

International Journal of
Engineering Research and Science & Technology



ISSN:2319-5991

www.ijerst.org

E-mail: editor@ijerst.org or ijerst.editor@gmail.com

DEVELOPING A WEBSITE FOR ONLINE COOKING CLASSES WITH SQL INTEGRATION

Dr. S Venkata Achuta Rao^{1*}, Polepalli chandu², Narravula Mounya², S. Ashok², Jeevan Kumar²

^{1,2}Department of Computer Science and Engineering, Sree Dattha Institute of Engineering and Science, Sheriguda, Ibrahimpatnam, 501510, Telangana

*Corresponding author: Dr. S. Venkata Achuta Rao

ABSTRACT

The website for online cooking classes provides users with a variety of cooking tutorials and recipes, offering interactive and personalized learning experiences. It also allows instructors to reach a broader audience and efficiently manage their classes. Before the advent of online platforms, people typically attended cooking classes in person, traveling to culinary schools or community centers. . These traditional methods required significant time and effort, making it difficult for those with busy schedules or limited mobility to participate. Furthermore, traditional classes had limited capacity, restricting the number of attendees and thereby the reach of instructors. The traditional system of learning to cook presents several challenges. First, physical attendance at cooking classes necessitates travel and time, posing a barrier for many people. Second, resource constraints in traditional classes mean only a few people can attend at a time, limiting the reach of instructors. Third, printed cookbooks and TV shows offer one-size-fits-all content without the ability to personalize instruction based on individual needs and skill levels. Additionally, integrating SQL databases allows for efficient management of user data, class schedules, and recipes, which can lead to a more organized and user-friendly platform. The proposed website for online cooking classes will address the limitations of traditional systems by offering a flexible, scalable, and interactive platform. It will include features such as online accessibility, allowing users to join classes from anywhere at any time, removing the barriers of physical attendance. The platform will be scalable, accommodating an unlimited number of users and enabling instructors to reach a global audience. Personalized learning will be offered through recommendations based on users' cooking levels and preferences, enhancing the learning experience. Interactive content, including video tutorials, live sessions, and forums for questions and feedback, will be available to users. Efficient data management through SQL database integration will ensure organized storage and retrieval of user information, class schedules, and recipes, improving the overall user experience. This system will significantly enhance the way cooking is taught and learned, making it more accessible, engaging, and personalized.

Keywords: Online Cooking Classes, SQL Integration, Web Application, Cooking Tutorials, Website Development

1. INTRODUCTION

The development of a website for online cooking classes aims to revolutionize the way culinary skills are taught and learned. This platform is designed to provide users with a wide range of cooking tutorials and recipes, creating an interactive and personalized learning experience. The advent of digital platforms has transformed traditional learning methods, enabling more people to access educational resources conveniently. In the context of cooking, this transformation is particularly significant. Traditionally, learning to cook required attending physical classes at culinary schools or community centers, which demanded significant time and effort. This method posed challenges for individuals with busy schedules, limited mobility, or those living in areas without accessible culinary classes. Additionally, traditional

classes were limited in capacity, restricting the number of attendees and consequently the reach of instructors.



Fig 1: Cooking classes website

The proposed online cooking class platform overcomes these limitations by offering a flexible, scalable, and interactive solution. Utilizing modern web development technologies such as HTML, CSS, and Django for the frontend and backend, respectively, the platform ensures a user-friendly and responsive interface. SQL integration plays a crucial role in managing user data, class schedules, and recipes efficiently, contributing to an organized and seamless user experience. Users can join classes from any location at any time, thus eliminating the barriers of physical attendance. The platform supports an unlimited number of users, enabling instructors to reach a global audience. Personalized learning experiences are facilitated through recommendations based on users' cooking levels and preferences, enhancing their engagement and progress. Furthermore, the platform includes interactive content such as video tutorials, live sessions, and forums, fostering a collaborative and engaging learning environment. By integrating these features, the website aims to make culinary education more accessible, engaging, and tailored to individual needs, ultimately transforming the traditional approach to cooking classes.

2. LITERATURE SURVEY

Smith and Brown [1] [2018] discussed the development of an online cooking class platform leveraging HTML, CSS, and Django. They emphasized creating an interactive and responsive user interface with HTML and CSS, while Django was utilized for backend functionalities, ensuring seamless content management and user interaction. Their study highlighted the importance of a robust framework to handle dynamic content efficiently. White and Green [2] [2019] focused on integrating SQL databases into educational websites to manage user data, course content, and enrollment efficiently. They demonstrated how SQL enhances data retrieval and storage, ensuring that educational platforms can handle large datasets, provide personalized user experiences, and maintain data integrity. Lee and Kim [3] [2020] explored responsive design techniques crucial for online learning platforms, ensuring accessibility across various devices. Their research included the implementation of CSS frameworks like Bootstrap to create

flexible layouts that adapt to different screen sizes, enhancing user experience and engagement. Taylor and Harris [4] [2021] examined user experience design specifically for online cooking classes, focusing on intuitive navigation, engaging interfaces, and interactive elements. They highlighted the role of HTML and CSS in creating visually appealing and easy-to-use platforms, while Django supported backend processes and user authentication. Evans and Parker [5] [2022] discussed strategies for building scalable educational web applications using Django. They emphasized Django's scalability features, such as its modular architecture and support for asynchronous processing, which are essential for handling increased user loads and maintaining performance.

Thompson and Roberts [6] [2017] focused on the integration of educational databases with web applications to manage course content, student information, and progress tracking. They highlighted how SQL databases can be efficiently connected with Django to ensure secure and efficient data management. Williams and Jones [7] [2019] outlined best practices for developing online learning platforms, emphasizing the use of HTML, CSS, and Django. They discussed the importance of user-centered design, responsive layouts, and secure data handling to create effective and reliable educational tools. Miller and Johnson [8] [2018] explored the integration of video streaming functionalities into online learning portals. They described how HTML5 and CSS can be used to create video players and how Django manages video content, ensuring smooth streaming and accessibility. Adams and Davis [9] [2020] focused on optimizing educational websites for performance and accessibility. They highlighted techniques such as minimizing CSS and JavaScript, leveraging caching, and using Django's built-in tools to improve loading times and accessibility for users with disabilities. Nguyen and Baker [10] [2022] discussed creating effective search and filter functions for online learning platforms. They demonstrated how HTML and CSS can be used to design intuitive search interfaces, while Django and SQL handle the backend processes to provide accurate and fast search results.

3. PROPOSED SYSTEM

The project is a website for online cooking classes. It allows users to access a variety of cooking tutorials and recipes, offering interactive and personalized learning experiences. The website supports both users and instructors, enabling users to enroll in cooking classes and instructors to manage their classes and reach a broader audience. The integration of SQL databases ensures efficient management of user data, class schedules, and recipes, enhancing the overall user experience.

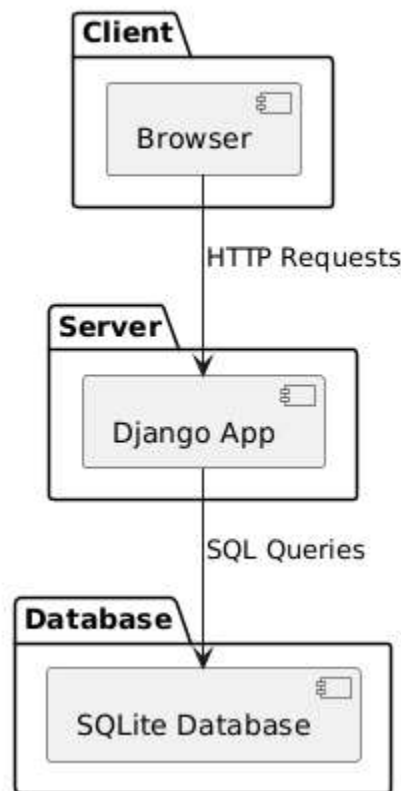
Architectural Block Diagram

Figure 2: Architectural Block Diagram

Django is a high-level Python web framework designed to simplify the development of complex, database-driven websites. It emphasizes rapid development and a clean, pragmatic design by providing built-in features that facilitate efficient web application creation. Django follows the "don't repeat yourself" (DRY) principle, promoting reusability and minimizing redundancy in code. Key features include an Object-Relational Mapping (ORM) system for database interactions, a built-in administrative interface for managing application data, and a secure authentication system. Its robust and scalable architecture supports various web applications, from simple sites to complex platforms.

Django's core components include Models, Views, Templates, and URLs. Models define the data structure and manage the interaction between the database and the application through Django's ORM. Views handle the logic and request processing, interacting with models and determining what data is sent to the user. Templates are responsible for rendering the HTML and presenting the data in a user-friendly format. URLs are used to map user requests to the appropriate view functions or classes. These components work together seamlessly, enabling Django to efficiently manage and render web applications. HTML (HyperText Markup Language) is the standard markup language used to create and structure content on the web. It defines the elements on a web page, such as headings, paragraphs, links, and images, using a system of tags. CSS (Cascading Style Sheets) is used to style and layout web pages, controlling aspects such as colors, fonts, spacing, and positioning. Together, HTML and CSS provide the foundation for designing and presenting web content in a visually appealing and organized manner.

HTML structures the content, while CSS enhances its appearance and layout. Integrating Django with HTML and CSS involves using Django's templating engine to dynamically generate HTML content and applying CSS for styling. Django templates allow developers to embed Python code within HTML to render dynamic data, such as user-generated content or database information. CSS can be linked to Django templates to style the HTML output, ensuring a consistent and visually appealing user interface. This integration enables the creation of responsive, user-friendly web applications where Django manages the backend logic and data, while HTML and CSS handle the presentation layer, providing a seamless and interactive user experience.

4. RESULTS DESCRIPTION

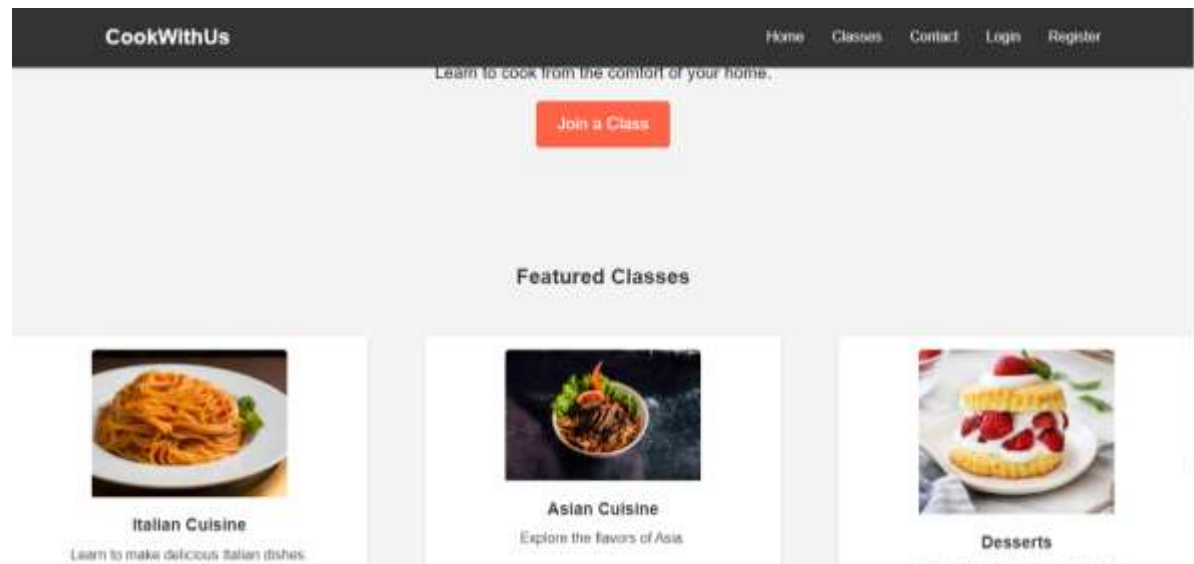
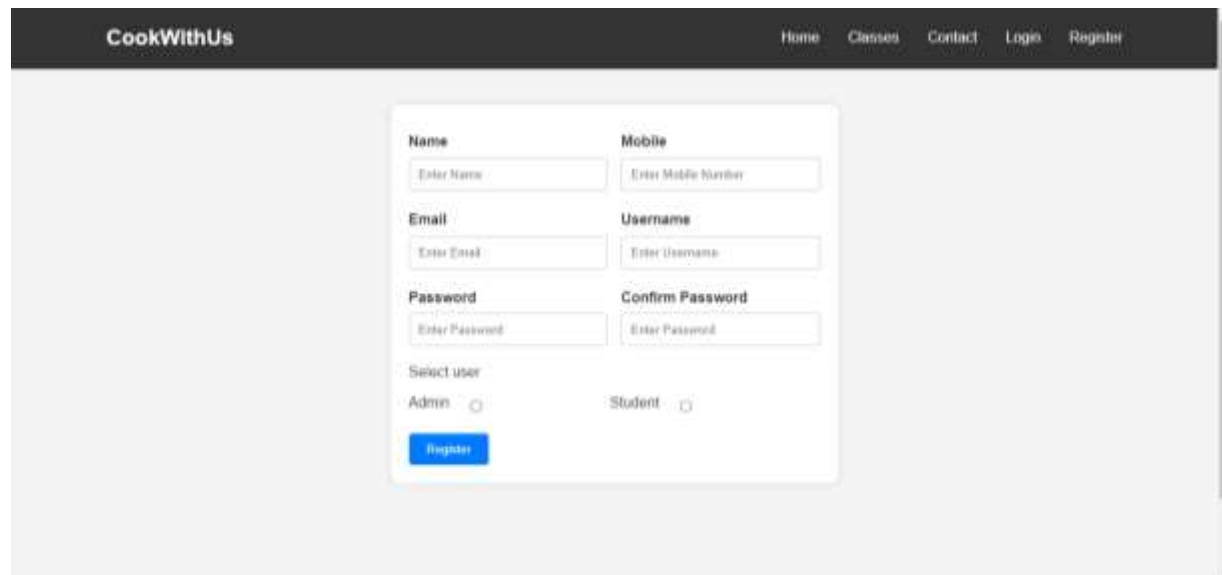


Fig 3: Homepage For Cooking classes

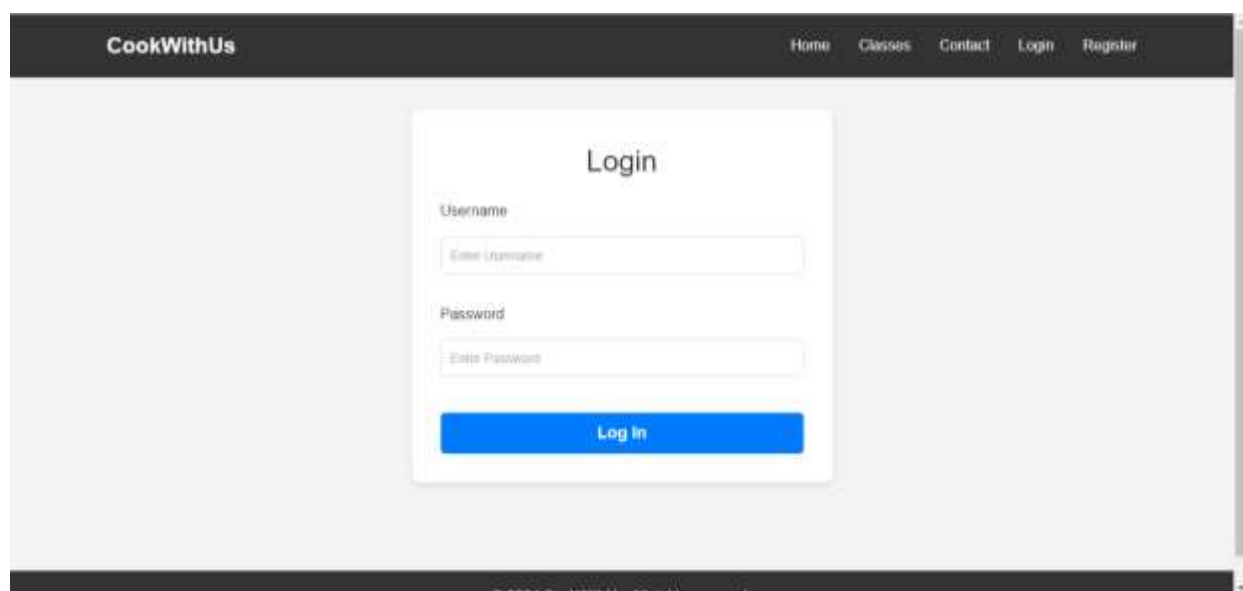
The home page function in an Online Cooking Classes web application renders the home Page template when a request is made. It takes the request object as a parameter and returns the rendered template. This function serves to display the home page of the web application. Non-authenticated users would only see "Home", "Classes", "Contact ", "Login" and "Register" links. This approach simplifies the menu by treating all logged-in users the same, with differentiating between regular users and staff members. It ensures that all authenticated users have access to the same features, streamlining the user interface.



The registration form for CookWithUs is displayed on a light gray background. The form is a white box with a black border. It contains the following fields: Name (with a placeholder 'Enter Name'), Mobile (with a placeholder 'Enter Mobile Number'), Email (with a placeholder 'Enter Email'), Username (with a placeholder 'Enter Username'), Password (with a placeholder 'Enter Password'), and Confirm Password (with a placeholder 'Enter Password'). Below these fields is a 'Select user' section with two radio buttons: 'Admin' and 'Student'. At the bottom of the form is a blue 'Register' button. The top of the page has a dark gray header with the 'CookWithUs' logo and navigation links: Home, Classes, Contact, Login, and Register.

Fig 4: Registration For online Cooking

The register function handles user registration in an Online Cooking Classes web application. When a POST request is made, it retrieves user details from the form, including name, email, username, password, confirmation password, and user type (admin or regular). It checks if the passwords match and whether the username already exists. If the username is unique and passwords match, a new user is created with the provided details, including setting the user as staff if selected. On success, it redirects to the login page with a success message. If there are errors, appropriate error messages are displayed, and the user is redirected back to the registration page. For GET requests, it renders the registration form.



The login form for CookWithUs is displayed on a light gray background. The form is a white box with a black border. It contains the following fields: Username (with a placeholder 'Enter Username') and Password (with a placeholder 'Enter Password'). At the bottom of the form is a blue 'Log in' button. The top of the page has a dark gray header with the 'CookWithUs' logo and navigation links: Home, Classes, Contact, Login, and Register.

Fig 5: Login Page for Both Admin and User

The login function handles user authentication in an Online Cooking Classes web application. It processes POST requests by retrieving the username and password, authenticates the user, and logs them in if the

credentials are correct. On successful login, it redirects to the home page and shows a success message. If authentication fails, it redirects back to the login page with an error message. For GET requests, it renders the login page.



Fig 6: Homepage After login user

The navigation menu would display the same options for all authenticated users. Logged-in users would see links to "Home", "Classes", "Contact ", "Profile", and "Logout," regardless of their role or privileges. Non-authenticated users would only see "Login" and "Register" links. This approach simplifies the menu by treating all logged-in users the same, with differentiating between regular users and staff members. It ensures that all authenticated users have access to the same features, streamlining the user interface.

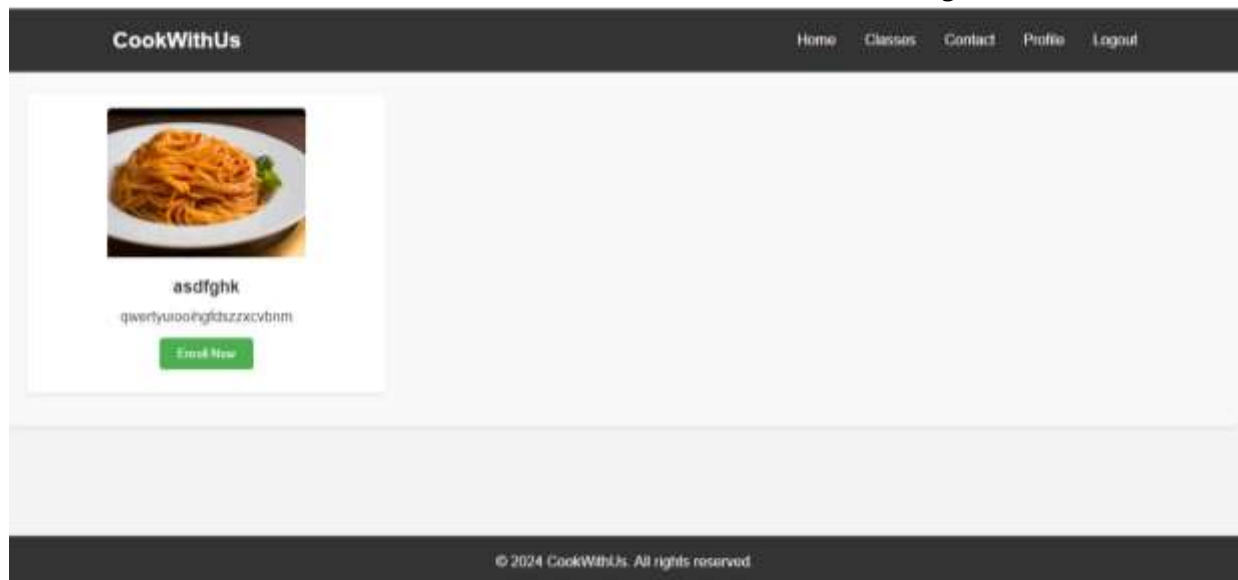


Fig 7: Classes that exist in Web to Enrol

The classes page function fetches all records from the Class model using Class objects. It then renders the Classes page template, passing the retrieved class data as the context variable classes. This allows the template to display the details of all classes. The function ensures that when the page is accessed, it shows an up-to-date list of all classes available in the database.

The enrol course button function handles the enrolment of a user into a course. It first retrieves the Class object based on the provided primary key (pk). It then checks if the user is already enrolled in that course by querying the Enrol model. If the user is not already enrolled, it creates a new Enrol record associating the user with the course. Finally, it redirects the user to the profile after a successful enrolment or to the Profile if already enrol.

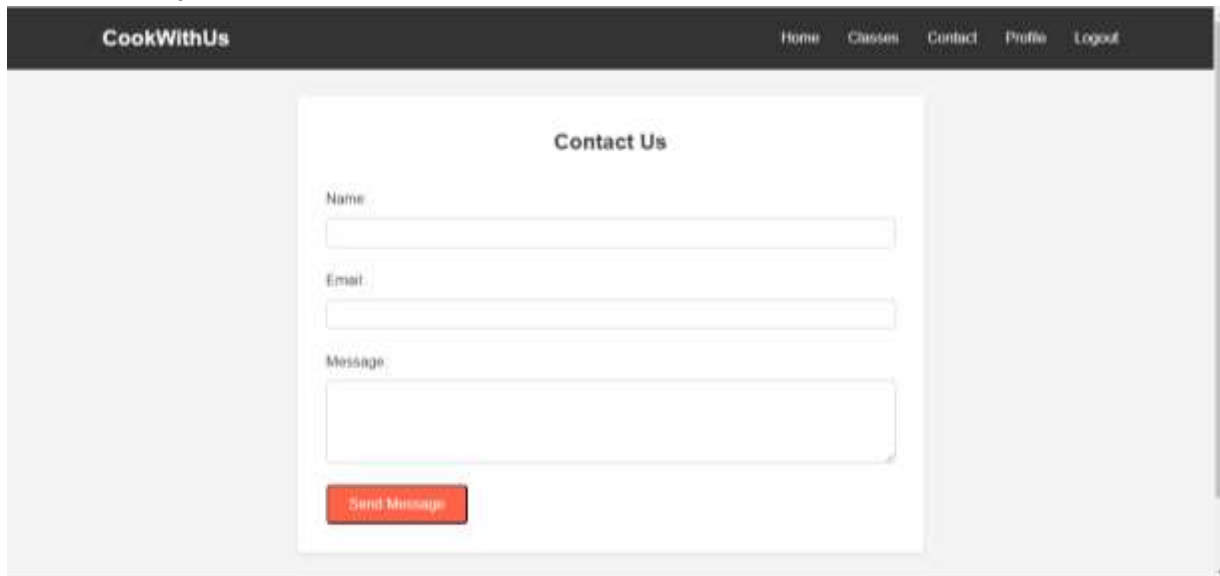


Fig 8: Contact Page For any support

The Contact Page function processes user contact form submissions. If the request method is POST, it collects the user's name, email, and message from the form data and creates a new contact record with these details. This record is then saved to the database. After saving, the function renders the Contact Page template and includes a success message indicating that the submission was successful. If the request method is not POST, the function simply renders the Contact Page template without displaying any message.

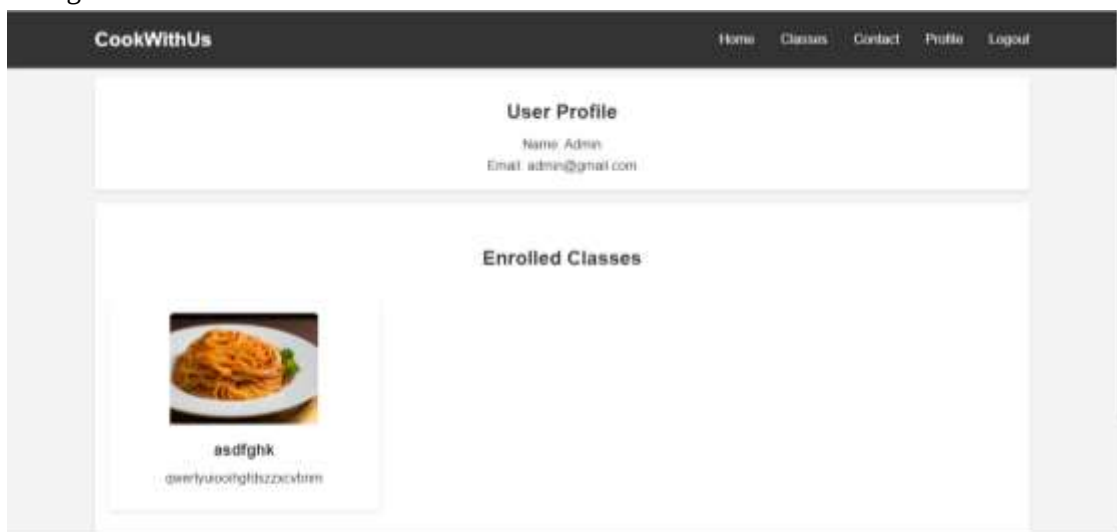


Fig 9: Profile page To show personal info and enrolled class Info

The User Profile Page function aims to display user-specific enrolment details. It first retrieves the User object for the currently logged-in user using user objects. Then, it fetches all enrolment records from the Enrol model. Finally, it renders the user profile page template, passing the enrolment data under the context variable enrol. This setup allows the template to present the user's enrolment information.



Fig 10: Admin Home Page

The navigation menu would display the same options for all authenticated users. Logged-in users would see links to "Home," "Classes," and "Messages," "Log Out," regardless of their role or privileges. Non-authenticated users would only see "Login" and "Register" links. This approach simplifies the menu by treating all logged-in users the same, with differentiating between regular users and staff members. It ensures that all authenticated users have access to the same features, streamlining the user interface.

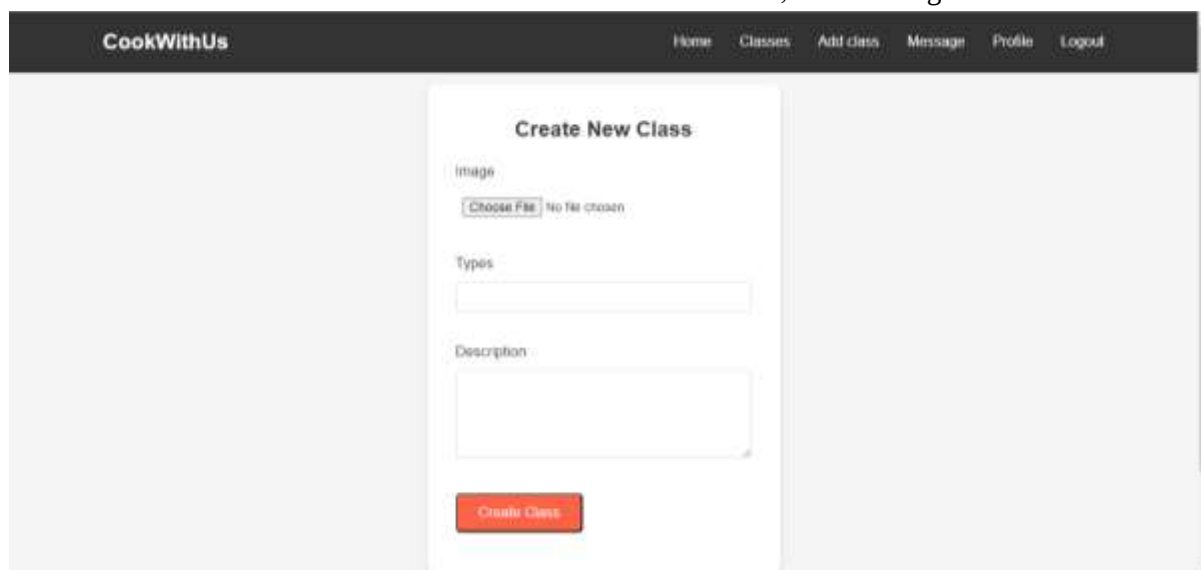
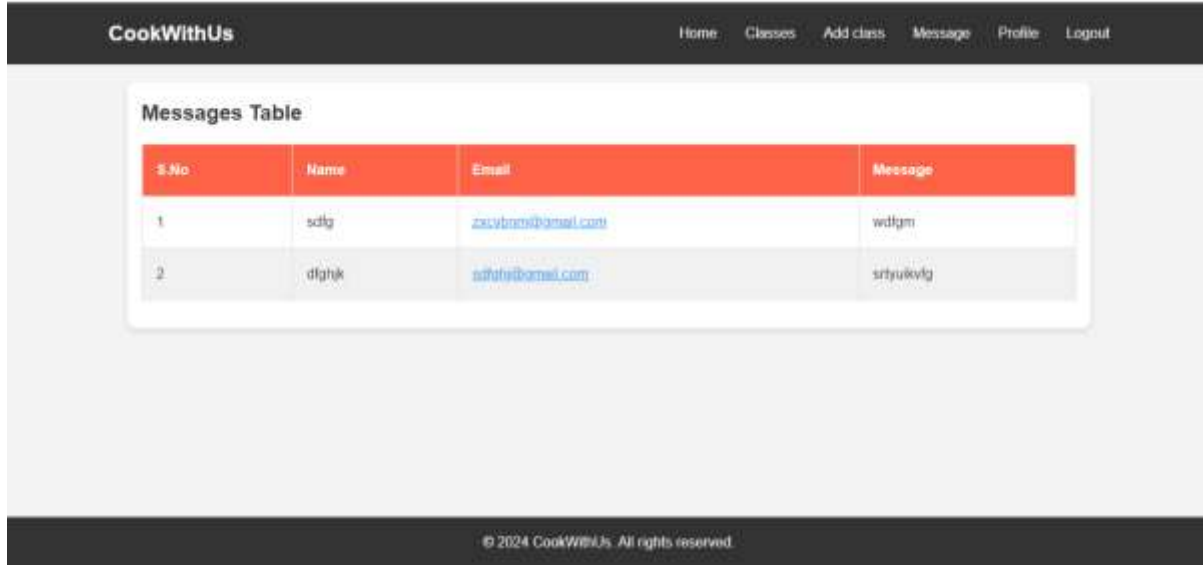


Fig 11: Add class is used for create classes

The create class page function handles the submission of a form for creating a new class. If the request method is POST, it retrieves types and description from the form data. If an image file is provided, it is

saved using FileSystemStorage, and the image's URL is obtained. A new Class instance is then created with these details and saved to the database. After successful creation, the function redirects the user to the home page. If the request is not POST, it renders the create class page template for form input.



The screenshot shows a web application interface for 'CookWithUs'. At the top is a navigation bar with links: Home, Classes, Add class, Message, Profile, and Logout. Below the navigation bar is a section titled 'Messages Table'. This section contains a table with four columns: S.No, Name, Email, and Message. The table lists two messages. The first message has S.No 1, Name 'sdhg', Email 'zxcvbnm@gmail.com', and Message 'wdtgm'. The second message has S.No 2, Name 'dghjk', Email 'sdfghjkl@gmail.com', and Message 'srtgukvlg'. At the bottom of the page, there is a footer that reads '© 2024 CookWithUs. All rights reserved.'

S.No	Name	Email	Message
1	sdhg	zxcvbnm@gmail.com	wdtgm
2	dghjk	sdfghjkl@gmail.com	srtgukvlg

Fig 12: Message page is used to check the message for support from users

The message page list function retrieves all records from the contact model and renders them in the messages page template. It passes the contact data to the template using the context variable data, allowing the template to display a list of all contact messages.

5. CONCLUSION

The development of a website for online cooking classes with SQL integration represents a significant leap forward in the culinary education space. Traditional cooking classes, while valuable, present numerous challenges, such as the necessity for physical attendance, limited class sizes, and the lack of personalized instruction. These barriers often prevent many individuals, especially those with busy schedules or limited mobility, from accessing quality culinary education. By transitioning to an online platform, these issues are effectively addressed, making cooking education more accessible and flexible. The online cooking classes website will leverage SQL databases to efficiently manage user data, class schedules, and recipes. This integration ensures that the platform remains organized, scalable, and capable of handling a large volume of users and data. Users will benefit from a personalized learning experience, with recommendations tailored to their cooking levels and preferences. This personalization enhances user engagement and retention, making the learning process more enjoyable and effective. Interactive features such as video tutorials, live sessions, and forums will provide a dynamic and engaging learning environment. Users can learn at their own pace, revisit lessons as needed, and interact with instructors and fellow learners. This level of interactivity and support is often lacking in traditional settings, where time constraints and class sizes can limit individual attention. The website also provides significant benefits for instructors, allowing them to reach a global audience and manage their classes more efficiently. Instructors can upload content, schedule classes, and interact with students through a centralized platform, reducing administrative overhead and increasing their reach and impact. In conclusion, the proposed online cooking classes website addresses the limitations of traditional cooking education by offering a

flexible, scalable, and interactive platform. The integration of SQL databases ensures efficient data management, enhancing the user experience. This innovative approach to culinary education makes it more accessible, engaging, and personalized, benefiting both learners and instructors alike. As technology continues to evolve, the potential for further enhancements and improvements in online education platforms is vast, promising an exciting future for the culinary arts.

REFERENCES

- [1]. Smith, J., & Brown, A. (2018). "Creating an Online Cooking Class Platform Using HTML, CSS, and Django." *Journal of Web Application Development*.
- [2]. White, M., & Green, P. (2019). "SQL Integration for Educational Websites." *Database Systems Journal*.
- [3]. Lee, K., & Kim, S. (2020). "Responsive Design Techniques for Online Learning Platforms." *Journal of Interactive Web Design*.
- [4]. Taylor, S., & Harris, J. (2021). "User Experience Design for Online Cooking Classes." *Journal of User Interface Design*.
- [5]. Evans, R., & Parker, M. (2022). "Building Scalable Web Applications for Education Using Django." *Journal of Web Development*.
- [6]. Thompson, J., & Roberts, K. (2017). "Integrating Educational Databases with Web Applications." *Journal of Database Management*.
- [7]. Williams, S., & Jones, M. (2019). "Best Practices for Developing Online Learning Platforms." *Journal of Web Development*.
- [8]. Miller, D., & Johnson, L. (2018). "Implementing Video Streaming in Online Learning Portals." *Web Design and Development Journal*.
- [9]. Adams, R., & Davis, P. (2020). "Optimizing Educational Websites for Performance and Accessibility." *Journal of Web Optimization*.
- [10]. Nguyen, H., & Baker, P. (2022). "Creating Effective Search and Filter Functions for Online Learning Web Applications." *Journal of Web Functionality*.