A
PROJECT REPORT
ON

Dreamcatchers

(Event Management System)

Carried Out at

CENTRE FOR DEVELOPMENT OF ADVANCED COMPUTING
KNOWLEDGE PARK, BANGALORE

UNDER THE SUPERVISION OF

Mr. Srinivas P. Vasu

PG-DAC, C-DAC Bangalore

Submitted By –

Anjali A Darokar	**(200250120007)**
Ankit Chourey	**(200250120008)**
Santosh Sharma	**(200250120085)**
Shivani Yadav	**(200250120091)**

PG DIPLOMA IN ADVANCED COMPUTING
C-DAC, BANGALORE.

ABSTRACT

1. Dreamcatchers is a web-based event creation and registration system or an event management website, end-users will have the ability to either create or register for events at locations of their choice, with management of the system falling into the Admin's hands. The system will have separate access for end-users and the Dreamcatchers admin.
2. Event management system is an online event management system software project that serves as an interface between the customer and vendor.
3. The system allows registered user login and new users are allowed to register on the application. The system helps in the management of events, users and the aspects related to them. This is proposed to be a web application.
4. Majorly there are two roles- one is the Administrator and the other one is the User. If the user wants to do any business then he/she can be a Vendor and can post their business advertisements.
5. Authentication is done using JWT tokens and if the user or the admin enters a wrong username or password which he/she has not used at the time of registration then he/she will be asked to enter the correct username or password.
6. Errors will be minimized through the use of drop-down buttons and radio buttons which eliminates the excessive use of text input. Management of data includes searching, adding, modifying and deleting.

Software Requirements:

Number	Description	Type
1	Operating System	Windows XP / Windows
2	Frontend	Bootstrap 3, HTML 5, CSS 3 and Angular 10
3	Backend	Java 8 (Spring Boot Framework, Hibernate Framework)
4	Database	MySQL
5	IDE	Visual Studio Code, Eclipse
6	Browser	Google Chrome

7. This website is designed to assist users in finding various vendors that provide different services related to celebrations like wedding, parties etc. and also to assist managers, event organizers, firms, and other users whose line of business deals with events management to manage their participants' data in an orderly manner.
8. Event management is a process of organizing a professional and focused event, for a particular target audience. It involves visualizing concepts, planning, budgeting, organizing and executing events such as wedding, musical concerts, corporate seminars, exhibitions, birthday celebrations, theme parties, etc.

FLOW OF EXECUTION OF PROJECT

We have followed the three tier architecture in our project- service and DAO layer is built as a RESTful web service. So basically, it is independent from the Presentation layer. The relational database is adopted because it is made up of a group of logically connected tables (data that has a relationship to other data).

1. From the client side, after the successful login, when the user clicks on the 'Post' button, she/he will be able to post an advertisement for her/his specific business.

2. After clicking 'Post' button 'onSubmit' function in PostAdvComponent class in post-adv.component.ts file is called.

3. The 'onSubmit' function extracts advertisement details from the form in the HTML file and calls the 'postAdd' function in vendor.service.ts file.

4. Then the userId and Advertise model will be passed to the 'postAdd' function.

5. In vendor.service.ts, the postAdd function will call 'insertAdvertise' function of the Advertisement Controller of Service layer which in turn call the 'addAdvertise' function of the AddService which will further call 'insertAdvertise' function of the Advertise DAO.

6. The 'insertAdvertise' function of the Advertise DAO will return the advertise object, on successfully posting the advertisement, to the addAdvertise function of the AddService which in turn will return this object to 'insertAdvertise' function of the Advertisement Controller. This object is finally returned to the Presentation layer.

7. In the Presentation layer, the alert flag saying "You are ready to provide services to your customers- GOOD LUCK!" will be displayed on the screen.

Some scenarios where we got stuck and how did we overcome

1. In frontend:

1.1. We wanted to send 'roles' string from frontend to the Service Layer (without user selecting or bothering about the role).

Solution:

a. Set the route in frontend with the role (used an array of roles).

b. Read the role from the path and sent it to the end point being called.

1.2. Displaying the list of advertisements along with the photos.

Solution:

a. Saved all the advertisements in an array.

b. Used proper indexing for each element of the array using ngFor directive.

2. In Backend :

2.1. We wanted authorization and authentication in our application.

Solution:

a. After a lot of research, we discovered the concept of JWT then we tried to understand its concepts and successfully implemented it.

2.2. While uploading the photos from the frontend, we did not know how to receive the photos in the backend.

Solution:

a. For this we used a multipart file type to receive the photos from the request body.

Learning during the Project

We learned:

1. To develop a Full Stack application and learned how the Presentation layer interacts with the Service layer.

2. About authorization and how to implement it through JWT.

3. How to send images across the Server and receive them i.e. saving and retrieving images from the database and displaying them in the frontend.

4. How to deal with each other while working in a team and work collaboratively.

5. To use the available online resources efficiently.