

**LEAVE MANAGEMENT SYSTEM**

*A project submitted in partial fulfillment of the requirements for the award of the degree of*

**Bachelor of Technology** **in**

**COMPUTER SCIENCE AND ENGINEERING**

***Submitted by:***

*Rohit Raj & Yugansh Goyal*

*Roll No.: 12011035 & 12011043*

*Branch: CSE*

***Supervised by:***

*Dr. Mukesh Mann*

*&*

*Mr. Sandeep Singh*

**ACKNOWLEDGEMENTS**

We would like to express our special thanks of gratitude to our teachers (Dr. Mukesh Mann & Mr. Sandeep Singh) who gave us the golden opportunity to do this wonderful project on the topic Leave Management System which also helped us in doing a lot of Research and we came to know about so many new things we are really thankful to them. Secondly, we would also like to thank our parents and friends who helped us a lot in finalizing this project within the limited time frame. We are overwhelmed in all humbleness and gratefulness to acknowledge my depth to all those who have helped us to put these ideas, well above the level of simplicity and into something concrete. Any attempt at any level can 't be satisfactorily completed without the support and guidance of my parents and friends. we would like to thank our parents who helped me a lot in gathering different information, collecting data and guiding me from time to time in making this project, despite of their busy schedules, they gave me different ideas in making this project unique. Thanking you,

**SELF DECLARATION**

We hereby declare that work contained in the project file titled “LEAVE MANAGEMENT SYSTEM” is original. We have followed the standards of research/project ethics to the best of our abilities. We have acknowledged all sources of information which we have used in the project.

**Name**: Rohit Raj & Yugansh Goyal

**Roll No.**: 12011035 & 12011043

Department of Computer Science and Engineering,

Indian Institute of Information Technology,

Sonipat-131201, Haryana, India.

**CERTIFICATE**

This is to certify that Mr. Rohit Raj & Mr. Yugansh Goyal has worked on the project entitled “LEAVE MANAGEMENT SYSTEM” under my supervision and guidance. The contents of the project, being submitted to the Department of Computer Science and Engineering, IIIT Sonipat, for the award of the degree of B.Tech in Computer Science and Engineering, are original and have been carried out by the candidate himself. This project has not been submitted in full or part for the award of any other degree or diploma to this or any other university.

Dr. Mukesh Mann & Mr. Sandeep Singh

Supervisor

Department of Computer Science and Engineering,

Indian Institute of Information Technology,

Sonepat-131201, Haryana, India

**ABSTRACT**

Name of the students: **Rohit Raj** & **Yugansh Goyal**

Roll No.: **12011035 & 12011043**

Degree for which submitted: **B. Tech (CSE)**

Department of **Computer Science and Engineering, IIIT Sonipat**.

Project Title: **LEAVE MANAGEMENT SYSTEM**

Name of the thesis supervisor: **Dr. Mukesh Mann** **& Mr. Sandeep Singh**

Month and year of project submission: **November 2022**

It is said that more than 80% of employee timesheets require stringent rectification because it is difficult for them to remember the timings. Even if employees work from home or at a remote location, keeping track of their working hours is critical. It is a direct measure of different departments’ productivity and provides simple inputs for areas that need to be improved.

Therefore, there is an urgent need of an online software which can track and manage the employees’ leaves.

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure No** | **Name of Figure** | **Page No** |
| 2.1 | E-Commerce Sales | 9 |
| 2.2 | API Working Diagram | 13 |
| 3.1 | Admin Use Case Diagram | 19 |
| 3.2 | User Use Case Diagram | 20 |
| 3.3 | User Use Case Diagram - 2 | 20 |
| 3.4 | User Use Case Diagram - 3 | 21 |
| 3.5 | User Use Case Diagram - 4 | 21 |
| 3.6 | Class Diagram | 23 |
| 3.7 | Admin Login Sequence Diagram | 24 |
| 3.8 | Code Snippet for Home page | 25 |
| 3.9 | Code Snippet for Hero section of home page | 26 |
| 3.10 | Code Snippet for Featured section of home page | 26 |
| 3.11 | Code Snippet for Add Item to Cart | 27 |
| 3.12 | Code Snippet for Removing Item from Cart | 27 |
| 3.13 | Code Snippet for Changing Item amount in Cart | 28 |
| 3.14 | Code Snippet for Cleaning the Cart | 28 |
| 3.15 | Code Snippet for Searching product using Name | 29 |
| 3.16 | Code Snippet for Filtering Search | 29 |
| 3.17 | Code Snippet for User Sign-Up | 30 |
| 3.18 | Code Snippet for Registered Sign-Up | 30 |
| 3.19 | Code Snippet for Logging Out of Account | 31 |
| 3.20 | Code Snippet for Sending Payment Request | 31 |
| 3.21 | Code Snippet for Storing Data for Confirmed Data  Order | 32 |

|  |  |  |
| --- | --- | --- |
| 4.1 | Upper Half of Home Page | 37 |
| 4.2 | Lower Half of Home Page | 38 |
| 4.3 | Shopping Cart Page | 38 |
| 4.4 | Sign up Page | 39 |
| 4.5 | Sign-In Page | 39 |
| 4.6 | Search options and Types of Filters | 40 |
| 4.7 | Fantasy Genre Filter Search | 40 |
| 4.8 | Admin Login Page | 41 |
| 4.9 | Admin Product Page | 42 |
| 4.10 | Admin Orders Page | 42 |
| 4.11 | Admin Users Page | 43 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **Table Number** | **Name of Table** | **Page Number** |
| Table 1.1 | Indian Retail Market | 4 |
| Table 1.2 | Indian Book Industry | 4 |
| Table 4.1 | Software Testing Template  and Results | 34 |

|  |  |
| --- | --- |
| Acknowledgments | i |
| Self-Declaration | ii |
| Certificate | iii |
| Abstract | iv |
| List of Abbreviations | v |
| List of figures | vi |
| List of Tables | viii |

|  |  |  |  |
| --- | --- | --- | --- |
| **TABLE OF CONTENTS** | | | |
| **Chapter 1** | **Introduction** | | 1-7 |
|  | 1.1 | Introduction | 2 |
|  | 1.2 | Problem Outline | 2 |
|  | 1.3 | Project Objective | 3 |
|  | 1.4 | Project Methodology | 3 |
|  | 1.5 | Scope of project work | 4 |
|  |  | 1.5.1 Limitations | 5 |
|  | 1.6 | Organization of Project | 6 |
|  | 1.7 | Summary | 7 |
| **Chapter 2** | **Study and Review of Literature** | | 8-16 |
|  | 2.1 | Introduction | 9 |
|  | 2.2 | Front-End Development | 9 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | 2.3 | UI Development | 10 |
|  | 2.4 | Back-End Development | 12 |
|  | 2.5 | Deployment | 15 |
|  | 2.6 | Summary | 16 |
| **Chapter 3** | **Implementation** | | 17-32 |
|  | 3.1 | Introduction | 18 |
|  | 3.2 | UML Analysis model | 18 |
|  |  | 3.2.1 Use case diagram | 18 |
|  |  | 3.2.2 Class diagram | 22 |
|  |  | 3.2.3 Sequence diagram | 23 |
|  | 3.3 | Code snippet | 24 |
|  |  | 3.3.1 Home page | 25 |
|  |  | 3.3.2 Add product to cart | 26 |
|  |  | 3.3.3 Remove product from cart | 27 |
|  |  | 3.3.4 Changing item amount | 28 |
|  |  | 3.3.5 Clearing the cart | 28 |
|  |  | 3.3.6 Searching and filtering products | 29 |
|  |  | 3.3.7 User sign up and sign in | 30 |
|  |  | 3.3.8 Payment request and storing data | 31 |
|  | 3.4 | Summary | 32 |
| **Chapter 4** | **Result and Conclusion** | | 33-44 |
|  | 4.1 | Testing | 34 |
|  | 4.2 | Result and Screenshots | 37 |
|  |  | 4.2.1 Home page | 37 |
|  |  | 4.2.2 Shopping Cart page | 38 |

|  |  |  |
| --- | --- | --- |
|  | 4.2.3 Sign up and Sign in page | 39 |
|  | 4.2.4 Filter and Search products page | 40 |
|  | 4.2.5 Admin login page | 41 |
|  | 4.2.6 Admin Accessible pages | 41 |
|  | 4.3 Conclusion and Future Scope | 43 |
|  | **References** | 45-46 |
|  | **Appendix** | 47 |

***I*.**

**INTRODUCTION**

The “Leave Management System” is an online software which keeps a track of and manages the employees’ leaves. The main idea is to reduce the load of the organizations by decreasing the paper work and ease the record maintenance by having a particular website for leaves management.

This project enhances the employee and the company to serve more quickly and efficiently. This software is developed in order to computerize the activities which take more time, if done manually.

It enables the employee and company staff to make things faster and can get information quickly. If they want any information about employee, they can access it quickly.

1. **Problem Outline**

According to research only 66% of employees who manually report their time on timesheets are accurate. This is due to the absence of automation. Companies and employees bear a significant financial burden as a result of timesheet inaccuracies.

Some people are paid more than the work they have done, while others are paid less than they have worked. For enterprises with a small number of employees to those with a large number of employees, the “Leave Management System” will enhance the accuracy rates.

1. **Project Objectives**

It is said that more than 80% of employee timesheets require stringent rectification because it is difficult for them to remember the timings. Even if employees work from home or at a remote location, keeping track of their working hours is critical. It is a direct measure of different departments’ productivity and provides simple inputs for areas that need to be improved.

Therefore, there is an urgent need of an online software which can track and manage the employees’ leaves.

The main objective of the proposed system is to decrease the paper work and easier record maintenance by having a particular website for leaves maintenance.

It computerizes the maintenance of the employee details and leave section in the company. It also includes the details about the employee personal.

This approach basically deals with the number of leaves taken by the employees in the organization.

1. **Methodology**

Front End was created and designed using Hypertext Markup Language (HTML), Cascading Style Sheets (CSS) and BOOTSTRAP.

Python and Django will be used to design a backend as we know that Python is a multi-purpose language and is widely used to write text. Web Scraping will be done by using Good soup which is a third party Python library for web scraping extracting data from HTML and XML files, and separating Html content from a website will be done using the XML library. By web scraping the data from website which is mostly random text in HTML which is converted to useful data. Most of this data is random data in HTML format and converted to organized data into a spreadsheet or website for use in various applications.

**II.**

**IMPLEMENTATION**

1. **Frontend**

User interacts with the front end of the website. The front end is also known as the ‘client side’. It includes all the things seen and accessed by the user like navigation options, pictures, color scheme, text, graphs and tables, etc. HTML, CSS and JavaScript are languages used for Front End development. The structure, design, behaviour, and content of all visual effects on browser screens when websites, web applications, or mobile applications are opened, using previous developers. The primary responsibility of developers while designing front end is to ensure performance and response (the website should respond correctly on all different devices with different screen sizes).

• HTML: HTML stands for Markup Hypertext Language. It is used to give structure to web pages and web page content, it uses hypertext to allow navigation to other pages by simply clicking on the link

• CSS: Cascading Style Sheets called CSS is a simple language designed to simplify the process of making web pages look great. CSS lets you apply styles to web pages. Most importantly, CSS lets you do this without the HTML that renders every web page.

• Bootstrap: Bootstrap is a widely used framework of HTML,CSS and Javascript which makes it extremely easy to design wonderful frontends and as it is a framework, all the basics have been laid for responsive web development, and all developers need to do to encrypt the grid in the predefined grid.

1. **Backend Web-Development**

Backend is part of website server. Maintains and organizes data, and ensures that everything around the website client works properly. It does not communicate directly with users and users interact only with the front-end, which is the accessible part. Features and functionalities developed by backend designers are accessed indirectly by users through the previous app. Tasks, such as writing APIs, creating libraries, and working with system components without the interaction of user or science program systems, are also included in the background. The services and tools used to customize the web application domain are described below.

● Python is an advanced programming language designed to be easy to read and easy to use. It is an open source, which means it can be used for free, even in commercial applications. Python is considered a writing language, such as Ruby or Perl, and is often used to Web applications and dynamic Web content.

●Django-crispy-forms is an application that helps to manage Django forms. It allows adjusting forms' properties (such as method, send button or CSS classes) on the backend without having to re-write them in the template. It usually saves program organizers hours or working days.

● Django is a high-quality Python web framework widely used for the development of secure websites. Django also provides a framework that protects the website to help developers avoid common security errors helps to avoid many common security errors for developers.

1. **UML ANALYSIS MODEL**

Modeling involves the designing of software systems before coding takes place. Modeling plays an important role in any software development project. It guarantees the completeness and correctness of a software system and the fulfillment of end-user’s expectations. In addition,

modeling serves as the only reference point to cross-check requirements before coding. A Unified Modeling Language (UML) based tool was used to model this application. UML diagrams give both static and dynamic views of an application and it is well suited for object oriented languages like Java and C#. The following sub-sections present the UML diagrams used to model this application.

1. USE CASE DIAGRAMS

The use case diagrams for this application illustrate the interactions that exist between users (actors) and use cases (actions) within the application. There are two actors identified for this application – administrator (admin) and employee actors. As a result, there are two use case diagrams for the software application – admin use case diagram and employee use case diagram. The admin is the head of the organization who performs various administrative tasks such as check applications, add/view employees, add/view other administrators while the employee can apply for leaves view/edit personal details.

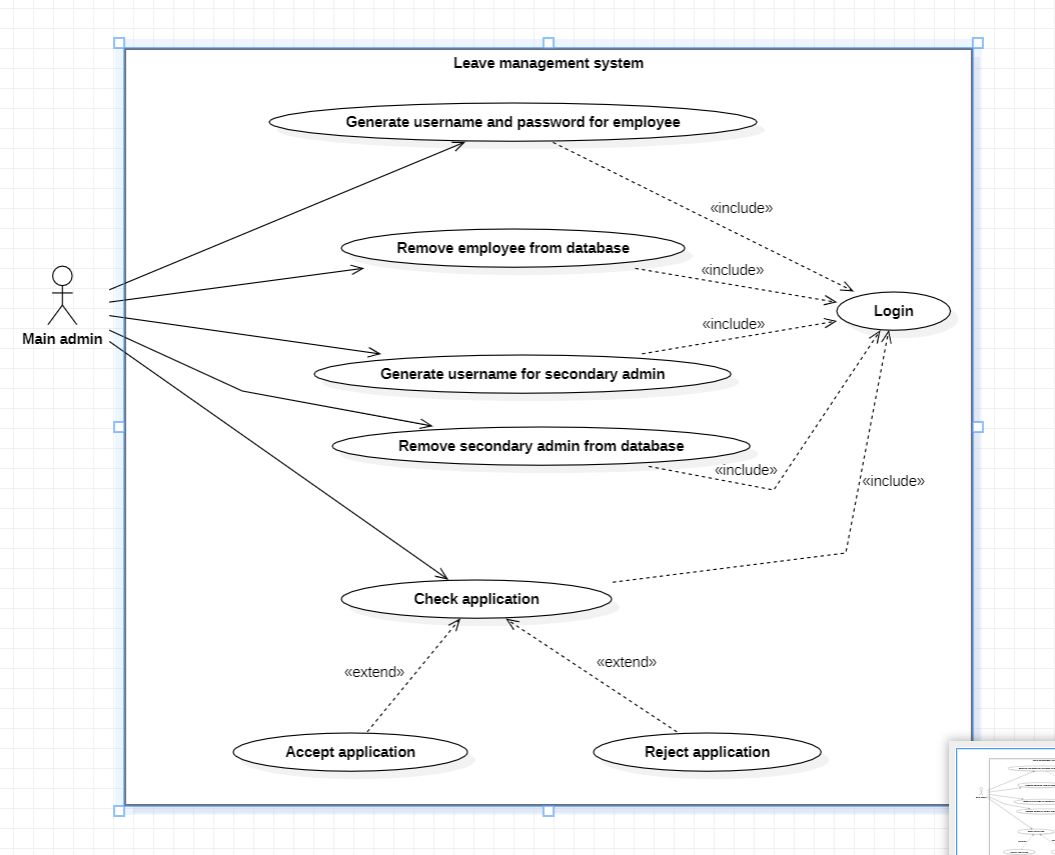


Fig. . Main admin Use Case Diagram

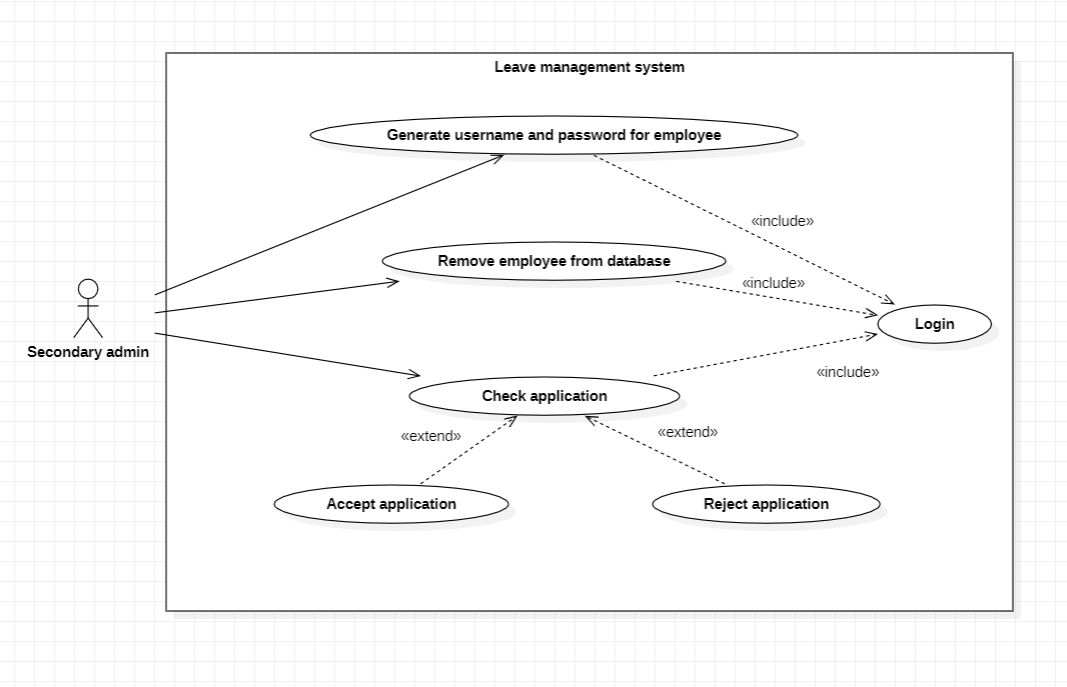


Fig. . Secondary Use Case Diagram

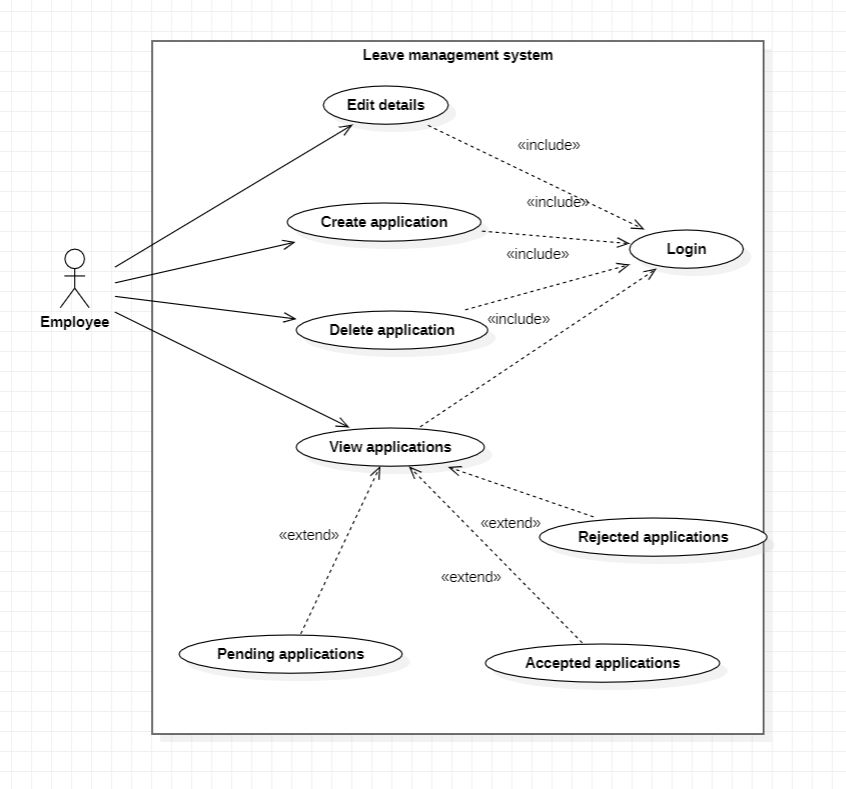


Fig. . Employee Use Case Diagram

1. DATAFLOW DIAGRAMS

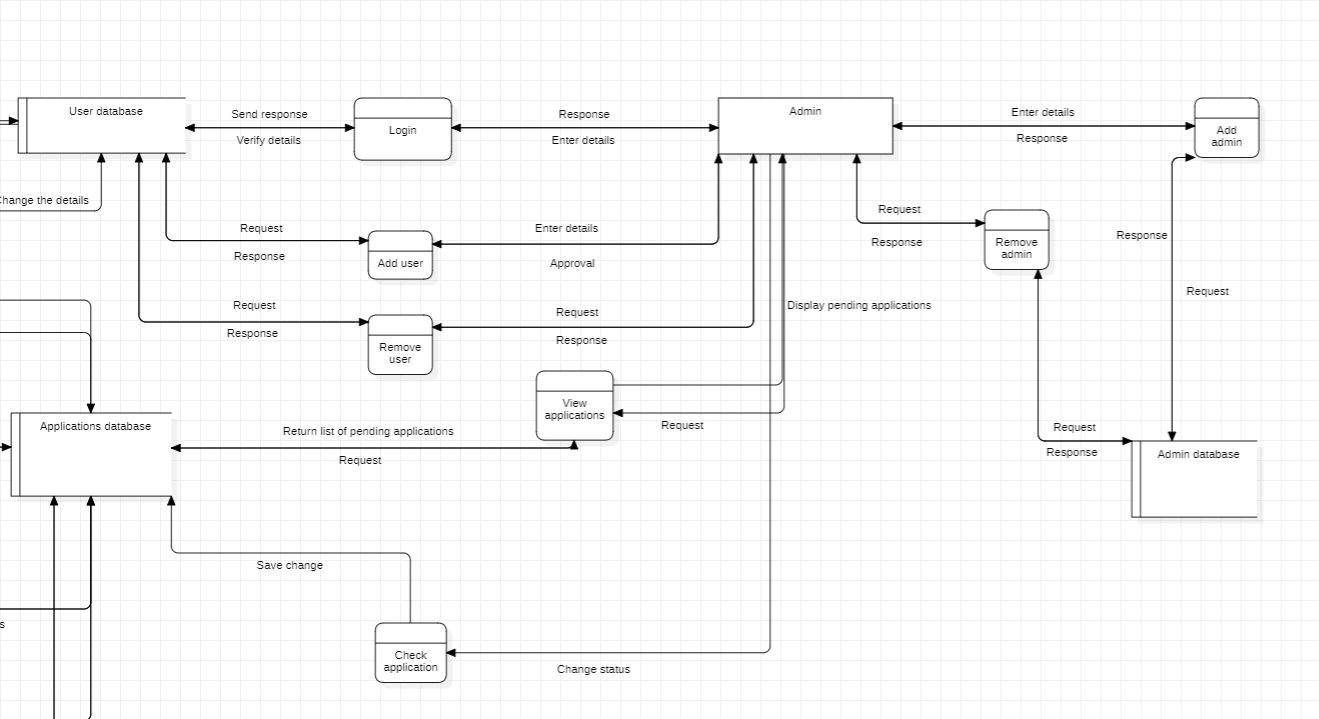


Fig. . Admin Data Flow Diagram

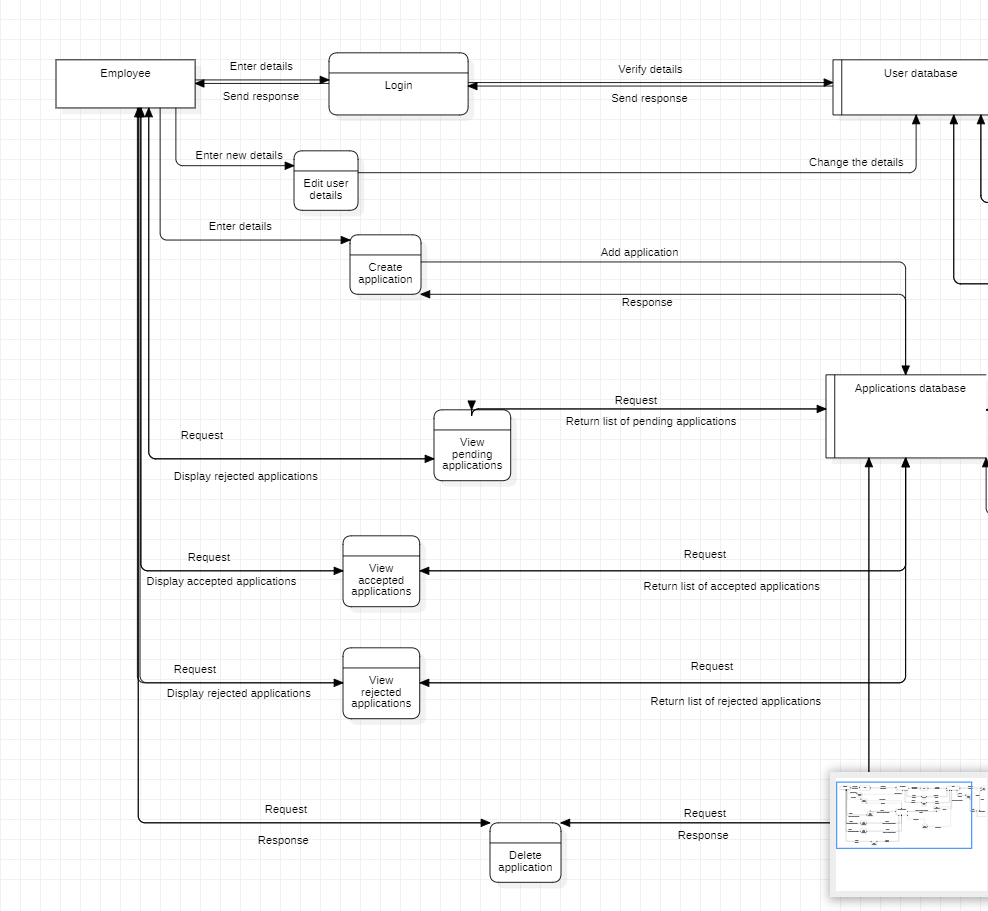


Fig. . Employee Data Flow Diagram

**III.**

**WEBSITE:**

**LEAVE MANAGEMENT SYSTEM**

1. **Main Features**

* Extremely effective and fast management portal
* It has a user login system so that every user has a unique account
* Can store datasets like total leaves requested till date, total leaves accepted/rejected, date of last leave accepted.
* Main admin can create multiple admins, if the organization is big, to ease the management.

1. **Applications**

* Employees can easily request for leaves, no need of pen-paper.
* Organizations can keep a track of the leaves an employee is taking.

1. **How to use the Leave Management System**

The user has to log into his account using the username and password provided by the admin.

Then the employee can send a leave application, view pending/accepted/rejected applications, edit personal details and can also keep a track of leaves.

The admin can view applications, list of employees, add an employee, can created and view other administrators.

The administrators created by the main admin also have the same powers but they can’t create or view administrators.

1. **Results and Screenshots**

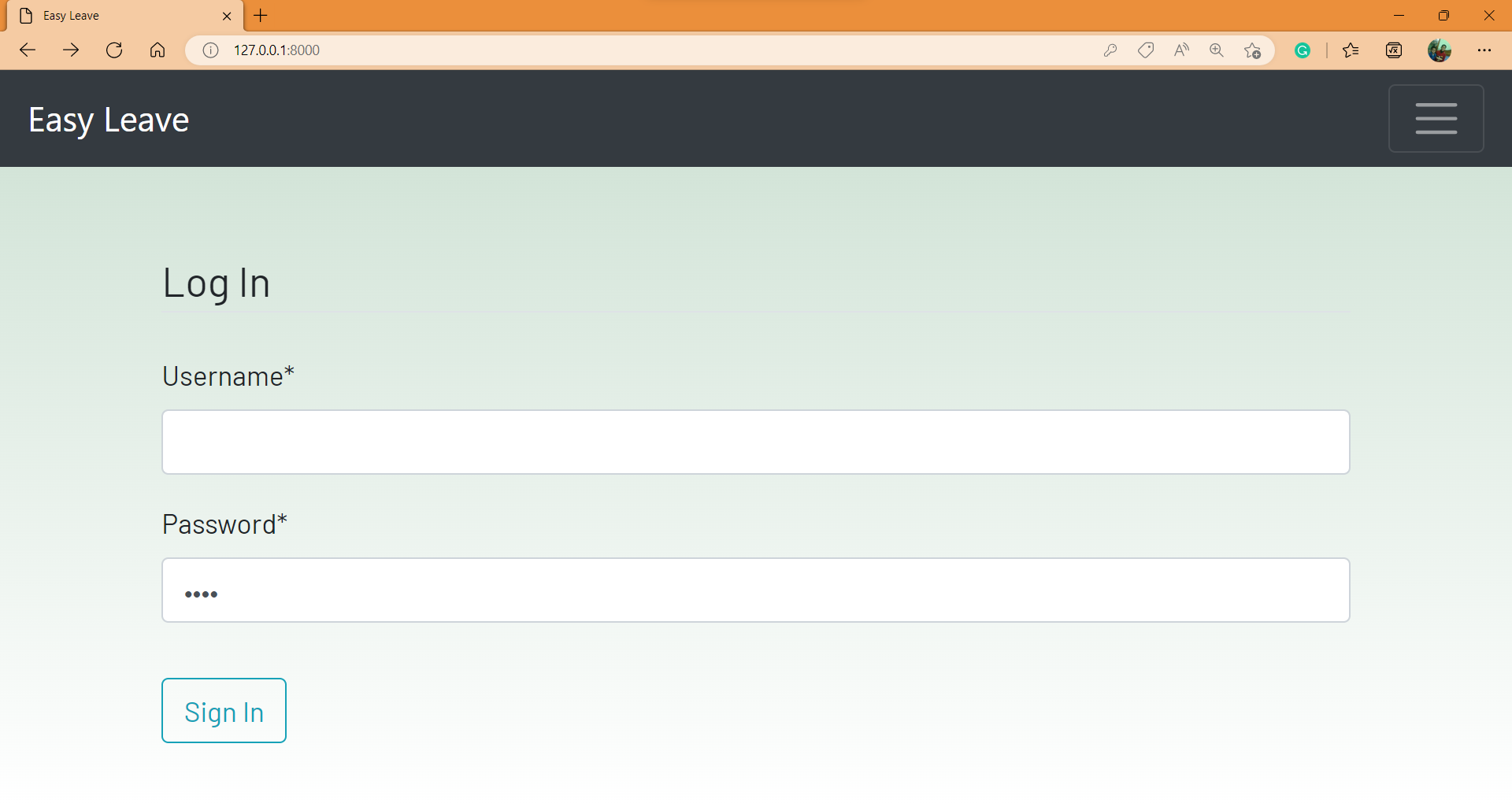


Fig. 6. Login page

**User Interface for Administrators**

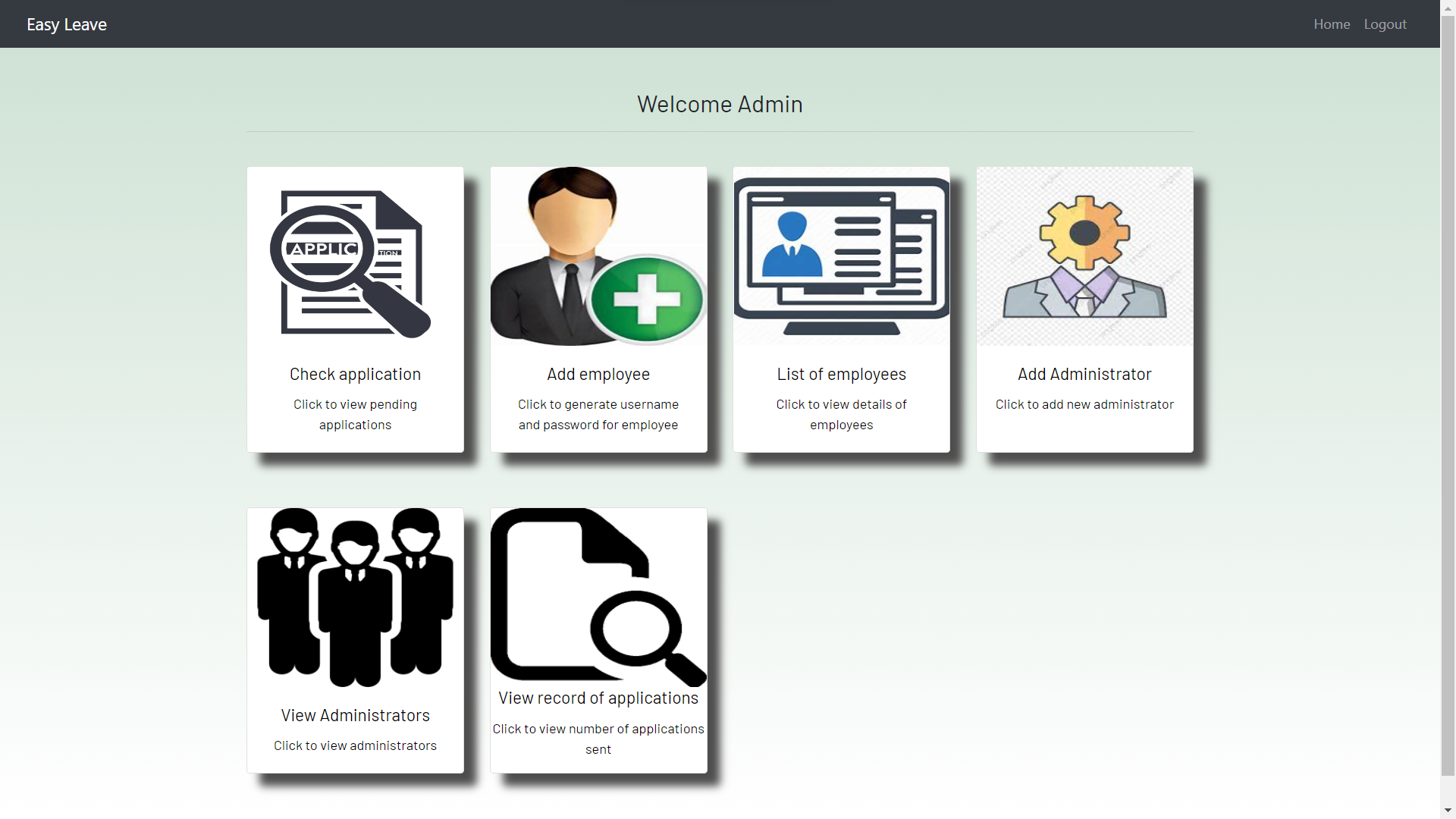


Fig. 7. Admin Dashboard

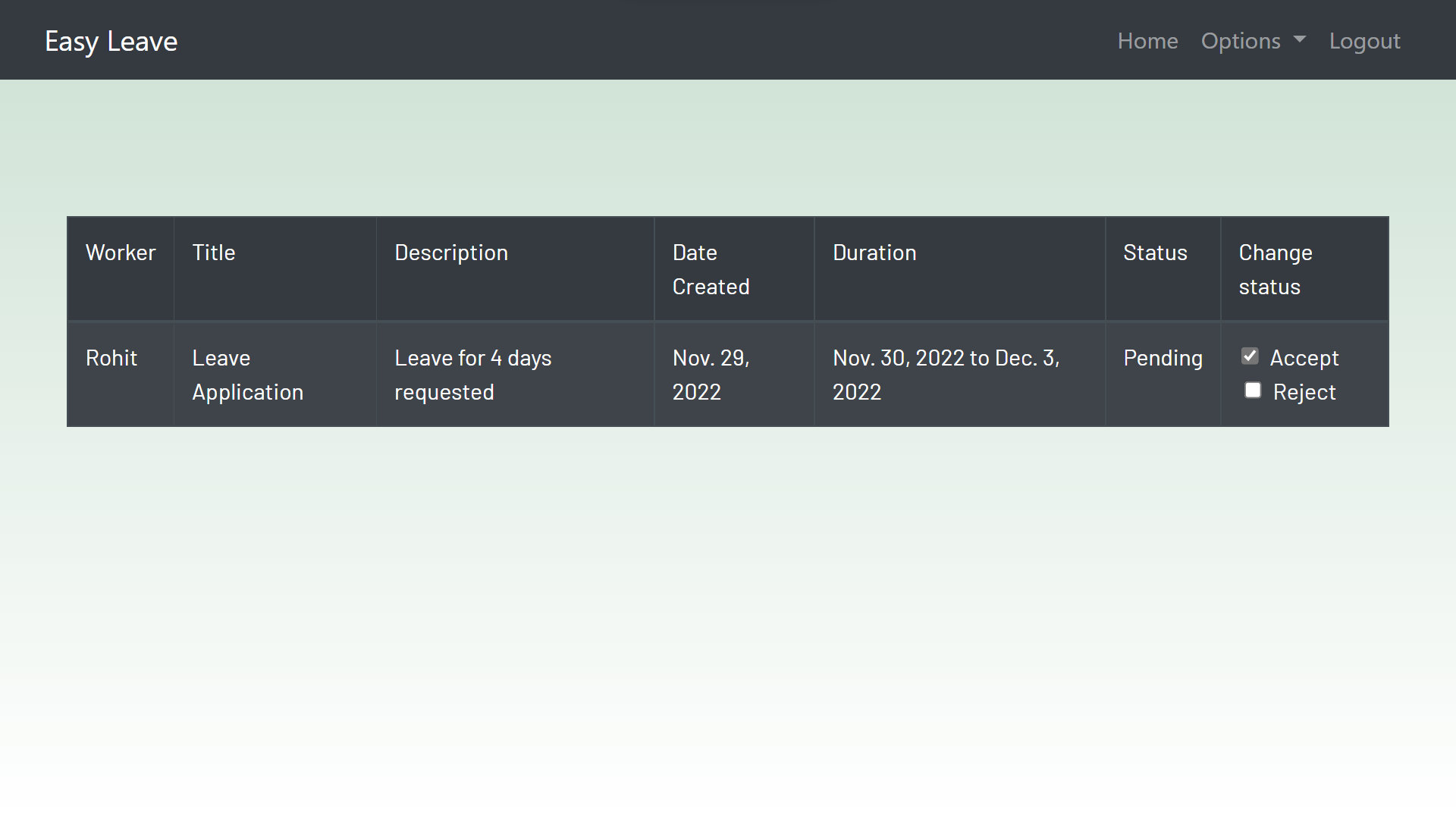


Fig. 8. Check Application

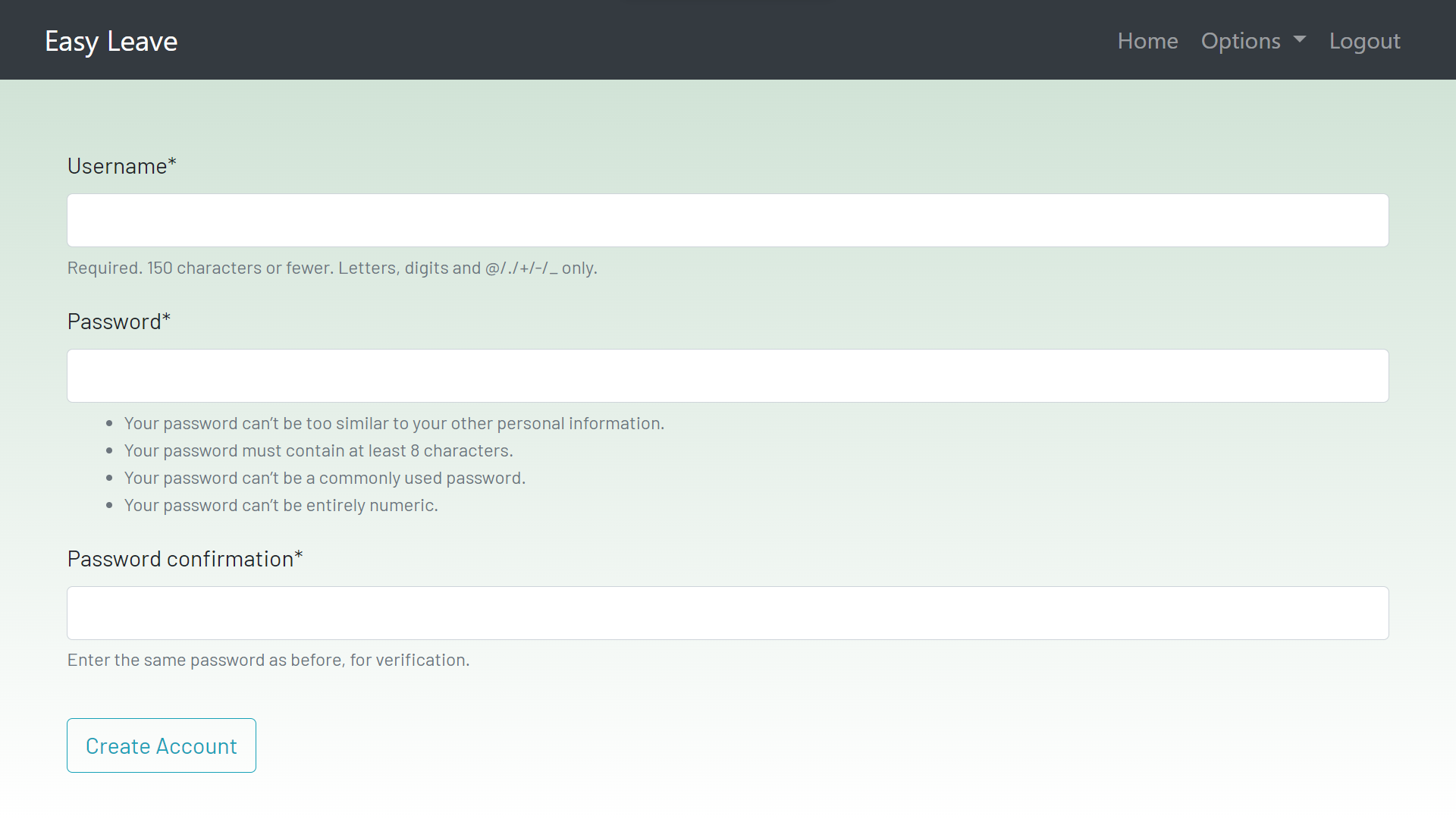


Fig. 9. Add Employee

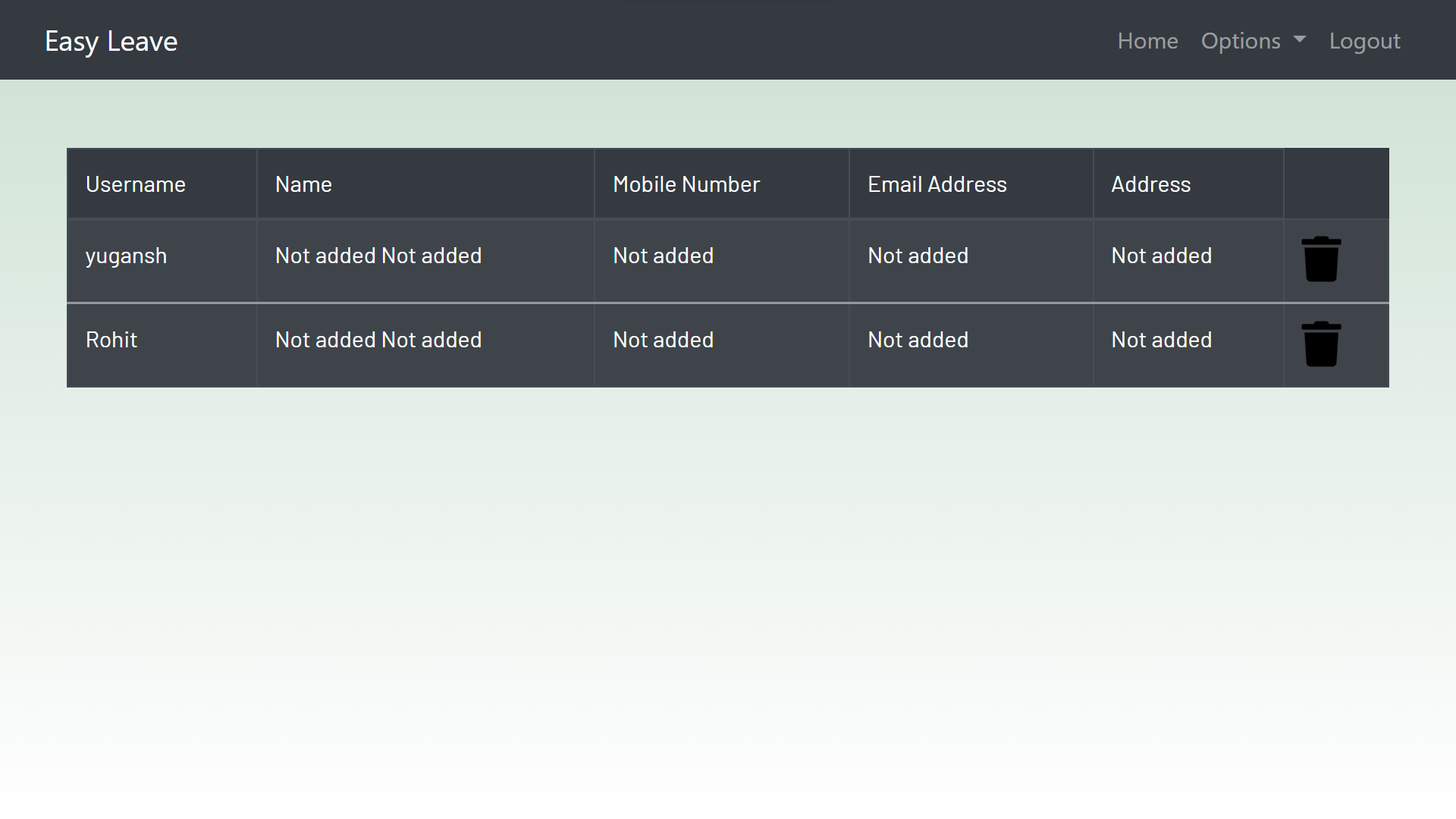


Fig. . View list of employees

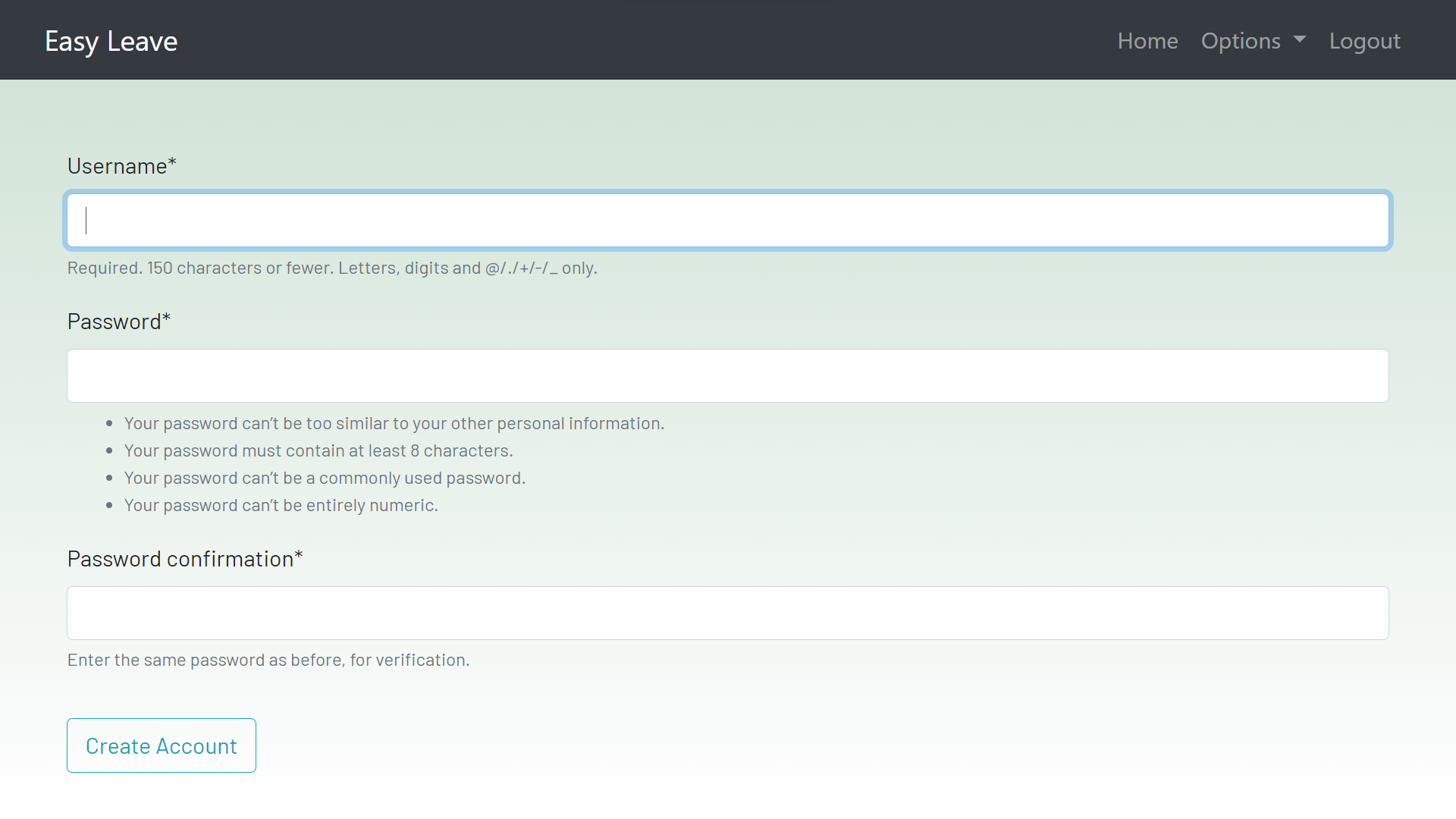


Fig. . Add more administrators

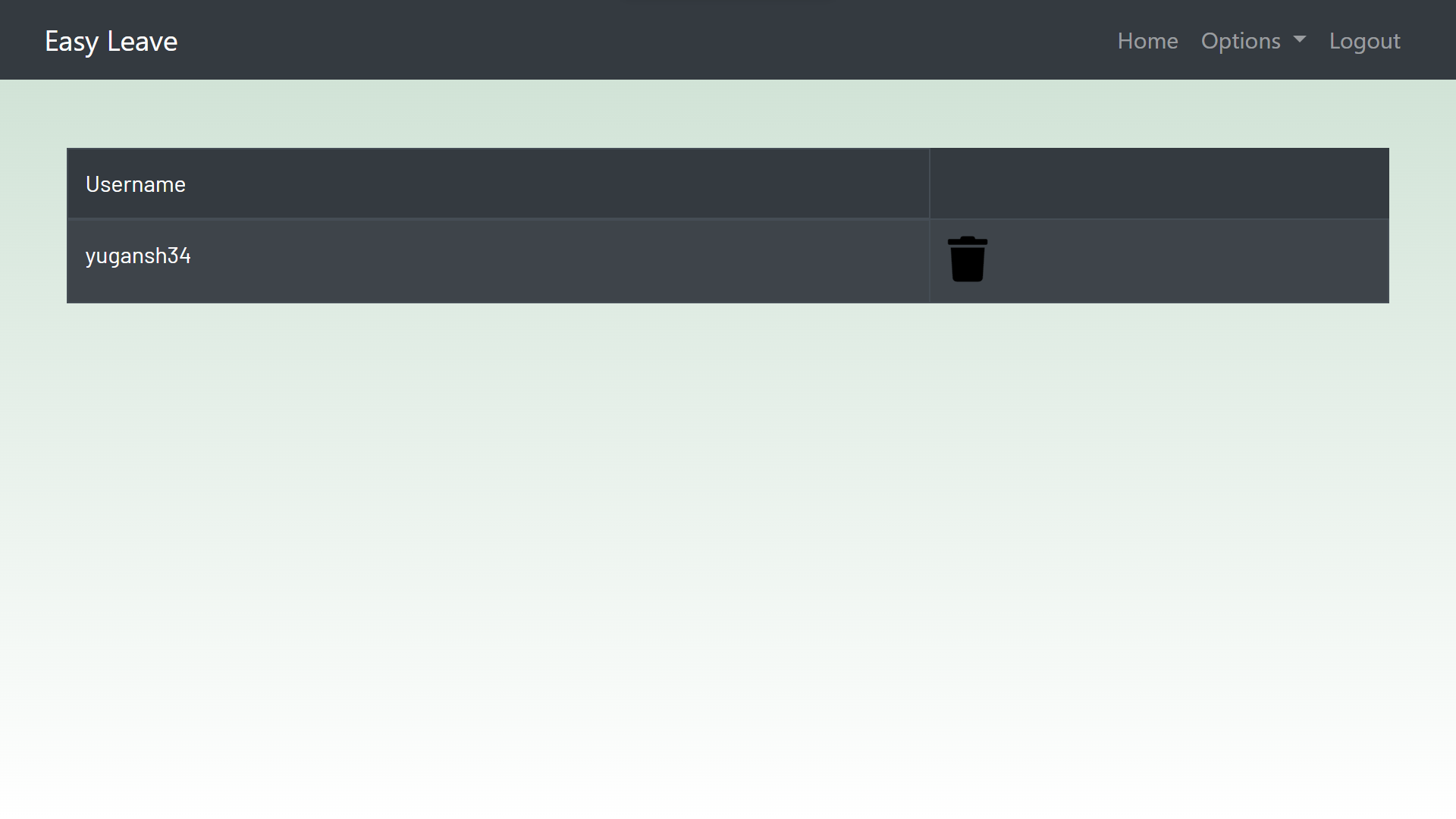


Fig. . View administrators

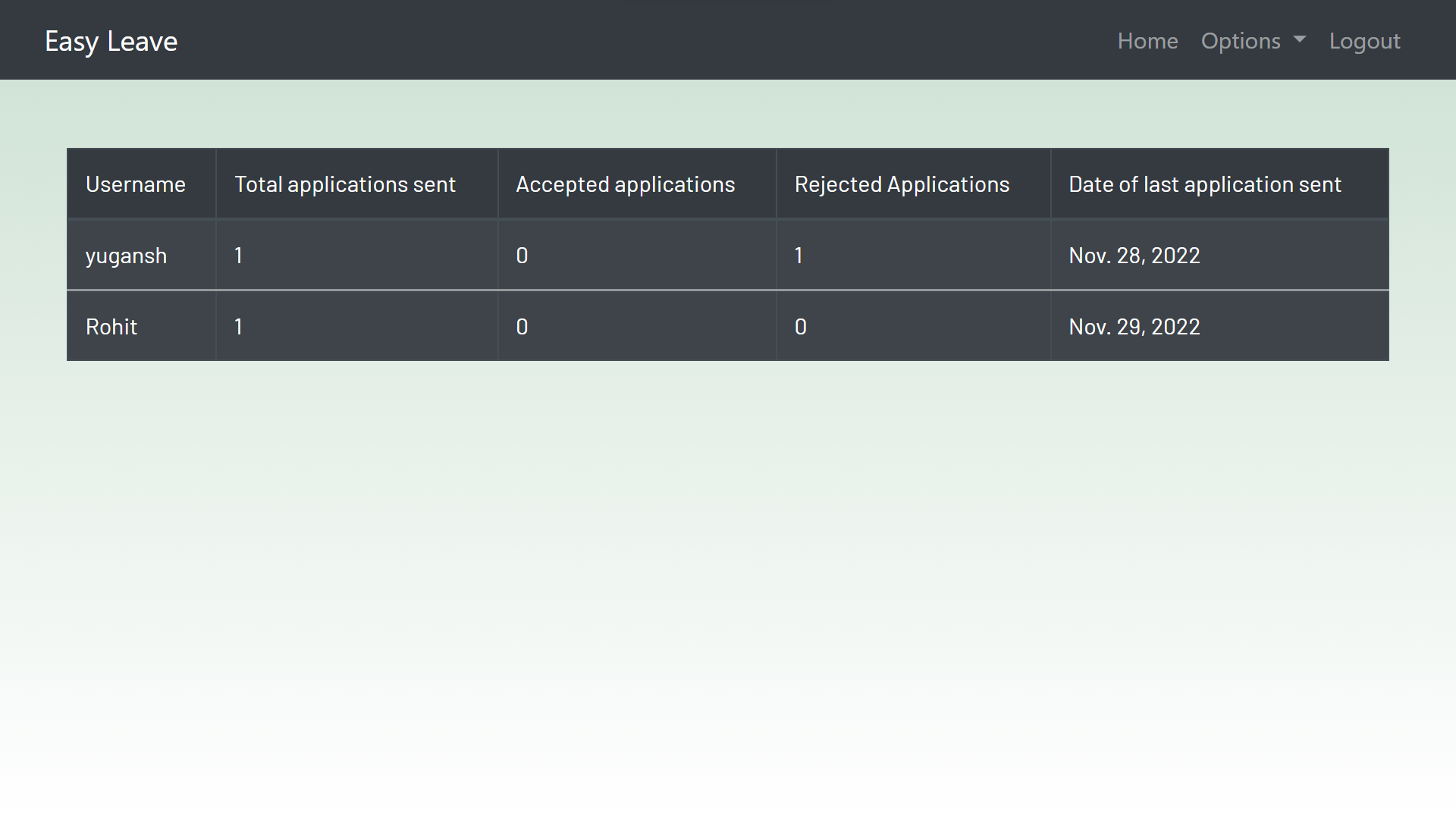


Fig. . View record of applications

**User Interface for Employees**

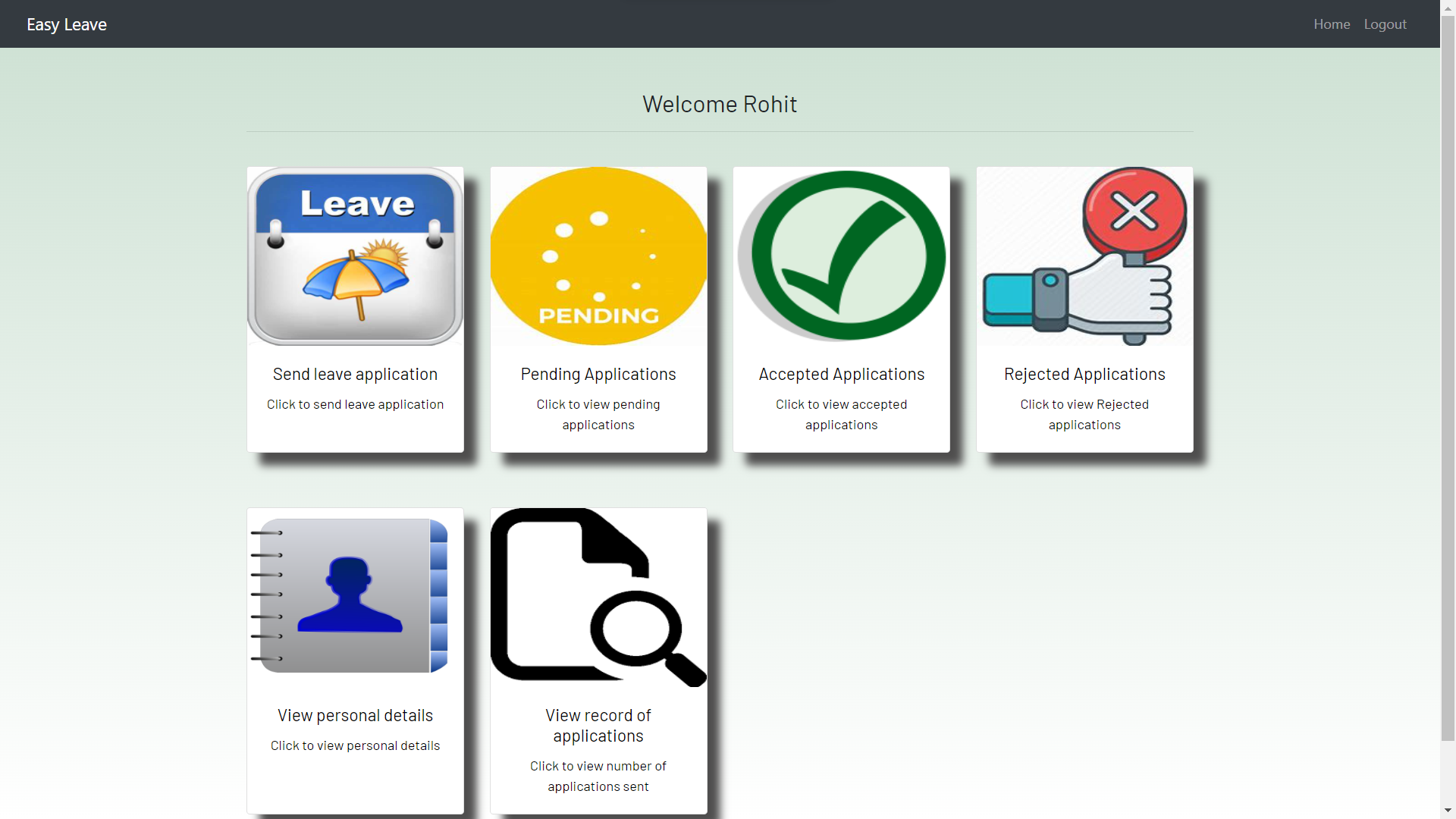


Fig. . Employee Dashboard

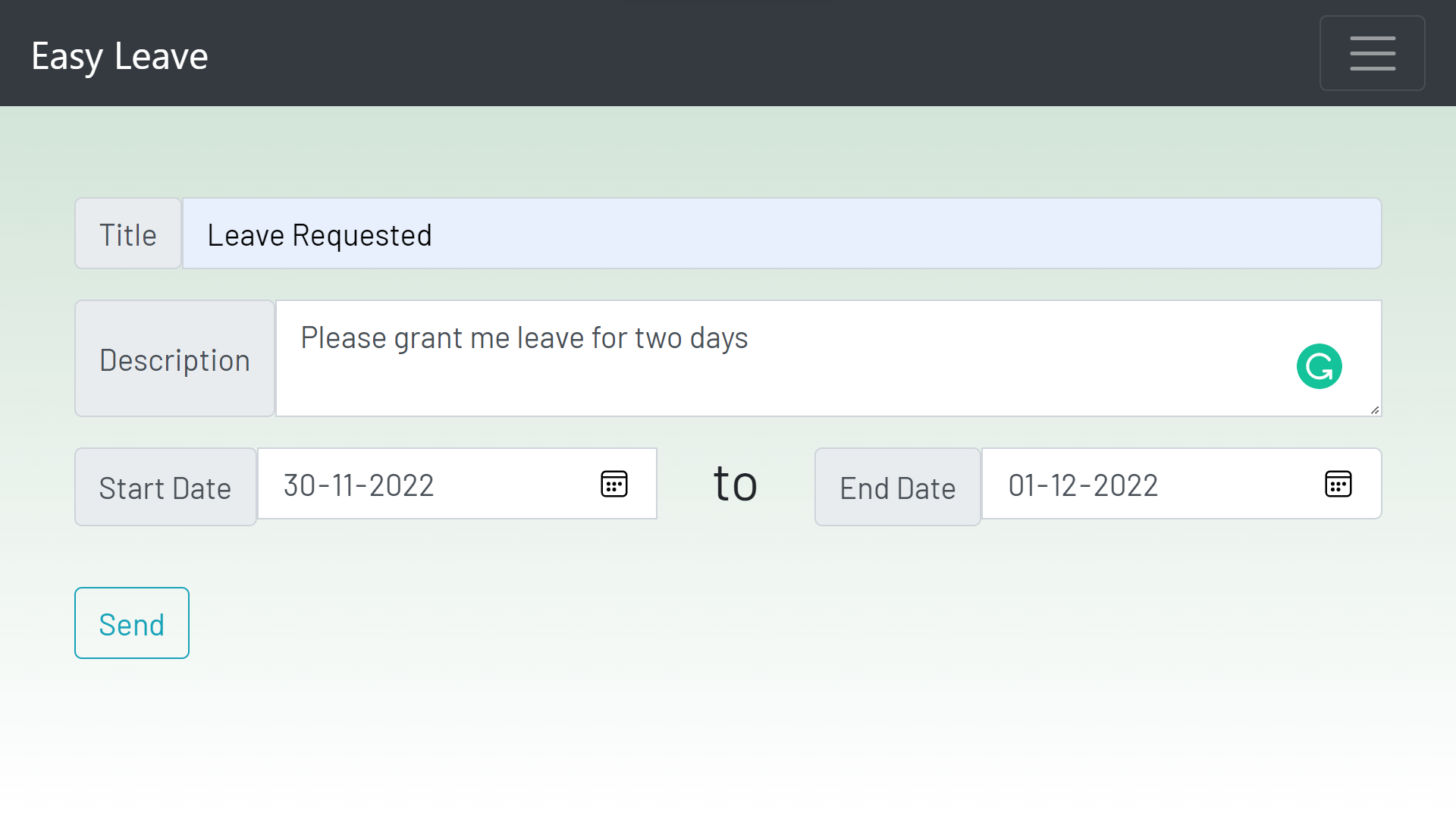


Fig. . Send Leave Application

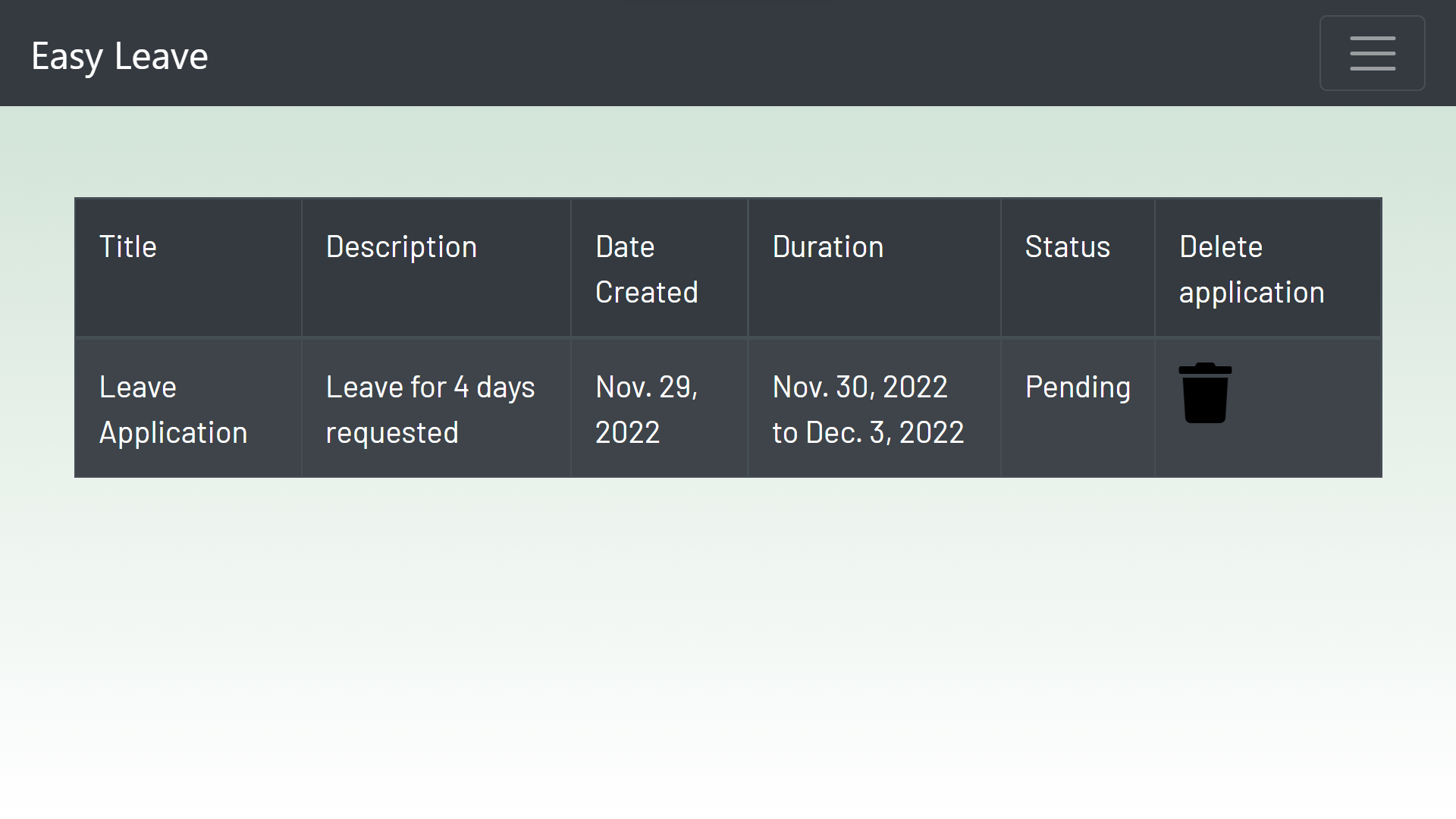


Fig. . Check Pending Applications

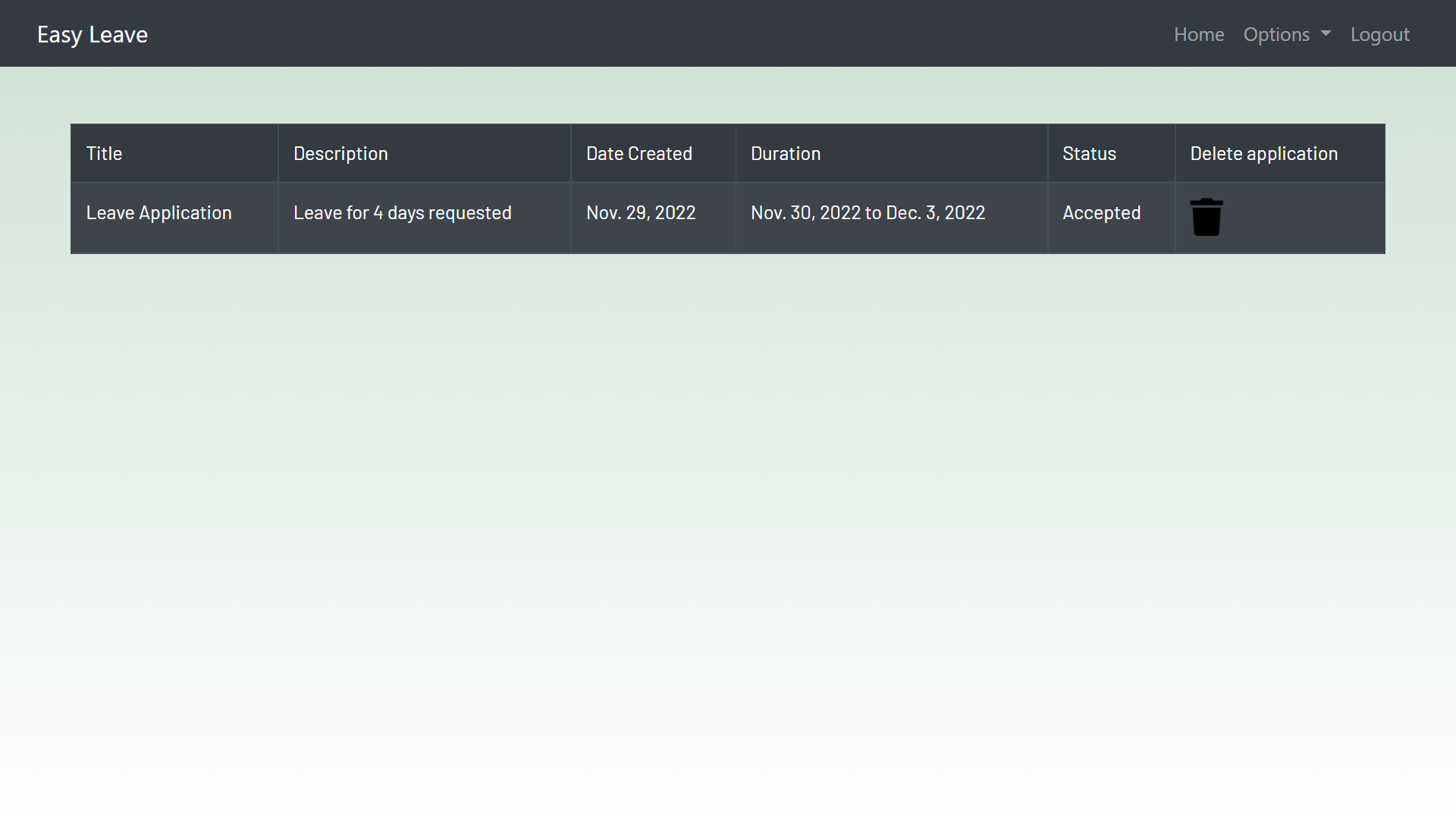


Fig. . Check for Accepted Applications

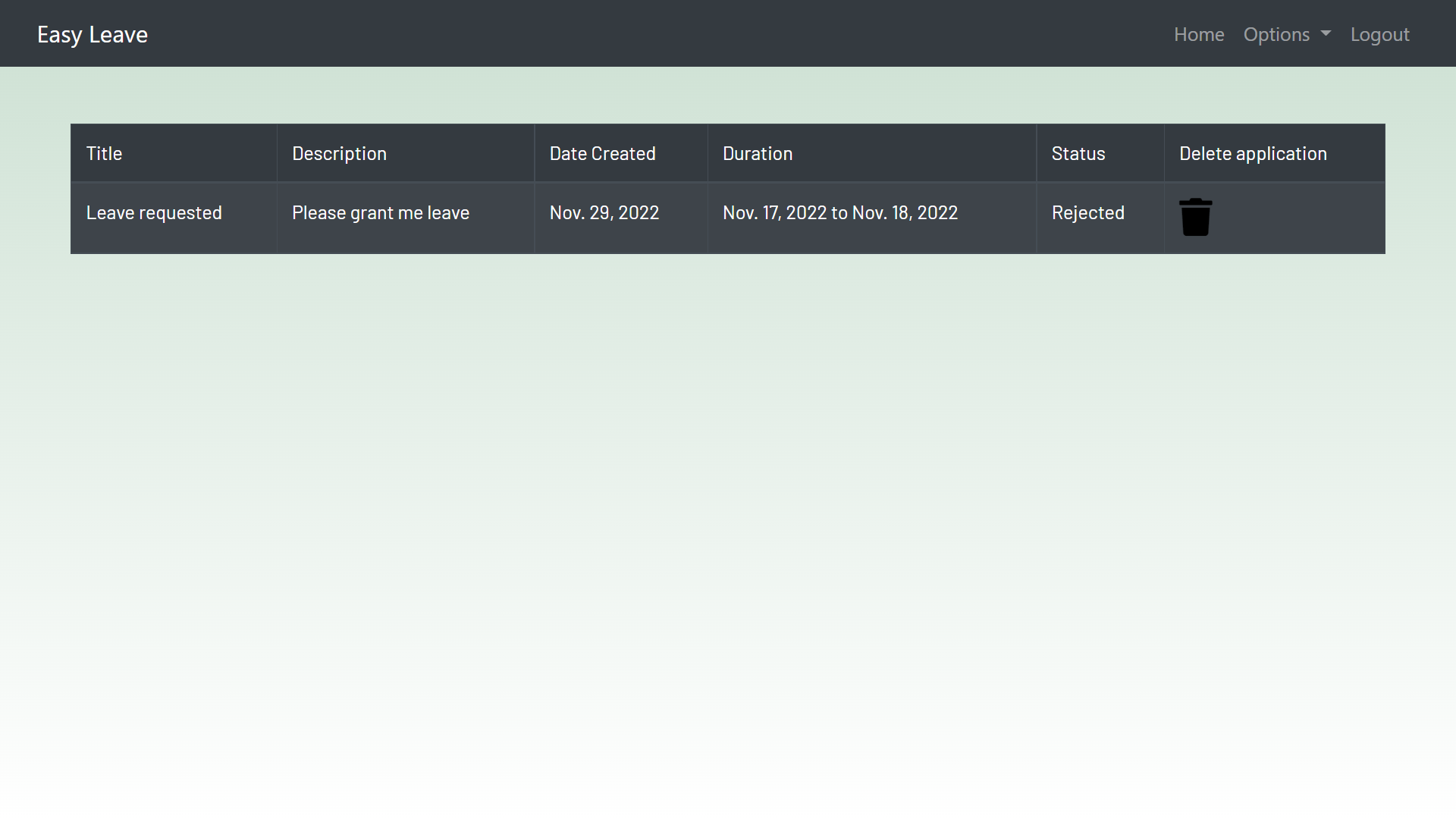


Fig. . Check for rejected applications

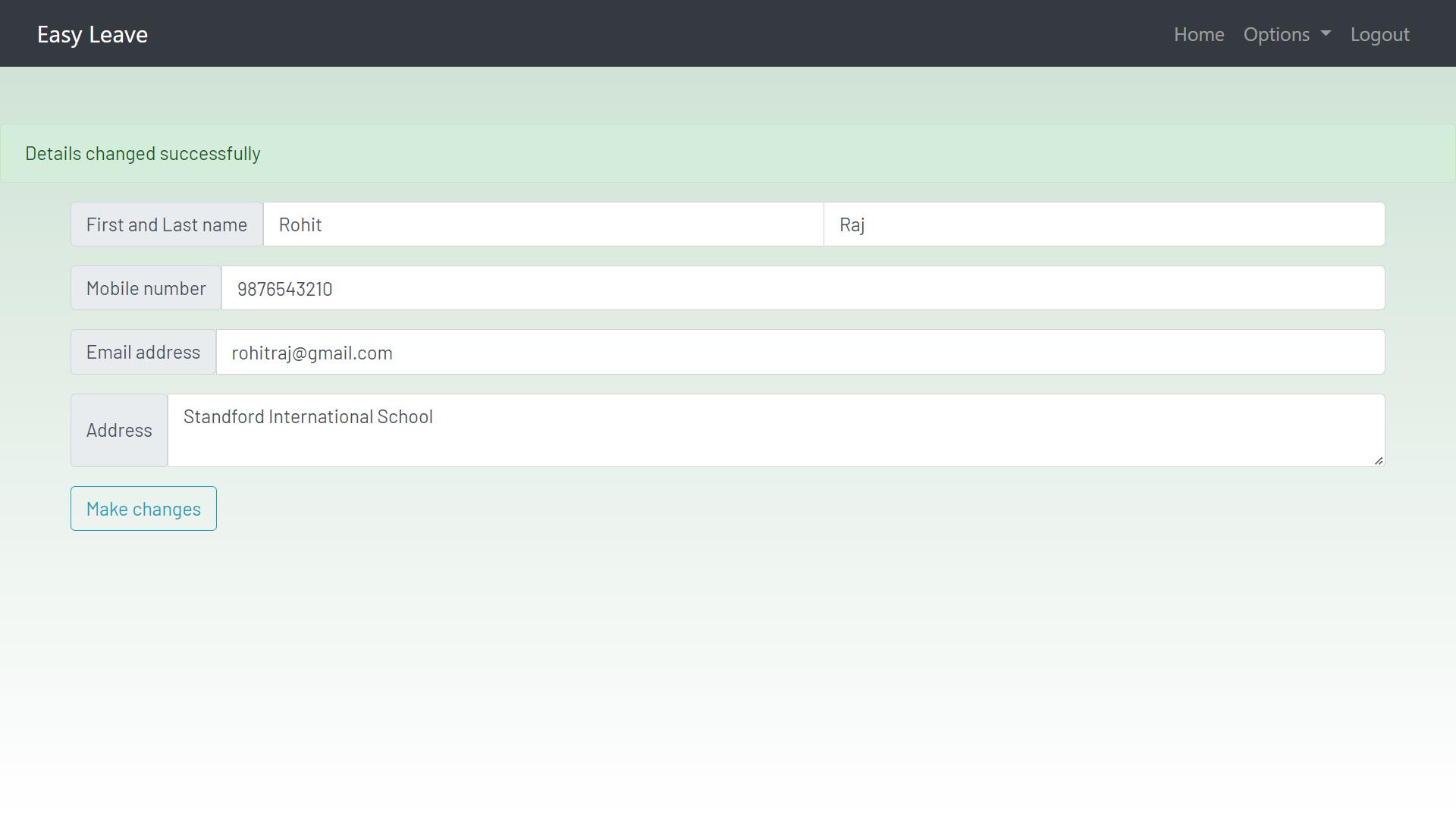


Fig. . View or Update details

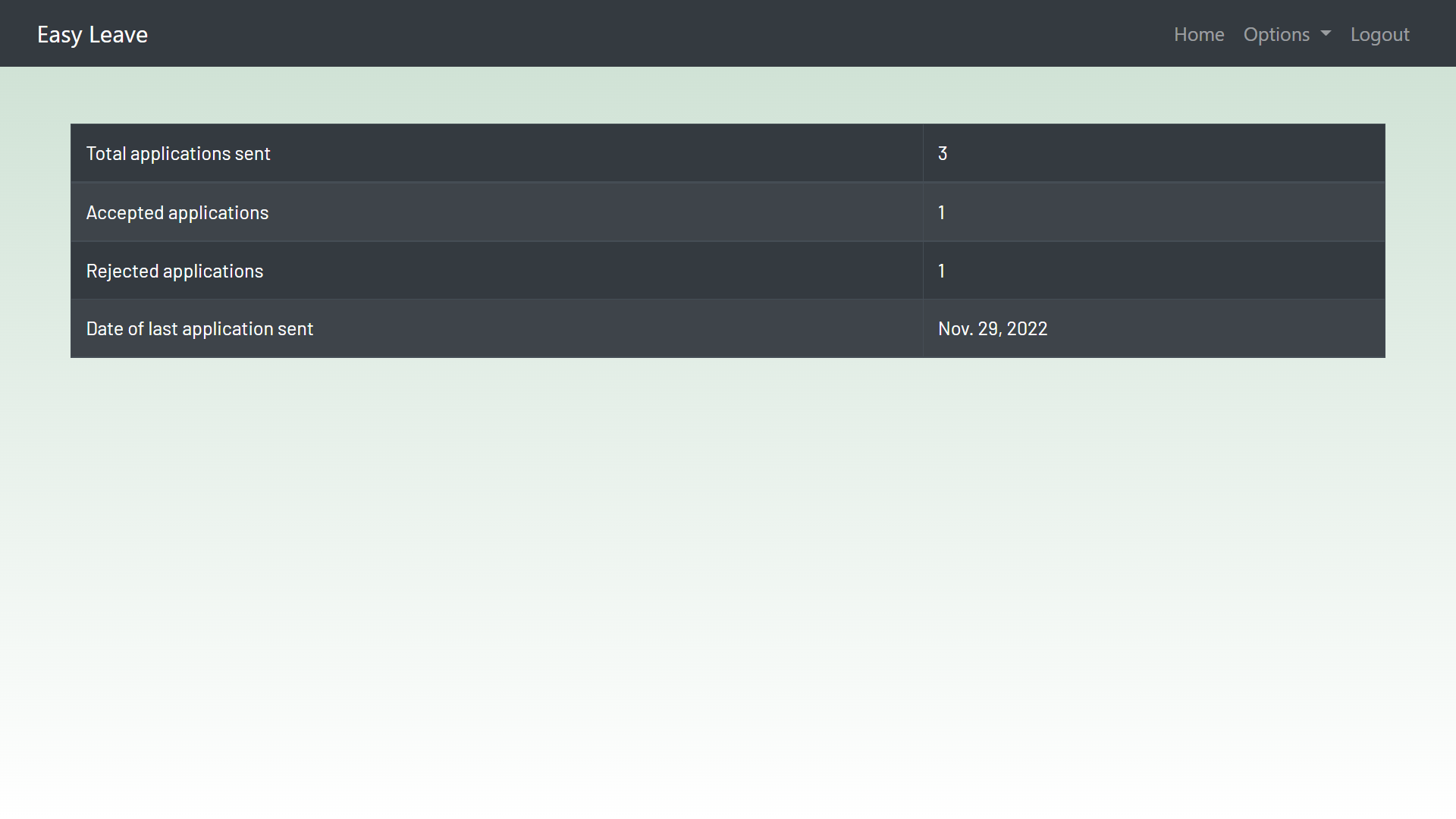


Fig. . View record of applications

**IV.**

**CONCLUSION & FUTURE SCOPE**

1. **Limitations & Drawbacks**

Every project has some limitations and drawbacks and here are some for our project:-

There are already some competitors and it is very difficult for a new player to quickly overthrow these softwares and establish itself.

1. **FUTURE SCOPE**

Although the requirements set out for the web application have been met, there are still some future scopes and areas which can be improved later on. A mobile version can be developed for the application so that the users can have access to the application from their mobile phones as well.

REFERENCES

[1]"An introduction to the Django Python web app framework", *Opensource.com*, 2022. [Online]. Available: <https://opensource.com/article/18/8/django-framework>.

[2]"youtube-search-python", *PyPI*, 2022. [Online]. Available: <https://pypi.org/project/youtube-search-python/>.

[3]"wikipedia", *PyPI*, 2022. [Online]. Available: <https://pypi.org/project/wikipedia/>.

[4]"Advantages And Disadvantages Of Online Learning", *eLearning Industry*, 2022. [Online]. https://elearningindustry.com/advantages-and-disadvantages-online-learning.

[5]R. Blog and I. education, "Importance of online education", *Times of India Blog*, 2022. [Online]. Available: https://timesofindia.indiatimes.com/readersblog/personal-blog/importance-of-online-education-36962/.