# COLLEGE FEST MANAGEMENT SYSTEM

## IGNITE

**A PROJECT REPORT SUBMITTED IN PARTIAL**

**FULFILMENT OF REQUIREMENT**

**FOR THE AWARD OF THE DEGREE**

## MASTER OF COMPUTER APPLICATION(MCA)

**OF**

**MAHATMA GANDHI UNIVERSITY, KOTTAYAM**

**BY**

**Abdul Hakeem M K**

**Reg No : 22PMC101**

MARIAN COLLEGE
KUTTIKKANAM
(AUTONOMOUS)

MAKING COMPLETE

## Marian College Kuttikanam Autonomous

**Peermade, Kerala – 685 531**

**2022**

A Project Report on

# COLLEGE FEST MANAGEMENT SYSYEM

## IGNITE

**SUBMITTED IN PARTIAL FULFILMENT OF REQUIREMENT FOR THE AWARD OF THE DEGREE**

## MASTER OF COMPUTER APPLICATION(MCA)

**OF**

**MAHATMA GANDHI UNIVERSITY, KOTTAYAM**

**By**

**Abdul Hakeem M K**

**Reg No. 22PMC101**

**Under the guidance of**

**MR . SATHEESH KUMAR S**

**Assistant Professor**

PG Department of Computer Applications

Marian College Kuttikkanam( Autonomous)

MARIAN COLLEGE
KUTTIKKANAM
(AUTONOMOUS)

MAKING COMPLETE

## Marian College Kuttikanam Autonomous

**Peermade, Kerala – 685 531**

**2022**

**PG DEPARTMENT OF COMPUTER  APPLICATIONS**

# Marian College Kuttikkanam Autonomous

**MAHATMA GANDHI UNIVERSITY, KOTTAYAM**

**KUTTIKKANAM – 685 531, KERALA.**

# CERTIFICATE

This is to certify that the project work entitled

## COLLEGE FEST MANAGEMENT SYSTEM

is a bonafide record of work done by

## ABDUL HAKEEM M K

**Reg. No 22PMC101**

In partial fulfillment of the requirements for the award of Degree of

## MASTER OF COMPUTER APPLICATIONS [MCA]

During the academic year 2022-2023

**MR . SATHEESH KUMAR S**

ASSISTANT PROFFESSOR

PG Department of Computer Applications

Marian College Kuttikkanam Autonomous

**Mr Win Mathew John**

Head of the Department

PG Department of Computer Applications

Marian College Kuttikkanam Autonomous

**Internal Examiner**

**External Examiner**

# ABSTRACT

Ignite is an advanced web-based platform designed to streamline the management of college tech fests. It serves as a one-stop solution for students, judges, and administrators, offering a range of features to facilitate event registration, information dissemination, result tracking, and profile management. Built with a combination of HTML, CSS, JavaScript, Bootstrap, Python, and Django, Ignite provides a seamless user experience and efficient backend functionality.

This abstract provides an overview of Ignite's key user roles and their corresponding capabilities. The three main user roles are students, judges, and admins. Students can easily register for events, access event details, view results, and manage their profiles. Judges, registered by the admin, enjoy the same features as students and additionally have the ability to upload event results. Admins, the super users, have full control over the system and can effortlessly add events, stages, judges, and edit various aspects through the user-friendly frontend interface.

Ignite's frontend has been developed using HTML, CSS, JavaScript, and Bootstrap to ensure an intuitive and visually appealing user interface. On the backend, Python and Django provide a robust and scalable foundation, enabling efficient data management, authentication, and seamless communication between the various components of the system.

Ignite aims to revolutionize the management of college tech fests by simplifying event registration, enhancing information accessibility, and automating result tracking. With its user-centric design and powerful features, Ignite empowers students, judges, and admins to effectively navigate and orchestrate successful tech fest experiences.

### 1. Admin Module:

- The admin module is responsible for overall system management and administrative tasks.
- It includes functionalities such as managing events, adding judges and managing judges, and managing users, also have access to results and stage management.

**2. Judge Module:**

- The judge module is designed for uploading the results of specific events.
- Judges can edit their profiles, including personal information and qualification.
- They can see the schedule of events, view results.

**3. Student Module**:

- The student module can registration with their credentials and can select the events they are interested.
- Students can edit profiles, including personal details and preferences.
- They can see the results of every events and details of each events.

# PROJECT REQUIREMENTS

**1. User Registration and Authentication:**

- Students should be able to register an account on the Ignite platform.
- Proper authentication and authorization mechanisms should be implemented to ensure secure access to different modules.

**2. Admin Module:**

- Admins should have the ability to manage user accounts, including creating, editing, and deleting accounts.
- Admins should be able to add judges and allocate them to specific events.

**3. Instructor Module:**

- Judges should be able to edit and manage their profiles, including personal information and qualification.
- The module should allow judges to view the results and event details.

**4. Student Module:**

- Students should have the ability to create and manage their profiles, including personal information and preferences.
- Students should be able find the events schedules and details of each events.
- The module should allow students to register for the events they are interested to participate.
- The should be able to see the results event ways.

**5. Result Upload:**

- Judges should have the ability to upload event results, including participant names and grades.

**6. Database Management:**

- The system should incorporate a database to store user information, instructor profiles, student bookings, and other relevant data.
- Proper data modeling and database management techniques should be employed to ensure data integrity and scalability.

**7. User Interface and User Experience:**

- The system should have an intuitive and user-friendly interface for easy navigation and interaction.
- The user interface should be responsive and accessible on different devices.

**8. Security and Privacy:**

- The system should implement appropriate security measures, including data encryption, to protect user information and ensure privacy.
- Access control and authorization mechanisms should be implemented to restrict access to sensitive functionalities and data.

**9. Admin Dashboard:**

- Admins should have access to a user-friendly dashboard to manage events, stages, judges, and other system components.

These requirements serve as a starting point for the Ignite college tech fest management system. Additional requirements may arise based on specific organizational needs and user feedback during the development and testing phases.

# FEATURES AND HIGHLIGHTS OF THE PROJECT

1. **User Registration and Login:**
   Students can easily register and create an account on the Ignite platform, providing them with personalized access to the system. They can log in using their credentials for subsequent visits.
2. **Event Registration and Details:**
   Students can browse through the list of tech fest events, view detailed information about each event (date, time, venue, rules, requirements, etc.), and register for the events of their interest. This ensures efficient event planning and participation.
3. **Result Tracking and Upload:**
   Judges, registered by the admin, can upload event results, including participant names and scores. This feature enables real-time result tracking and transparency, enhancing the overall competitiveness and fairness of the tech fest.
4. **Profile Management:**
   Students and judges have the ability to manage their profiles within the system. They can edit personal information, areas of interest, contact details, and other relevant information, ensuring up-to-date and accurate user profiles.
5. **Admin Dashboard:**
   The admin has access to a comprehensive dashboard that provides centralized control over the system. From the dashboard, the admin can manage events, stages, judges, and other crucial aspects of the tech fest.
6. **Event Creation and Management:**
   The admin can effortlessly create new events, specifying event details, rules, requirements, and other necessary information. This feature simplifies event organization and ensures all relevant information is readily available to students and judges.
7. **Judge Management:**
   The admin can add, remove, and manage judges for different events. This feature enables efficient coordination between judges and the admin, ensuring a smooth judging process throughout the tech fest.

8. **User-Friendly Interface:**
   Ignite offers an intuitive and visually appealing user interface, designed using HTML, CSS, JavaScript, and Bootstrap. The interface is easy to navigate and enhances the overall user experience.
9. **Performance and Scalability:**
   The system is designed to handle a large number of concurrent users without compromising performance. It is scalable to accommodate an increasing number of events, users, and data, ensuring a smooth and efficient tech fest management experience.
10. **Security and Data Privacy:**
    Ignite incorporates robust security measures, including user authentication and data encryption, to safeguard user information and ensure data privacy.
11. **Accessibility:**
    The system adheres to accessibility standards, ensuring that users with disabilities can access and use the platform effectively.

These features and highlights of the Ignite College Tech Fest Management System contribute to a seamless and efficient management experience for students, judges, and admins, ultimately enhancing the success and organization of college tech fests.

# TECHNICAL ASPECTS

-Architecture of your project

-Class Diagram

**Django Framework**:

Django is the primary framework used for building the LMS project. It provides a robust foundation for developing web applications, offering features such as URL routing, database connectivity, authentication, and templating.

**Python:**

The LMS project is written in Python, a versatile and powerful programming language known for its simplicity and readability. Python is used to implement the backend logic, handle data processing, and perform various system-level operations.

**HTML/CSS/JavaScript**:

The project utilizes front-end technologies such as HTML, CSS, and JavaScript to develop the user interface and enhance user interactions. These technologies are essential for creating responsive and visually appealing web pages.

**Database Management System (DBMS):**

The project utilizes a DBMS to store and manage data related to users, courses, content, assessments, and grades. Popular choices for DBMS in Django projects include PostgreSQL, MySQL, and SQLite.

**Presentation Layer:**

This layer handles the user interface and user interactions.

It includes components such as HTML/CSS templates, JavaScript, and views in Django.

User actions trigger requests to the application layer for processing.

**Application Layer:**

This layer contains the business logic and orchestrates the flow of data and operations.

It includes components such as Django views, forms, and services.

Views receive requests from the presentation layer, interact with the models and services, and prepare data for rendering.

**Model Layer:**

This layer represents the data model and interacts with the database.

It includes components such as Django models and database management systems (e.g., PostgreSQL, MySQL).

Models define the structure and relationships of the data entities, such as students, instructors, slots, and bookings.

**Services Layer:**

This layer encapsulates reusable and domain-specific functionalities.

It includes components such as service classes or modules.

Services handle complex business logic, data manipulation, and integration with external services or APIs.

**Database Layer:**

This layer stores and manages the persistent data.

It includes the database management system (DBMS) and associated database.

The DBMS interacts with the model layer to perform CRUD (Create, Read, Update, Delete) operations on the data.

**External Services and APIs:**

This layer represents any external services or APIs integrated into the project, such as geocoding services or email notification services.

It includes components responsible for interacting with these external services.

## THIRD PARTY LIBRARIES

**Django REST Framework:**

This library is used to build APIs for the LMS project, enabling seamless communication between the front-end and backend.

**jQuery:**

jQuery is a JavaScript library used to simplify DOM manipulation and handle AJAX requests within the project's front-end components.

**Bootstrap:**

Bootstrap is a popular CSS framework used for responsive web design and UI components, making it easier to create visually appealing and mobile-friendly interfaces.

**Pillow**

A powerful library for image processing and manipulation.

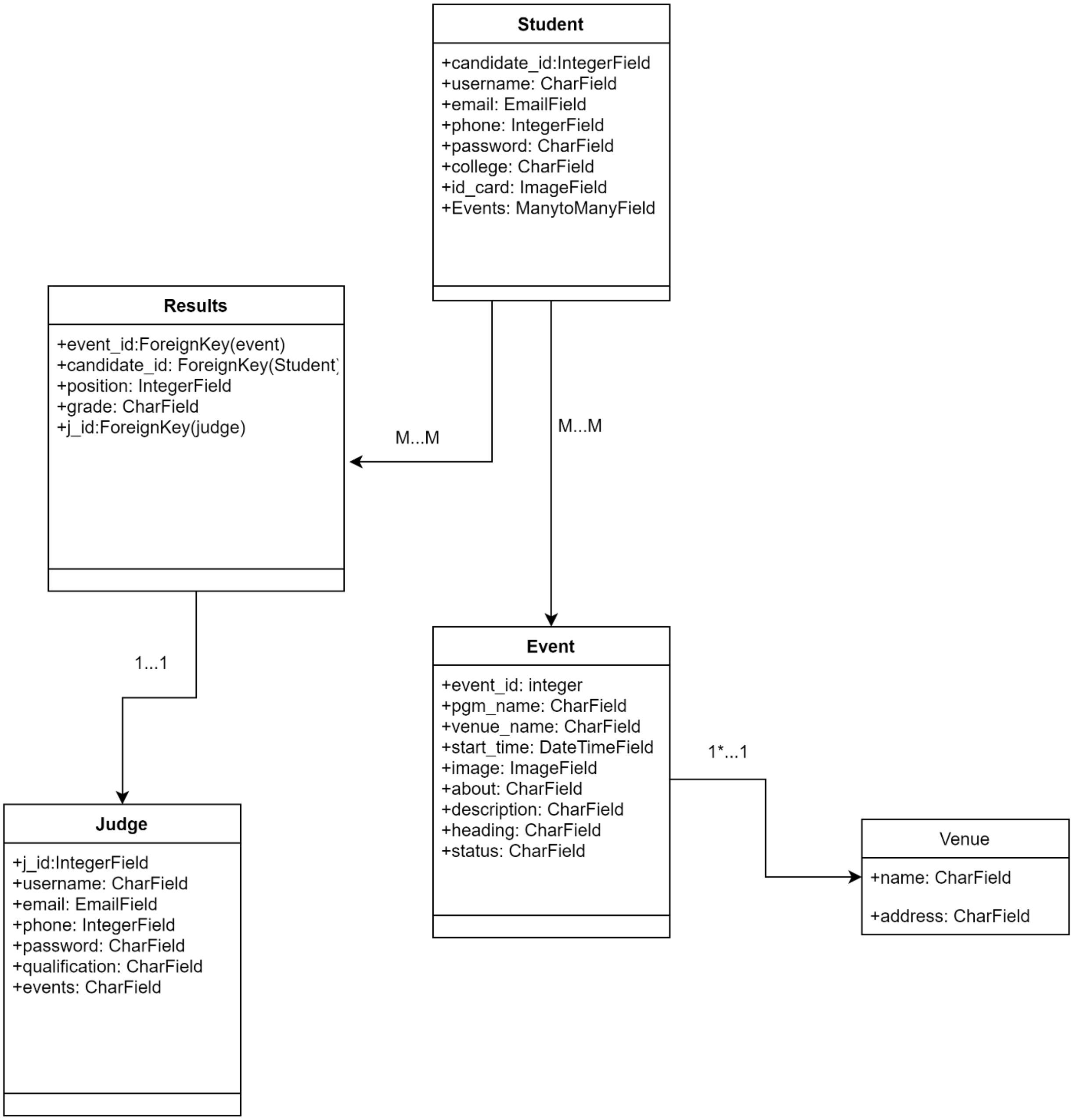
## Libraries

### Jazzmin:

Django Jazzmin is a third-party package that provides a modern and customizable administration interface for Django, a popular Python web framework. Jazzmin aims to enhance the default Django admin interface with improved features, aesthetics, and user experience.

With Django Jazzmin, you can customize the appearance of the admin interface by changing colors, fonts, icons, and other visual elements. It also provides additional functionality, such as draggable model ordering, a user-friendly filter sidebar, and improved search capabilities. The package includes various theme options and allows you to create your own themes as well.

# CLASS DIAGRAM



# CHALLENGES FACED DURING THE DEVELOPMENT

### Profile Updation and Edition:

Providing a user-friendly interface for updating and editing user profiles requires designing intuitive forms, handling data validation, and ensuring proper data persistence.

**Solution:** Use Django's form and model form features to create forms for profile updation and edition. Implement client-side and server-side validation to ensure data integrity. Leverage Django's form handling capabilities to bind form data to model instances and save updates to the database.

### Testing and Validation:

Validating the functionality and ensuring the robustness of the project requires comprehensive testing at different stages of development. It's important to cover edge cases, handle input validation, and perform integration testing.

**Solution:** Write unit tests and integration tests using Django's testing framework. Create test cases that cover different scenarios, including edge cases and invalid inputs. Use tools like Selenium for browser automation to simulate user interactions and test the application's functionality end-to-end.

### Data Modeling:

Designing an efficient database schema to handle user profiles, time slots, and instructor information can be challenging. You need to carefully plan the relationships between different entities and ensure proper indexing and querying for efficient data retrieval.

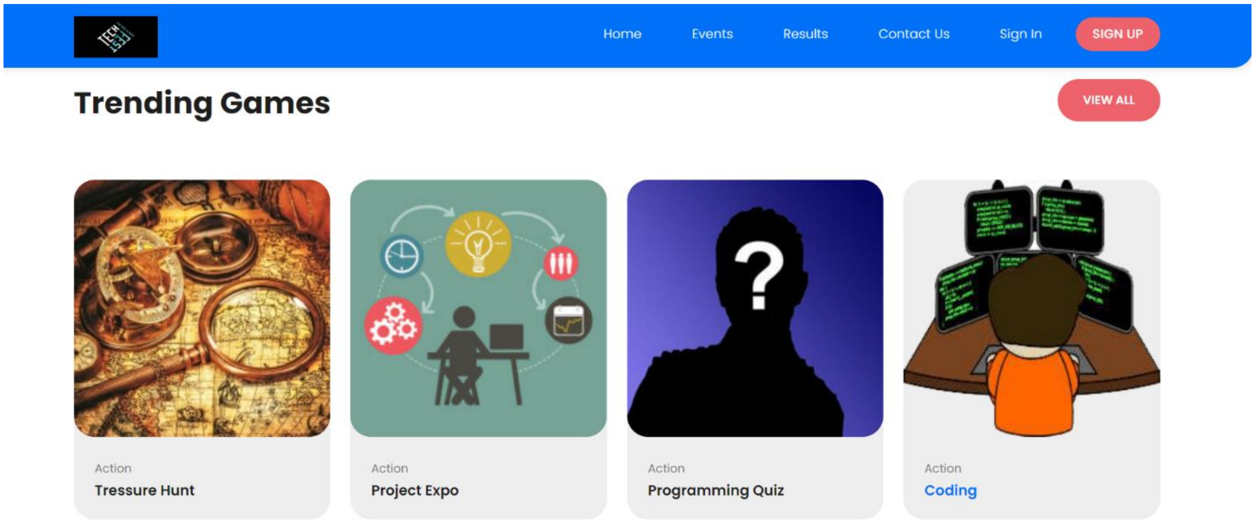
**Solution**: Spend time on data modeling before starting development. Identify the entities, relationships, and attributes needed for the project. Use Django to define models and establish relationships between them. Consider the performance implications of your design choices.

### User Feedback and Iterative Improvrements:

Gathering user feedback and incorporating it into iterative improvements can be a challenge. Ensuring effective communication channels for users to provide feedback and managing the feedback loop to prioritize and address issues requires careful coordination

# IMPORTANT SCREENSHOTS WITH EXPLANATION

# HOME PAGE:



Trending Games

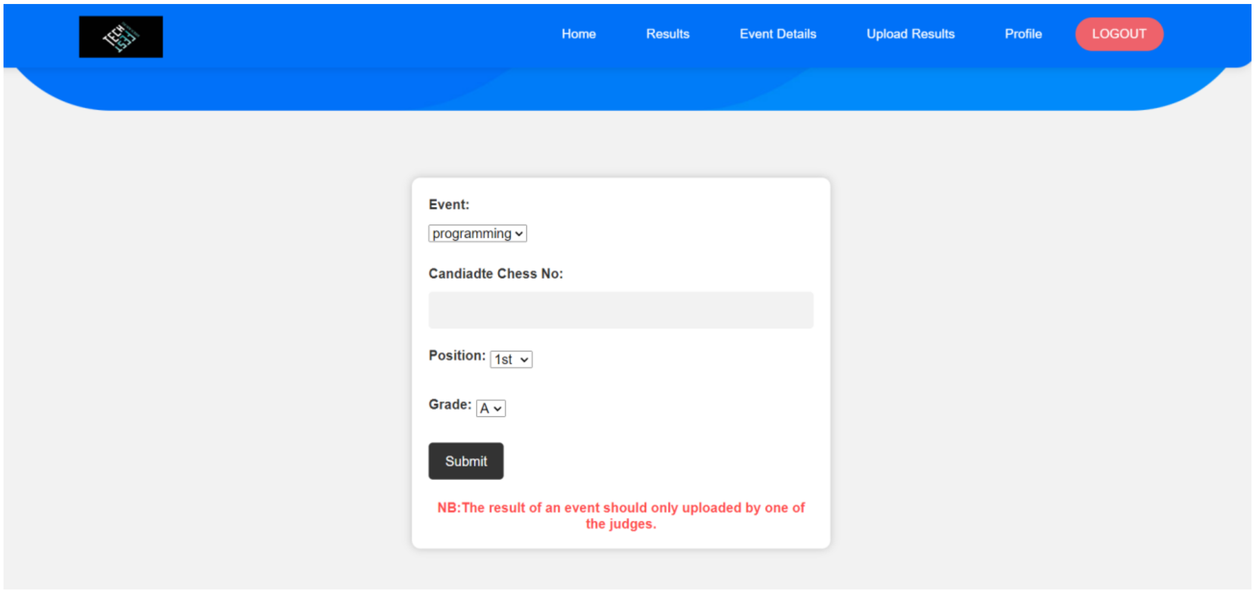
Action
Tressure Hunt

Action
Project Expo

Action
Programming Quiz

Action
Coding

# RESULT UPLOADING PAGE:



Event: programming

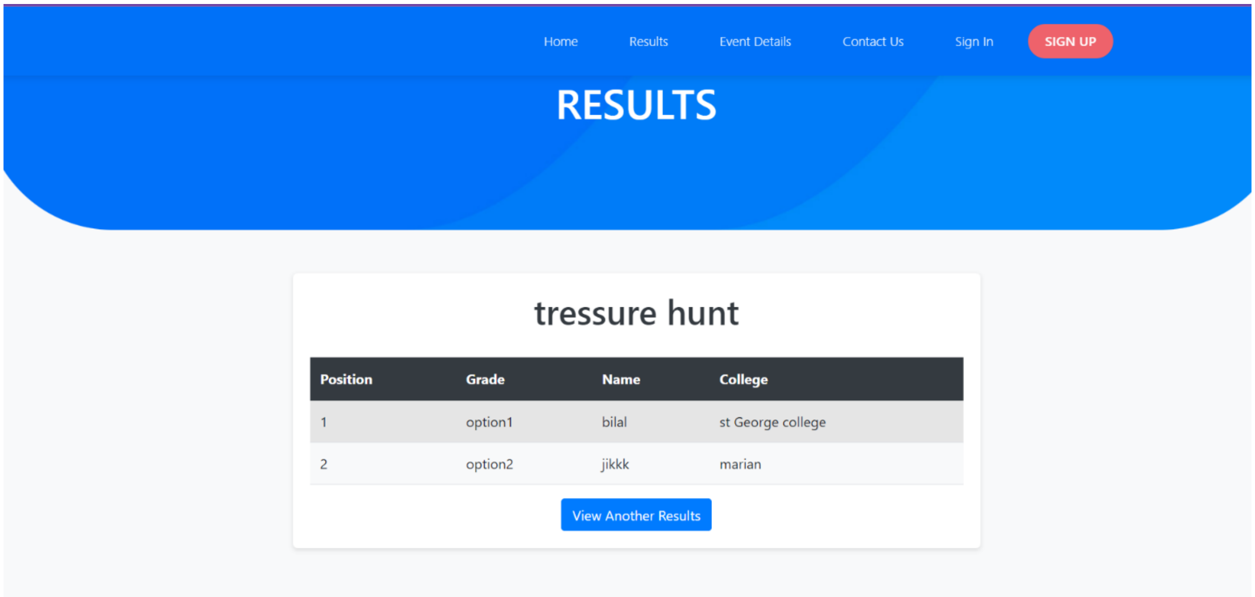
Candiadte Chess No:

Position: 1st

Grade: A

NB:The result of an event should only uploaded by one of the judges.

# RESULT VIEW PAGE

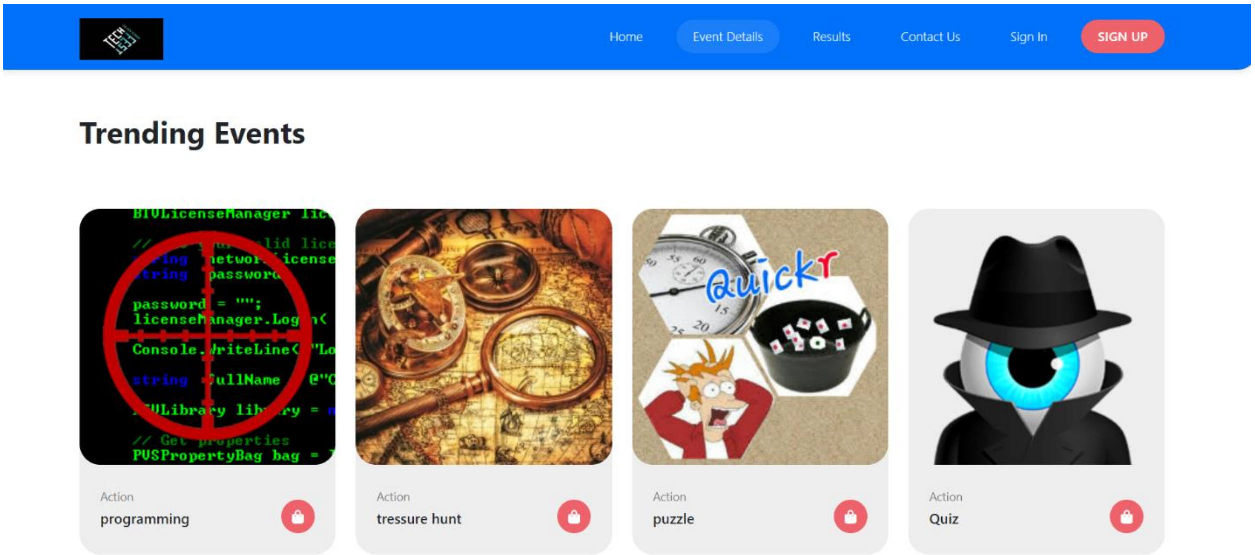


RESULTS

tressure hunt

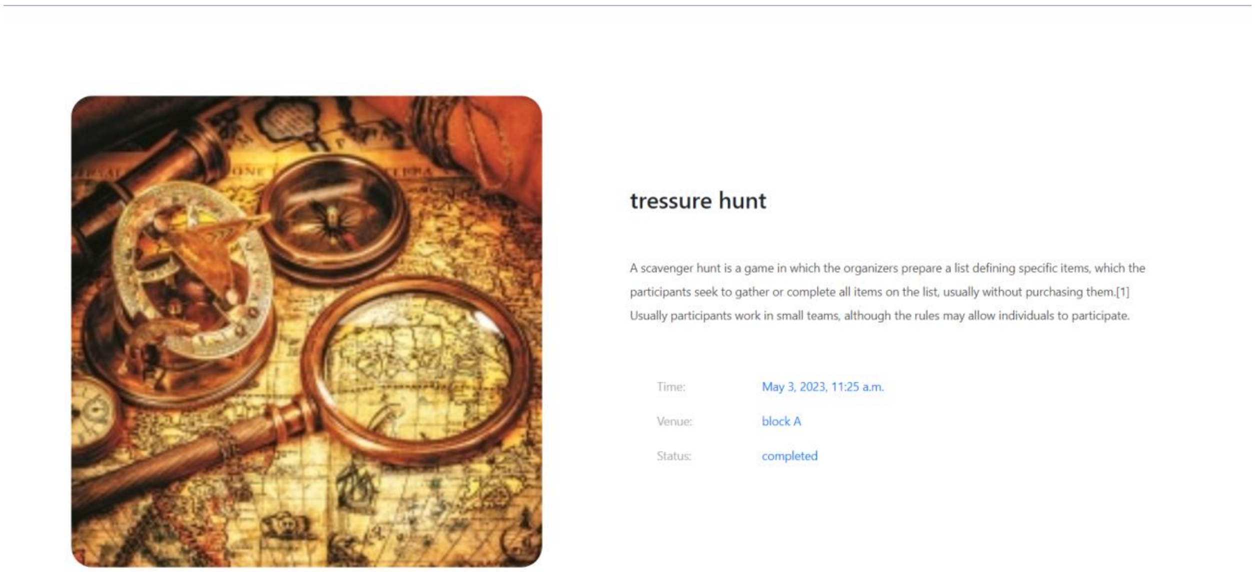
| Position | Grade | Name | College |
|---|---|---|---|
| 1 | option1 | bilal | st George college |
| 2 | option2 | jikkk | marian |

# EVENTS PAGE:



Trending Events

## EVENT DETAILS PAGE:

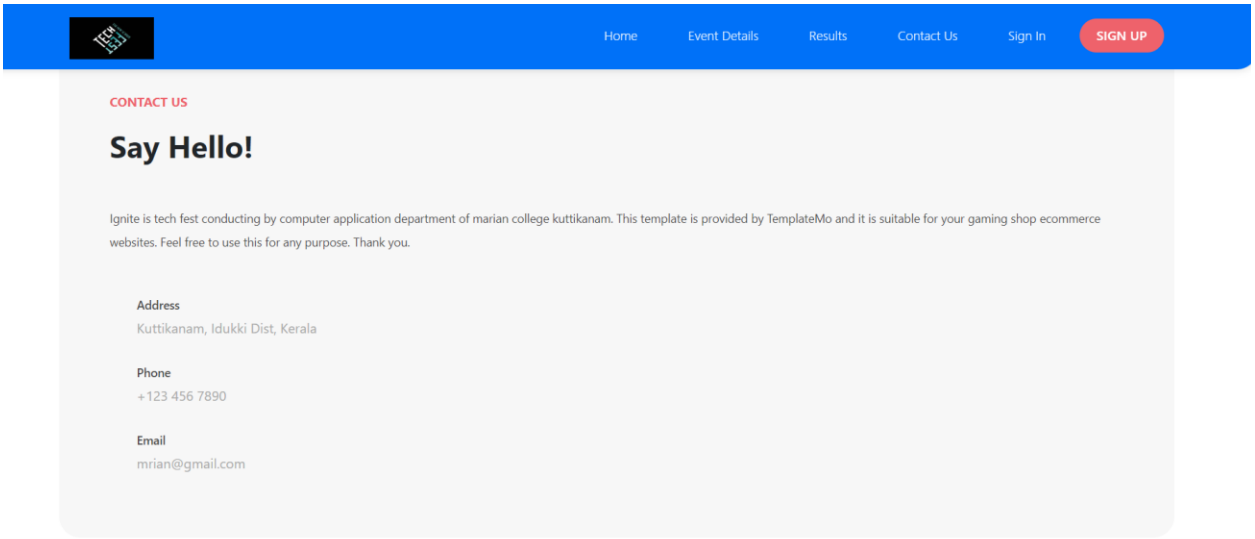


tressure hunt

A scavenger hunt is a game in which the organizers prepare a list defining specific items, which the participants seek to gather or complete all items on the list, usually without purchasing them.[1] Usually participants work in small teams, although the rules may allow individuals to participate.

| | |
|---|---|
| Time: | May 3, 2023, 11:25 a.m. |
| Venue: | block A |
| Status: | completed |

## CONTACT US PAGE:



CONTACT US

Say Hello!

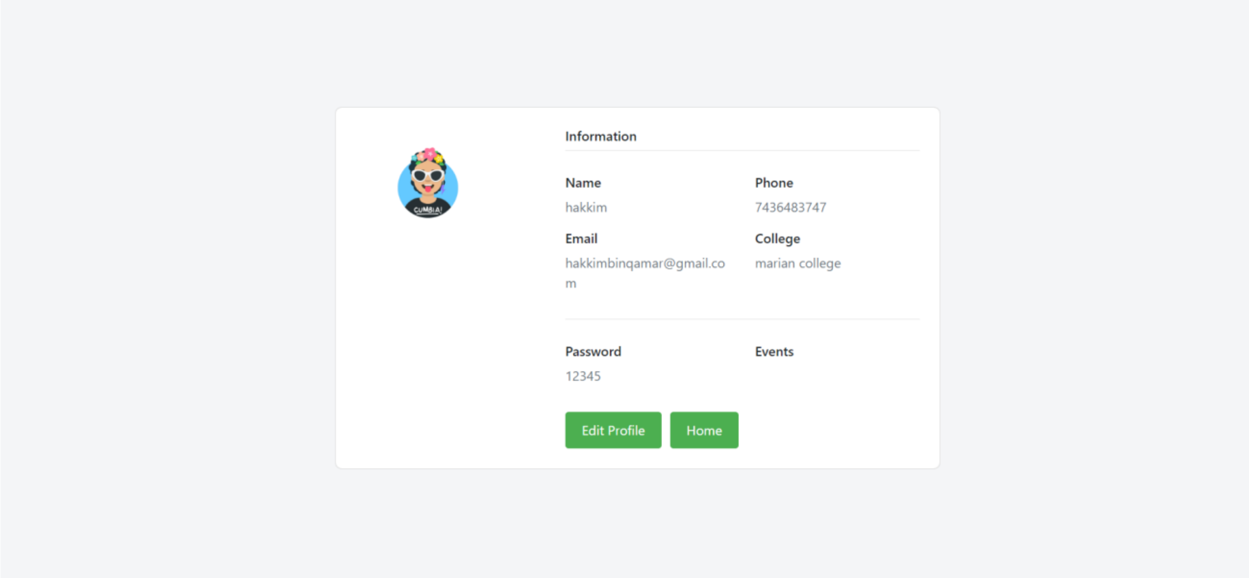
Ignite is tech fest conducting by computer application department of marian college kuttikanam. This template is provided by TemplateMo and it is suitable for your gaming shop ecommerce websites. Feel free to use this for any purpose. Thank you.

Address
Kuttikanam, Idukki Dist, Kerala

Phone
+123 456 7890

Email
mrian@gmail.com

## PROFILE:



Information

Name
hakkim

Phone
7436483747

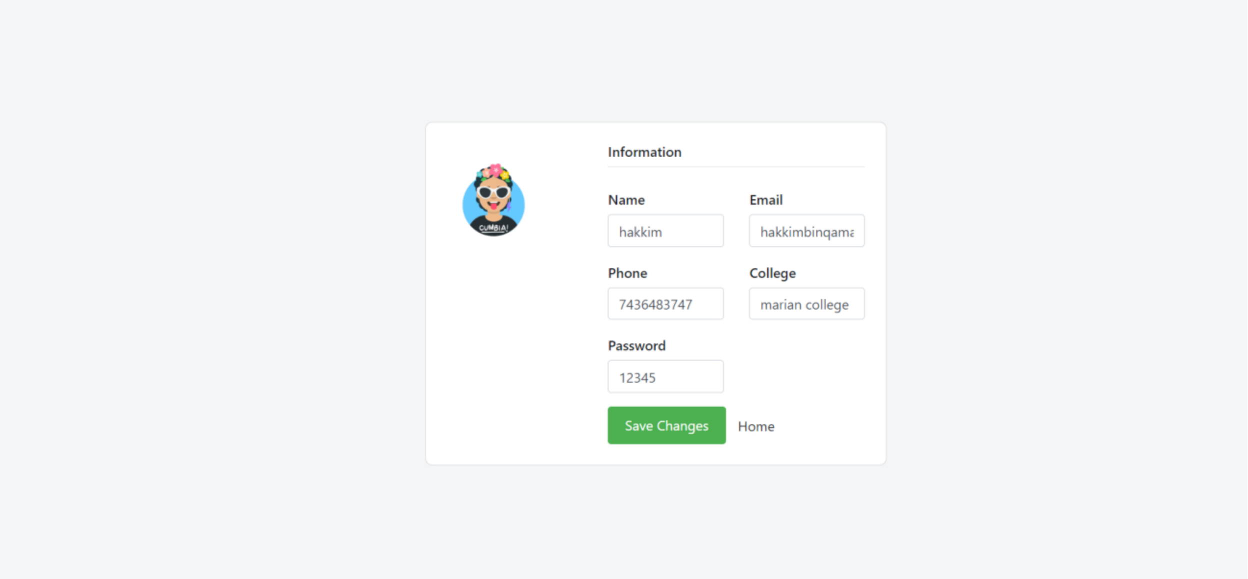
Email
hakkimbinqamar@gmail.com

College
marian college

Password
12345

Events

## STUDENT PROFILE EDTING :



Information

Name
hakkim

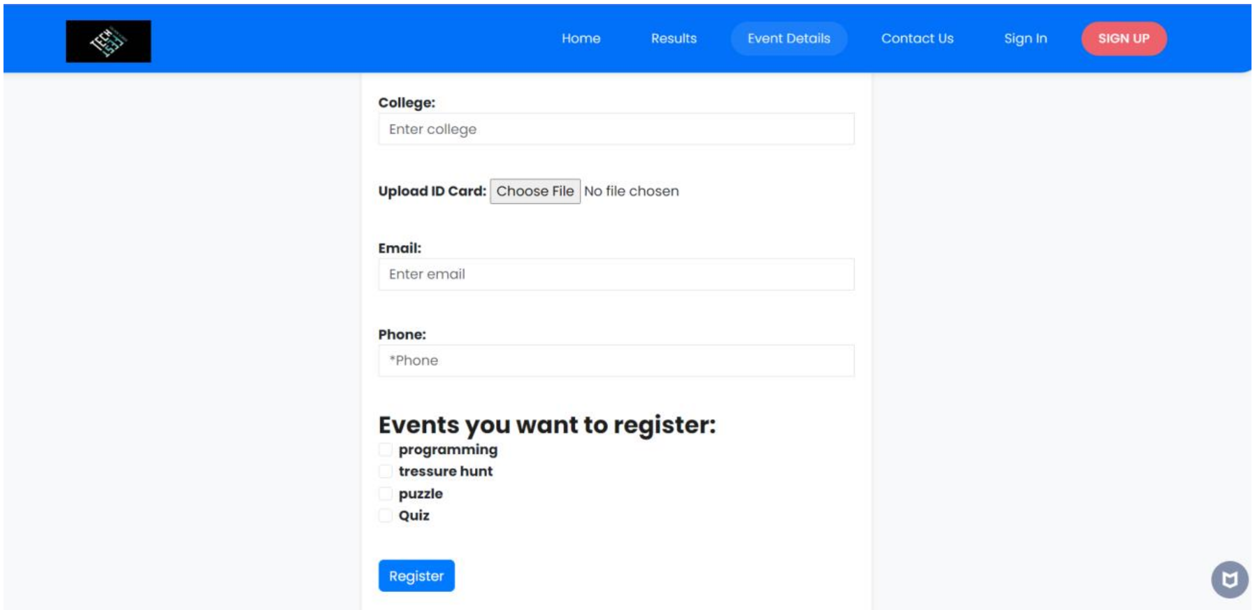
Email
hakkimbinqama

Phone
7436483747

College
marian college

Password
12345

## REGISTRATION FOR STUDENT:



College:

Enter college

Upload ID Card: Choose File No file chosen

Email:

Enter email

Phone:

*Phone

**Events you want to register:**

□ programming

□ tressure hunt

□ puzzle

□ Quiz

Register

# FUTURE ENHANCEMENTS

There are several future enhancements you can consider for your college fest project in Python Django. Here are some suggestions:

1. **Online Payments:**

Integrate a payment gateway to allow users to make online payments for services like license fees, booking time slots, or instructor fees. Implement secure payment processing and provide users with a seamless and convenient payment experience.

2. **Notification System:**

Implement a notification system to keep users informed about important updates, such as appointment confirmations, changes in time slots, or document verification status. Use Django's built-in messaging framework or integrate with email or SMS services to send notifications to users.

3. **Mobile Application:**

Develop a mobile application companion for your project to provide users with a mobile-friendly interface and enhanced accessibility. The app can include features like profile management, time slot booking, instructor search, and push notifications.

4. **Rating and Feedback:**

Allow users to rate and provide feedback on their learning experience with instructors. Implement a rating system and a feedback mechanism to collect user reviews. Display average ratings and reviews to help other users make informed decisions when selecting instructors.

5. **Analytics and Reporting:**

Incorporate analytics and reporting features to track key metrics and generate insights about user activities, popular time slots, instructor availability, and overall system performance. Use data visualization tools to present the information in a visually appealing and understandable manner.

6. **Multilingual Support:**

Add support for multiple languages to cater to a wider user base. Allow users to select their preferred language and provide translations for the user interface, instructional content, and communication messages.

### 7. Gamification Elements:

Incorporate gamification elements into the application to make the learning process more engaging and rewarding. For example, you can introduce achievement badges, progress tracking, or leaderboards to motivate users and make the learning experience enjoyable.

Consider prioritizing these enhancements based on your users' needs, market demand, and available resources. Regularly gather user feedback and conduct usability testing to identify areas for improvement and guide your future enhancements.

# CONCLUSION

In conclusion, the Ignite College Tech Fest Management System is a comprehensive solution designed to streamline the management of college tech fests. It caters to the needs of students, judges, and administrators, offering a range of features and functionalities to enhance the overall tech fest experience.

The system provides a user-friendly interface developed using HTML, CSS, JavaScript, and Bootstrap, ensuring an intuitive and visually appealing design. The backend is powered by Python and Django, enabling efficient data management, authentication, and seamless communication between various system components.

Key features of Ignite include user registration and login, event registration and details, result tracking and upload, profile management, and an admin dashboard for centralized control. The system incorporates security measures such as user authentication and data encryption to protect user information.

Technical aspects of the project encompass frontend technologies, backend technologies (Python, Django, Django Templates, Django ORM), database management systems, authentication and security implementations, API integrations, deployment options, testing, and version control.

Overall, the Ignite College Tech Fest Management System aims to revolutionize the management of college tech fests by simplifying event registration, enhancing information accessibility, automating result tracking, and providing a user-centric experience for all stakeholders. With its robust features and technical foundation, Ignite empowers colleges to organize successful and well-orchestrated tech fests, fostering innovation and collaboration among students.

# REFERENCES

Stack Overflow: http://stackoverflow.com/

Django Jazzmin Documentation:
https://django-jazzmin.readthedocs.io/

ChatGPT

https://djangopackages.org/