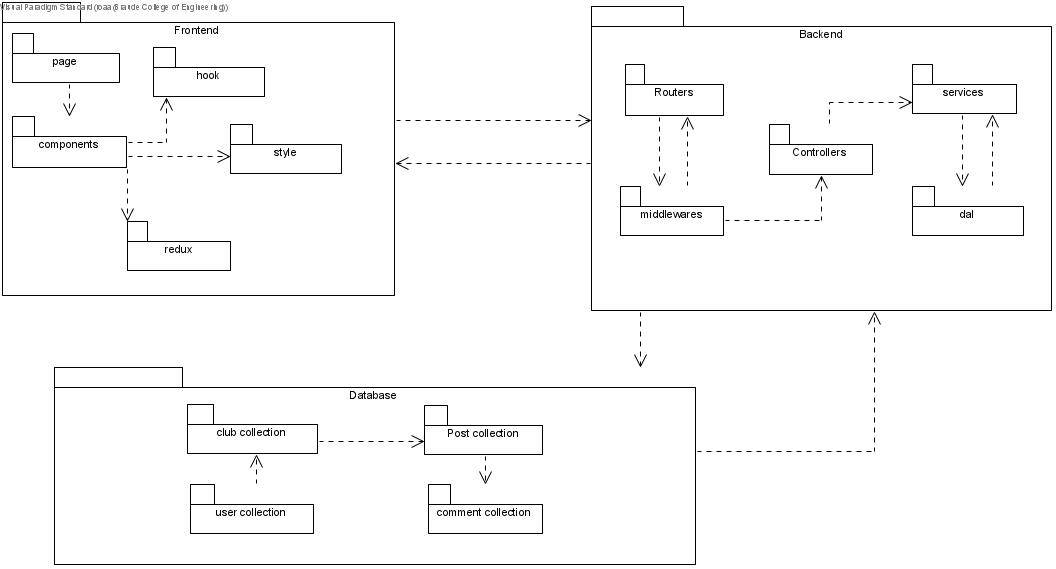
****

Figure 2

**5.4 Activity Diagram:**

5.4.1 This activity diagram describes the flow of adding a new club:

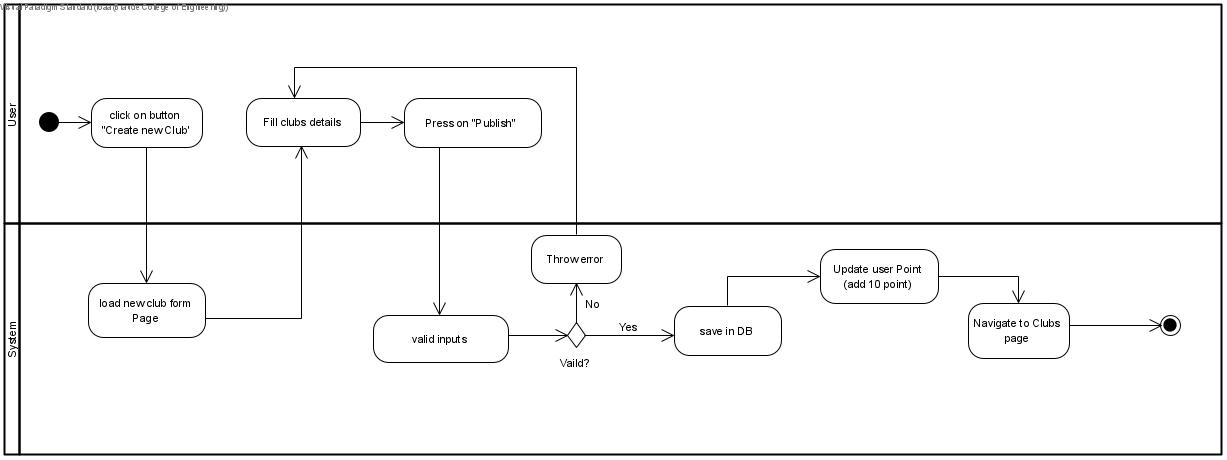


Figure 3

5.4.2 This activity diagram describes the flow of adding a new comment to a post:

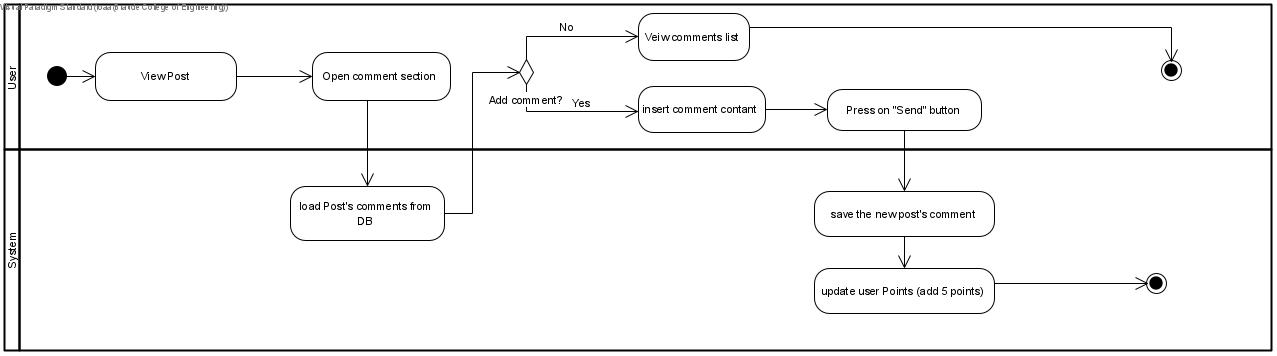


Figure 4

5.4.3 This activity diagram describes the flow of a moderator checking the reliability of a post in the club:

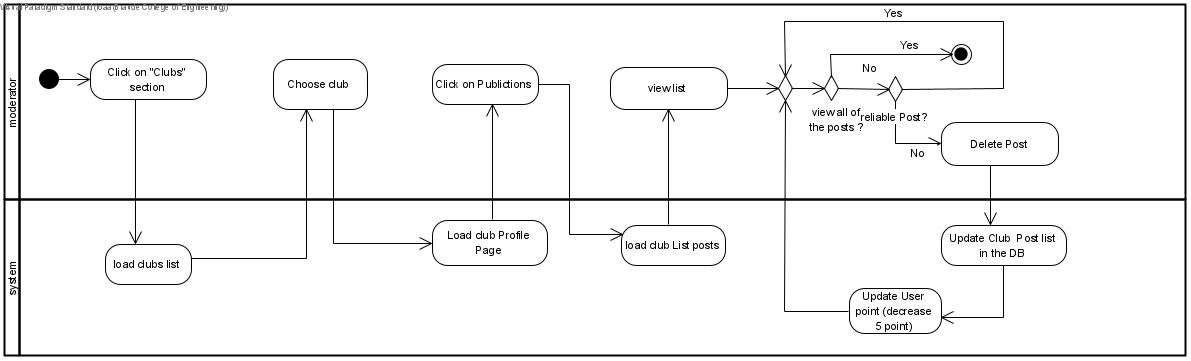


Figure 5

**6. Implementation process:**

During the implementation of our web development, we took our designs and plans and turned them into real, working sites. We used tools like HTML, typescript, and CSS to create the pages that users interact with. We used ReactJs and NodeJs to develop the front-end and back-end to make sure everything runs smoothly.

We started by breaking down our website design into small components. Each component was carefully planned so they could work together smoothly. This helped us keep our code organized and make sure each part of the website does its job properly.

To set up the development environment, we used the Visual Studio Code, which provided a wide range of tools that helped to keep the code organized.

As part of our integration process, we created user accounts and checked that our website worked right with the necessary services to ensure the website runs smoothly.

As we worked on building our website, we made sure to test our code carefully to ensure it works properly and wouldn't cause any issues. We used different methods, like unit testing, and checked each component separately, to make sure everything was right. This way, we can be confident that our website will run smoothly for users.

We worked on making our website run faster and smoother so that users have a better time using it and an excellent user experience. We made our code better and used techniques to make it work faster. By making performance our top priority, we wanted to give users a fast and smooth interface, making them satisfied with our website overall.

During the implementation phase, we carefully wrote code, connected services, tested everything, and made it run smoothly. This set a strong base for the next steps of testing and making improvements.

**7. Challenges during the implementation process:**

* Choose the most appropriate development platform: we needed to find a platform that could handle the complex interactions and data management required for users to connect, share experiences, and offer support effectively.

We ultimately decided on using React, Node.js, and MongoDB. React provided us with a dynamic and interactive user interface, making it easy for users to engage with the community features. Node.js allowed us to build a scalable backend system to handle user authentication, data storage, and real-time interactions. MongoDB, a NoSQL database, was chosen for its flexibility and ability to handle large amounts of unstructured data efficiently.

* Implementing the moderator role: adding a moderator to our project was challenging. A moderator's job is to make sure the information shared in our health community is trustworthy. We had to figure out how we could implement this role in our project. Then, we had to build tools for moderators to check posts and take action if needed. By doing this, we succeed in updating the users’ points and by that, we encourage the users to share reliable information. Even though it was hard, having a moderator helps keep our community safe and reliable.
* Finding the right software: During the project, we had technical issues in finding the right software to run the model. We ended up using Visual Studio Code. We thought about several ones but didn't find them suitable. But Visual Studio Code turned out to be the right tool for the job. It helped us write and test the code effectively, making the implementation stage smoother.

**8. Testing plan:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test. number** | **Test description** | **Expected Result** | **Comments** |
| 1 | The system allows users to sign-up for the system | The user will be automatically logged in | Display a message “Successfully signed up” |
| 2 | The system doesn't allow the user to sign-up with an existing email | The sign-up failed, and an error message will be displayed | Display an error message “This email already exists” |
| 3 | The system allows existing users to log in to the system | The user will be logged in, and the home page is loaded |  |
| 4 | The system doesn't allow non existing users to log-in to the system | The log-in failed, and an error message will be displayed | Display an error message “This username is not found” |
| 5 | The system doesn't allow a user to log in with a wrong password/empty details | The log-in failed, then an appropriate  message will be displayed | The fields will be marked in red |
| 6 | The system allows the user to add post | The post is loaded and displayed |  |
| 7 | The system doesn't allow the user to create an empty post | Failed to create a post | The required fields will be marked in red |
| 8 | The system allows the user to a delete a post | The post is removed from the community | The post is deleted and is not found anymore |
| 9 | The system allows the user to update a post | The post is updated and loaded again in the community |  |
| 10 | The system allows the user to add a comment | The comment is loaded and displayed |  |
| 11 | The system doesn't allow the user to add an empty comment | Failed to create a comment and the required fields will be marked in red |  |
| 12 | The system allows the user to delete a comment | The comment is removed from the list | The comment is deleted and is not found anymore |
| 13 | The system allows the user to update a comment | The comment is updated and loaded again list |  |

**8.1. Login screen:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | case | Input | Result |
| 1 | Username field validates empty input | Empty username | Displays the appropriate error message |
| 2 | Password field validates empty input and displays the appropriate error message | Empty password | Displays the appropriate error message |
| 3 | Login button triggers the form validation | username: reem4 Password: Rk1234 | Pass |
| 4 | Login button is pressed with valid email and password, the `loginUser` method is called with the correct parameters | username: reem4 Password: Rk1234 | Pass |
| 5 | Join now button navigates to the `SignupScreen` |  | Pass |

**8.2. Sign-up screen:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | case | Input | Result |
| 1 | Field validate empty input | Empty fields | Displays the appropriate error message |
| 2 | Email field validates the input according to the defined | Email without @ | Displays the appropriate error message |
| 3 | Password field validates the input according to the defined rules | weak password | Displays the appropriate error message |
| 4 | Clicks the create account button adds the user to the user to the system | First name  Last name  Username  Email  Password  Confirm Password | Pass |
| 5 | The password field doesn’t match the Confirm Password field | Password:Ab1234  Confirm password:123 | Displays the appropriate error message |
| 7 | The system doesn't allow the user to sign-up with an existing username | Enter an existing username: reem4 | Displays the appropriate error message |

**8.3. Unlogged user screen:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | case | Input | Result |
| 1 | Guest tries to add like to post | Clicks on like button | The operation will fail since the guest isn't logged in |
| 2 | Guest tries to add comment and fails | Clicks on comment button | The operation will fail since the guest isn't logged in |
| 3 | Guest tries to join club | Clicks on subscribe button | The operation will fail since the guest isn't logged in |
| 4 | Guest tries subscribe to author page | Clicks on subscribe button | The operation will fail since the guest isn't logged in |
| 5 | Guest can view club profile | Clicks on club card | Pass |
| 6 | Guest can view author profile | Clicks on author card | Pass |

**8.4. Logged user screen:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | case | Input | Result |
| 1 | User can add like to post | Clicks on like button | Pass |
| 2 | User can add comment to post | Clicks on comment button | Pass |
| 3 | User can’t add an empty comment to posts | Empty fields comment | Displays the appropriate error message |
| 4 | User can view comments | Clicks on subscribe button | Pass |
| 5 | User can delete comment | Clicks on delete button | Pass |
| 6 | User create a new club with empty input | Empty fields | Displays the appropriate error message |
| 7 | User can view the clubs | Clicks on clubs button | Pass |
| 8 | User can view club profile (posts, moderators and about club) | Clicks on club card | Pass |
| 9 | User can subscribe to a club | Clicks on subscribe button | Pass |
|  | User can delete his club | Clicks on delete button | Pass |
| 10 | User can view all the Authors in the system | Clicks on author button | Pass |
| 11 | User can view authors profile (posts, comments and about author) | Clicks on author card | Pass |
| 11 | User can edit his profile | Clicks on edit profile button | Pass |
| 12 | User can join/create zoom meeting | Clicks on zoom button | Pass |
| 13 | User can add a new post | Clicks on Create new post button | Pass |
| 14 | User tries to add post with empty input | Empty fields | Displays the appropriate error message |
| 15 | User can delete a post | Clicks on delete button | Pass |

**8.5. Moderator screen:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | case | Input | Result |
| 1 | Moderator can delete post from club | Clicks on delete button | Pass |
| 2 | Moderator can delete comment | Clicks on delete button | Pass |

**9. Results and conclusion:**

Our health community project had a lot of changes and improvements. It's proven to be effective in bringing together people who need support with those who can provide it. Whether someone is seeking advice, sharing experiences, or offering support, our platform connects them with the right people. After detailed planning we’ve successfully made a user-friendly platform that helps people to get reliable advice and share experiences.

We did research and looked closely at what people wanted. Then, we used that information to make our website just right for them. We listened to what they needed and made sure our website went above and beyond what they expected by providing reliable information to them.

As we worked on our project, we faced some tough problems that needed smart solutions and teamwork. We thought ahead to fix any issues and make sure our website worked smoothly. The diagrams helped us to understand how everything should work together. This helped us develop a website that is easy to use, so people can make sure they can rely on the information that are provided in the website.

To make sure our project was good and it was working as expected, we did lots of testing in each stage. We used different ways to find and fix any problems with the website, like bugs or things that weren't working as expected.

In summary, our social network for healthy life effectively makes sure that the information that the people are providing are relatable and other users can rely on them. This way the people can learn from other users’ experiences and share advice in an easy and effective way. Also, it has features to encourage the users to be active.

The hard work we put into making this website, which makes interactions more effective for users, shows in how well it turned out. We're sure our community will be really useful, making things better for everyone who uses it.

**10. Lessons that we learned:**

While working on the community, we learned some important things that helped us understand how to make software better. First, we found out how important it is to do good research and listen to the people who will use this website. This helped us make the community fit what they wanted and make them satisfied when they use it.

Also, we found out how helpful diagrams are in making the websites’ functionalities and design better and easier to use. In addition, We found out that managing time well is really important. We also learned that we should always be ready for problems and things we didn't expect.

Lastly, we learned how important it is to test the website well to make sure it works great and makes users happy.

**11. User Guide:**

**11.1 Operational Process:**

**11.1.1 Login:**

On this screen, the user can log in using their email. This brings the user to enter his details and then the user needs to click on the blue button to login. After that, the process will check the login parameters if valid or not.

**11.1.2 Log-in:**

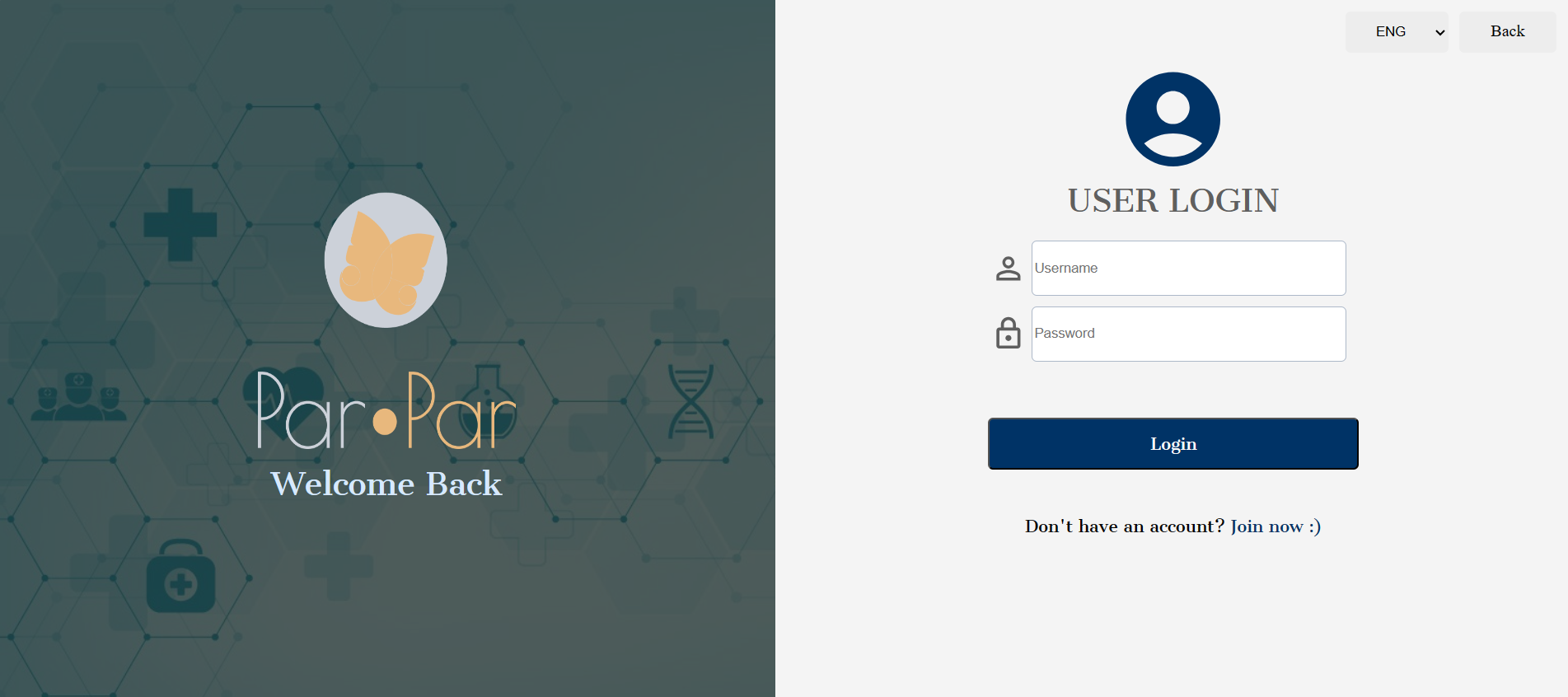
****

Figure 6

**11.1.3 Register:**

If the user is new, he can register to the web by clicking “join now”. After clicking on “join now”, the register screen will be loaded, and the user needs to enter his details. The next step is to click on create account in order to create a new user.

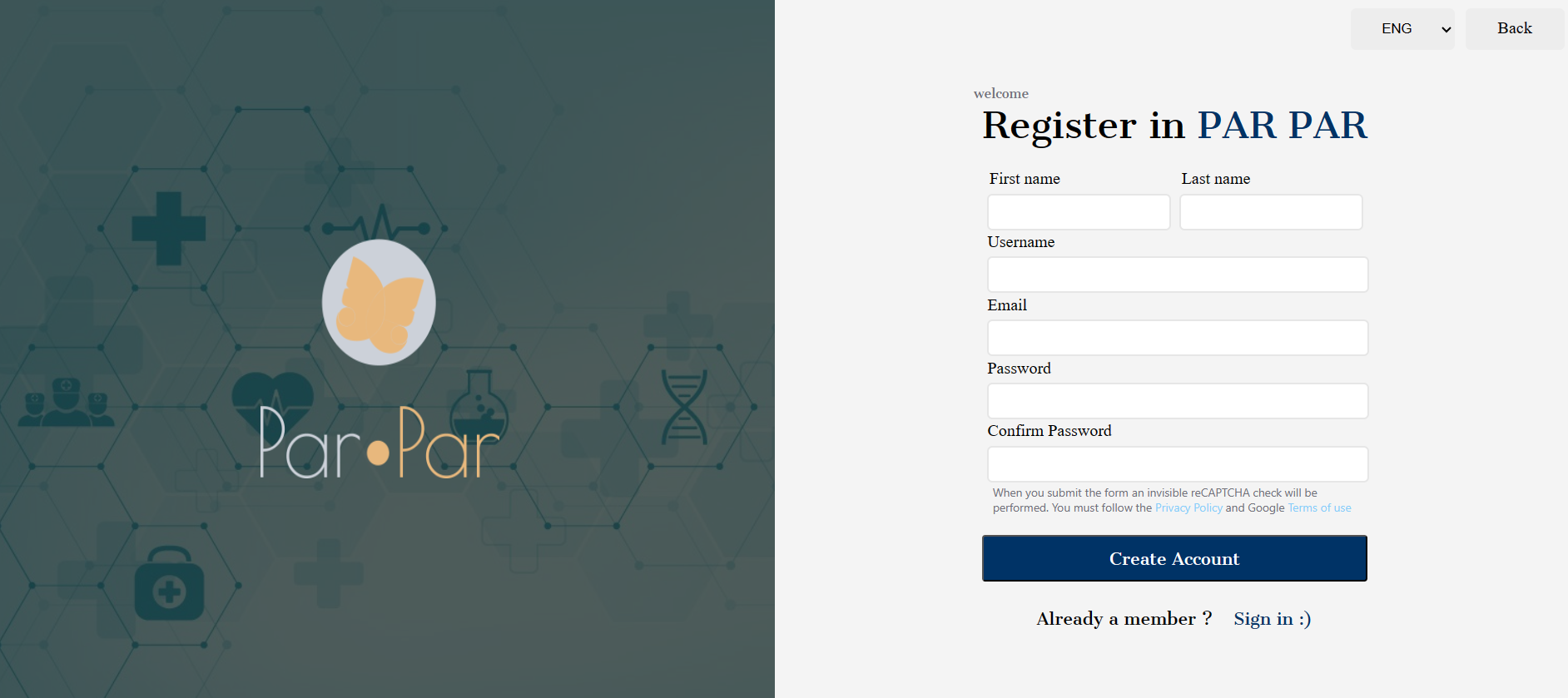
****

Figure 7

In this website, we have screens for logged users and unlogged users.

**11.2. Unlogged user:**

**11.2.1 Home screen (posts screen):**

On this screen, an unlogged user can view posts, comments,clubs, authors, active users and clubs.

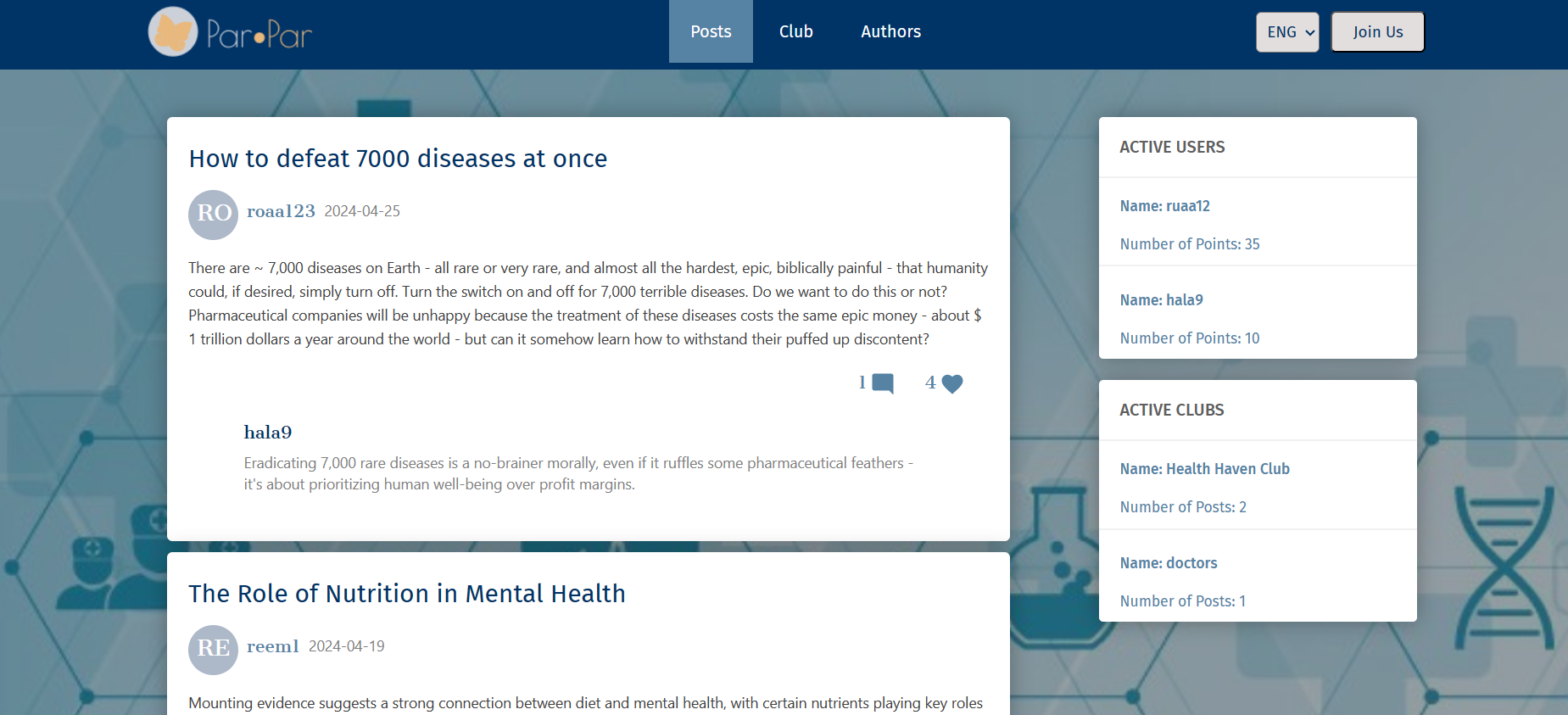


Figure 8

**11.2.2 Club screen:**

By clicking on club in the header, the club page will be loaded where all the clubs will be displayed. In addition, the number of members in each club.

If the unlogged user tries to click on the subscribe button, the login page will be loaded.

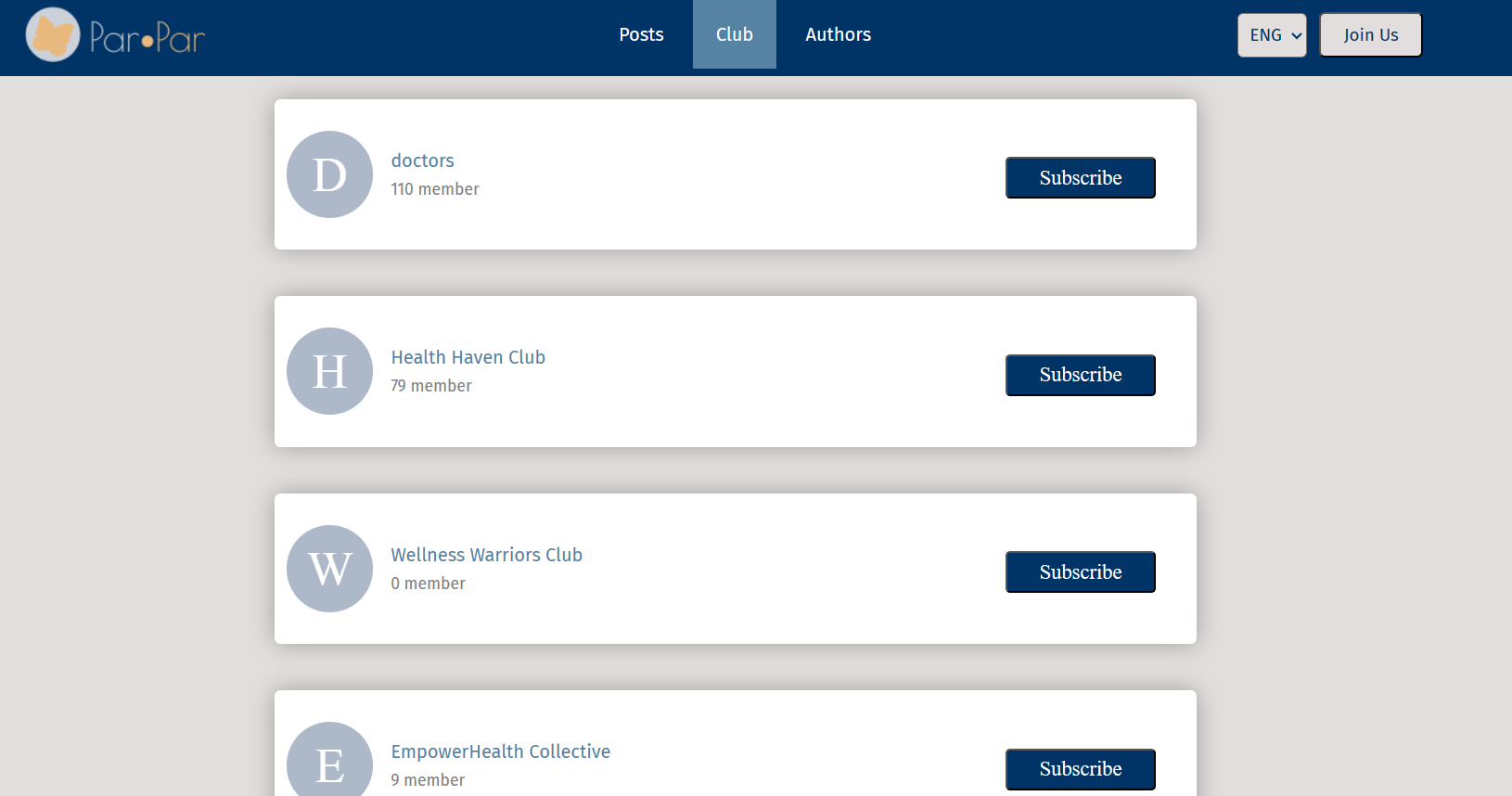


Figure 9

**11.2.3 View club profile - About club:**

By clicking on a club's name the user can view the club’s profile. The club's profile screen has 3 fields. The default screen is the “About club”. In this screen the user can view about a club, their interests and the registered date.

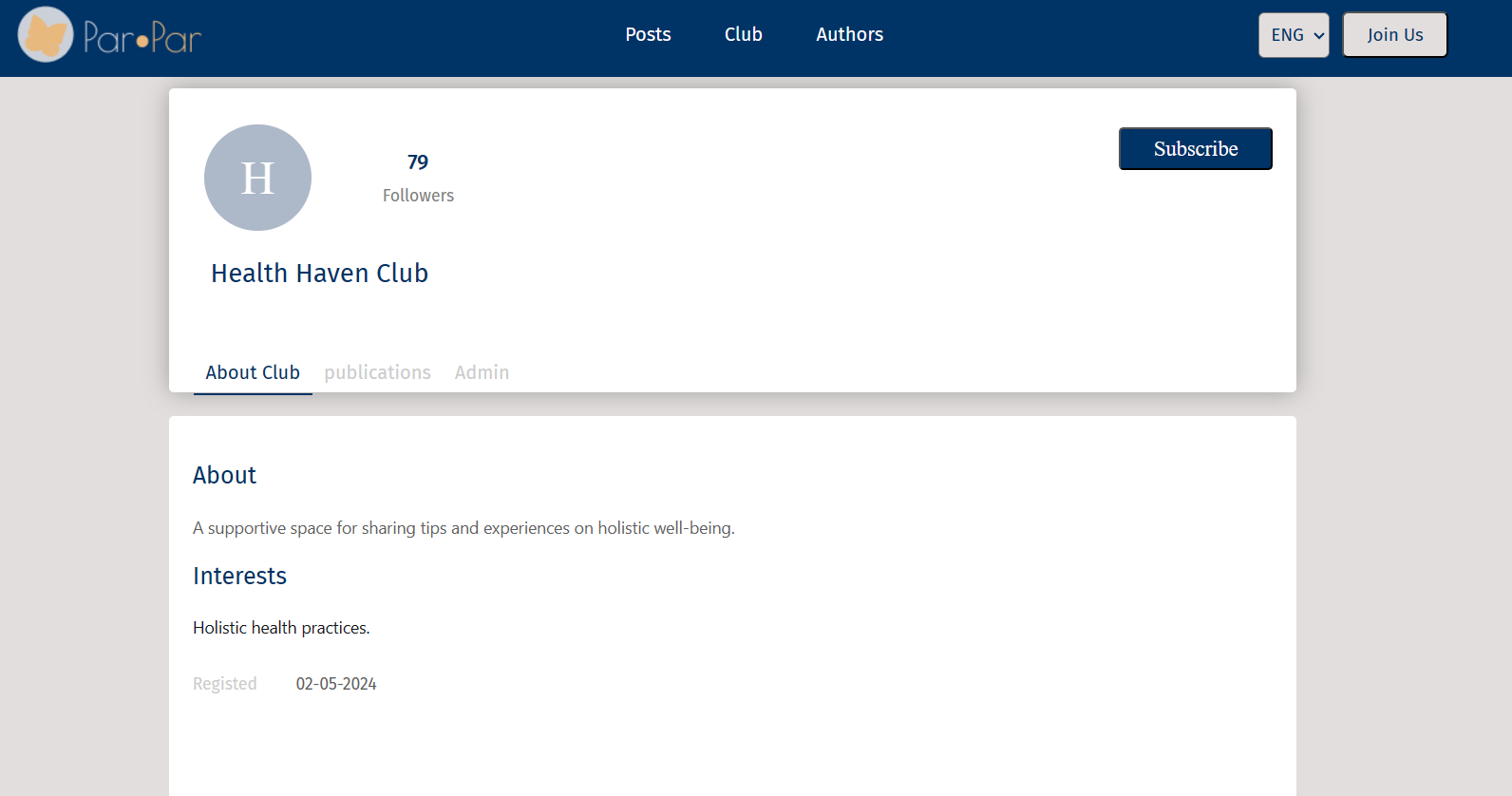


Figure 10

**11.2.4 View club profile - Publications:**

By clicking on Publications, the user can view the club's posts.

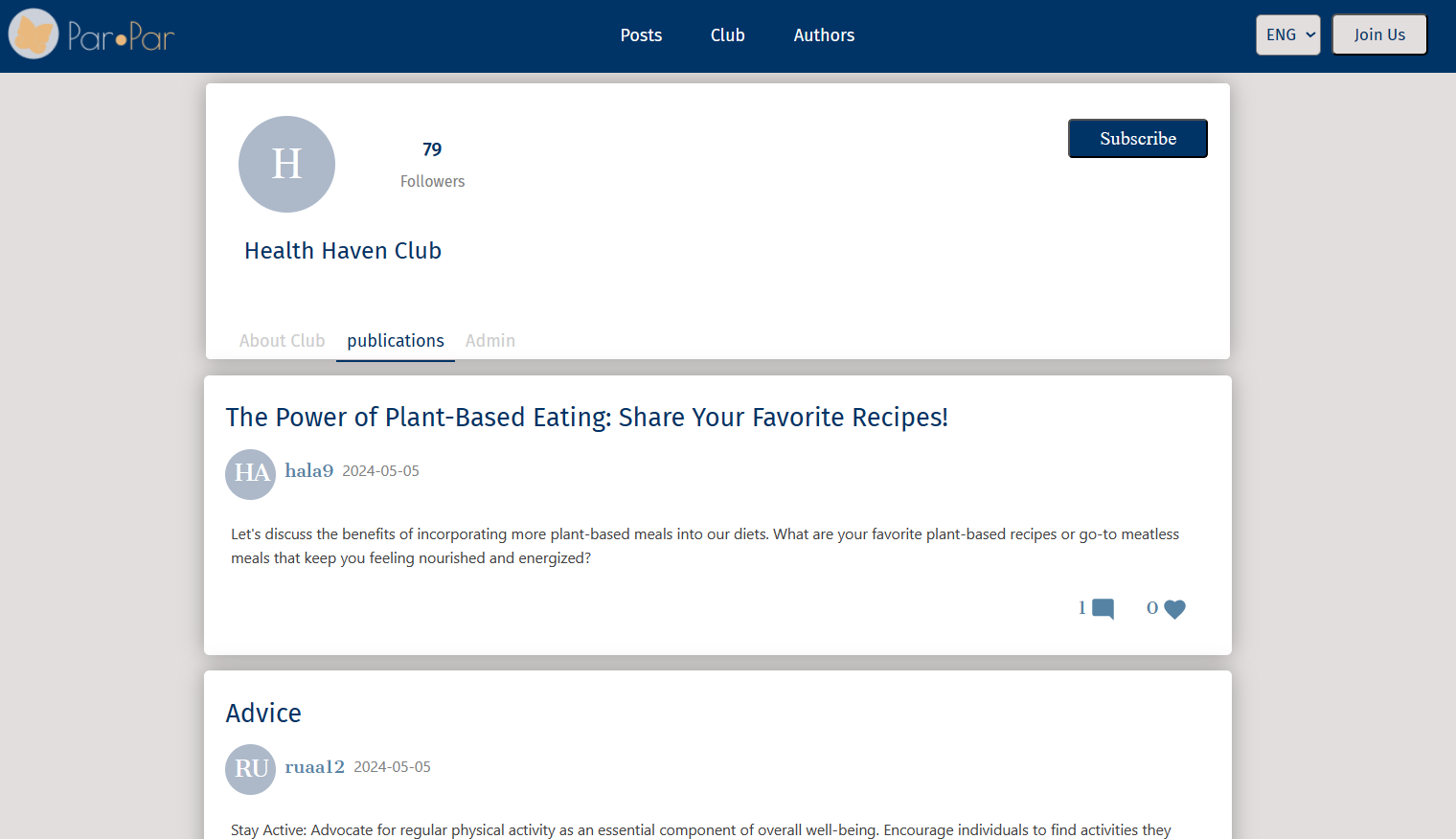


Figure 11

**11.2.5 View club profile - Admin:**

the user can view the club's Admin By clicking on Admin.

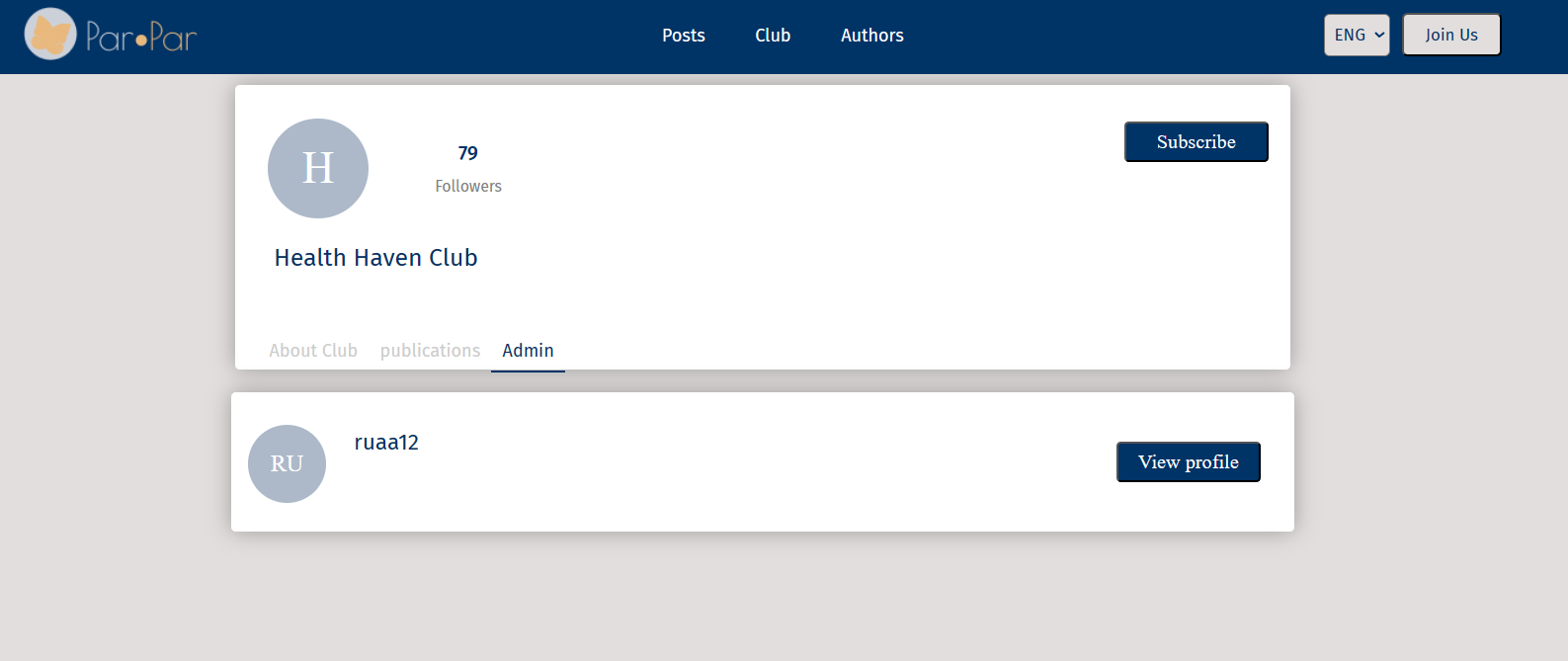


Figure 12

**11.2.6 Author screen:**

By clicking on author in the header,the author page will be loaded where all the authors will be displayed. In addition, the number of points of each author.

If the unlogged user tries to click on the subscribe button, the login page will be loaded.

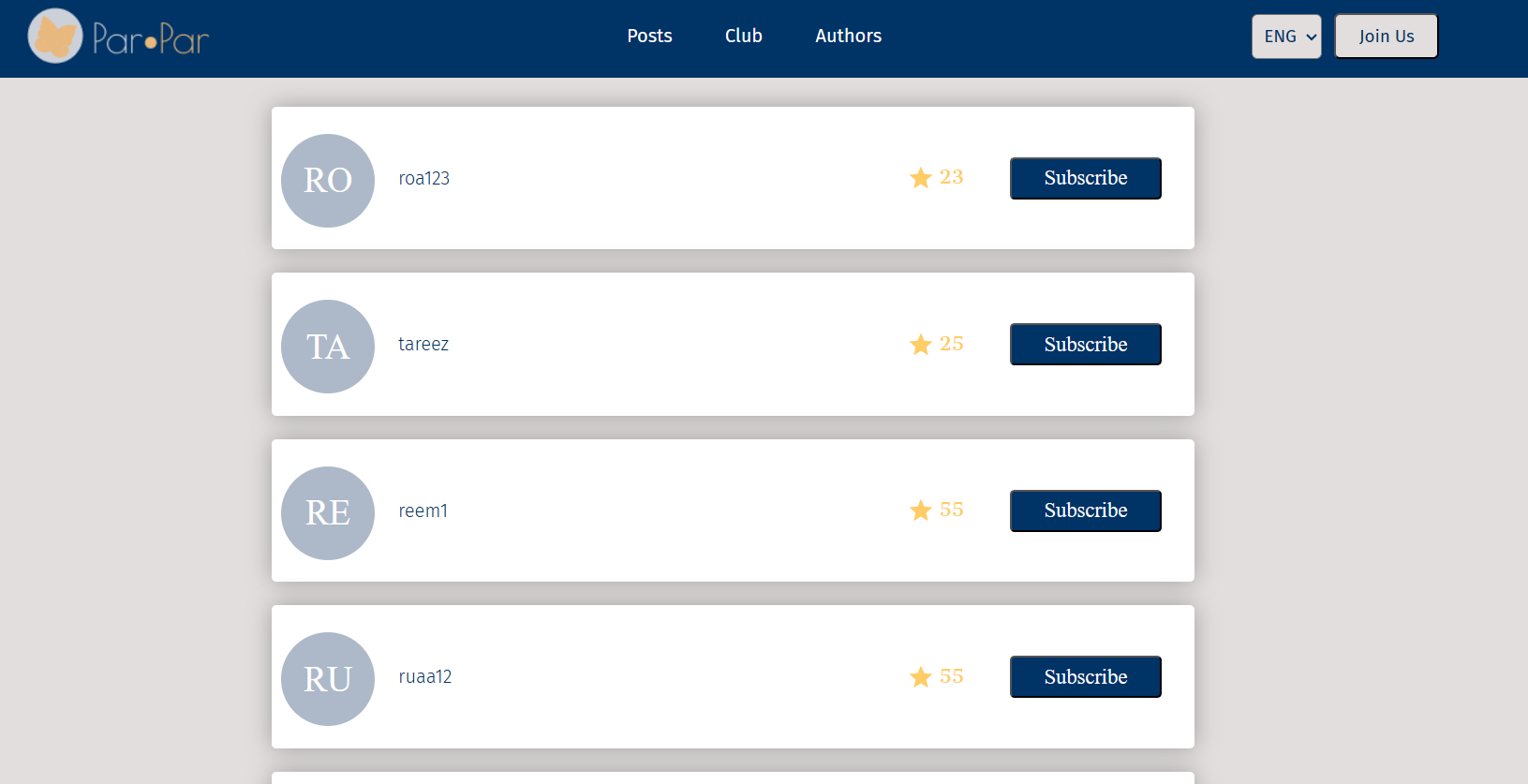


Figure 13

**11.2.7 View author profile - Publications:**

By clicking on an author's name, the user can view the author's profile. The author's profile screen has 3 fields. The default screen is the “Publications”. In this screen the user can view the author's posts and the comments on each post.

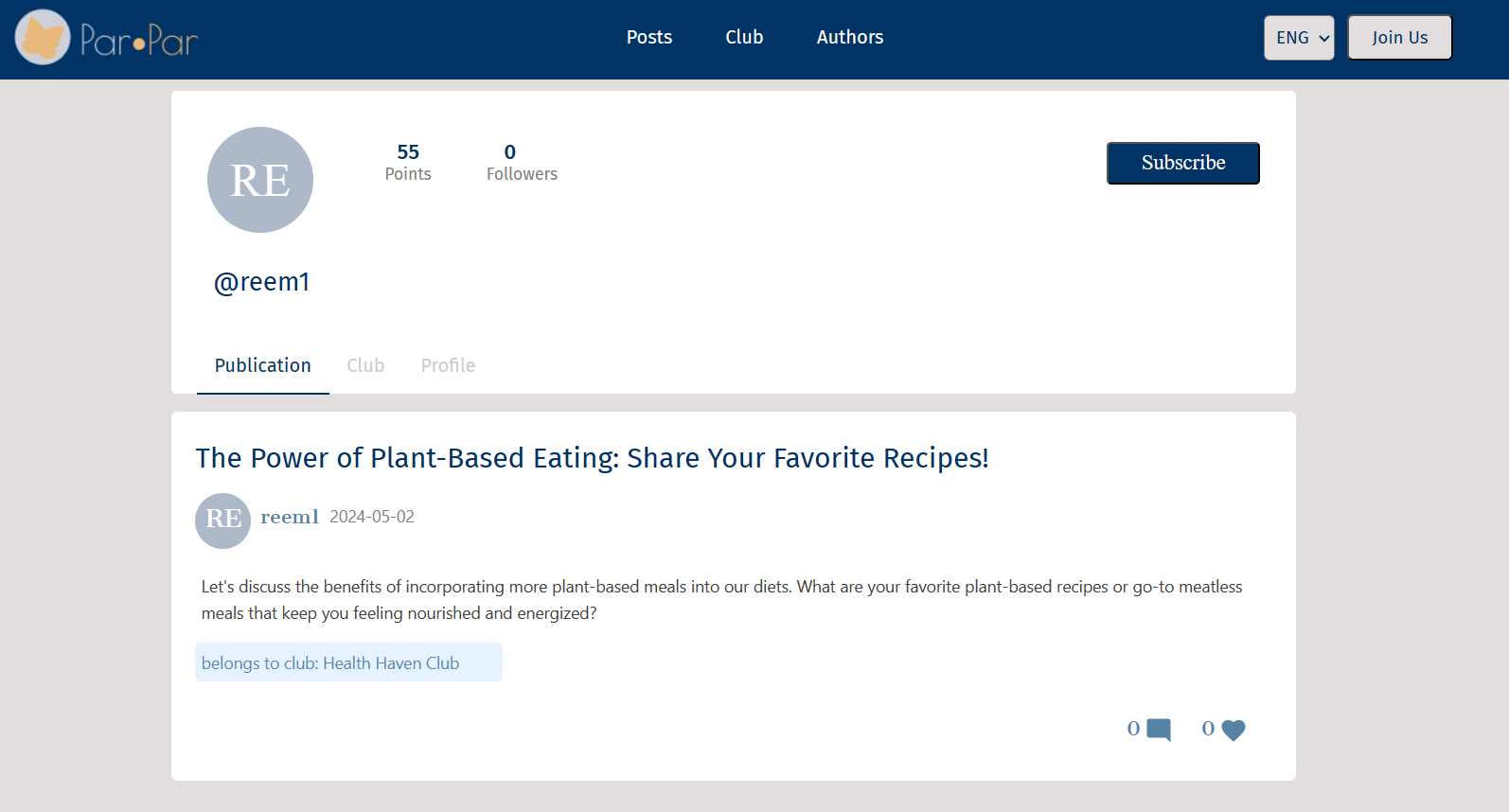


Figure 14

**11.2.8 View author profile - club:**

By clicking on club, the user can view the author’s clubs.

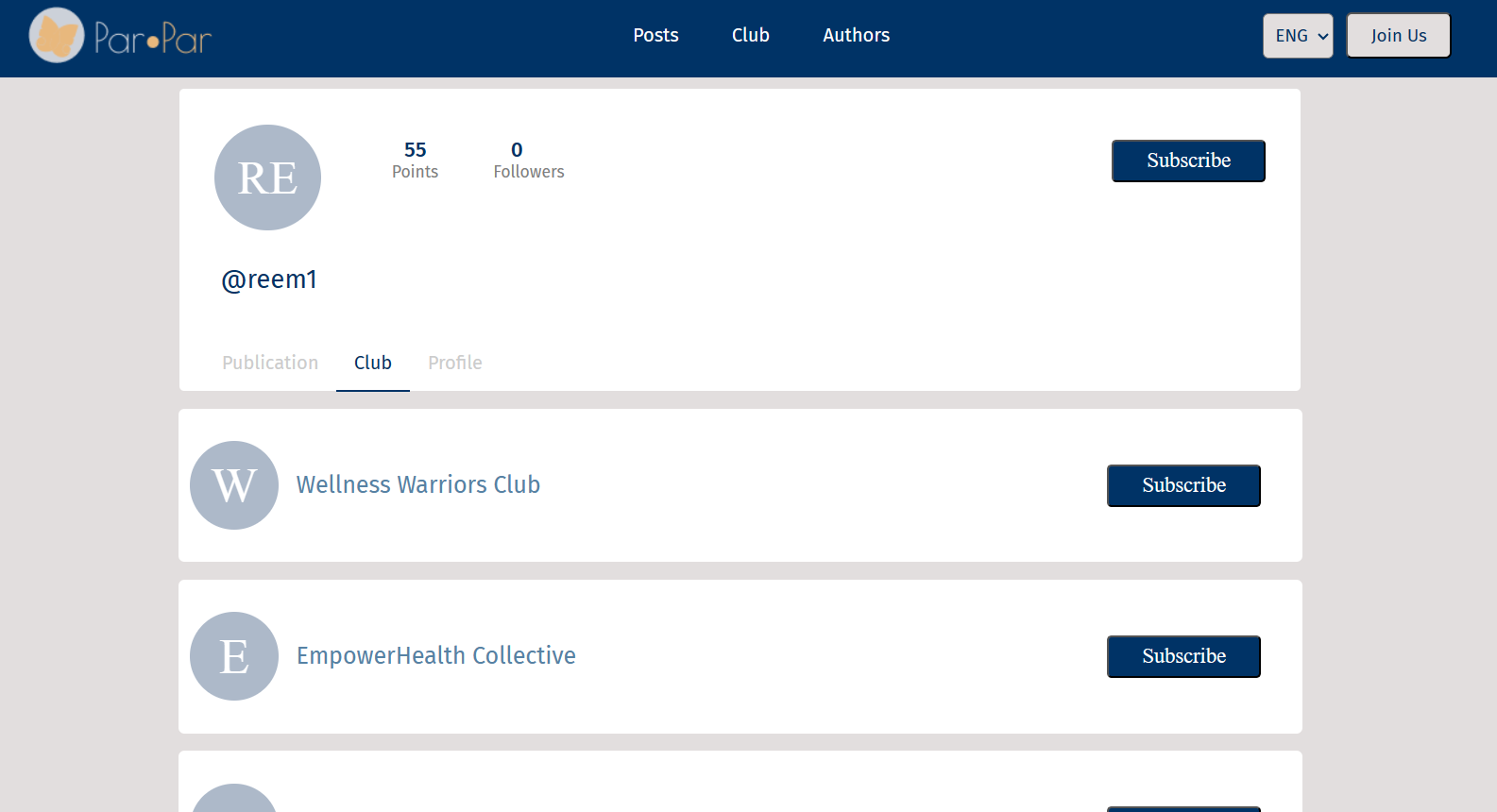


Figure 15

**11.2.9 View author profile - profile:**

the user can view the author's profile by clicking on profile where he can view more details about the author.

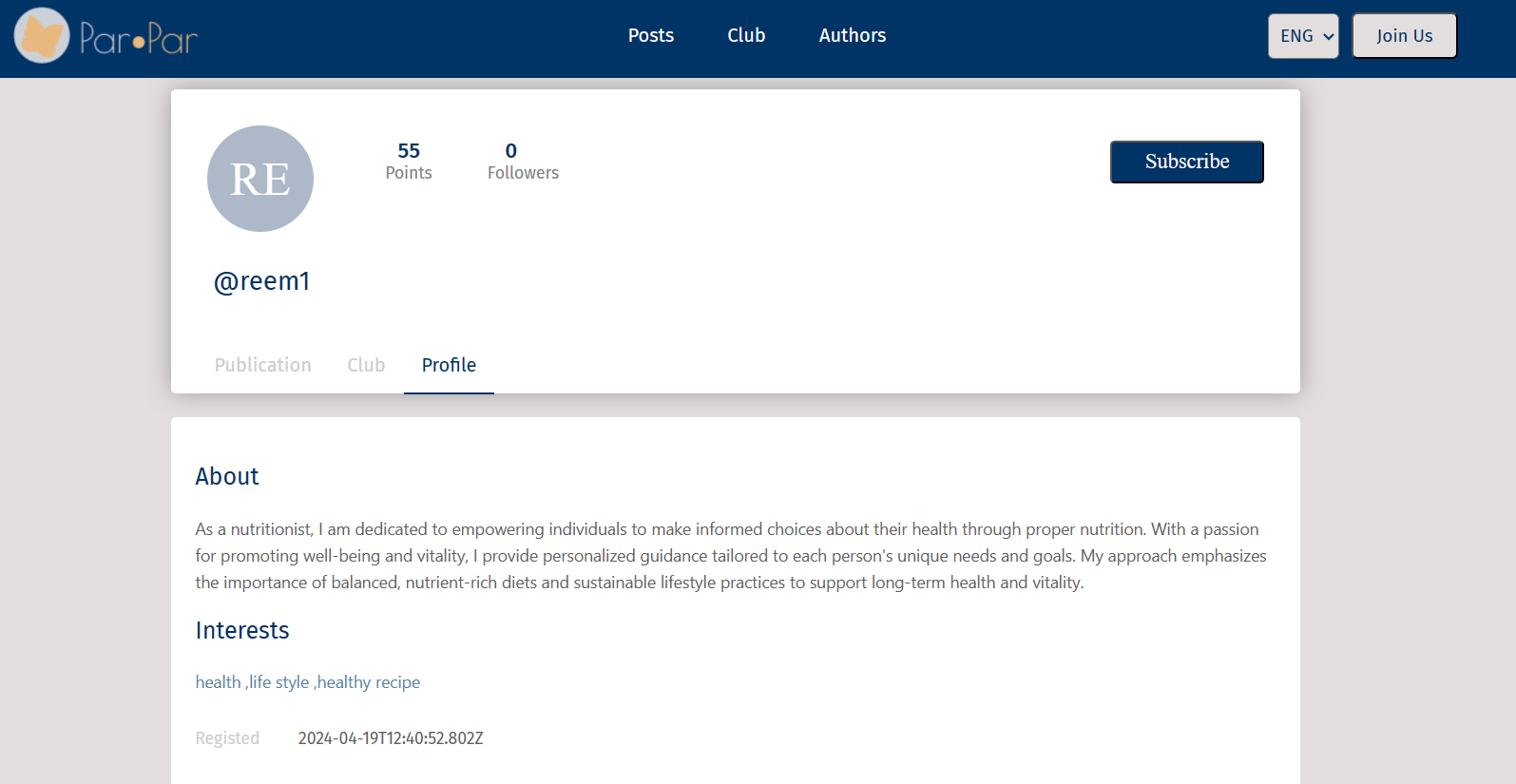


Figure 16

**11.3. Logged users:**

Logged user can do all the features that the unlogged user can do, in addition:

**11.3.1 Home screen (posts screen):**

In this page the user can add comments/likes to a post. Also, he can delete a post that he created.

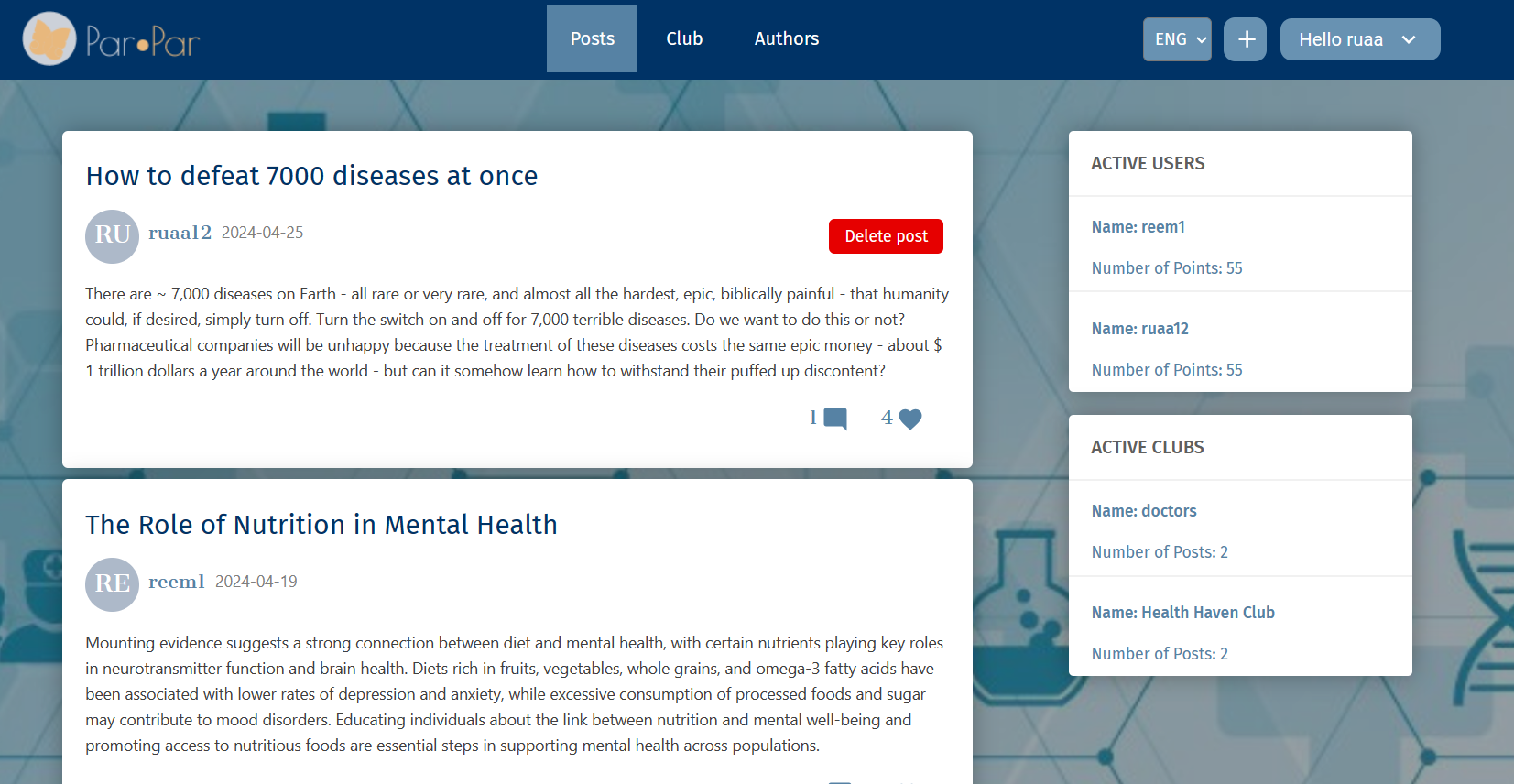


Figure 17

**11.3.2 Club screen:**

The user can click on the blue button to subscribe/unsubscribe to a club.

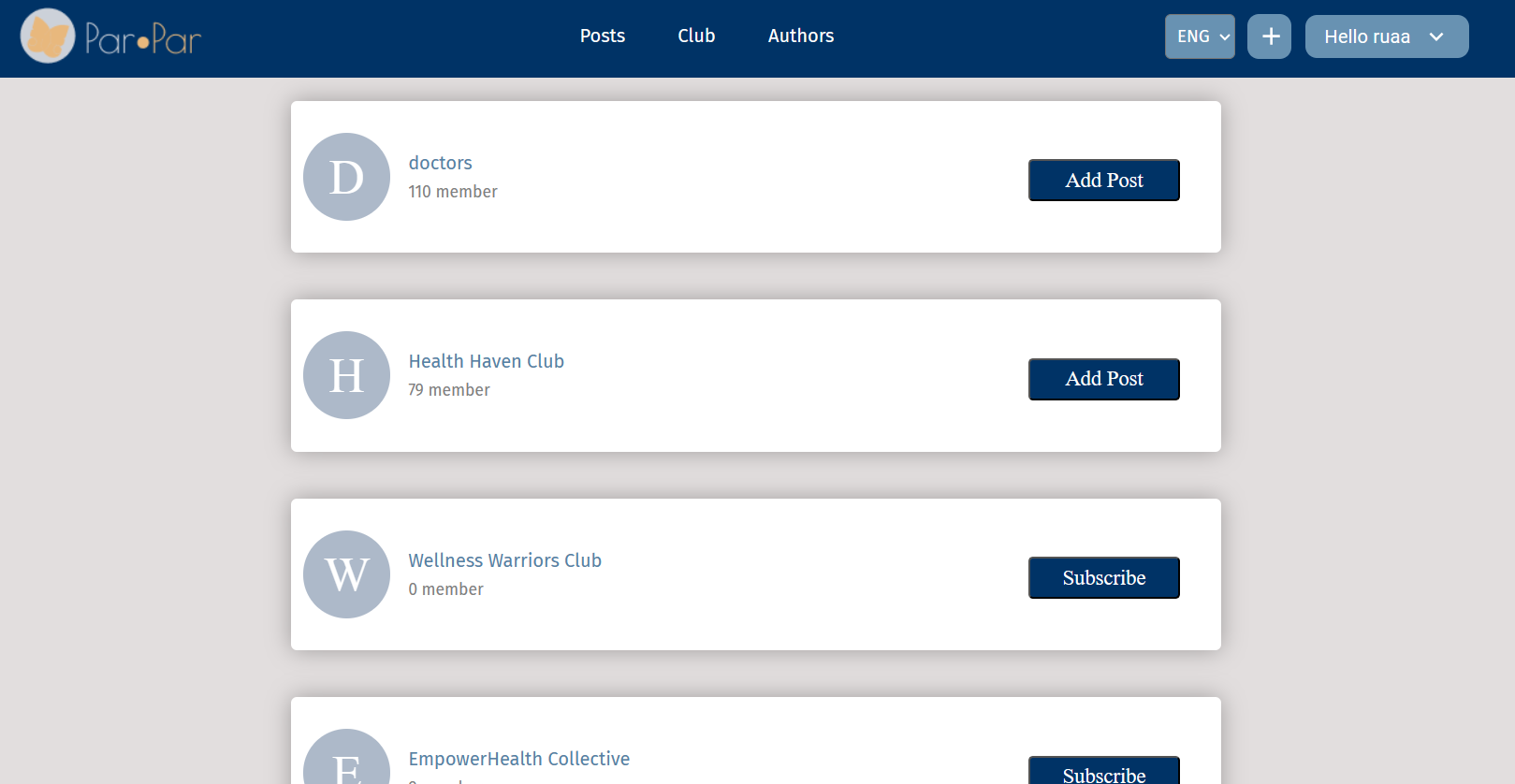


Figure 18

**11.3.3 View club profile - Publications:**

The user can add comments/likes to posts in the clubs.

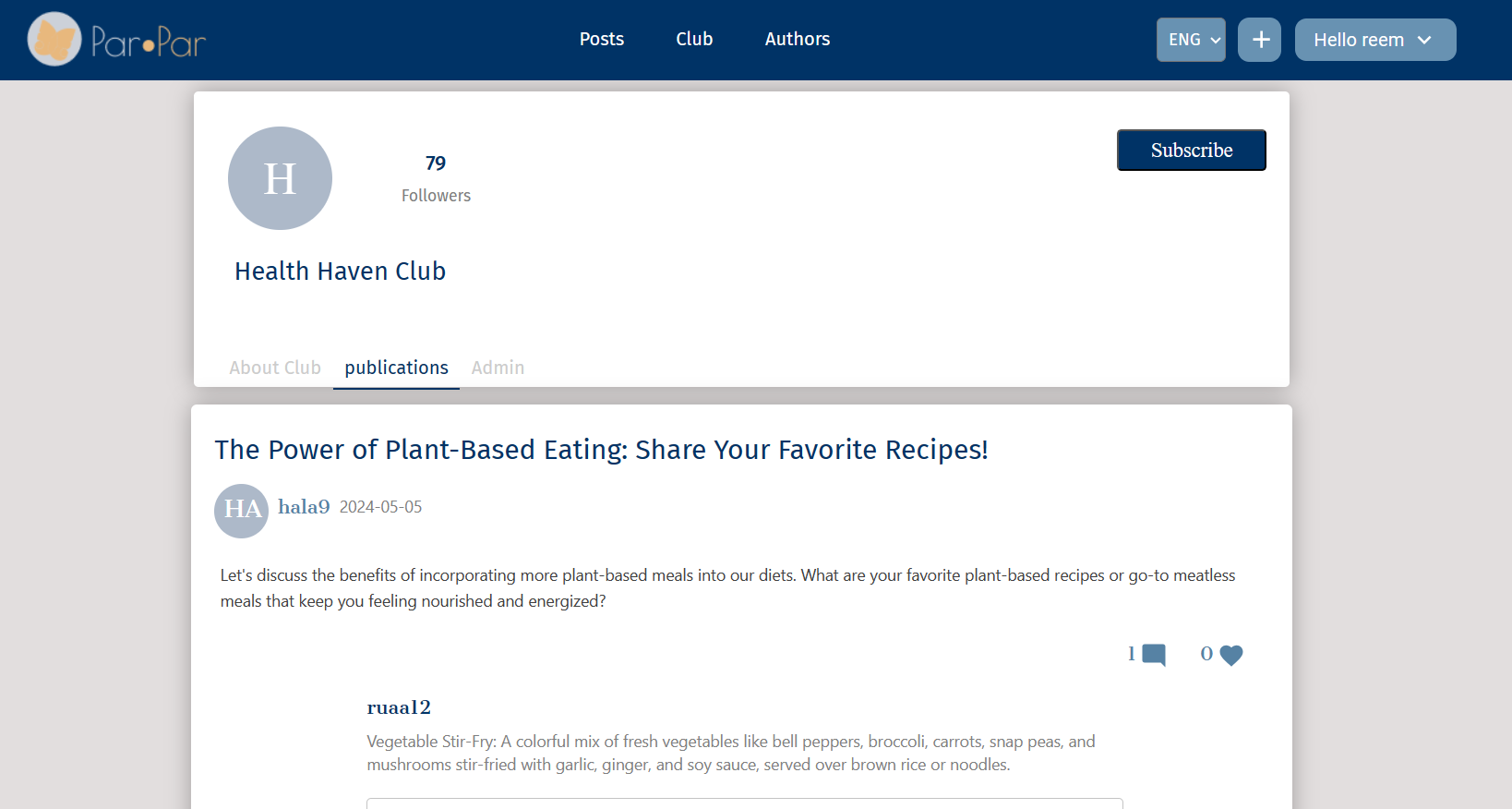


Figure 19

**11.3.4 Author screen:**

The user can click on the blue button to subscribe/unsubscribe to an author.

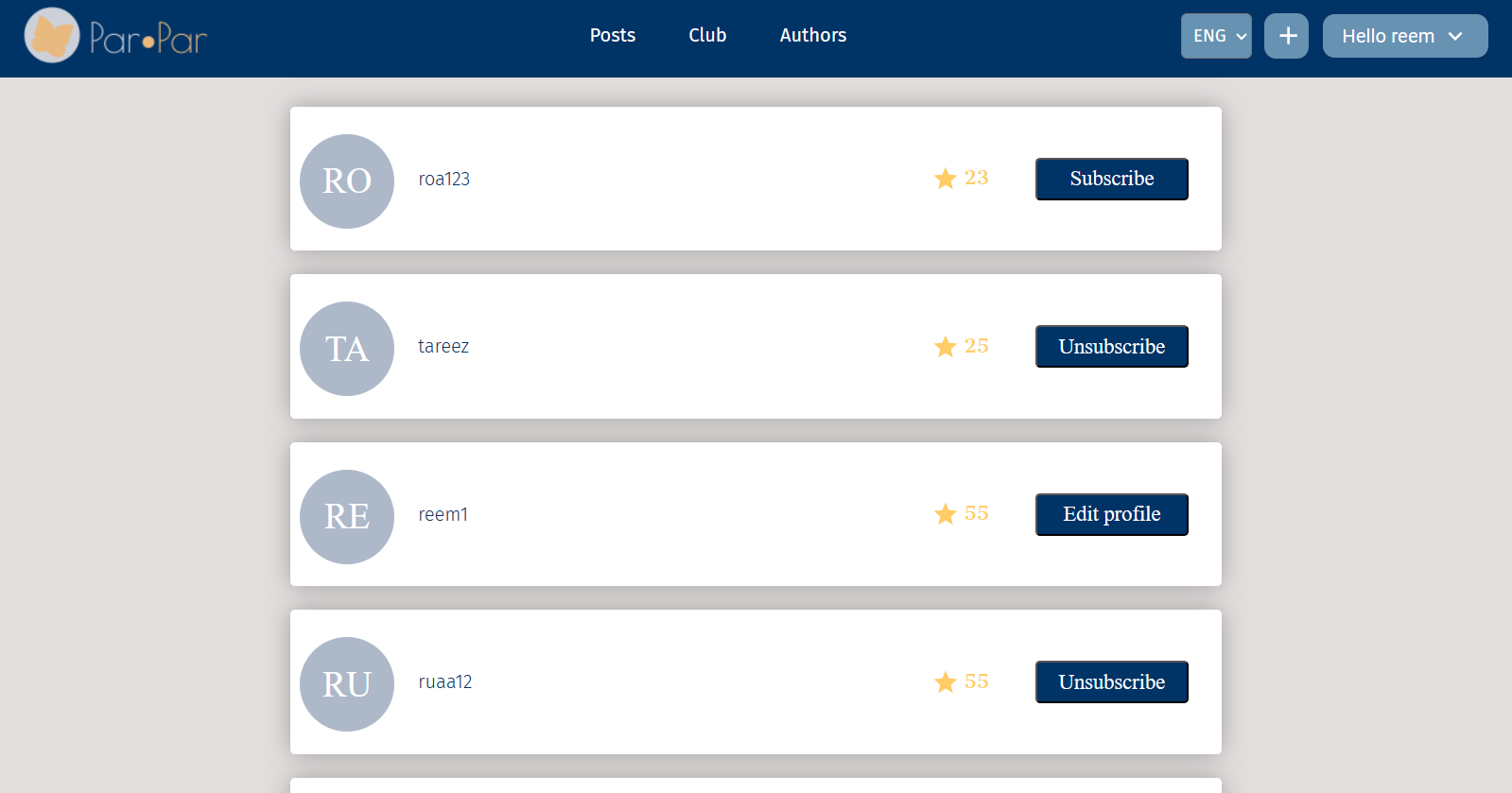


Figure 20

**11.3.5 View author profile - Publications:**

The user can add comments/likes to posts.

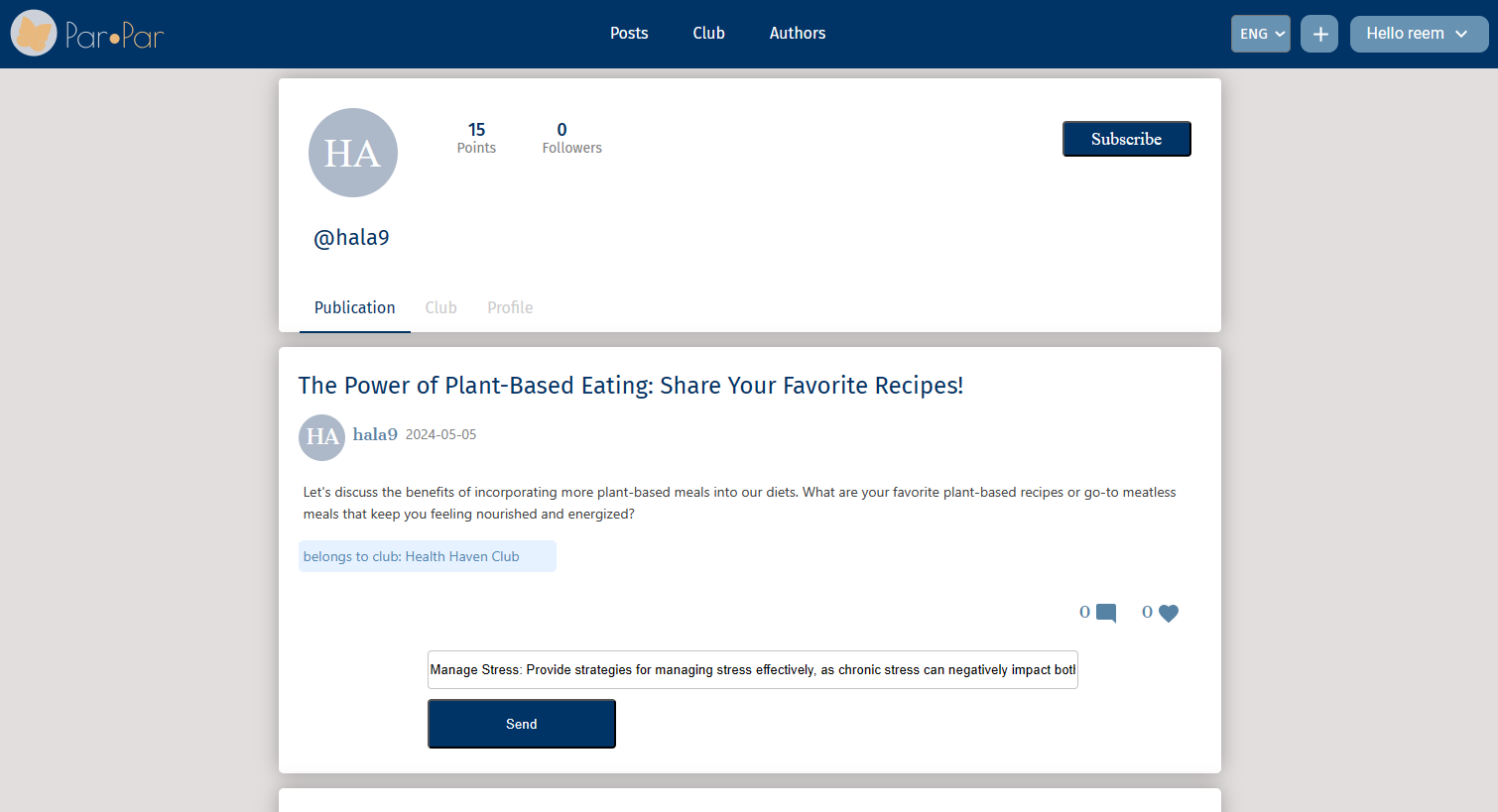


Figure 21

**11.3.6 View Profile:**

The user can click on the “my profile” button, after clicking on hello in the header and his profile page will load. The user can click on the “edit profile” button and then the edit profile page will load, or he can click on the ZOOM button to start a new meeting in ZOOM.

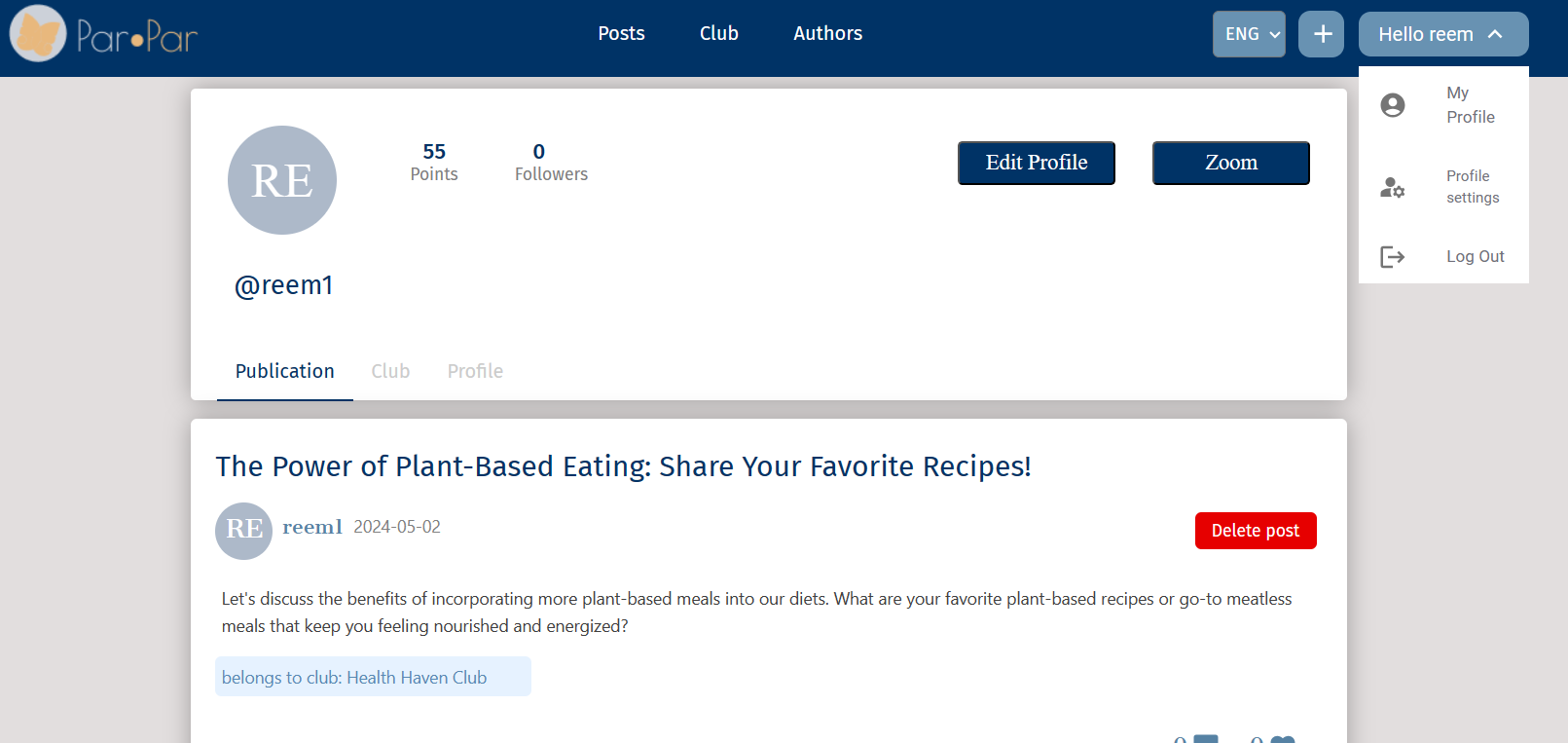


Figure 22

**11.3.7 View Profile - Publications:**

The user's profile screen has 3 fields. The default screen is the “Publications”. In this screen the user can view his posts and the comments on each post. He can also delete posts.

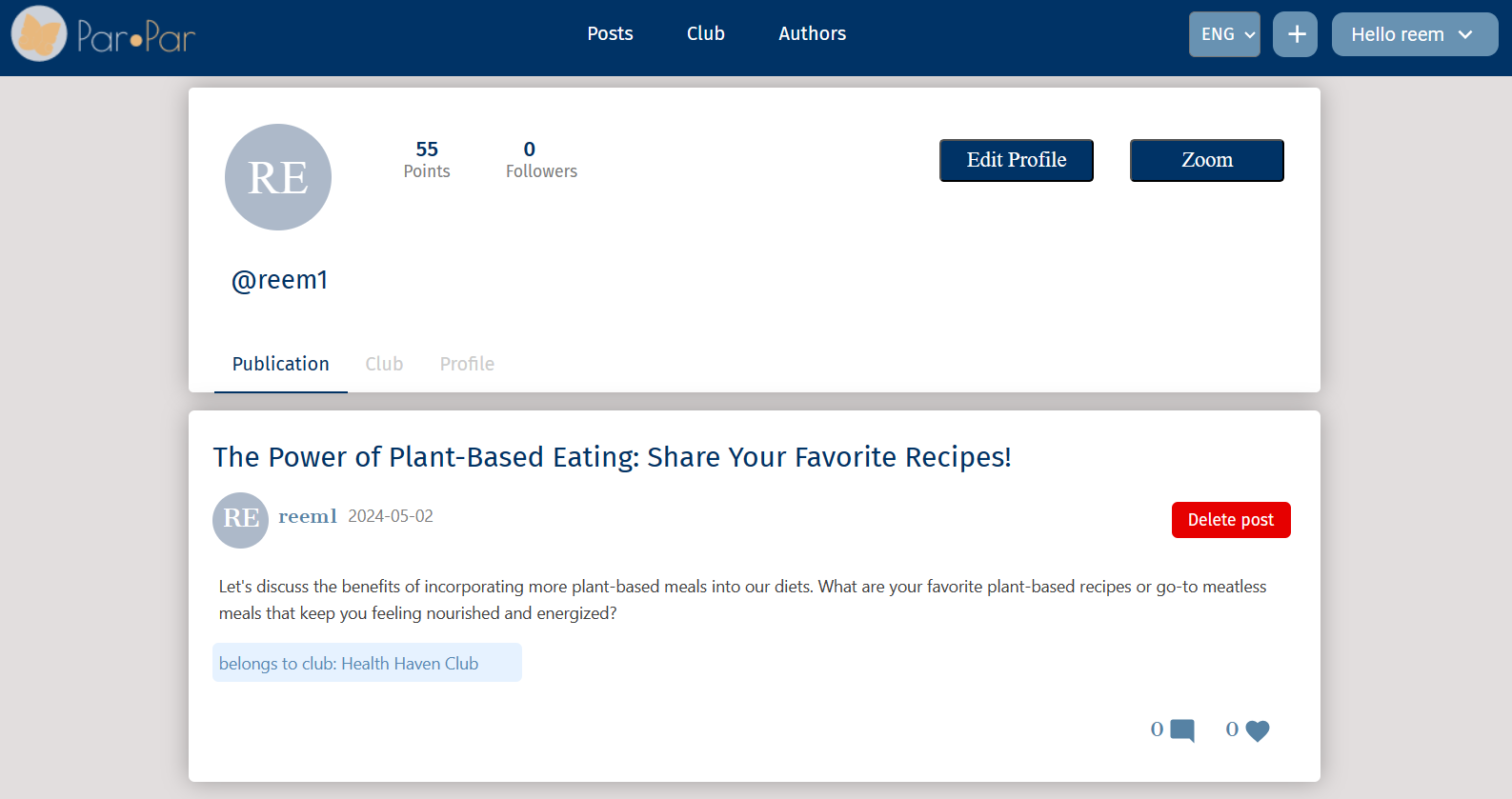


Figure 23

**11.3.8 View Profile - Clubs:**

The user can view the clubs that he created/subscribed to. He can view the club’s profile by clicking on the name.

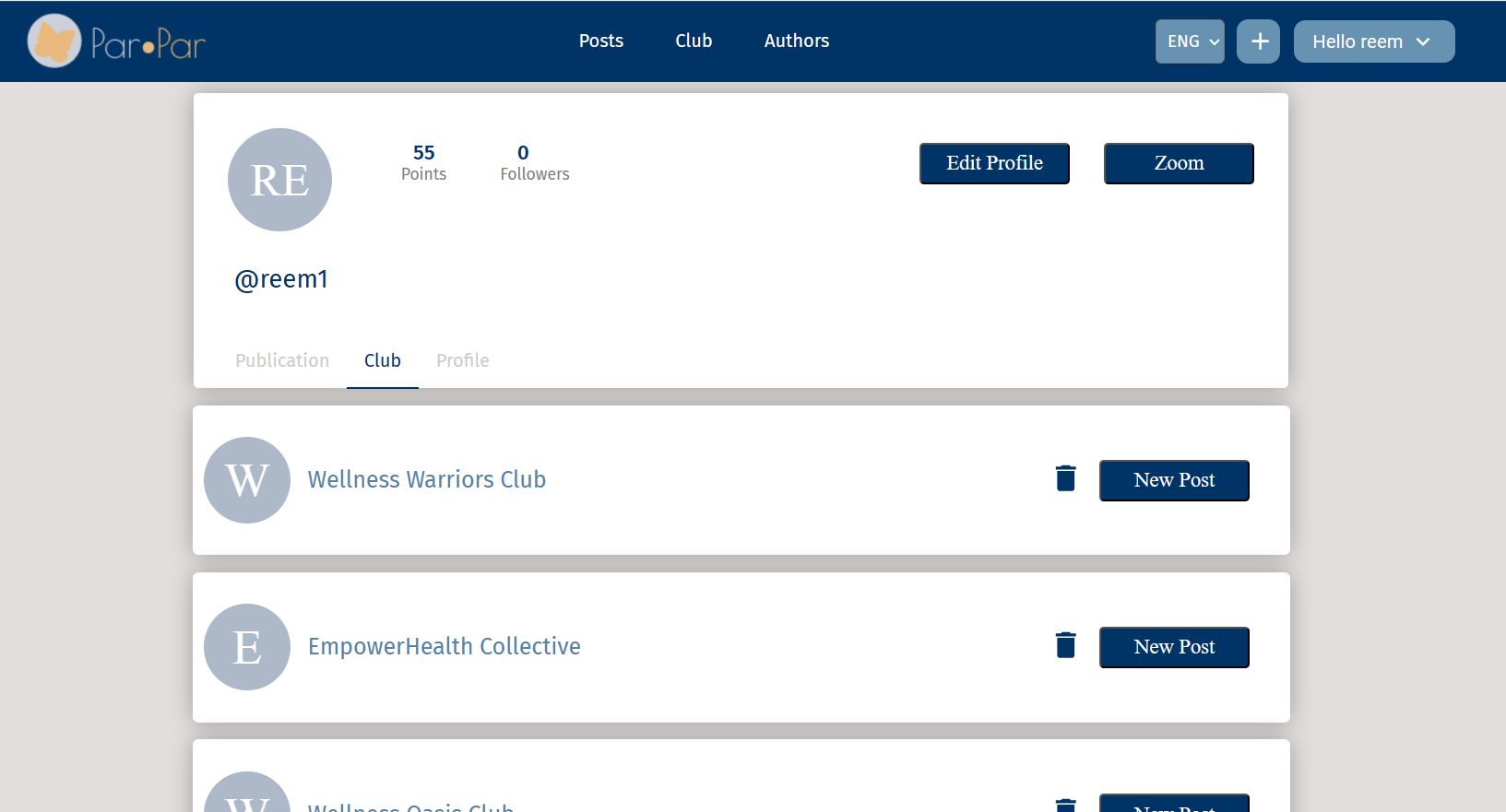


Figure 24

**11.3.9 View Profile - Profile:**

The user can view more details on the user.

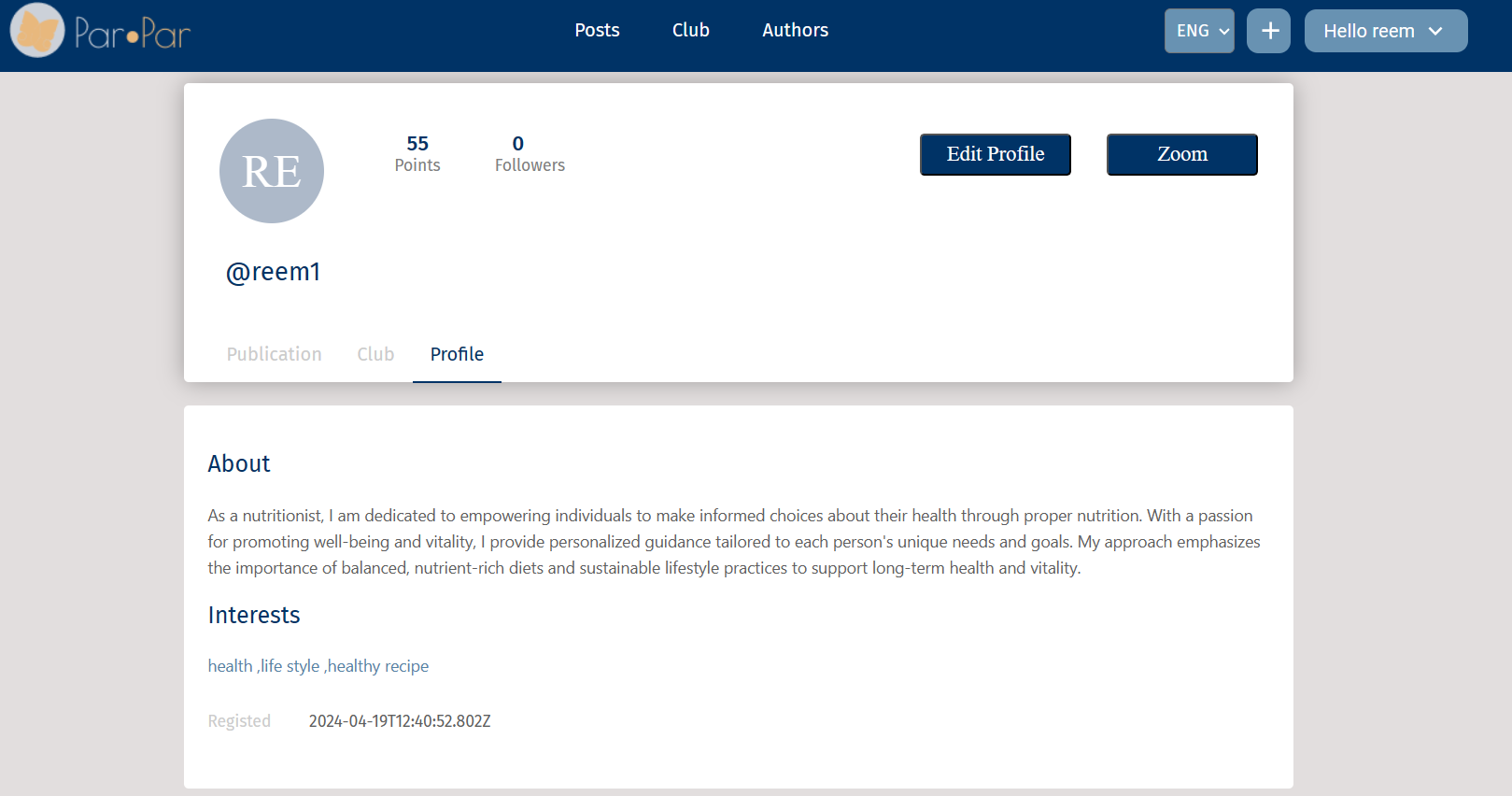


Figure 25

**11.3.10 Profile Settings - Edit Profile:**

The user can edit his profile by clicking on the “profile settings” in the header, and then the edit profile will load.

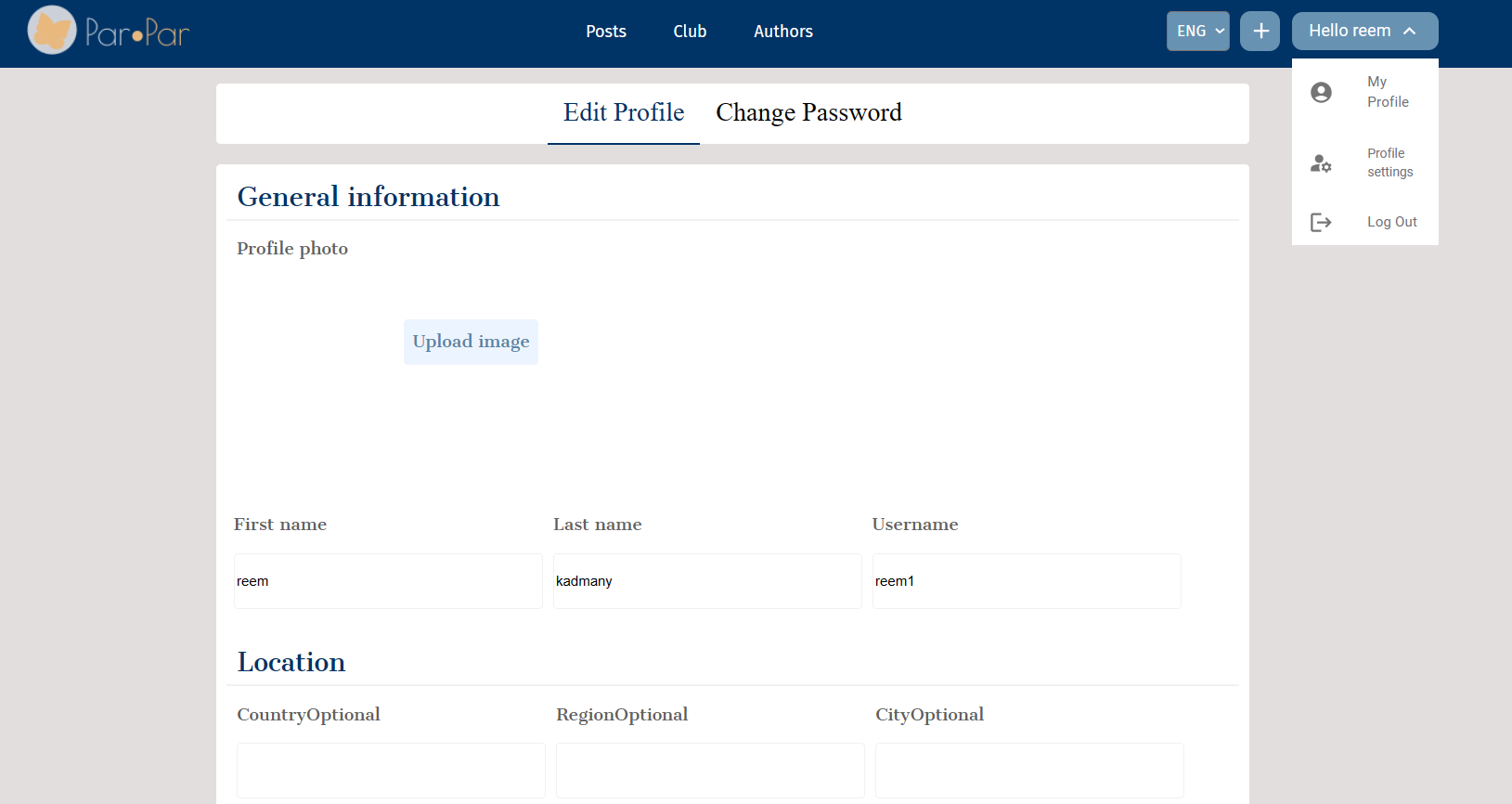


Figure 26

**11.3.11 Profile Settings - Change Password:**

The user can change his password by clicking on “Change Password” in the profile settings button in the header.

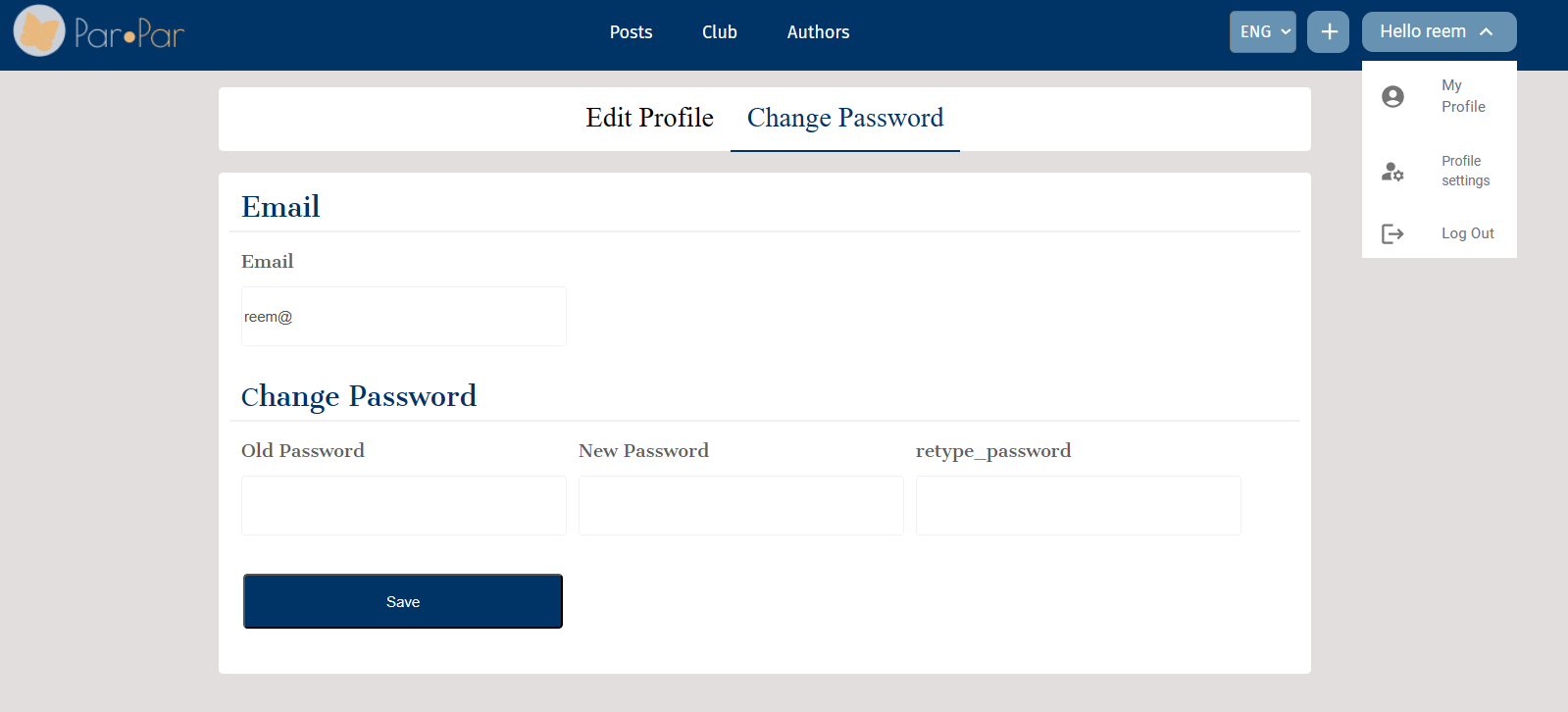


Figure 27

**11.4 Admin:**

Admin can do all the features that the logged user can do, in addition:

In the club profile, the user can edit the club that he created by clicking on “edit club”.

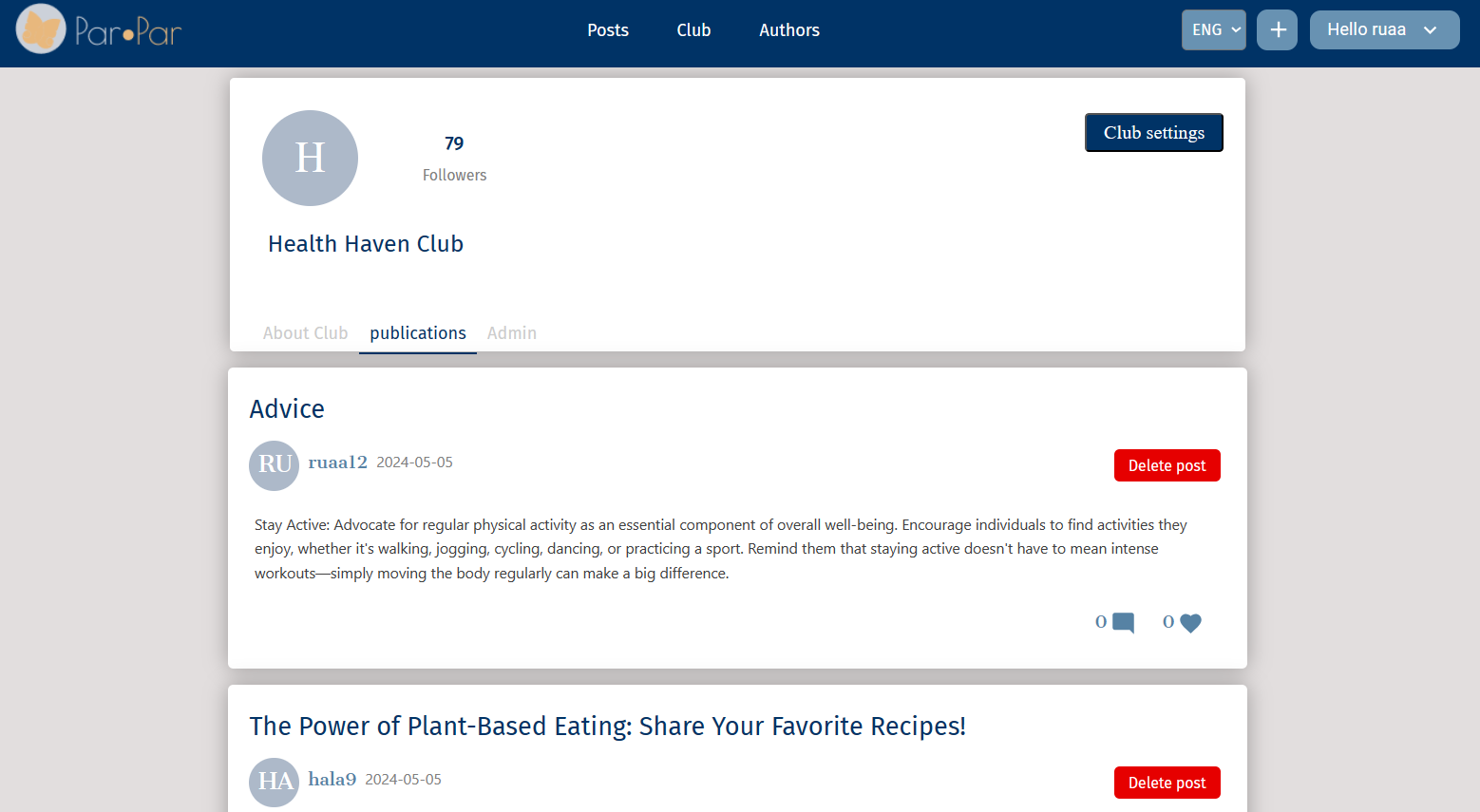


Figure 28

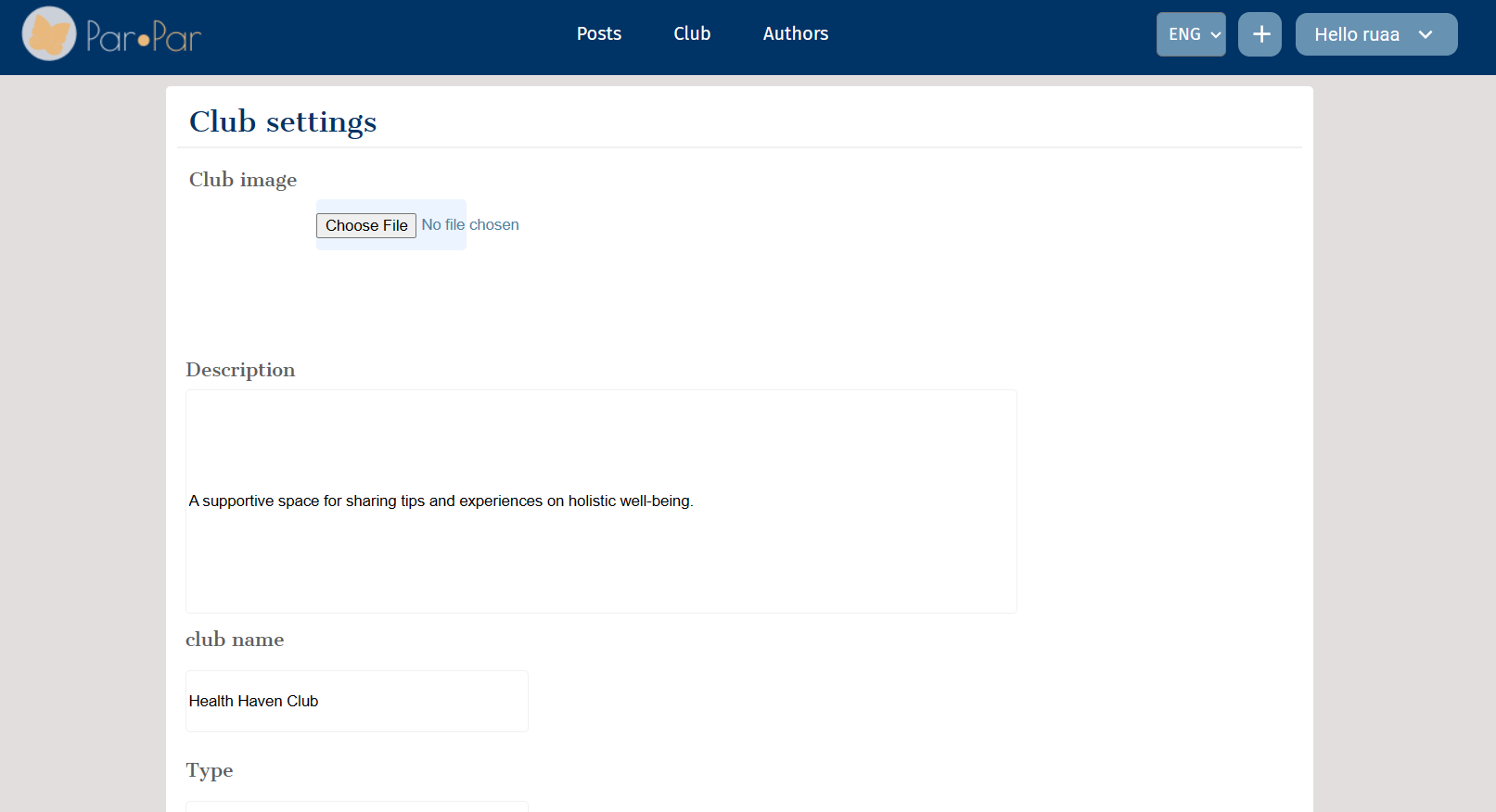


Figure 29

**11.5 Moderator:**

Moderatorcan do all the features that the logged user can do, in addition:

He can delete a post that has unreliable information from: posts page/club page.

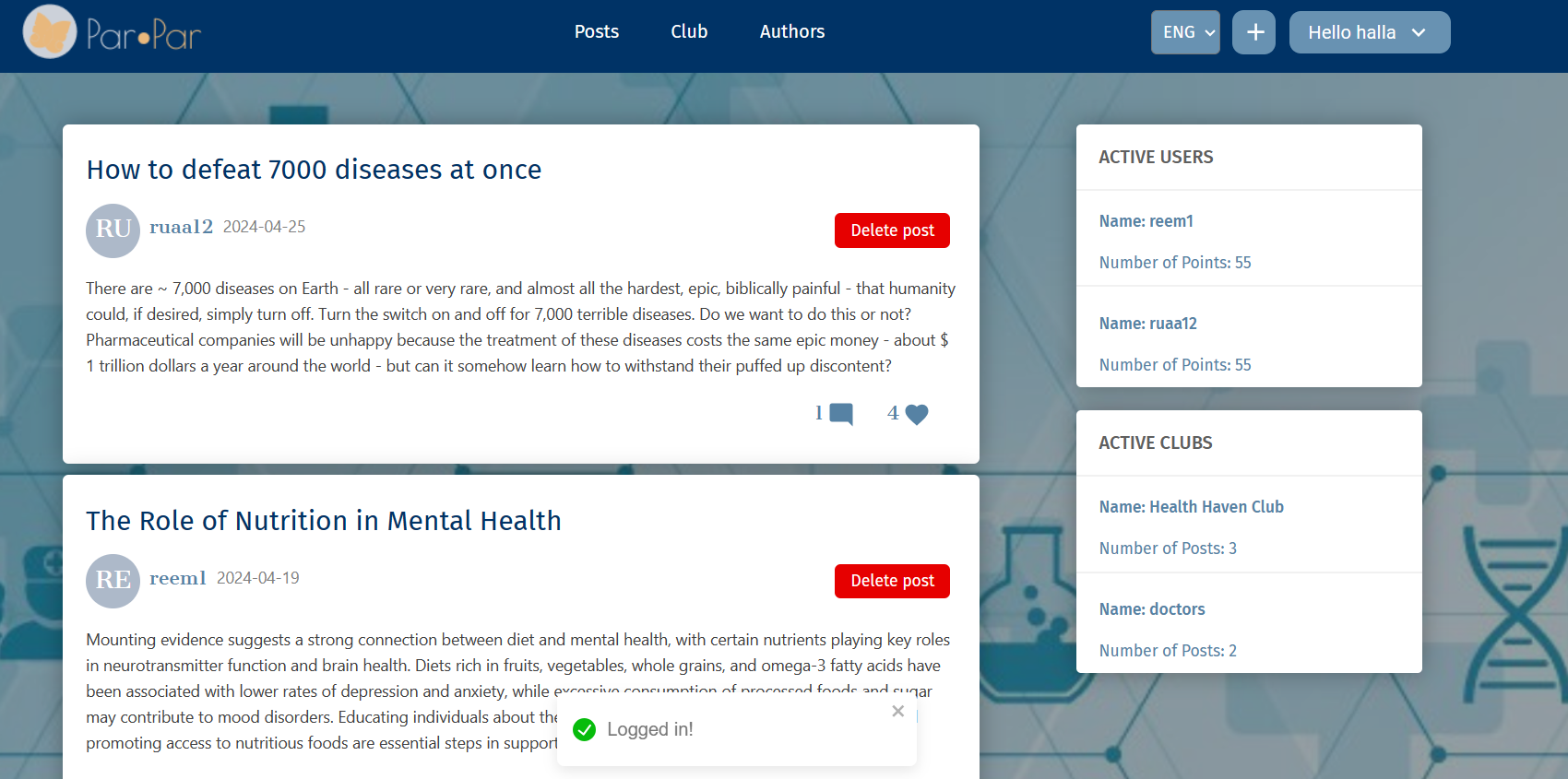


Figure 30

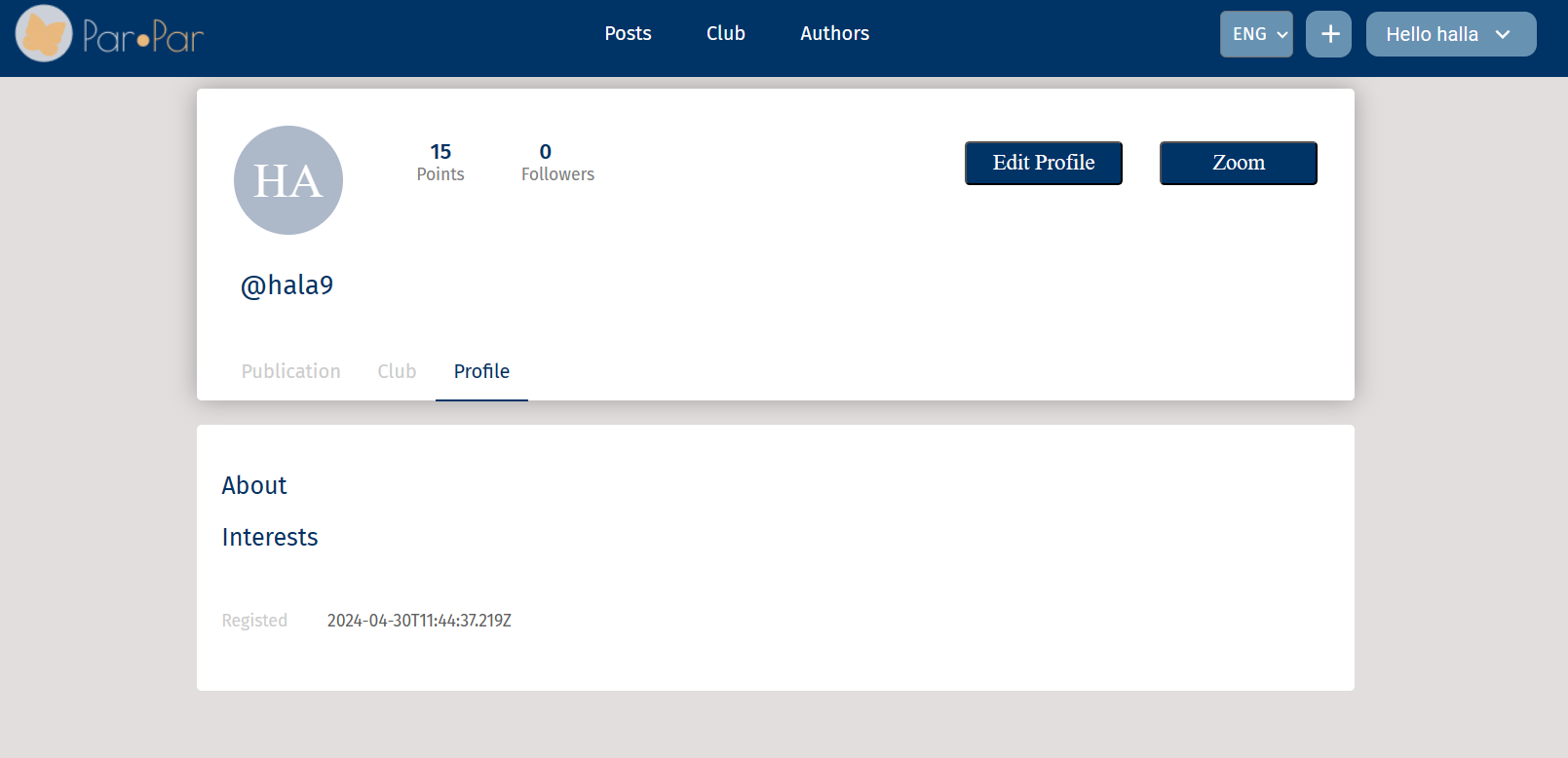


Figure 31

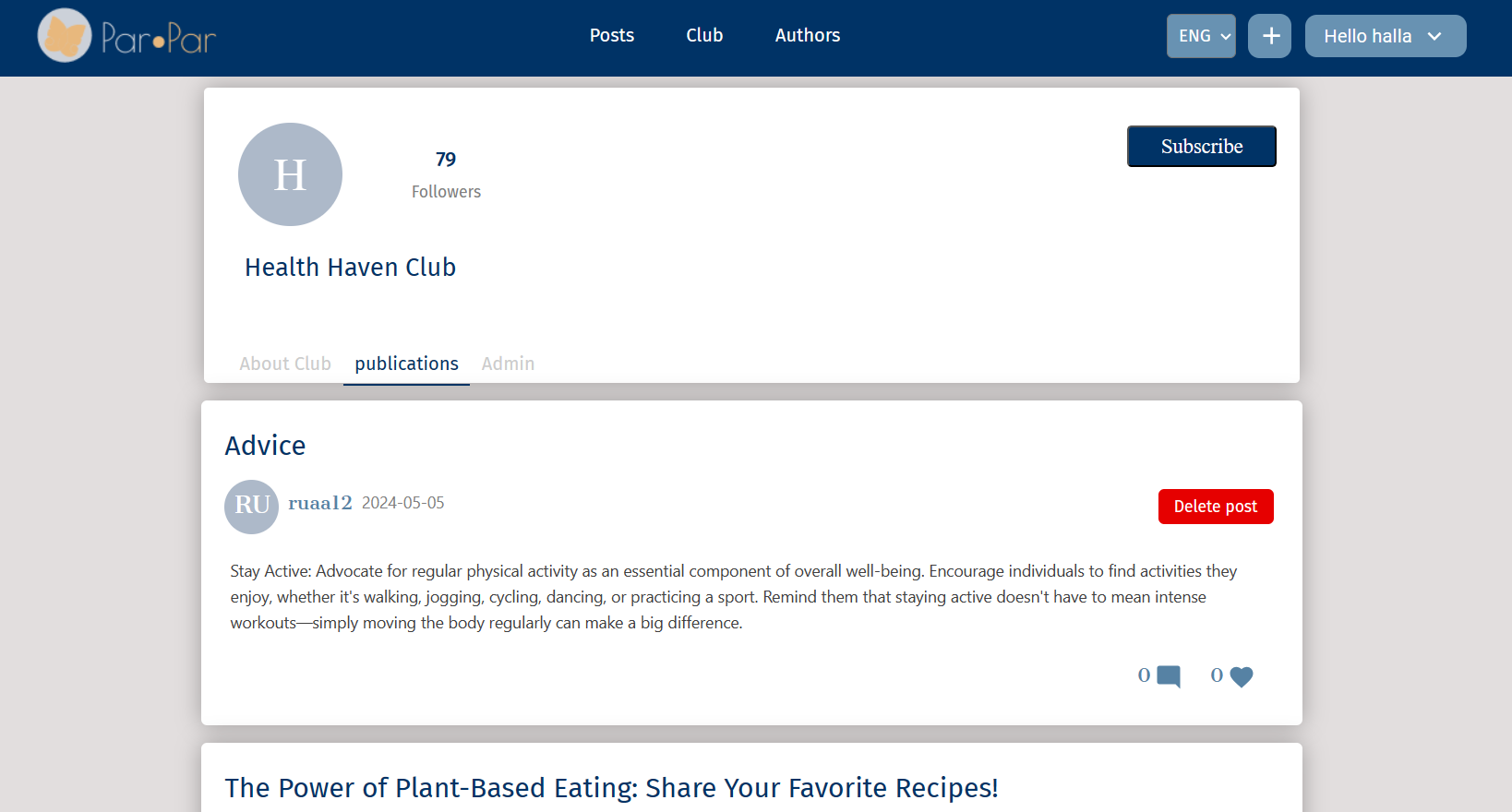


Figure 32

**12. Maintenance Guide:**

**12.1. Software Requirements:**

* Visual Studio Code: Download and install the latest version of Visual Studio Code.

**12.2. Installation Process:**

* Click on the link and download the project from GitHub: https://github.com/roaab3/SocialnetworkForHealth.git
* Extract the ZIP file to a directory on your computer and open it onVisual Studio Code.
* For the server side run the command “npm i” to install all the packages that the project needs. To run the server, run the command: “npx ts-node src/app.ts”.
* For the front side, run the command “npm i” to install all the packages that the project needs. To start the front side, run the command: “npm start”.