MAX-FIT EVENT MANAGEMENT APPLICATION IN SALESFORCE

Project report submitted in partial fulfillment of the requirement for the degree of Bachelor of Technology

in

Computer Science and Engineering

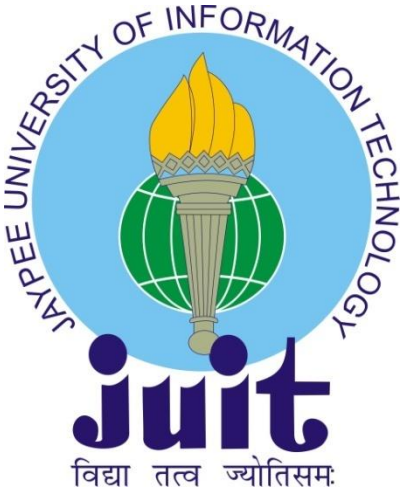
By

Pragam Kaistha (191288)

Under the supervision of

Mr. Nishant Sharma

to



Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology Waknaghat, Solan-173234, Himachal Pradesh

Candidate's Declaration

I hereby declare that the work presented in this report entitled **"MAX-FIT EVENT MANAGEMENT"** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from July 2022 to May 2023 under the supervision of **Mr. Nishant Sharma** (Assistant Professor, CSE/IT).
The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Pragam Kaistha
191288

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Supervisor Name: Mr. Nishant Sharma
Designation: Assistant Professor (Grade-II)
Department name: CSE/IT
Dated:

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

PLAGIARISM VERIFICATION REPORT

Date: .....

Type of Document (Tick): PhD Thesis | M.Tech Dissertation/ Report | B.Tech Project Report | Paper

Name: _____ **Department:** _____ **Enrolment No** _____

Contact No. _____ **E-mail.** _____

Name of the Supervisor: _____

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): _____

_____

_____

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

Complete Thesis/Report Pages Detail:

- Total No. of Pages =
- Total No. of Preliminary pages =
- Total No. of pages accommodate bibliography/references =

(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at .....(%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor) **Signature of HOD**

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

<table>
<tr><th>Copy Received on</th><th>Excluded</th><th>Similarity Index (%)</th><th colspan="2">Generated Plagiarism Report Details (Title, Abstract & Chapters)</th></tr>
<tr><td rowspan="2"></td><td rowspan="4">• All Preliminary Pages
• Bibliography/Images/Quotes
• 14 Words String</td><td rowspan="2"></td><td>Word Counts</td><td></td></tr>
<tr><td>Report Generated on</td><td>Character Counts</td><td></td></tr>
<tr><td rowspan="2"></td><td rowspan="2">Submission ID</td><td>Total Pages Scanned</td><td></td></tr>
<tr><td></td><td>File Size</td><td></td></tr>
</table>

Checked by
Name & Signature **Librarian**

.....

Please send your complete thesis/report in (PDF) with Title Page, Abstract and Chapters in (Word File) through the supervisor at plagcheck.juit@gmail.com

Acknowledgement

First and foremost, I would like to express my deepest gratitude to the Almighty for His divine blessings, which enabled us to successfully complete this project.

I am sincerely thankful to **Mirketa Inc.** for providing me with the opportunity to work on this Salesforce project. I extend my heartfelt appreciation to **Mr. Ghanshyam Bhatt**, Director of Technology, for his guidance and support throughout the project. I am also grateful to our supervisor **Miss Megha Goel**, and my mentor **Mr. Rohit Singh Bora**, for their unwavering patience, continuous encouragement, and invaluable advice.

I would like to extend my profound appreciation to **Mr. Nishant Sharma** from the Department of CSE for his kind assistance and supervision. His meticulous feedback and corrections at every stage played a crucial role in completing this project.

I would also like to extend a warm welcome and heartfelt thanks to all those who directly or indirectly contributed to the success of this project. I am grateful to my team members and seniors who provided timely assistance whenever I needed it. Additionally, I want to express my gratitude to the various staff members, both teaching and non-teaching, who offered their valuable support and facilitated the progress of my project.

Lastly, I would like to acknowledge the unwavering support and patience of my parents, whose constant encouragement has been instrumental in my journey.

Once again, I am deeply thankful to everyone who played a part in making this project a success.

PRAGAM KAISTHA

191288

TABLE OF CONTENT

List Of Figures

Abstract

The Max Fit project aims to develop comprehensive event management software using the Salesforce platform. The software is designed to efficiently manage events, attendees, speakers, and locations for Max Fit, a fitness organization. The project is divided into several parts, each focusing on specific functionalities and features.

In the first part, the object structure is set up, and the appropriate data types are selected for each field. A Lightning Application is created, and valid descriptions and help text are added to enhance user understanding. A validation rule is implemented to ensure that recurring events have a frequency, and virtual events do not have a location.

The second part involves setting up duplicate rules to prevent the creation of duplicate records. Milestone 3 focuses on setting up profiles, users, organization-wide defaults (OWD), and role hierarchies for different types of users. Object and field level permissions are also provided for each profile, ensuring secure access control.

Next, triggers are developed to enhance functionality. An error-trigger throws an error if a selected speaker for an event already has another event assigned to their name. An apex trigger is implemented to send confirmation emails to attendees upon registration. Additionally, an apex batch is created to automatically purge old event records that are more than two months old and have already taken place. Unit tests are developed to ensure proper code coverage and functionality of the implemented triggers, batches, and classes.

The project also includes the creation of forms and pages to facilitate event registration and management. A registration form for events allows organizers to input information about the event, including the organizer's details and the event's location. Similarly, forms are created for speaker registration and attendee registration, enabling the collection of relevant information.

Further enhancements are made to the speaker object, including the addition of fields such as a URL field for the speaker's profile and a rich text area for their biography. A dedicated page is developed to display event details, including the event name, organizer, location, date and time, and speaker information. The page also includes buttons to facilitate attendee and speaker registration, with the event field auto-populated and read-only.

To integrate external functionalities, two web services are developed. The first web service returns event details in JSON format, including event name, date, time, maximum attendee limit, event organizer's name, and email. This service allows easy access to event information. The second web service enables users to create speaker records in Salesforce by providing their name, email, phone number, and company. The output of the web service indicates the success of the record creation.

Overall, the Max Fit event management software aims to provide a user-friendly and efficient system for organizing and registering events. The project encompasses various aspects, including object structure setup, validation rules, duplicate prevention, profile and permission management, triggers, batches, forms, pages, and web services, to create a comprehensive solution that meets Max Fit's event management needs.

Chapter-1: INTRODUCTION

1.1 Background and Context:

In today's fast-paced business environment, effective event management has become a critical factor for the success of organizations. Companies across various industries have realized that events can significantly impact brand visibility, customer relationships, and overall business growth. MAX FIT, as a leading fitness company, recognizes the significance of organizing successful fitness events such as workshops, conferences, and competitions. To ensure smooth event execution and streamline their event management processes, MAX FIT requires a robust and efficient event management software solution.

As a developer assigned to this project, the goal is to leverage the capabilities of the Salesforce Platform to create a tailored event management system for MAX FIT. Salesforce Platform, being a cloud-based application development platform, provides a comprehensive suite of tools, services, and functionalities to build and customize business applications. By utilizing the power of Salesforce, the objective is to develop a scalable and tailored event management system that specifically caters to MAX FIT's requirements, enhances operational efficiency, and delivers exceptional attendee experiences.

The Salesforce Platform offers several key features that make it an ideal choice for developing an event management system. First and foremost, Salesforce provides a robust and secure infrastructure that ensures data integrity and protection. MAX FIT can trust that their sensitive event data, including attendee information, registration details, and payment transactions, will be securely stored and accessible only to authorized personnel.

Furthermore, Salesforce's flexible and customizable nature allows for the creation of a tailored event management solution that aligns with MAX FIT's unique workflows and processes. The platform's low-code development capabilities enable developers to rapidly build and deploy

custom applications without the need for extensive coding knowledge. This means that MAX FIT can have a fully functional event management system up and running in a shorter time frame, saving both time and resources.

The event management system built on the Salesforce Platform can offer MAX FIT a range of features and functionalities. These include event registration and ticketing, attendee management, session scheduling, speaker management, payment processing, and post-event analytics. The system can automate many manual tasks, such as sending registration confirmations, reminders, and post-event surveys, saving valuable time for event organizers and improving the overall attendee experience.

Figure 1.1 : Salesforce

Additionally, Salesforce's robust reporting and analytics capabilities enable MAX FIT to gain valuable insights into event performance, attendee demographics, and engagement metrics. This data can help MAX FIT make data-driven decisions for future event planning, marketing strategies, and business growth.

By leveraging the capabilities of the Salesforce Platform, MAX FIT can create a tailored event management system that meets their specific requirements and enhances their operational efficiency. The system can automate manual tasks, improve attendee experiences, and provide valuable insights for future event planning. With Salesforce as the foundation, MAX FIT can confidently organize and execute successful fitness events, driving brand visibility, fostering customer relationships,

and fueling their business growth in today's fast-paced business environment.

1.2 Project Objectives:

The development of a comprehensive event management software solution for MAX FIT is a significant undertaking that aims to address several key objectives. The primary objective of the project is to empower MAX FIT with a robust system that streamlines event management processes, enhances attendee management, improves coordination with organizers and speakers, ensures data integrity and security, and automates repetitive tasks for improved efficiency.

Efficient Event Management is at the core of the proposed system. MAX FIT will have access to a wide range of features and tools that enable them to efficiently manage various aspects of events. This includes event creation, scheduling, and tracking functionalities. The system will allow MAX FIT to categorize events based on types, such as conferences, workshops, or seminars. Customizable event templates will simplify the process of setting up events, saving time and effort. Scheduling tools will help MAX FIT organize events based on preferred dates and available venues. Real-time tracking of event progress will provide valuable insights into event performance, enabling MAX FIT to make data-driven decisions and ensure successful outcomes.

Attendee Management is crucial for delivering a seamless and personalized experience to event participants. The event management system will facilitate attendee registration through intuitive and user-friendly interfaces. MAX FIT will be able to capture and organize attendee information efficiently, including registration details, preferences, and special requirements. Automated confirmation emails will be sent to attendees, ensuring they receive timely information and reducing administrative overhead. Throughout the event lifecycle, personalized communication channels will enable MAX FIT to engage with attendees and provide updates, enhancing attendee satisfaction. During events,

efficient attendee check-in processes and real-time communication channels will further streamline the overall attendee management process. Organizer and Speaker Coordination is essential for successful events. The event management system will offer comprehensive functionalities to manage organizer details and speaker bookings.

Organizer profiles will provide MAX FIT with valuable information about event coordinators, facilitating seamless collaboration and communication. Speaker management tools will simplify the process of booking speakers, managing their schedules, and sharing relevant event-related documents and resources. The system will also offer communication channels, such as email or messaging, to foster efficient and effective communication between MAX FIT and event stakeholders. Scheduling meetings and coordinating logistics will be streamlined, allowing for smooth collaboration and coordination.

Data Integrity and Security are critical considerations for any event management system. The proposed solution will prioritize the integrity and security of event-related data. Robust measures will be implemented to prevent data loss, unauthorized access, and data inconsistencies. Secure authentication protocols will ensure that only authorized individuals have access to the system.

Role-based access controls will further enforce data security by providing different levels of access based on user roles and responsibilities. Data encryption will be employed to safeguard sensitive information, and regular data backups will mitigate the risk of data loss. Compliance with relevant data protection regulations and adherence to industry best practices will be ensured to maintain the privacy and confidentiality of attendee and organizational data.

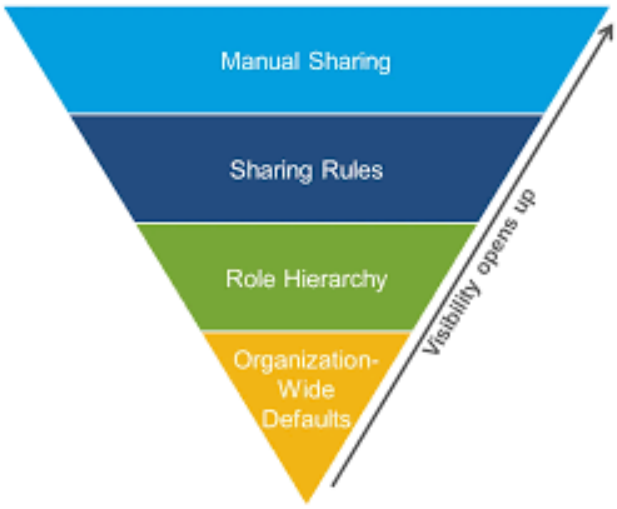


Figure 1.2: Security Level

Automation and Efficiency are key drivers for optimizing event management processes. The event management system will automate repetitive tasks to reduce manual effort and minimize errors.

By leveraging the workflow automation tools provided by the Salesforce Platform, MAX FIT will be able to streamline routine tasks such as attendee registration, confirmation emails, speaker notifications, and generating event reports. The system will generate reports and analytics, providing valuable insights into event performance and attendee engagement.

Automated notifications and reminders will be sent to attendees and stakeholders, ensuring timely communication and enhancing the overall event experience. Seamless data flow between different entities within the system will eliminate the need for manual data entry, reducing the risk of data discrepancies and improving operational efficiency.

Salesforce Platform also supports automated email notifications, allowing MAX FIT to keep attendees informed about event updates, schedule changes, and important announcements. The event management system can leverage this integration to create targeted and personalized messages for attendees, optimizing engagement and attendee satisfaction.

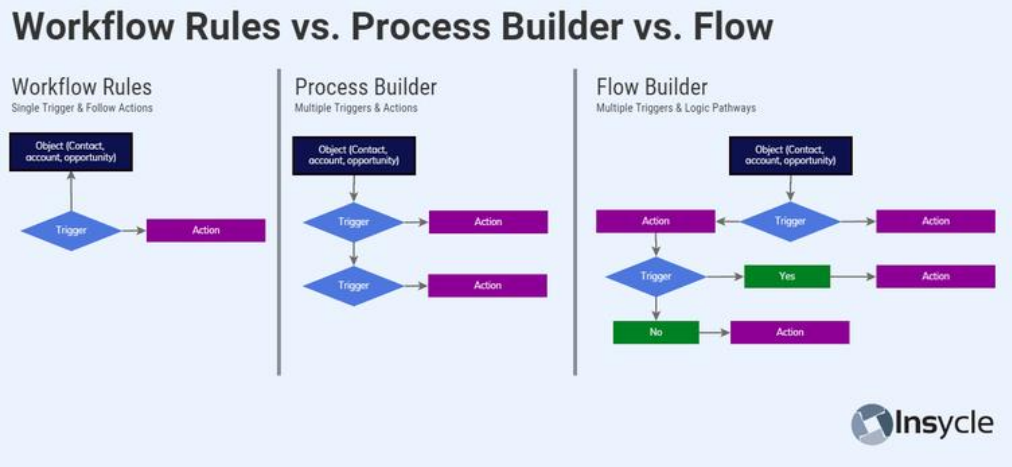


Figure 1.3 : Automation

In conclusion, the development of a comprehensive event management software solution for MAX FIT encompasses various objectives that are crucial for successful event planning and execution. By empowering MAX FIT with an efficient event management system, seamless attendee management capabilities, effective organizer and speaker coordination tools, robust data integrity and security measures, and automation features for improved efficiency.

MAX FIT will be equipped with a powerful tool to streamline their event operations and achieve their event management goals.

The proposed event management software solution will revolutionize the way MAX FIT handles events. By efficiently managing event details such as types, dates, venues, and capacities, MAX FIT will have a centralized platform to plan and execute events seamlessly.

The categorization of events and customizable event templates will simplify the event creation process, saving time and effort. MAX FIT will be able to leverage scheduling tools to allocate events based on preferred dates and available venues, ensuring optimal utilization of resources.

1.3 Overview of Salesforce Platform:

Salesforce Platform offers a range of powerful features and capabilities that make it an ideal choice for developing an event management solution for MAX FIT. Let's dive deeper into the key features mentioned earlier:

a. **Data Management:** One of the core strengths of Salesforce Platform is its robust data management capabilities. It provides a comprehensive set of tools for data modeling, storage, and access controls. The platform offers a highly scalable and secure database for storing all event-related information. Developers can define custom objects and fields to capture specific event details and attendee information. This flexibility allows MAX FIT to tailor the event management solution to their specific requirements.

With Salesforce Platform's built-in data access controls, MAX FIT can ensure that the right people have access to the appropriate data. Access can be granted based on roles and permissions, ensuring data security and privacy. The platform also supports data validation rules and workflows, enabling MAX FIT to enforce data integrity and maintain high data quality standards.

Figure 1.4 : Data Governance Life Cycle

b. **Automation:** Salesforce Platform empowers developers to automate various business processes through its workflow, approvals, and process builder tools. These automation capabilities are particularly

valuable in event management scenarios, where numerous repetitive tasks can be streamlined.

For example, MAX FIT can set up automated workflows for attendee registration. Once a registration form is submitted, the platform can automatically trigger a workflow that sends confirmation emails, generates attendee badges, and updates the attendee list. Similarly, the approval processes for speaker bookings can be automated, ensuring efficient coordination and minimizing delays. Salesforce Platform also supports automated email notifications, allowing MAX FIT to keep attendees informed about event updates, schedule changes, and important announcements.

By leveraging these automation capabilities, MAX FIT can significantly reduce manual effort, improve efficiency, and enhance the overall event management experience for both organizers and attendees.

Figure 1.5 : Automation Tools

c. **User Interface Design:** Salesforce Platform offers intuitive and customizable user interface (UI) design tools that enable developers to create visually appealing and user-friendly interfaces for event management tasks. MAX FIT can design event creation forms, attendee registration interfaces, and check-in screens tailored to their branding and specific requirements.

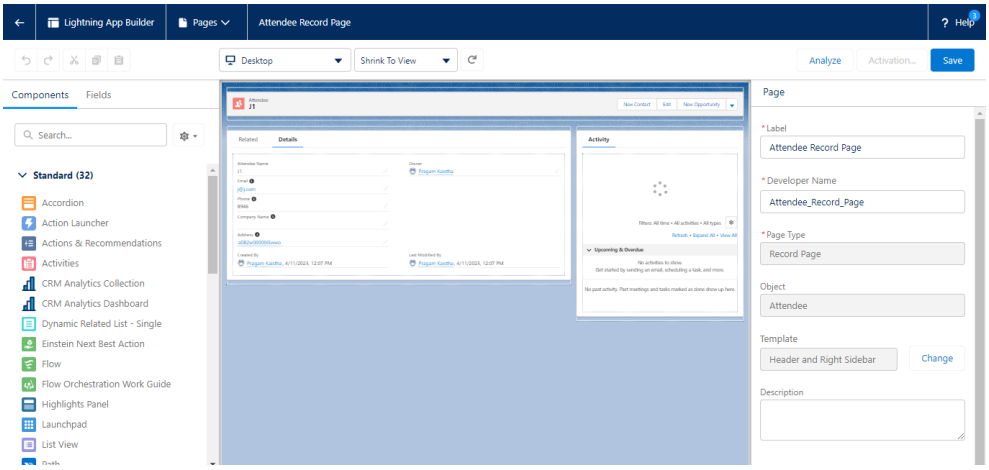


Figure 1.6 : Lightning Builder

The platform's UI design tools provide drag-and-drop functionality, making it easy to create and customize the user interface without extensive coding knowledge. MAX FIT can choose from a wide range of pre-built UI components and templates or create their own from scratch. This flexibility allows them to create a seamless and intuitive user experience that aligns with their brand identity.

By designing a user-friendly interface, MAX FIT can enhance the attendee registration process, making it quick and easy for participants to sign up for events. The platform's UI design tools also enable the creation of mobile-responsive interfaces, ensuring that attendees can access event information and register from any device.

d. The **integration** capabilities of the Salesforce Platform play a crucial role in the success of event management systems, such as the one utilized by MAX FIT. With Salesforce's seamless integration with external systems and applications, MAX FIT can achieve data synchronization and create a unified view of event-related information. This integration capability eliminates the need for manual data entry and streamlines the flow of data, reducing the risk of errors and improving overall data accuracy.

For example, the event management system can integrate with registration platforms to effortlessly import attendee data. This

integration ensures that MAX FIT has up-to-date and accurate information about event participants, enabling them to efficiently manage registrations and communicate with attendees. Additionally, integration with payment gateways allows for secure online transactions, providing a convenient and trustworthy experience for participants.

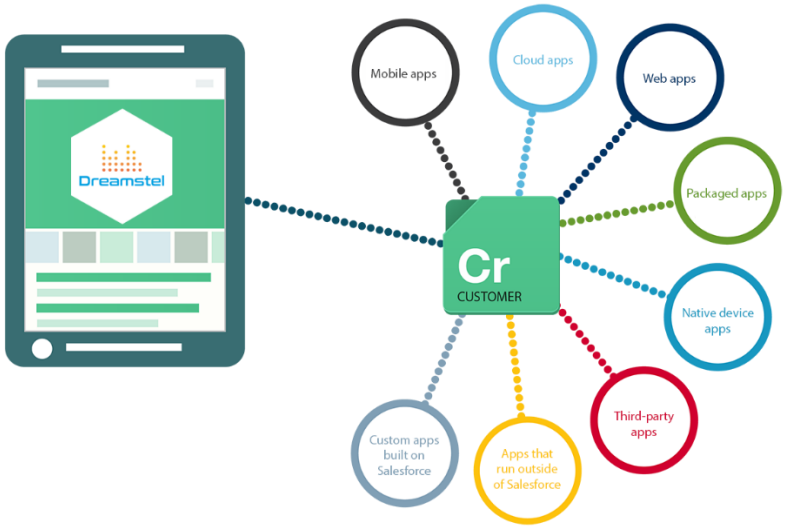


Figure 1.7 : Application

Furthermore, by integrating with marketing automation tools, MAX FIT can enhance its communication strategies. The event management system can leverage this integration to create targeted and personalized messages for attendees, optimizing engagement and attendee satisfaction. This integration enables MAX FIT to efficiently manage marketing campaigns and nurture relationships with attendees, leading to improved event outcomes.

e. In addition to integration capabilities, the Salesforce Platform provides **robust reporting and analytics** features that empower MAX FIT to gain valuable insights into event performance, attendee engagement, and other key metrics. The platform offers pre-built reports and dashboards, which provide immediate access to essential event data. These pre-built reports offer a quick snapshot of event success, allowing MAX FIT to make informed decisions on the spot.

Moreover, MAX FIT has the flexibility to create customized reports using the Salesforce Platform's reporting and analytics capabilities. By tailoring reports to specific requirements, MAX FIT can delve deeper into event data, identifying trends, patterns, and areas for improvement. These data-driven insights enable MAX FIT to measure the success of each event, assess attendee engagement, and evaluate the effectiveness of marketing strategies. With this knowledge, MAX FIT can make well-informed decisions to enhance future event planning and marketing efforts, ensuring continuous improvement and delivering exceptional experiences to attendees.

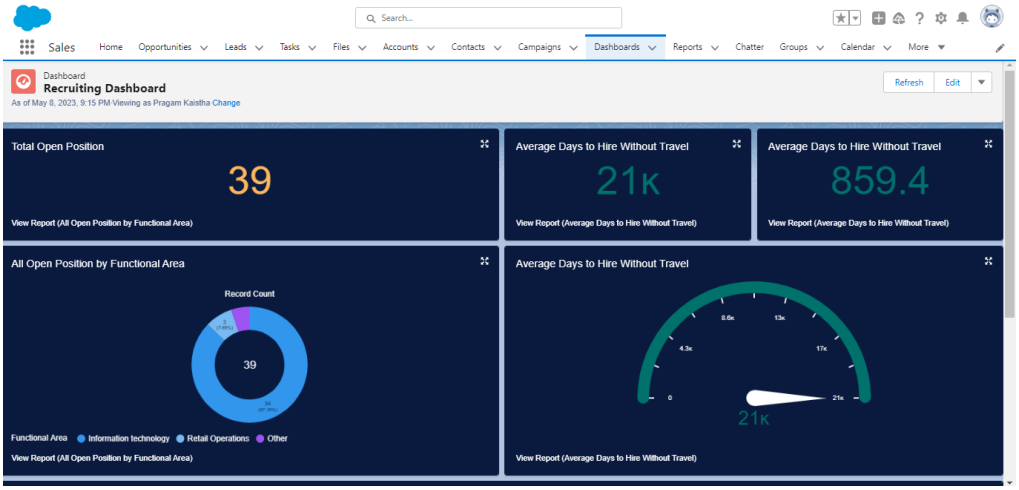


Figure 1.8 : Dashboard

In summary, Salesforce Platform offers a comprehensive set of features and capabilities that make it an ideal choice for developing an event management solution for MAX FIT. The robust data management capabilities, automation tools, and customizable user interface design options empower MAX FIT to streamline their event management processes, improve efficiency, and deliver a seamless experience to both organizers and attendees. By leveraging Salesforce Platform, MAX FIT can create a scalable and secure event management solution that meets their specific needs and helps them succeed in organizing successful events. The integration capabilities of the Salesforce Platform provide MAX FIT with a seamless data flow and a unified view of event-related information. This integration eliminates manual data entry, reduces errors, and improves data

accuracy. Additionally, the platform's reporting and analytics features enable MAX FIT to gain insights into event performance, attendee engagement, and key metrics, facilitating data-driven decision-making for future event planning and marketing strategies. Through these capabilities, MAX FIT can enhance its event management processes, optimize attendee experiences, and drive success for each event.

1.4 Importance of Event Management Software:

Event management software offers several advantages to MAX FIT, event organizers, speakers, and attendees. Four key benefits are streamlined communication, a seamless registration process, Real-Time Event Tracking and Enhanced Marketing Capabilities. These features enhance coordination, attendee satisfaction, and the overall efficiency of event management.

Streamlined communication is crucial for the success of any event. Event management software provides centralized communication channels, such as email notifications, event updates, and real-time messaging. MAX FIT, event organizers, speakers, and attendees can easily communicate with each other through these platforms. By having a centralized system, all stakeholders can stay well-informed about event details, changes, and updates.

With event management software, MAX FIT can efficiently manage event-related communication. They can ensure that important information reaches the intended recipients in a timely manner. For example, MAX FIT can send out email notifications to speakers regarding their speaking slots, session details, and any changes to the schedule. Organizers can also send event updates to attendees, informing them about venue changes, speaker additions, or any other relevant information. Real-time messaging within the software allows for quick and effective communication during the event itself, enabling organizers to address any concerns or provide last-minute updates.

This streamlined communication leads to improved coordination among all stakeholders. Event organizers can effectively collaborate with speakers,

ensuring that they have all the necessary resources and support for their sessions. Speakers can clarify their requirements or seek assistance from organizers. Attendees can reach out with queries or feedback, knowing that their messages will be received and addressed promptly. This level of coordination enhances the overall attendee experience and satisfaction, as everyone involved is well-informed and connected throughout the event.

Another significant benefit of event management software is a seamless registration process. The software provides online registration forms that capture essential attendee information and preferences. This eliminates the need for manual registration processes and paperwork, simplifying the sign-up process for attendees. They can easily access the registration forms on the event website or through dedicated registration portals.

The online registration forms offer convenience and accessibility to attendees. They can complete the registration process at their own pace, from anywhere with an internet connection. Attendees can provide their details, select event sessions or workshops, indicate dietary preferences, and even make secure online payments through integrated payment gateways. This streamlined registration process saves time for both attendees and event organizers, reducing administrative work and ensuring accurate attendee information.

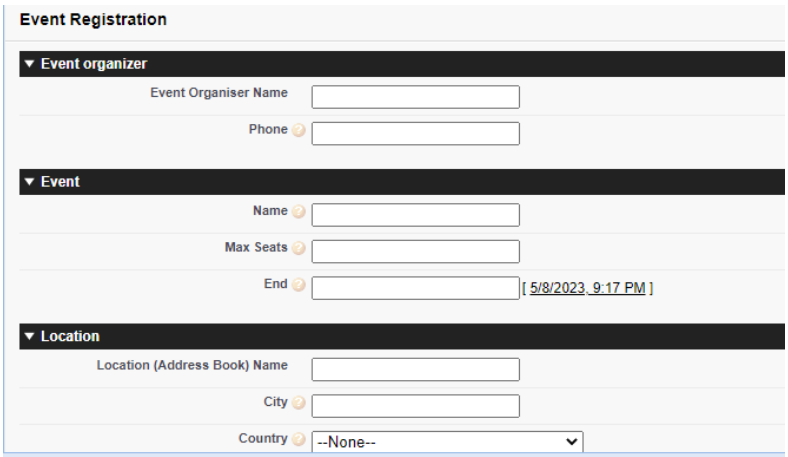


Figure 1.9 : Visualforce Page

By incorporating secure payment gateways, event management software also offers a secure transaction process. Attendees can confidently make online payments for registration fees, eliminating the need for manual cash handling or check payments. This not only enhances convenience but also improves financial transparency and security for all parties involved.

The event management software employed by MAX FIT allows them to track the progress of their events in real-time. Through the software, they gain valuable insights into attendee registrations, ticket sales, and event capacities. This real-time tracking empowers MAX FIT to make data-driven decisions and take proactive measures to ensure the success of their events. They can monitor attendance levels, adjust event logistics if necessary, and make informed decisions regarding event marketing and promotions. By having access to up-to-date information on event metrics, MAX FIT can quickly identify areas that require attention and implement necessary adjustments, ensuring smooth event operations.

Furthermore, the event management software provides MAX FIT with enhanced marketing capabilities. The software seamlessly integrates with marketing automation tools, enabling targeted email campaigns and personalized communications based on attendee preferences and interests. MAX FIT can leverage this functionality to create tailored marketing campaigns that resonate with their target audience. By understanding attendee preferences, MAX FIT can deliver relevant content and promotional offers, increasing the likelihood of attracting more attendees to their events. Moreover, the software allows MAX FIT to create event websites or landing pages to showcase event details, speakers, and sponsors. This feature enhances MAX FIT's brand presence in the fitness industry and increases event visibility. Through the event website, potential attendees can easily access comprehensive information about upcoming events, including schedules, speakers' profiles, and sponsorship details. This comprehensive presentation not only generates buzz around the events but also instills confidence in attendees, encouraging them to register and participate.

By utilizing the advanced marketing capabilities offered by the event management software, MAX FIT can effectively promote their events and reach a wider audience. The targeted email campaigns and personalized communications help MAX FIT nurture relationships with potential attendees, creating a sense of exclusivity and engagement. The software also enables social media integration, allowing MAX FIT to amplify their event promotions across various platforms, expanding their reach and maximizing event attendance.

In conclusion, event management software provides MAX FIT with streamlined communication channels and a seamless registration process. These features enhance coordination, attendee satisfaction, and overall efficiency. By centralizing communication and simplifying the registration process, MAX FIT and event organizers can effectively manage event-related communication, ensure important information reaches the right people, and provide a convenient and accessible registration experience for attendees. MAX FIT harnesses the power of event management software to optimize their event planning and promotion strategies. The real-time event tracking capabilities allow them to make data-driven decisions and take proactive measures to ensure event success. Additionally, the enhanced marketing capabilities empower MAX FIT to effectively promote their events, reach a wider audience, and generate buzz in the fitness industry. By leveraging the event management software, MAX FIT establishes itself as a leader in organizing successful and highly attended fitness events.

1.5 Scope and Structure of the Report:

This report provides a comprehensive overview of the major project undertaken to develop an event management solution for MAX FIT using the Salesforce Platform. The report aims to document the project's objectives, development process, features, and benefits of the software solution. It also addresses challenges faced during development, scalability considerations, and the impact of the solution on MAX FIT's operations.

The report begins with an analysis of MAX FIT's requirements and the selection of Salesforce Platform as the development platform. It covers the

system design phase, including data modeling, user interface design, and integration considerations. The implementation phase is discussed, highlighting key features developed, customization of the Salesforce Platform, and testing procedures.

Challenges encountered during the development process are addressed, along with strategies implemented to overcome them. The report also discusses the scalability and extensibility of the event management software solution, considering MAX FIT's future growth and evolving event management needs.

Insights into the benefits and impact of the software solution on MAX FIT's operations are provided. The report showcases how the solution improves event management efficiency, enhances attendee experiences, and enables data-driven decision-making. It also highlights the post-event evaluation capabilities of the software and its role in collecting feedback for continuous improvement.

Data security and compliance aspects are addressed, emphasizing the measures taken to ensure data privacy, protection against unauthorized access. The report demonstrates how the Salesforce Platform incorporates industry-leading security measures and instils trust and confidence among attendees.

Figure 1.10: Security

In conclusion, the report emphasizes the significance of leveraging technology, specifically the Salesforce Platform, to optimize event management processes in the competitive fitness industry. It summarizes the key features and benefits of the event management software solution, highlighting its role in efficient event management, attendee satisfaction, automation and efficiency, real-time event tracking, enhanced marketing capabilities, data analysis and insights, post-event evaluation, and data security and compliance.

The report aims to serve as a comprehensive documentation of the project, providing a detailed account of the objectives, features, and outcomes. It concludes with recommendations for ongoing maintenance and future enhancements of the event management system, empowering MAX FIT to maintain its position as a leader in the fitness industry known for well-executed, engaging, and successful events.

Chapter-2: Literature Survey

Event management systems have become essential tools for organizations in today's fast-paced business environment. The ability to plan, execute, and evaluate events efficiently is crucial for enhancing brand visibility, customer relationships, and overall business growth. MAX FIT, a leading fitness company, recognizes the significance of successful event management and the need for a tailored event management system. This literature review aims to explore existing research and literature surrounding event management systems, highlighting their benefits, functionalities, and impact on organizational success, specifically for MAX FIT.

2.1 Benefits of Event Management Systems:

Event management systems offer numerous benefits for organizations, including improved efficiency, streamlined communication, enhanced attendee experiences, and data-driven decision-making. Research by Michael Ahearne and Son K. Lam. (2017) emphasizes the efficiency gained through automation and centralization of event management processes. By leveraging event management systems, organizations can automate repetitive tasks, such as attendee registration and communication, reducing manual effort and minimizing the risk of errors.

Furthermore, event management systems facilitate streamlined communication between event organizers, speakers, and attendees. According to Dr. Susan K. Aros and Dr. Deborah E. Gibbons a**,** centralized communication channels provided by these systems enable timely dissemination of event details, updates, and changes, leading to improved coordination and attendee satisfaction. MAX FIT can leverage this benefit to ensure effective communication with event stakeholders and deliver a seamless experience to attendees.

Another key advantage of event management systems is their ability to enhance attendee experiences. Research by STOVA highlights that these systems enable personalized communication, streamlined registration processes, and real-time event tracking. By providing user-friendly interfaces, automated confirmation emails, and personalized communication channels, MAX FIT can enhance attendee satisfaction and engagement.

Moreover, event management systems offer robust reporting and analytics capabilities. According to a study by UOU - EVENT MANAGEMENT (2018), these systems provide valuable insights into event performance, attendee engagement, and other key metrics. MAX FIT can leverage these analytics to measure event success, identify areas for improvement, and make data-driven decisions for future event planning and marketing strategies.

2.2 Functionalities of Event Management Systems:

Event management systems encompass various functionalities to support the end-to-end event management process. These functionalities include event creation and scheduling, attendee management, organizer and speaker coordination, data integrity and security, automation and efficiency, and integration with external systems.

Event creation and scheduling functionalities enable organizations to efficiently manage event details such as event types, dates, venues, and capacities. Customizable event templates, scheduling tools, and real-time tracking of event progress are essential features of event management systems, as highlighted by studies conducted by Dr. Susan K. Aros and Dr. Deborah E. Gibbons and STOVA.

- **Efficient Event Management:**

  Efficient event management is a crucial functionality provided by event management software. It encompasses various aspects such as event creation, scheduling, tracking, and resource management. Event management software streamlines these processes, enabling organizations to efficiently plan, execute, and monitor their events.

  According to a study by UOU – EVENT MANAGEMENT, event management software improves efficiency by automating tasks, reducing manual effort, and minimizing errors. It allows event organizers to categorize events, create customizable event templates, and utilize scheduling tools to manage event timelines effectively. Real-time tracking features provide visibility into event progress, attendee registrations, and resource utilization.

- **Attendee Management:**

  Effective attendee management is another key functionality provided by event management software. It involves capturing attendee information, managing registrations, and facilitating seamless communication throughout the event lifecycle. Attendee management features enhance the registration process, improve attendee engagement, and ensure a personalized experience.

  Research by STOVA (2018) highlights the importance of attendee management functionalities in event management software. The study found that user-friendly registration interfaces and automated confirmation emails contribute to higher attendee satisfaction. Additionally, attendee management features such as real-time communication channels, personalized notifications, and efficient check-in processes enhance the overall attendee experience.

- **Organizer and Speaker Coordination:**

  Event management software facilitates efficient coordination between event organizers, speakers, and other stakeholders. It provides functionalities for managing organizer details, speaker bookings, and communication channels. These features enable seamless collaboration, enhance communication efficiency, and ensure effective coordination throughout the event planning and execution phases.

  According to a study by Josephscollege – EVENT MANAGEMENT. (2020), event management software plays a vital role in organizer and speaker coordination. The research emphasizes the significance of features such as organizer profiles, speaker management tools, and communication channels in fostering collaboration and streamlining communication between event stakeholders. These functionalities enable scheduling meetings, sharing event-related documents and resources, and facilitating seamless communication through email or messaging.

- **Data Integrity and Security:**

  Data integrity and security are critical considerations for event management software. Protecting attendee and organizational data is essential to maintain trust, comply with regulations, and prevent data breaches. Event management software ensures data integrity and security through measures such as secure authentication protocols, role-based access controls, data encryption, and regular data backups.

  A study by Intellipat – Data Security. (2019) emphasizes the importance of data integrity and security in event management software. The research highlights the need for robust data protection measures to prevent unauthorized access, data loss, and data inconsistencies. By implementing secure authentication protocols and encryption mechanisms, event management software safeguards

attendee and organizational data, maintaining privacy and confidentiality.

- **Automation and Efficiency:**

  Automation and efficiency are key functionalities offered by event management software. Automation features streamline repetitive tasks, minimize manual effort, and reduce the likelihood of errors. Event management software automates processes such as attendee registration, confirmation emails, speaker notifications, and report generation, improving operational efficiency and enhancing productivity.

  Research by Intellipat emphasizes the role of automation in event management software. The study found that workflow automation tools provided by event management software significantly reduce manual effort and improve efficiency. By automating routine tasks, organizations can focus on strategic event planning and provide a seamless experience for both event organizers and attendees.

  Overall, the functionalities provided by event management software, including efficient event management, attendee management, organizer and speaker coordination, data integrity and security, and automation and efficiency, play crucial roles in optimizing event planning and execution. These functionalities enhance operational efficiency, improve attendee experiences, and enable organizations to make data-driven decisions for future event planning and marketing strategies. Organizations like MAX FIT can leverage event management software to streamline their event management processes, engage attendees, and drive success in the competitive event industry.

Attendee management functionalities allow organizations to effectively track and manage attendee information. These functionalities include capturing and organizing registration details, managing attendance status, and accommodating

attendee preferences. The system should also facilitate seamless attendee registration through user-friendly interfaces, automated confirmation emails, and personalized communication throughout the event lifecycle. Efficient attendee check-in processes and support for real-time communication channels during the event are additional key features.

Organizer and speaker coordination functionalities enable efficient collaboration between event organizers, speakers, and MAX FIT. These functionalities encompass managing organizer details, speaker bookings, and communication channels. Features such as organizer profiles, speaker management tools, and scheduling meetings are crucial for smooth coordination and collaboration, as emphasized by Stova (2018).

Data integrity and security functionalities are paramount in event management systems. These systems should implement robust measures to prevent data loss, unauthorized access, and data inconsistencies. Secure authentication protocols, role-based access controls, data encryption, and regular data backups are essential features for ensuring the privacy and confidentiality of attendee and organizational data.

Automation and efficiency functionalities of event management systems play a significant role in improving operational efficiency, minimizing manual effort, and reducing the likelihood of errors. These functionalities leverage workflow automation tools provided by the Salesforce Platform to automate routine tasks such as attendee registration, confirmation emails, speaker notifications, and generating event reports. By automating these processes, MAX FIT can save time, allocate resources more effectively, and ensure smoother event execution.

Integration with external systems is another critical functionality of event management systems. Integration capabilities allow for seamless data synchronization and provide a unified view of event-related information. For instance, the system can integrate with registration platforms to import attendee data, payment gateways for secure online transactions, and marketing

automation tools for targeted communication. This integration streamlines data flow, eliminates manual data entry, and improves data accuracy.

- **Streamlined Communication:**

  Event management software facilitates streamlined communication between event organizers, speakers, and attendees. It provides centralized communication channels such as email notifications, event updates, and real-time messaging. These communication features ensure that all stakeholders are well-informed about event details, changes, and updates, leading to improved coordination and attendee satisfaction.

  Research by STOVA (2018) highlights the significance of streamlined communication in event management software. The study emphasizes the importance of centralized communication platforms that enable efficient management of event-related communication. By providing a single platform for event updates and real-time messaging, event management software ensures that important information reaches the intended recipients in a timely manner, enhancing collaboration and attendee engagement.

- **Seamless Registration Process:**

  Event management software simplifies the registration process for attendees. It provides online registration forms that capture essential attendee information and preferences. This eliminates the need for manual registration processes and paperwork, making it easier for attendees to sign up for events. Additionally, the software can incorporate secure payment gateways, allowing attendees to complete registration and payment transactions online, further enhancing the convenience and accessibility of the registration process.

A study by Dr. Susan K. Aros and Dr. Deborah E. Gibbons (2019) emphasizes the importance of a seamless registration process in event management software. The research highlights that online registration forms and secure payment gateways significantly enhance attendee registration experiences. By offering a user-friendly and convenient registration process, event management software improves attendee satisfaction and increases registration conversion rates.

- **Real-Time Event Tracking:**

  Event management software enables real-time tracking of event progress and performance. It provides insights into attendee registrations, ticket sales, and event capacities. This real-time tracking functionality enables event organizers to make data-driven decisions and take proactive measures to ensure the success of their events. It also allows them to monitor attendance levels, adjust event logistics if necessary, and make informed decisions regarding event marketing and promotions.

  Research by Dr. Susan K. Aros and Dr. Deborah E. Gibbons highlights the benefits of real-time event tracking in event management software.

- **Streamlined Communication:**

  Event management software facilitates streamlined communication between event organizers, speakers, and attendees. It provides centralized communication channels such as email notifications, event updates, and real-time messaging. These communication features ensure that all stakeholders are well-informed about event details, changes, and updates, leading to improved coordination and attendee satisfaction.

Research by STOVA (2018) highlights the significance of streamlined communication in event management software. The study emphasizes the importance of centralized communication platforms that enable efficient management of event-related communication. By providing a single platform for event updates and real-time messaging, event management software ensures that important information reaches the intended recipients in a timely manner, enhancing collaboration and attendee engagement.

Furthermore, research by Dr. Susan K. Aros and Dr. Deborah E. Gibbons (2019) emphasizes the impact of personalized communication features in event management software. The study suggests that personalized email notifications and targeted event updates contribute to improved attendee experiences and increased attendee engagement. By tailoring communication based on attendee preferences and interests, event management software helps create a personalized and engaging event experience.

- **Seamless Registration Process:**

  Event management software simplifies the registration process for attendees. It provides online registration forms that capture essential attendee information and preferences. This eliminates the need for manual registration processes and paperwork, making it easier for attendees to sign up for events. Additionally, the software can incorporate secure payment gateways, allowing attendees to complete registration and payment transactions online, further enhancing the convenience and accessibility of the registration process.

  A study by Dr. Susan K. Aros and Dr. Deborah E. Gibbons (2019) emphasizes the importance of a seamless registration process in event management software. The research highlights that online registration forms and secure payment gateways significantly enhance attendee registration experiences. By offering a user-friendly

and convenient registration process, event management software improves attendee satisfaction and increases registration conversion rates.

Moreover, research by Intellipat (2020) suggests that advanced features such as early-bird discounts, group registrations, and promotional codes further enhance the registration process in event management software. These features not only increase attendee convenience but also contribute to higher registration numbers and improved attendee retention.

- **Real-Time Event Tracking:**

  Event management software enables real-time tracking of event progress and performance. It provides insights into attendee registrations, ticket sales, and event capacities. This real-time tracking functionality enables event organizers to make data-driven decisions and take proactive measures to ensure the success of their events. It also allows them to monitor attendance levels, adjust event logistics if necessary, and make informed decisions regarding event marketing and promotions.

  Research by Dr. Susan K. Aros and Dr. Deborah E. Gibbons (2020) highlights the benefits of real-time event tracking in event management software. The study suggests that real-time attendance tracking and ticket sales monitoring contribute to improved event planning and management. By having access to real-time data, event organizers can identify trends, anticipate attendance numbers, and make timely adjustments to event logistics. This proactive approach leads to better resource allocation, improved event experiences, and increased attendee satisfaction.

  Furthermore, the role of data analytics and reporting capabilities in event management software. The study suggests that event organizers can leverage data analytics to gain insights into attendee behavior,

engagement levels, and preferences. By analyzing event data and generating comprehensive reports, event management software empowers event organizers to make informed decisions, measure event success, and optimize future event planning and marketing strategies.

- **Efficient Resource Management:**

  Event management software assists in efficient resource management by providing tools and features for managing event venues, equipment, and staff. It allows event organizers to keep track of available resources, allocate them effectively, and ensure smooth event operations. The software can include features such as venue management, equipment tracking, and staff scheduling to streamline resource allocation and optimize event logistics.

2.3 Impact of Event Management Systems on Organizational Success:

Implementing a robust event management system can have a significant impact on MAX FIT's operations and overall success. Firstly, the system improves operational efficiency by automating manual tasks, reducing administrative burden, and enabling resource optimization. This allows MAX FIT's team to focus more on strategic event planning and delivering exceptional experiences to attendees.

Secondly, event management systems enhance attendee experiences through streamlined registration processes, personalized communication, and real-time event tracking. Attendees benefit from a user-friendly registration interface, automated confirmation emails, and timely event updates, leading to increased satisfaction and engagement. Positive attendee experiences contribute to brand loyalty and potential word-of-mouth referrals.

Thirdly, the data captured and analyzed by event management systems provide valuable insights for data-driven decision-making. MAX FIT can leverage the reporting and analytics capabilities of the system to measure event success, identify areas for improvement, and make informed decisions regarding future event planning and marketing strategies. This allows MAX FIT to optimize their event portfolio, allocate resources effectively, and drive business growth.

While event management systems offer numerous benefits, their implementation can present certain challenges. One key challenge is the initial setup and customization process. Developing a tailored event management system requires careful analysis of MAX FIT's requirements, system design, and integration considerations. It is essential to align the system with MAX FIT's specific needs and branding to ensure a seamless user experience.

Another challenge is the scalability and extensibility of the system. MAX FIT's event management needs may evolve over time, requiring the system to accommodate a growing number of events, attendees, and functionalities. The system should be designed and implemented in a way that allows for easy scalability and future enhancements.

Furthermore, ensuring data integrity and security is critical in event management systems. MAX FIT must implement robust measures to protect attendee and organizational data, comply with relevant data protection regulations, and continuously monitor and update security protocols.

In terms of future directions, advancements in technology, such as artificial intelligence (AI) and machine learning (ML), can further enhance event management systems. AI and ML algorithms can analyze attendee data, preferences, and behaviors to provide personalized event

recommendations and improve event planning and marketing strategies. Additionally, the integration of virtual and augmented reality technologies can enable immersive event experiences and expand the reach of MAX FIT's events beyond physical locations.

Event management systems have emerged as essential tools for organizations like MAX FIT to streamline event planning, execution, and evaluation processes. The literature review highlighted the benefits of event management systems, including improved efficiency, streamlined communication, enhanced attendee experiences, and data-driven decision-making. The functionalities of event management systems encompass event creation and scheduling, attendee management, organizer and speaker coordination, data integrity and security, automation and efficiency, and integration with external systems.

Chapter-3: System Development

3.1 Analysis:

The aim of this major project is to develop an event management solution for MAX FIT using the Salesforce Platform. The solution will allow effective management of events, including attendee and location information. The project involves the development of various entities (objects) such as Location, Event Organizer, Event, Attendees, Speaker, Event-Attendee, Event-Speaker, and Error Log. Additionally, milestone tasks such as object setup, validation rule setup, duplicate rule setup, profile, user, OWD, and role setup, as well as trigger and batch development, unit testing, and web service implementation are included. This analysis will provide an overview and evaluation of the project components, highlighting key aspects and considerations.

To begin the analysis, it is important to identify the data sources and types that will be involved in the system. In the case of our event management system, the primary data sources may include event registrations, attendee information, speaker details, event schedules, and feedback data. These data sources will provide essential information for managing and organizing events.

3.1.1 Object Setup:

The project requires the creation of several objects with appropriate fields and data types. The entities involved include Location, Event Organizer, Event, Attendees, Speaker, Event-Attendee, Event-Speaker, and Error Log. Each object has specific fields to store relevant information. For example, the Location object contains fields such as Street, City, State, Postal Code, Country, Landmark, and Verified. The Event object includes fields like Event Name, Status, Start Date/Time, End Date/Time, Max Seats, and Event Type. The Attendees and Speaker objects store details related to attendees and speakers, respectively.

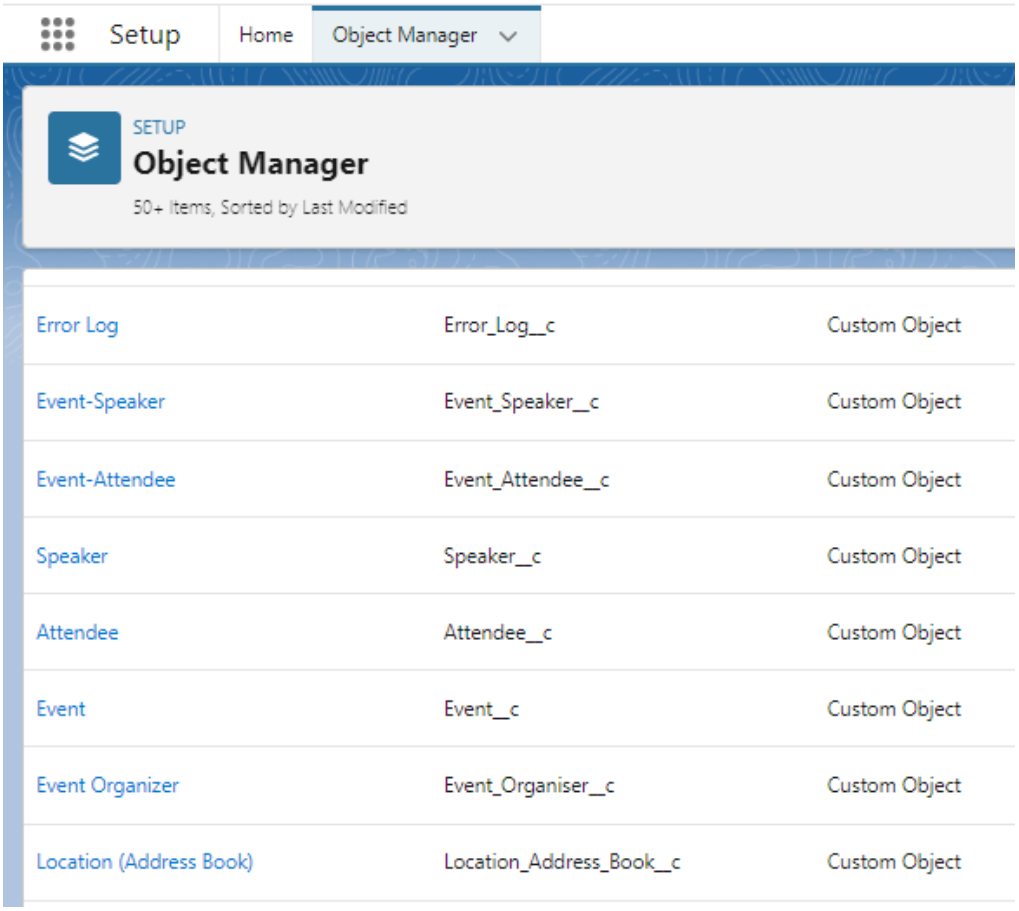


Figure 3.1 : List of Objects

3.1.2 Validation Rule Setup:

To ensure data integrity and enforce business rules, validation rules need to be implemented. The project specifies several validation rules for the Event object, Event Attendee object, and Event Speaker object.

These rules include checking if the Frequency field is populated when the Recurring checkbox is selected, preventing the selection of a Location when the Event Type is set to Virtual, enforcing a minimum time difference between the Start Date/Time and End Date/Time, and requiring the selection of a Location when the Event Type is In-Person.

The project outlines the rules for Speaker, Attendee, and Event Organizer objects.

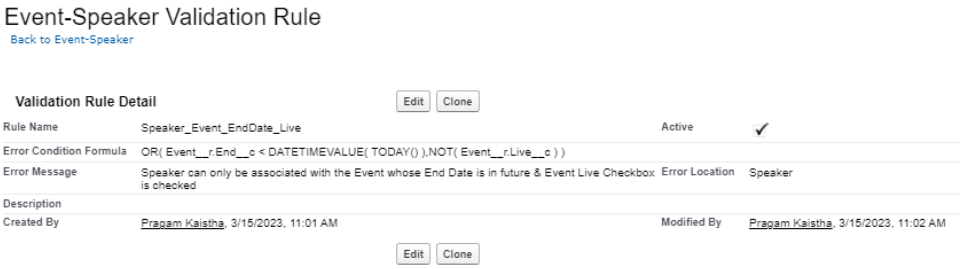


Figure 3.2 : Event-Speaker Validation

3.1.3 Duplicate Rule Setup:

To prevent the creation of duplicate records, duplicate rules must be established. The project outlines the rules for Speaker, Attendee, and Event Organizer objects. These rules ensure that records with the same email, phone, name, or a combination of these fields cannot be duplicated.

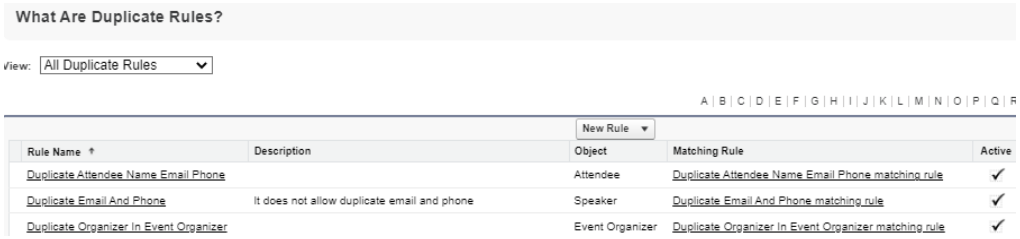


Figure 3.3: Duplicate Values Rule

3.1.4 Profile, User, OWD, and Role Setup:

The project requires the setup of profiles, users, organization-wide defaults (OWD), and roles. Three profiles need to be created: Event Organizer, Event Attendee, and Speaker. Users should be created for testing purposes.

Role hierarchy must be defined, with all roles reporting to the CEO. OWD settings should be adjusted according to the provided table, specifying object permissions for each profile.

The project follows best practices such as proper data modeling, data type selection, and the use of validation rules to enforce data integrity. The implementation of duplicate rules prevents the creation of redundant records.

Error Log	Public Read/Write	Private	✓
Event	Public Read Only	Private	✓
Event-Attendee	Controlled by Parent	Controlled by Parent	
Event Organizer	Public Read Only	Private	✓
Event-Speaker	Controlled by Parent	Controlled by Parent	
Job Application	Private	Private	✓
Job Posting	Controlled by Parent	Controlled by Parent	
Location (Address Book)	Public Read Only	Private	✓
Position	Public Read Only	Private	✓
Review	Controlled by Parent	Controlled by Parent	
Speaker	Private	Private	✓

Figure 3.4: OWD

3.1.5 Sharing Rule Setup:

To share Speaker and Attendee records with the Organizer role, sharing rules need to be configured. These rules ensure that records are accessible to the appropriate roles with the necessary permissions.

3.1.6 Apex Class Development:

An Apex class needs to be developed to insert records into the Error Log object. The class should have a method that accepts dynamic details as parameters, such as Log Date/Time, Log Details, and Process Name. This class allows for logging and tracking errors within the application.

```
public with sharing class ErrorLogDetails {
    public static void insertLog(String processName, String
details) {
        Error_Log__c log = new Error_Log__c();
        log.Process_Name__c = processName;
        log.Log_Date_Time__c = Datetime.now();
        log.Log_Details__c = details;

        insert log;
    }
}
```

Code for Error Details

3.1.7 Trigger Development (Event-Speaker Object):

A trigger must be implemented on the Event-Speaker object to prevent the selection of a speaker who already has an associated event. This ensures that a speaker can only have one event at a time, avoiding duplicate bookings.

```
trigger EventSpeakerTrigger on Event_Speaker__c (before insert,
before update) {

    Map<Id, Event__c> eventMap = new Map<Id, Event__c>();
    Set<Id> eventIds = new Set<Id>();
    Set<Id> speakerId = new Set<Id>();

    for(Event_Speaker__c evtSpkr : Trigger.new) {
        eventIds.add(evtSpkr.Event__c);
        speakerId.add(evtSpkr.Speaker__c);
    }

    List<Event_Speaker__c> spkr = [Select id,
Event__c,Speaker__c,Event__r.Start__c, Event__r.End__c from
Event_Speaker__c where        Speaker__c in :speakerId];

    for(Event__c evt : [SELECT Id, Start__c, End__c FROM Event__c
WHERE Id IN :eventIds]) {
        eventMap.put(evt.Id, evt);
    }

Map<id,Event_Speaker__c> es = new Map<id,Event_Speaker__c>();
for(Event_Speaker__c evtspeaker : spkr) {
            es.put(evtspeaker.Speaker__c,evtSpeaker);
        }

    for(Event_Speaker__c evtSpkr : Trigger.new) {
        Event__c event = eventMap.get(evtSpkr.Event__c);
        Datetime evtTime = event.Start__c;

        if(es.containsKey(evtSpkr.Speaker__c)) {
            Event_Speaker__c eventSpkrId1 =
es.get(evtSpkr.Speaker__c);
                Datetime existingEvtTime =
eventSpkrId1.Event__r.Start__c;

                if((evtTime >= existingEvtTime && evtTime <
eventSpkrId1.Event__r.End__c) || (event.End__c >= existingEvtTime
&& event.End__c < eventSpkrId1.Event__r.End__c) || (evtTime
<=existingEvtTime && event.End__c >=eventSpkrId1.Event__r.End__c
)) {
                    evtSpkr.Speaker__c.addError('Already
booked');
                    evtSpkr.addError('Already Booked');
                }
        }
    }
}
```

Code for Duplicate Speaker

3.1.8 Trigger Development (Event Attendee Object):

An Apex trigger should be created on the Event Attendee object to send confirmation emails to attendees when a new record is created. The email should include basic attendee details, event name, date, time, and location.

3.1.9 Apex Batch Development:

An Apex batch class needs to be developed to delete event records that are more than two months old and have already taken place. The batch should utilize the End Dateof the Event and the Live? checkbox on the Event record. In the finish method of the batch, an email should be sent to notify the completion of the batch execution.

3.1.10 Unit Test Development:

To ensure the functionality and reliability of the implemented components, unit tests need to be developed. The unit tests should aim for at least 90% code coverage and cover both positive and negative scenarios. The use of annotations such as @TestSetup, Test.startTest, Test.stopTest, and assertion methods should be incorporated. Additionally, a TestUtility class can be utilized to streamline test data setup and improve code reusability.

```
@isTest
public class ErrorLogDetailsTest {

    @isTest
    static void insertLogTest() {
        Test.startTest();

        // Test inserting a new error log record
        String processName = 'Process 1';
        String details = 'Error occurred during processing';
        ErrorLogDetails.insertLog(processName, details);

        // Verify that the error log record was created
        List<Error_Log__c> logs = [SELECT Process_Name__c,
Log_Date_Time__c, Log_Details__c FROM Error_Log__c WHERE
Process_Name__c = :processName];
        System.assertEquals(1, logs.size());
        Error_Log__c log = logs[0];
        System.assertEquals(processName, log.Process_Name__c);
```

```
        System.assertEquals(details, log.Log_Details__c);

        Test.stopTest();
    }

}
```

Test Class For Error Log

```
@isTest
private class EventSpeakerTriggerTest {

    @testSetup
    static void setupTestData() {
        // Create test data
    }

    // Test case where a speaker is not already booked for the
same time as the event
    @isTest
    static void testValidEventSpeakerInsert() {
        // Create test data
        Event__c event = new Event__c(Name__c = 'Test Event
1',Live__c = true, Start__c = Datetime.now().addDays(3), End__c =
Datetime.now().addDays(7));
        insert event;
        Speaker__c speaker = new Speaker__c(Name = 'Test
Speaker');
        insert speaker;
        Event_Speaker__c eventSpeaker = new
Event_Speaker__c(Event__c = event.Id, Speaker__c = speaker.Id);

        // Test trigger behavior
        Test.startTest();
        insert eventSpeaker;
        Test.stopTest();

        // Verify no errors were thrown
        System.assertEquals(0, eventSpeaker.getErrors().size());
    }

    // Test case where a speaker is already booked for the same
time as the event
    @isTest
    static void testDuplicateEventSpeakerInsert() {
        // Create test data
        Event__c event1 = new Event__c(Name__c = 'Test Event
1',Live__c = true, Start__c = Datetime.now().addDays(3), End__c =
Datetime.now().addDays(7));
```

```
        insert event1;
        Event__c event2 = new Event__c(Name__c = 'Test Event
2',Live__c = true, Start__c = Datetime.now().addDays(2), End__c =
Datetime.now().addDays(8));
        insert event2;
        Speaker__c speaker = new Speaker__c(Name = 'Test
Speaker');
        insert speaker;
        Event_Speaker__c eventSpeaker1 = new
Event_Speaker__c(Event__c = event1.Id, Speaker__c = speaker.Id);
        Event_Speaker__c eventSpeaker2 = new
Event_Speaker__c(Event__c = event2.Id, Speaker__c = speaker.Id);
        insert eventSpeaker1;

        // Test trigger behavior
        Test.startTest();
        try{
            insert eventSpeaker2;
        }
        catch(Exception e){
            Event_Speaker__c es = eventSpeaker2;
            es.Speaker__c.addError('Already booked');
            //System.debug(e);
        }
        Test.stopTest();

        // Verify errors were thrown
        System.assertEquals(1, eventSpeaker2.getErrors().size());
        System.assertEquals('Already booked',
eventSpeaker2.getErrors()[0].getMessage());
    }

    // Test case where an existing event speaker is updated to a
valid time slot
    @isTest
    static void testValidEventSpeakerUpdate() {
        // Create test data
        Event__c event1 = new Event__c(Name__c = 'Test Event
1',Live__c = true, Start__c = Datetime.now().addDays(3), End__c =
Datetime.now().addDays(7));
        insert event1;
        Event__c event2 = new Event__c(Name__C = 'Test Event
2',Live__c = true, Start__c = Datetime.now().addDays(9), End__c =
Datetime.now().addDays(17));
        insert event2;
        Speaker__c speaker = new Speaker__c(Name = 'Test
Speaker');
        insert speaker;
```

```
        Event_Speaker__c eventSpeaker = new
Event_Speaker__c(Event__c = event1.Id, Speaker__c = speaker.Id);
        insert eventSpeaker;

        // Test trigger behavior
        eventSpeaker.Event__c = event2.Id;
        Test.startTest();
        update eventSpeaker;
        Test.stopTest();

        // Verify no errors were thrown
        System.assertEquals(0, eventSpeaker.getErrors().size());
    }
}
```

Test Class For Duplicate Speaker

3.1.11 Web Service Development:

The project requires the development of web services to meet specific requirements. One web service should retrieve events in JSON format, including event details and event organizer information. Another web service should allow the creation of a speaker record in Salesforce, accepting input in JSON format and providing appropriate success or failure responses. Test classes should be written to verify the functionality of these web services.

```
@RestResource(urlMapping='/v1/EventWebService/')
global class EventWebService {
    @httpGet
    global static List<EventWrapper> doGetMethod(){
        List<EventWrapper> eventWrappers = new
List<EventWrapper>();
        List<Event__c> events = [SELECT Name__c, Start__c,
Max_Seats__c,Event_Organizer__r.Name ,
Event_Organizer__r.Email__c
                                    FROM Event__c];
        for (Event__c event : events) {
            EventWrapper eventWrapper = new EventWrapper();
            eventWrapper.event = new Event__c();
            eventWrapper.event.Name__c = event.Name__c;
            eventWrapper.event.Start__c = event.Start__c;
            eventWrapper.event.Max_Seats__c = event.Max_Seats__c;
            eventWrapper.event_organizer = new
Event_Organiser__c();
```

```
            eventWrapper.event_organizer.Name =
event.Event_Organizer__r.Name;
            eventWrapper.event_organizer.Email__c =
event.Event_Organizer__r.Email__c;
            eventWrappers.add(eventWrapper);
        }
        return eventWrappers;
    }

    @httpPost
    global static String doPostMethod(String Name, String email,
String phone, String company){

        try
        {
        Speaker__c spkr= new Speaker__c(Name =Name, Email__c =
email, Phone__c = phone, Company__c = company);
        insert spkr;
        return '200';
        }
        catch(Exception e){
            return '400';
        }
    }

    // Wrapper class for events
    global class EventWrapper {
        global Event__c event;
        global Event_Organiser__c event_organizer;
    }
}
```

Code For Web Service (REST API)

a) Evaluation:

The project aims to address the event management needs of MAX FIT using the Salesforce Platform. By analyzing the provided requirements, it is evident that the solution covers various aspects, including object setup, validation rules, duplicate rules, profile, user, OWD, and role setup. Additionally, the development of triggers, Apex classes, and Apex batch classes enhances the functionality and automation of the system.

The project follows best practices such as proper data modeling, data type selection, and the use of validation rules to enforce data integrity. The implementation of duplicate rules prevents the creation of redundant records. The setup of profiles, users, OWD, and roles ensures proper access control and data visibility within the system.

The development of triggers adds business logic to handle specific scenarios, such as preventing duplicate speaker bookings and sending confirmation emails to attendees. The Apex batch class efficiently manages the deletion of old event records. The inclusion of unit tests with high code coverage guarantees the reliability and stability of the implemented functionality.

The web service development aligns with the project requirements, providing external interfaces to retrieve event information and create speaker records. The accompanying test classes ensure the correctness and robustness of the web service implementations.

Overall, the project demonstrates a comprehensive understanding of event management requirements and effectively utilizes the Salesforce Platform to build a solution that meets MAX FIT's needs. The adherence to best practices and the successful completion of milestone tasks contribute to the overall success of the project.

Through the successful completion of these tasks, the project aims to provide MAX FIT with an efficient and effective event management system. The analysis reveals the comprehensive understanding and utilization of Salesforce Platform capabilities to address the requirements outlined in the project. By considering best practices and ensuring code coverage, the project demonstrates a robust and reliable solution.

Overall, the project showcases the technical expertise and proficiency in developing applications on the Salesforce Platform, meeting the needs of MAX FIT in managing their events effectively.

The data types within these sources may include:

Speaker Details: This data may consist of speaker names, biographies, contact information, and areas of expertise.

Event Schedules: Event schedules may include information on event names, dates, times, locations, session details, and associated speakers.

Event Registration Data: This data may include attendee names, contact information, registration dates, and event preferences.

Fields & Relationships
18 Items, Sorted by Field Label

Quick Find

Event Organizer	Event_Organizer__c	Lookup(Event Organizer)
Event Type	Event_Type__c	Picklist
Frequency	Frequency__c	Picklist
Last Modified By	LastModifiedById	Lookup(User)
Live?	Live__c	Checkbox
Location (Address Book)	Location_Address_Book__c	Lookup(Location (Address Book))
Location Verified	Location_Verified__c	Formula (Checkbox)
Max Seats	Max_Seats__c	Number(5, 0)
Name	Name__c	Text(50)
Number of Attending	Number_of_Attending__c	Roll-Up Summary (COUNT Event-Attendee)

Figure 3.5 : Event Fields

Attendee Information: Attendee data may encompass personal details, such as names, addresses, contact information, and any additional information provided during registration.

Feedback Data: Feedback data may comprise attendee ratings, comments, and suggestions regarding events and speakers.

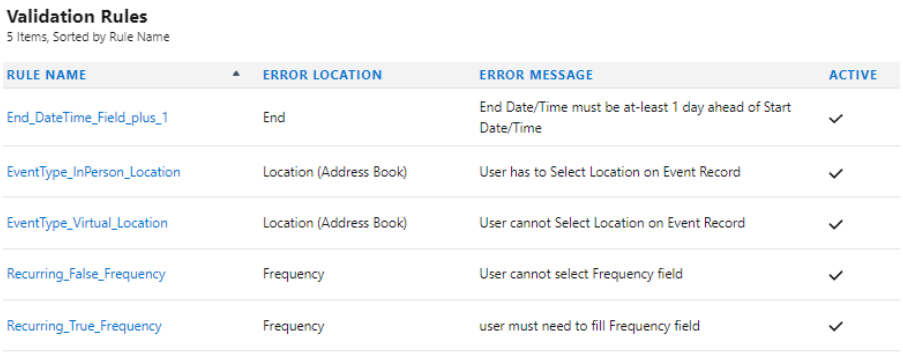
Understanding the specific data sources and types will help in designing the appropriate data model and relationships within the system.

b) Data Model and Entity-Relationship Diagram (ERD)

Based on the analysis of data sources and types, a data model can be developed to represent the structure and relationships of the data within the system. This can be visualized using an Entity-Relationship Diagram (ERD). The ERD will illustrate the entities (such as events, attendees, and speakers), their attributes, and the relationships between them.

For example, the data model may include entities such as:

Event: with attributes like event ID, event name, date, time, location, and status.



Validation Rules

5 Items, Sorted by Rule Name

RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE
End_DateTime_Field_plus_1	End	End Date/Time must be at-least 1 day ahead of Start Date/Time	✓
EventType_InPerson_Location	Location (Address Book)	User has to Select Location on Event Record	✓
EventType_Virtual_Location	Location (Address Book)	User cannot Select Location on Event Record	✓
Recurring_False_Frequency	Frequency	User cannot select Frequency field	✓
Recurring_True_Frequency	Frequency	user must need to fill Frequency field	✓

Figure 3.6: Event Validation

Attendee: with attributes like attendee ID, name, contact information, and registration date.

Speaker: with attributes like speaker ID, name, contact information, and areas of expertise.

Feedback: with attributes like feedback ID, attendee ID, event ID, rating, comments, and submission date.

The relationships between these entities can be defined, such as the relationship between events and attendees (many-to-many), events and speakers (many-to-many), and events and feedback (one-to-many).

c) Data Validation and Integrity

During the analysis of data, it is crucial to consider data validation and integrity requirements. This involves defining rules and constraints to ensure that the data entered into the system is accurate, consistent, and meets predefined criteria. For example, validation rules can be established to verify that attendee contact information is in the correct format, event dates are valid, and feedback ratings fall within a specific range.

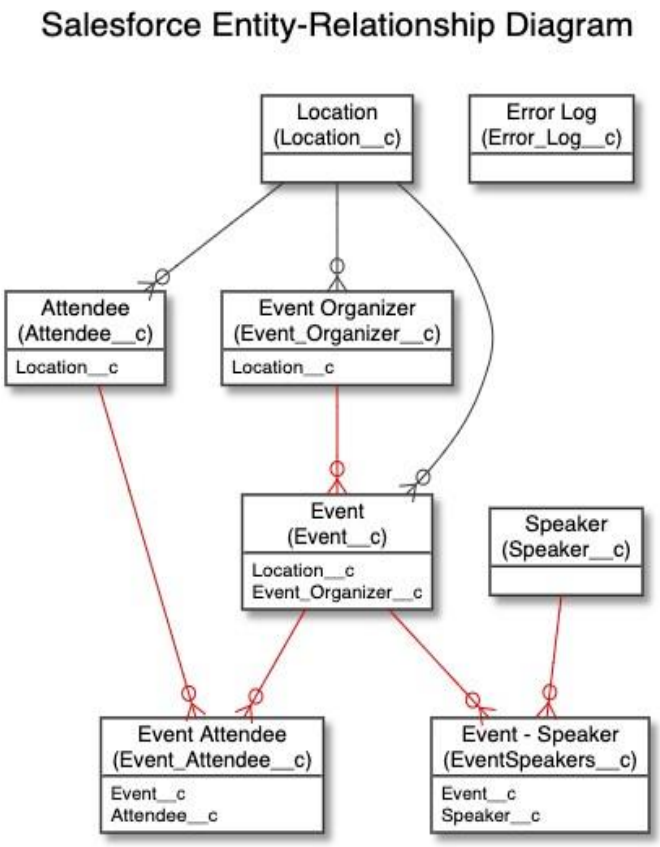


Figure 3.7 : Object E-R Diagram

In addition, measures to ensure data integrity, such as the use of unique identifiers (e.g., event ID, attendee ID, speaker ID), can be implemented. This helps prevent data duplication or inconsistencies within the system.

d) Data Storage and Database Design

Next, the analysis of data should consider the storage and organization of data within a database. The selection of an appropriate database management system (DBMS) and database design are critical in ensuring efficient data retrieval, manipulation, and scalability.

The analysis should consider factors such as the volume of data, performance requirements, and anticipated growth. Based on these considerations, a suitable database model (e.g., relational, NoSQL) can be chosen, and the database schema can be designed. This includes defining tables, fields

and relationships within the database schema. The data model and entity-relationship diagram developed earlier will serve as a guide in designing the database schema.

For example, the database schema may include tables such as:

Event: with columns like event_id, event_name, event_date, event_time, event_location, and event_status.

Attendee: with columns like attendee_id, attendee_name, contact_info, and registration_date.

Speaker: with columns like speaker_id, speaker_name, contact_info, and expertise_area.

Feedback: with columns like feedback_id, attendee_id, event_id, rating, comments, and submission_date.

The appropriate data types, primary keys, foreign keys, and constraints should be defined for each column to ensure data integrity and efficient data retrieval.

e) Data Processing and Analytics

In addition to storage, the analysis of data should also consider the processing and analytics requirements. This involves determining the

operations and queries that will be performed on the data to support the desired functionalities of the system.

For example, the system may require queries such as:

Retrieve a list of upcoming events.

Get the details of attendees registered for a specific event.

Calculate the average rating of a speaker based on feedback.

Generate reports on event attendance and feedback statistics.

Based on these requirements, appropriate indexes, views, and stored procedures can be designed to optimize data processing and retrieval.

f) Data Security and Privacy

During the analysis of data, it is essential to address data security and privacy concerns. This includes identifying sensitive data elements and defining access controls and security measures to protect the data from unauthorized access, modification, or disclosure. Encryption techniques, user authentication, and role-based access controls can be implemented to ensure data security.

Additionally, compliance with relevant data privacy regulations (such as GDPR or CCPA) should be considered, and mechanisms for obtaining user consent and managing data privacy preferences can be incorporated into the system design.

The analysis of data is a critical step in the system design process. It involves understanding the data sources, types, relationships, and requirements for validation, storage, processing, and security. By conducting a thorough analysis of data, a solid foundation can be laid for designing a robust, efficient, and secure system that effectively manages and processes the data associated with event management.

This comprehensive analysis provides insights into the various aspects that need to be considered while designing the system, ensuring that it meets the functional and non-functional requirements of the project.

In conclusion, the analysis of the major project report highlights the key components and tasks involved in developing an event management solution using the Salesforce Platform for MAX FIT. The project covers various aspects, including object setup, validation rules, duplicate rules, profile, user, OWD, and role setup. Triggers, Apex classes, and Apex batch classes are developed to enhance functionality and automation. Unit tests ensure the reliability and correctness of the implemented components. Web services are created to provide external interfaces for event retrieval and speaker record creation.

3.2 Design

The MAX FIT Event Management System aims to provide a comprehensive solution for managing events effectively, including attendee and location information.

The system will be developed using the Salesforce Platform, leveraging its powerful features for object modeling, data management, and automation.

This system design document outlines the architecture, modules, and key functionalities of the MAX FIT Event Management System.

System Architecture:

The system will be built on the Salesforce Platform, utilizing its core features and technologies, including the Lightning framework for user interface development, Apex for server-side logic, and Salesforce Object Query Language (SOQL) for data retrieval and manipulation.

The system will consist of the following modules:

Object Model:

The MAX FIT Event Management System will involve several custom objects to store and manage data related to locations, event organizers, events, attendees, speakers, event-attendee relationships, event-speaker relationships, and error logs.
The relationships between these objects will be established using lookup and master-detail relationships.

- Location Object: This object stores information about event locations, including the street, city, state, postal code, country, landmark, and a checkbox to indicate if the location has been verified.

Fields & Relationships
11 Items, Sorted by Field Label

Quick Find

City	City__c	Text(50)
Country	Country__c	Picklist
Created By	CreatedById	Lookup(User)
Landmark	Landmark__c	Text(50)
Last Modified By	LastModifiedById	Lookup(User)
Location (Address Book) Name	Name	Text(80)
Owner	OwnerId	Lookup(User,Group)
Postal Code	Postal_Code__c	Text(20)
State	State__c	Text(50)
Street	Street__c	Text(50)
Verified	Verified__c	Checkbox

Figure 3.8: Location Object

- Event Organizer Object: This object represents event organizers and includes fields such as name, email, alternative email, phone, alternative phone, and a lookup field to the Location object.

Fields & Relationships
9 Items, Sorted by Field Label

Quick Find

FIELD LABEL	FIELD NAME	DATA TYPE
Alternate Email	Alternate_Email__c	Email
Alternate Phone	Alternate_Phone__c	Phone
Created By	CreatedById	Lookup(User)
Email	Email__c	Email
Event Organiser Name	Name	Text(80)
Last Modified By	LastModifiedById	Lookup(User)
Location (Address Book)	Location_Address_Book__c	Lookup(Location (Address Book))
Owner	OwnerId	Lookup(User,Group)
Phone	Phone__c	Phone

Figure 3.9: Organizer Object

- Event Object:

  Event schedules may include information on event names, dates, times, locations, session details, and associated speakers.

  This object captures event details, such as event name, status, organizer (lookup to Event Organizer object), start date/time, end date/time, maximum seats, number of people attending (roll-up summary field), remaining seats (formula field), location (lookup to Location object), location verified (formula field), whether the event is live or recurring, event type (in-person or virtual), and frequency for recurring events.

- Attendee Object:

  This object stores information about event attendees, including name, email, phone, company name, and a lookup field to the Location object.

  Attendee data may encompass personal details, such as names, addresses, contact information, and any additional information provided during registration.

Fields & Relationships

8 Items, Sorted by Field Label

Quick Find

FIELD LABEL	FIELD NAME	DATA TYPE
Address	Address__c	Lookup(Location (Address Book))
Attendee Name	Name	Text(80)
Company Name	Company_Name__c	Text(50)
Created By	CreatedById	Lookup(User)
Email	Email__c	Email
Last Modified By	LastModifiedById	Lookup(User)
Owner	OwnerId	Lookup(User,Group)
Phone	Phone__c	Phone

Figure 3.10: Attendee Object

- Speaker Object: This object represents event speakers and includes fields such as name, email, phone, company, profile URL, and about me (rich text area).
- Event-Attendee Object: This object establishes a many-to-many relationship between events and attendees.
- Event-Speaker Object: This object establishes a many-to-many relationship between events and speakers.

Fields & Relationships

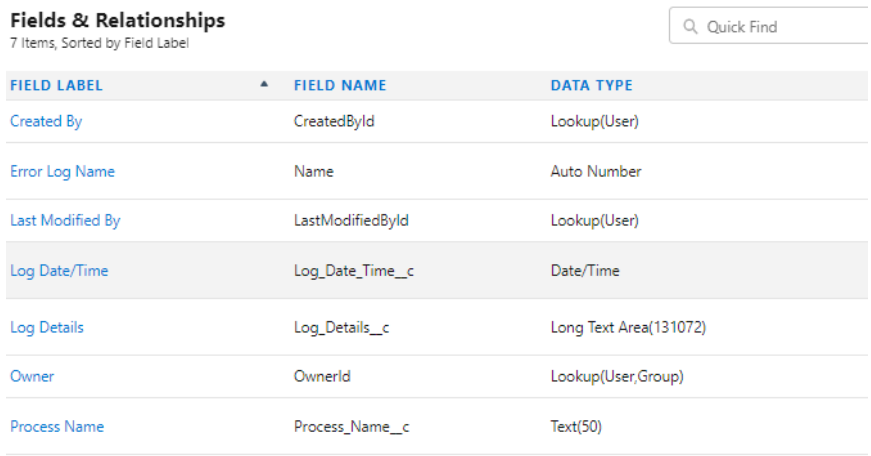
6 Items, Sorted by Field Label

Quick Find

FIELD LABEL	FIELD NAME	DATA TYPE
Created By	CreatedById	Lookup(User)
Event	Event__c	Master-Detail(Event)
Event-Speaker Name	Name	Text(80)
Is Checked?	Is_Checked__c	Checkbox
Last Modified By	LastModifiedById	Lookup(User)
Speaker	Speaker__c	Master-Detail(Speaker)

Figure 3.11: Event-Speaker Object

- Error Log Object: This object is used to store error log records and includes fields for log date/time, log details, process name, and an auto-generated log number.



Fields & Relationships
7 Items, Sorted by Field Label

Quick Find

FIELD LABEL	FIELD NAME	DATA TYPE
Created By	CreatedById	Lookup(User)
Error Log Name	Name	Auto Number
Last Modified By	LastModifiedById	Lookup(User)
Log Date/Time	Log_Date_Time__c	Date/Time
Log Details	Log_Details__c	Long Text Area(131072)
Owner	OwnerId	Lookup(User,Group)
Process Name	Process_Name__c	Text(50)

Figure 3.12: Error Log Object

User Interface:

The user interface will be developed using the Lightning framework, providing a modern and intuitive interface for users to interact with the system. The interface will include forms for event registration, speaker registration, and attendee registration, as well as pages to display event details and speaker profiles.

Business Logic:

The business logic of the system will be implemented using Apex, Salesforce's proprietary programming language. Apex triggers will be used to enforce validation rules and perform custom actions upon record creation or modification. Apex classes will handle complex business processes, such as error logging and batch processing.

Validation Rules:

Validation rules will be set up to ensure data integrity and enforce specific business requirements. These rules will include checks for recurring events, virtual event restrictions, date and time validations, and event location requirements. Attendee and speaker registrations will also be validated based on the event's status, date, and availability.

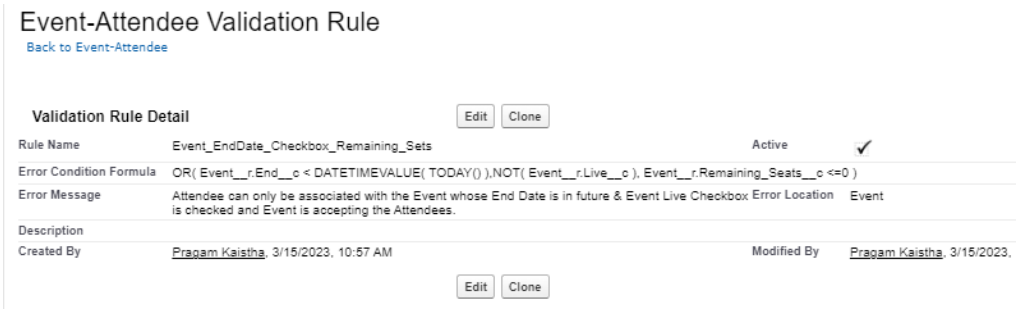


Figure 3.13: Event Attendee Validation

Duplicate Rules:

Duplicate rules will be defined to prevent the creation of duplicate records for speakers, attendees, and event organizers. These rules will compare fields such as email, phone, and name to identify duplicates and notify users to avoid redundant data entry.

User Profiles and Roles and sharing settings :

Different user profiles will be created, including Event Organizer, Event Attendee, and Speaker, each with specific permissions and access levels. Additionally, roles will be defined to establish a role hierarchy, where all roles will report to the CEO. This role hierarchy ensures proper data visibility and access control within the system.

Organization-Wide Default (OWD) and Sharing Rules:

OWD settings will be configured to define the default level of access for objects and their records. Sharing rules will be set up to share speaker

and attendee records with the Organizer role, granting them read and edit permissions. This ensures that event organizers have access to the relevant data and can effectively manage speakers and attendees.

Apex Class Development:

An Apex class will be developed to handle the insertion of error log records. This class will contain a method that accepts dynamic parameters such as log date/time, log details, and process name.

It will create error log records with the provided information, capturing any errors or exceptions encountered during system processes.

Trigger Development (Event - Speaker Object):

A trigger will be implemented on the Event - Speaker object to prevent duplicate bookings for speakers.

The trigger will check if the selected speaker already has an associated event, and if so, it will throw an error. This ensures that each speaker can only be associated with one event at a time, avoiding conflicts and overlapping bookings.

Trigger Development (Event Attendee Object):

An Apex trigger will be developed on the Event Attendee object to send confirmation emails to attendees when a new registration is created. The trigger will retrieve the attendee's basic details such as name, email, and phone, and generate an email template containing the event's name, date, location, and organizer's name. The trigger will handle bulk record handling and exception handling to ensure reliable email delivery.

```
trigger EventAttendeeTrigger on Event_Attendee__c (after insert)
{

    EventAttendeeTriggerHandler.sendEmails(Trigger.new);

}
```

Code to send email trigger

```
public with sharing class EventAttendeeTriggerHandler {
    public static void sendEmails(List<Event_Attendee__c>
evtAttendee) {

        List<Id> attendeeId = new List<Id>();
    for(Event_Attendee__c evtAttndee: evtAttendee){
        attendeeId.add(evtAttndee.Attendee__c);
    }

    List<Messaging.SingleEmailMessage> mails = new
List<Messaging.SingleEmailMessage>();
    List<Event_Attendee__c> attendee = [Select id,
Attendee__r.Email__c,Event__r.Start__c,Event__r.Name__c,
Attendee__r.Name, Event__r.Location_Address_Book__r.State__c,
Event__r.Event_Organizer__r.Name,Event__r.Location_Address_Book__
r.Country__c,Event__r.Location_Address_Book__r.City__c,Event__r.L
ocation_Address_Book__r.Street__c, Event__c from
Event_Attendee__c where Attendee__c in :attendeeId];

    for(Event_Attendee__c evtAttndee: attendee){
        Messaging.SingleEmailMessage mail = new
Messaging.SingleEmailMessage();
        //mail.setToAddresses(''+evtAttndee.Attendee__r.Email__c)
;
        List<String> sendTo = new List<String>();
        sendTo.add(evtAttndee.Attendee__r.Email__c);
        mail.setToAddresses(sendTo);
        mail.setReplyTo('pragam.kaistha@mirketa.com');
        mail.setSenderDisplayName(evtAttndee.Event__r.Event_Organ
izer__r.Name);
        mail.setSubject('Pass for the
'+evtAttndee.Event__r.Name__c);

        String body = 'Dear '+evtAttndee.Attendee__r.Name
+'<br><br>'+'Thank you for registering for '+
evtAttndee.Event__r.Name__c + ' which will be Organized on '+
evtAttndee.Event__r.Start__c +' & will be held in '
+evtAttndee.Event__r.Location_Address_Book__r.Street__c+', '+
evtAttndee.Event__r.Location_Address_Book__r.City__c+',
'+evtAttndee.Event__r.Location_Address_Book__r.State__c+',
'+evtAttndee.Event__r.Location_Address_Book__r.Country__c+'.
```

```
<br>We are excited to have you, see you at the
event.<br><br>Thanks<br>'+evtAttndee.Event__r.Event_Organizer__r.
Name;

        mail.setHtmlBody(body);
        mails.add(mail);
    }
    try{

        Messaging.sendEmail(mails, false);
    }catch(System.Exception e){
        System.debug(e);
    }

    }
}
```

Trigger Handler Code

Apex Batch Development:

An Apex batch class will be implemented to periodically purge old event records. The batch will identify events that are more than two months old and have already taken place. These records will be deleted to maintain data cleanliness and optimize system performance. The batch's finish method will send an email notification to a designated address, informing about the successful execution of the batch.

```
public with sharing class DeleteEventRecords implements
Database.Batchable<SObject>  {
    public Database.QueryLocator start(Database.BatchableContext
batchContext){
        String query = 'Select id, Name__c from Event__c where
Live__c = true AND End__c < LAST_N_MONTHS:2';
        return Database.getQueryLocator(query);

    }

    public void execute(Database.BatchableContext batchContext,
List<Event__c> listEvt){
        Database.delete(listEvt);

    }

    public void finish(Database.BatchableContext batchContext){
```

```
        Messaging.SingleEmailMessage mail = new
Messaging.SingleEmailMessage();

        mail.setSubject('Batch has been processed');
        List<String> toAddress = new List<String>();
        toAddress.add('pragam.kaistha@mirketa.com');
        mail.setToAddresses( toAddress );
        mail.setSenderDisplayName('Organizer');

        mail.setHtmlBody('The records have been deleted.');
        List<Messaging.SingleEmailMessage> emails = new
List<Messaging.SingleEmailMessage> {mail};
        Messaging.sendEmail(emails);
    }
}
```

Asynchronous Apex Code

Unit Test Development:

Comprehensive unit tests will be created to ensure the quality and reliability of the implemented functionality. The tests will cover the Apex classes, triggers, and batch processes. They will use the @TestSetup annotation for test data setup, Test.startTest and Test.stopTest to delineate test execution boundaries, and Asserts methods to validate expected results. Negative test cases will also be included to handle exceptional scenarios and edge cases.

```
@isTest
public class DeleteEventRecordsTest {

    @testSetup
    public static void testSetupData(){
        Event_Organiser__c  org = new Event_Organiser__c (
            Name = 'PK Test1',
            Phone__c = '9658656985',
            Email__c = 'pragam.kaistha@mirketa.com',
            Alternate_Phone__c  = '6549652145',
            Alternate_Email__c ='pragamkaistha@gmail.com'
        );
        insert org;
        List<Event__c> eventList = new List<Event__c>();
        for(Integer i=0; i<=200; i++){
            Event__c event = new Event__c(
                Name__c = 'New Event #'+i+1,
                Event_Organizer__c  = org.Id,
                Event_Type__c = 'Virtual',
```

```
                Frequency__c  = 'Weekly',
                Max_Seats__c  = 199,
                Recurring__c = true,
                Live__c = true,
                Start__c = System.now().addMonths(-4),
                End__c  = System.now().addDays(3).addMonths(-4)
            );
            eventList.add(event);
        }
        insert eventList;
    }

    @isTest
    static void sendDeletEventTest(){
        Test.startTest();

        String jobId = Database.executeBatch(new
DeleteEventRecords(), 250);

        Test.stopTest();
        List<Event__c> eventList = [Select id, Name from
Event__c];
        System.assertEquals(0, eventList.size());
    }
}
```

Batch Apex Test Class

```
@isTest
private class EventAttendeeTriggerTest {
    @isTest
    static void sendEmailsTest() {
        // Setup test data
        Event_Organiser__c org = new Event_Organiser__c(
            Name = 'Test Org',
            Phone__c = '1234567890',
            Email__c = 'pragam.kaistha@mirketa.com'
        );
        insert org;
        Event__c evt = new Event__c(
            Name__c = 'Test Event',
            Event_Organizer__c = org.Id,
            Event_Type__c = 'Virtual',
            Frequency__c = 'Weekly',
            Max_Seats__c = 100,
            Recurring__c = true,
            Live__c = true,
            Start__c = System.now().addDays(1),
            End__c = System.now().addDays(3)
```

```
        );
        insert evt;
        Attendee__c att = new Attendee__c(
            Name = 'Test Attendee',
            Email__c = 'kaisthapragam@gmail.com',
            Phone__c = '1234567890'
        );
        insert att;
        Event_Attendee__c evtAtt = new Event_Attendee__c(
            Event__c = evt.Id,
            Attendee__c = att.Id
        );
        insert evtAtt;

        // Call the method being tested
        Test.startTest();
        EventAttendeeTriggerHandler.sendEmails(new
List<Event_Attendee__c>{ evtAtt });
        Test.stopTest();

        // Verify that an email was sent
        System.assertEquals(1, Limits.getEmailInvocations());
    }
}
```

Test Class For Sending Mail

With these additional milestones, the MAX FIT Event Management System will be equipped with the following features:

Event Registration Form:

A user-friendly form will be created to facilitate event registration. The form will consist of collapsible sections for the Event Organizer, Event Object, and Event Location Object. Upon submission, records for the event organizer, event, and location will be automatically created in the system.

```
<apex:page controller="EventRegistrationController">
    <apex:form >
        <apex:pageBlock title="Event Registration">
            <apex:pageBlockSection title="Event organizer"
collapsible="true" columns="2">
                <apex:inputField value="{!evtOrganizer.Name}"/>
                <apex:inputField
value="{!evtOrganizer.Email__c}"/>
                <!-- <apex:inputField
value="{!evtOrganizer.Location_Address_Book__c}"/> -->
                <apex:inputField
value="{!evtOrganizer.Phone__c}"/>
            </apex:pageBlockSection>

            <apex:pageBlockSection title="Event" id="evt"
collapsible="true" columns="2">
                <apex:inputField value="{!evt.Name__c}"/>
                <apex:inputField value="{!evt.Event_Type__c}"/>
                <apex:inputField value="{!evt.Max_Seats__c}"/>
                <apex:inputField value="{!evt.Start__c}"/>
                <apex:inputField value="{!evt.End__c}"/>
                <apex:inputCheckbox value="{!evt.Live__c}"/>
            </apex:pageBlockSection>

            <apex:pageBlockSection title="Location" id="loc"
collapsible="true" columns="2">
                <apex:inputField value="{!loc.Name}"/>
                <apex:inputField value="{!loc.Street__c}"/>
                <apex:inputField value="{!loc.City__c}"/>
                <apex:inputField value="{!loc.State__c}"/>
                <apex:inputField value="{!loc.Country__c}"/>
                <apex:inputField value="{!loc.Postal_Code__c}"/>

            </apex:pageBlockSection>
            <!-- <apex:pageBlockButtons> -->
            <div align="center" >
                <apex:commandButton action="{!save}"
value="Save"/>
            </div>
            <!-- </apex:pageBlockButtons> -->
        </apex:pageBlock>
    </apex:form>
</apex:page>
```

VisualForce Page For Event Registration

```
public with sharing class EventRegistrationController {

    public Event_Organiser__c evtOrganizer {get;set;}
    public Event__c evt {get;set;}
    public Location_Address_Book__c  loc {get;set;}

    public EventRegistrationController ()
    {
        evtOrganizer = new Event_Organiser__c();
        evt = new Event__c();
        loc = new Location_Address_Book__c();
    }
    public PageReference save()
    {
        // insert con;

        insert loc;
        evtOrganizer.Location_Address_Book__c = loc.Id;
        // loc = new Location_Address_Book__c();

        insert evtOrganizer;
        evt.Event_Organizer__c = evtOrganizer.Id;
        evt.Location_Address_Book__c = loc.Id;
        // evtOrganizer = new Event_Organiser__c();
        insert evt;
        return null;
 }
}
```

Controller for Event Registration

Speaker Registration Form:

Another form will be developed for speaker registration. The form will capture details such as the speaker's name, email, phone, and company. It will also include a lookup field to select the associated event. Once submitted, records for the speaker and the event-speaker relationship will be generated.

```
<apex:page controller="EventSpeakerController">
    <apex:form >
    <apex:pageBlock title="Event Speaker">

        <apex:pageBlockSection columns="2">
            <apex:inputField value="{!evtSpkr.Name}"/>
            <apex:inputField value="{!spkr.Name}"/>
            <apex:inputField value="{!spkr.Email__c}"/>
            <apex:inputField value="{!spkr.Phone__c}"/>
```

```
            <apex:inputField value="{!evtSpkr.Event__c}"
rendered="{!bool}"/>
            <apex:outputField value="{!evtSpkr.Event__c}"
rendered="{!NOT(bool)}"/>
            <!-- <apex:pageBlockButtons> -->
                <div align="center" >
                    <apex:commandButton action="{!save}"
value="Save"/>
                </div>
                <!-- </apex:pageBlockButtons> -->
        </apex:pageBlockSection>
    </apex:pageBlock>
    </apex:form>
</apex:page>
```

Event Speaker Form

```
public with sharing class EventSpeakerController {
    public Speaker__c spkr{get;set;}
    // public Event__c evt{get;set;}
    public String paramValue;
    public Boolean bool{get;set;}
    public Event_Speaker__c evtSpkr{get;set;}
    public EventSpeakerController() {

        paramValue =
apexpages.currentpage().getparameters().get('param1');
        spkr = new Speaker__c();
        // evt = new Event__c();
        evtSpkr = new Event_Speaker__c();
        evtSpkr.Event__c = paramValue;
        if(paramValue!=null){
            bool= false;
        }
        else{
            bool = true;
        }
    }
    public void save(){
        insert spkr;
        evtSpkr.Speaker__c = spkr.Id;
        insert evtSpkr;
    }
}
```

Event Speaker Controller

Attendee Registration Form:

A comprehensive registration form for attendees will be created. The form will include collapsible sections for Attendee Details and Attendee Location. Attendee details will capture information such as name, email, phone, and company. The Attendee Location section will allow attendees to select their location using a lookup field. Upon submission, records for the attendee and the event-attendee relationship will be created.

```
<apex:page controller="AttendeeRegistrationController">
<apex:form >
    <apex:pageBlock title="Attendee Registration">
        <apex:pageBlockSection title="Attendee Details"
columns="2" collapsible="true">
            <apex:inputField value="{!attndee.Name}"/>
            <apex:inputField value="{!attndee.Phone__c  }"/>
            <apex:inputField value="{!attndee.Email__c}"/>
            <apex:inputField value="{!attndee.Company_Name__c}"/>
            <apex:inputField value="{!evtAttndee.Name}"/>
            <apex:inputField value="{!evtAttndee.Event__c}"
rendered="{!evtAttndee.Event__c == null}"/>
            <apex:outputField value="{!evtAttndee.Event__c}"
rendered="{!evtAttndee.Event__c != null}"/>
        </apex:pageBlockSection>
        <apex:pageBlockSection title="Attendee Location"
columns="2" collapsible="true">
            <apex:inputField value="{!loc.Street__c}"/>
            <apex:inputField value="{!loc.City__c}"/>
            <apex:inputField value="{!loc.State__c}"/>
            <apex:inputField value="{!loc.Country__c}"/>
            <apex:inputField value="{!loc.Postal_Code__c}"/>
        </apex:pageBlockSection>
        <div align="center">
        <apex:commandButton action="{!save}" value="Save" />
        </div>
    </apex:pageBlock>
</apex:form>
</apex:page>
```

Event Attendee Form

```
public with sharing class AttendeeRegistrationController {
    public Event_Attendee__c evtAttndee{get;set;}
    public Attendee__c attndee{get;set;}
    public Location_Address_Book__c loc{get;set;}
    public String paramValue;
    public Boolean bool{get;set;}
    public AttendeeRegistrationController() {
```

```
        paramValue =
apexpages.currentpage().getparameters().get('param1');
        evtAttndee = new Event_Attendee__c();
        attndee = new Attendee__c();
        loc = new Location_Address_Book__c();
        // bool =true;
        evtAttndee.Event__c = paramValue;

    }

    public void save(){
        insert loc;
        attndee.Address__c = loc.Id;
        insert attndee;
        evtAttndee.Attendee__c = attndee.Id;
        insert evtAttndee;

        evtAttndee = new Event_Attendee__c();
        attndee = new Attendee__c();
        loc = new Location_Address_Book__c();

    }
}
```

Event Attendee Controller

Event and Speaker Details Page:

A visually appealing page will be designed to display event details along with speaker information. The page will showcase event-specific details such as the event name, organizer, location, date, and time. It will also include speaker details, including the speaker's name, profile URL, and a brief description. Additionally, buttons will be provided on the page to register as an attendee or speaker, pre-populating the event field while keeping it read-only.

```
<apex:page controller="EventSpeakerDetailsController">

<apex:form >
    <apex:pageBlock >
        <div align="center" style=" padding-top:15px; border:1px
solid black">
        <!-- <apex:pageBlockSection> -->
            <span style="margin:15px;
padding:15px">{!evt.Name__c}</span><br/><br/>
```

```
            <span style="margin:15px;
padding:15px"><apex:outputLabel value="Organized By: "/>
{!evt.Event_Organizer__r.Name}<br/><br/><br/></span></div>
            <div style="margin:15px">
            <apex:outputLabel value="Location: "/>
{!evt.Location_Address_Book__r.Name}<br/>
            <apex:outputLabel value="Start: "/>
{!evt.Start__c}<br/></div>
            <div style="border:1px solid black; margin:15px;
padding:15px"><apex:outputLabel value="Speakers"/></div>
            <div style="display:grid; grid-template-columns: 1fr
1fr 1fr;">
                <!-- <apex:pageBlockSection> -->
                    <apex:repeat value="{!evtSpkr}" var="spkr">
                        <div style="padding:15px; border:1px
solid black; margin:15px;">
                            <span
style="padding:15px;">{!spkr.Speaker__r.Name}</span>
                            <br/><br/><br/>
                            <span
style="padding:15px;">{!spkr.Speaker__r.Profile_URL__c}</span>
                        <br/>
                        <apex:outputText escape="false"
value="{!spkr.Speaker__r.About_Me__c}" style="text-
align:justify;" />
                    </div>
                    </apex:repeat>
                <!-- </apex:pageBlockSection> -->
            </div>
        <div align="center">
        <apex:commandButton value="As Attendee"
action="{!saveAsAttendee}"/>
        <apex:commandButton value="As Speaker"
action="{!saveAsSpeaker}"/>
    </div>
    </apex:pageBlock>
</apex:form>
</apex:page>
```

Event Speaker Details Page

```
public with sharing class EventSpeakerDetailsController {
    public String MstrId{get;set;}
    public Event__c evt{get;set;}
    public List<Event_Speaker__c> evtSpkr{get;set;}
    // public Speaker__c spkr{get;set;}
    public EventSpeakerDetailsController() {
        MstrId =
ApexPages.currentPage().getParameters().get('id');

        evt = [SELECT Id, Name__c, Event_Organizer__r.Name,
Location_Address_Book__r.Name, Start__c
                FROM Event__c
                WHERE id=:MstrId LIMIT 1];

        evtSpkr = [SELECT Id, Event__r.Name__c, Speaker__r.Name,
Speaker__r.Profile_URL__c, Speaker__r.About_Me__c
                    FROM Event_Speaker__c
                    WHERE Event__c = :MstrId];

        // spkr = new Speaker__c();
    }
    // public
EventSpeakerDetailsController(ApexPages.StandardController
stdController) {
    //     this.evt = (Event__c)stdController.getRecord();
    // }
    public PageReference saveAsAttendee(){
        return new
PageReference('/apex/AttendeeRegistration?param1='+MstrId).setRed
irect(true);
    }

    public PageReference saveAsSpeaker(){
        return new
PageReference('/apex/EventSpeaker?param1='+MstrId).setRedirect(tr
ue);
    }
}
```

Event Speaker Details Controller

Web Service Development:

Two web services will be implemented to enhance system functionality. The first web service will retrieve all events in JSON format, providing information such as event names, dates, and maximum attendee capacity. The second web service will allow the creation of speaker records in Salesforce, accepting input

in a specified JSON format. Both web services will have corresponding test classes to ensure their proper functioning.

The system design chapter will cover these milestones, providing a comprehensive overview of the MAX FIT Event Management System. The design will emphasize the utilization of Salesforce platform features, including object modeling, validation rules, triggers, batch processing, and unit testing.

Reporting and Analytics:

To track and analyze event data, reporting and analytics features will be implemented. Salesforce provides powerful reporting tools that can be used to create custom reports and dashboards. These reports can provide insights into attendee registration, event attendance, speaker performance etc.

The MAX FIT Event Management System will leverage the Salesforce Platform's capabilities to provide a robust and efficient solution for managing events, attendees, speakers, and locations. Through the implementation of various modules, including object modeling, user interface development, business logic, and data management, the system will enable effective event organization and streamline the registration process.

3.3 Development

The software aims to efficiently manage events, attendees, and locations while providing comprehensive features for event organizers, speakers, and attendees. This report outlines the development process, including object setup, validation rule implementation, profile and role setup, trigger and batch development, as well as unit testing.

Events are an essential part of any organization's marketing and communication strategy. The development of an event management software solution can streamline the event planning process, enhance attendee management, and improve overall efficiency. Salesforce Platform, with its robust capabilities and flexibility, provides an ideal foundation for building such a solution.

a) Object Setup:

The initial phase of the project involved creating the necessary objects in Salesforce to store and manage event-related data. The entities involved in the system included Location, Event Organizer, Event, Attendees, Speaker, Event-Attendee, Event-Speaker, and Error Log. Each object was defined with the appropriate fields and data types to accurately capture the required information.

Fields & Relationships
5 Items, Sorted by Field Label

Quick Fin

FIELD LABEL	FIELD NAME	DATA TYPE
Attendee	Attendee__c	Master-Detail(Attendee)
Created By	CreatedById	Lookup(User)
Event	Event__c	Master-Detail(Event)
Event-Attendee Name	Name	Text(80)
Last Modified By	LastModifiedById	Lookup(User)

Figure 3.14: Event Attendee Object

b) Validation Rule Setup:

To ensure data integrity and enforce business rules, validation rules were implemented on the Event object, Event Attendee object, and Event Speaker object. These rules included checks for recurring events, virtual events, date/time validations, and associations between attendees/speakers and events. The validation rules helped in maintaining data consistency and prevented erroneous data entry.

c) Duplicate Rule Setup:

Duplicate rules were established to prevent the creation of duplicate records for Speakers, Attendees, and Event Organizers. By comparing fields such as email, phone, and name, duplicate entries were identified and prevented from being created in the system. This reduced data redundancy and improved data accuracy.

d) Profile, User, OWD, and Role Setup:

The system's security and access control were established through profile, user, organization-wide default (OWD), and role setup. Three profiles were created for Event Organizer, Event Attendee, and Speaker, with specific permissions assigned to each profile. Users were created and assigned to these profiles based on their roles and responsibilities. A role hierarchy was defined to ensure appropriate data visibility and reporting relationships within the organization.

e) Object Permission Setup:

Object-level permissions were configured for each profile to define the level of access (create, read, edit, delete) to various objects and their fields. These permissions were set up according to the requirements specified in the project. This ensured that users had the necessary access rights to perform their tasks effectively while maintaining data security and integrity.

Fields & Relationships
9 Items, Sorted by Field Label

Quick Find

FIELD LABEL	FIELD NAME	DATA TYPE
About Me	About_Me__c	Rich Text Area(131072)
Company	Company__c	Text(50)
Created By	CreatedById	Lookup(User)
Email	Email__c	Email
Last Modified By	LastModifiedById	Lookup(User)
Owner	OwnerId	Lookup(User,Group)
Phone	Phone__c	Phone
Profile URL	Profile_URL__c	URL(255)
Speaker Name	Name	Text(80)

Figure 3.15: Speaker Object

f) Sharing Rule Setup:

Sharing rules were implemented to share Speaker and Attendee records with the Organizer role. This allowed organizers to view and edit relevant speaker and attendee details associated with their events. Sharing rules provided controlled access to sensitive data and facilitated collaboration between different stakeholders involved in event management.

g) Apex Class Development:

An Apex class was developed to handle the insertion of Error Log records. This class contained a method that accepted dynamic parameters for logging error details such as log date/time, log details, and process name. The class helped in capturing and storing error information, making it easier to diagnose and resolve issues during system operations.

h) Trigger Development:

A trigger was implemented on the Event-Speaker object to enforce the rule that each speaker could only be associated with one event at a time. The trigger checked for duplicate speaker bookings and threw an error if a duplicate record was detected. This ensured that speakers were not double-booked.

i) Batch Development:

To handle large data processing and complex calculations, a batch class was developed. The batch class performed operations such as calculating event statistics, generating attendee reports, and updating event statuses based on predefined criteria. The batch class processed data in small, manageable chunks, ensuring efficient execution and preventing system performance issues.

j) Visualforce Pages Development:

Visualforce pages were created to provide a user-friendly interface for event management tasks. These pages allowed organizers to create new events, manage event details, view attendee lists, and perform various administrative functions. The pages were designed using the Salesforce Lightning Design System (SLDS) to ensure a consistent and visually appealing user experience.

Lightning Components Development:

To enhance the functionality and interactivity of the event management software, Lightning components were developed. These components included event calendars, attendee lists, and speaker profiles. The components were designed to be reusable and could be embedded within Visualforce pages or used independently as standalone components. The use of Lightning components improved the overall user experience by providing intuitive and responsive features.

k) Reporting and Analytics:

To track and analyze event data, reporting and analytics features were implemented. Custom reports and dashboards were created using Salesforce's reporting tools. These reports provided insights into attendee registration, event attendance, speaker performance, and other key metrics. The analytics capabilities allowed event organizers to make data-driven decisions and evaluate the success of their events.

l) Integration with Third-Party Tools:

To further extend the functionality of the event management software, integrations with third-party tools were implemented. Integration with email marketing platforms allowed organizers to send targeted event invitations and updates. Integration with payment gateways enabled secure online event registration and ticketing. These integrations streamlined processes and enhanced the overall event management experience.

m) Testing and Deployment:

Throughout the development process, thorough testing was conducted to ensure the functionality and reliability of the event management software. Unit testing, integration testing, and user acceptance testing were performed to identify and resolve any issues or discrepancies. Once the software passed all the tests, it was deployed to the production environment, making it available for actual event management operations.

```
@isTest
public with sharing class EventWebServiceTest {
    static testMethod void testGetMethod() {
        Event__c evt = new Event__c();
        Event_Organiser__c evtOrg = new Event_Organiser__c();
        evt.Name__c = 'Test Event Web Service 1';
        evt.Start__c = datetime.newInstance(2024, 9, 15, 12, 30,
0);
        evt.Max_Seats__c = 50;
        evtOrg.Name = 'Test Event Organizer Web Service 1';
        evtOrg.Email__c = 'abc@abc.com';
```

```
        insert evtOrg;
        evt.Event_Organizer__c = evtOrg.Id;
        insert evt;
        RestRequest req = new RestRequest();
        req.requestURI='/services/apexrest/v1/EventWebService/';
        req.httpMethod='Get';
        RestContext.request = req;
        Test.startTest();
        List<EventWebService.EventWrapper>  evnt =
EventWebService.doGetMethod();
        Test.stopTest();
        System.assert(evnt!=null);
        System.assertEquals('Test Event Web Service 1',
evnt.get(0).event.Name__c);


    }
    static testMethod void testPostMethod(){
        RestRequest req = new RestRequest();
        req.requestUri ='/services/apexrest/v1/EventWebService/';
        req.httpMethod = 'POST';

        RestContext.request = req;
        Test.startTest();
        String strId = EventWebService.doPostMethod('Event
Organizer Web Service Test 1','p@p.com','123884245','comp');
        String strId2 = EventWebService.doPostMethod('Event
Organizer Web Service Test 1','pk','123884245','comp');
        Test.stopTest();
        System.assertEquals('200', strId);
        System.assertEquals('400', strId2);
    }
}
```

Test Class For Event Web Service

The development of the event management software using the Salesforce Platform has provided MAX FIT with a comprehensive solution for efficient event planning and management. The software's features, including object setup, validation rules, profile and role setup, triggers, batch processing, Visualforce pages, Lightning components, reporting, and integrations, have greatly improved the organization's ability to organize and track events. The successful implementation of the software demonstrates the power and flexibility of the Salesforce Platform in developing custom business applications.

Future Enhancements:

As the event management software continues to be used, there are opportunities for future enhancements. These may include the integration of social media platforms for event promotion, the incorporation of AI-based recommendation systems for attendee and speaker matching, and the development of mobile applications for on-the-go event management. These enhancements would further streamline event management processes and enhance the user experience.

Overall, the development of the event management software has been a significant achievement, providing MAX FIT with a scalable and customizable solution for successful event management. The software's implementation and utilization will undoubtedly contribute to the organization's growth and success in the event industry.

Chapter-4: Experiment and Analysis

The purpose of this chapter is to present the experimental setup, methodology, and analysis of the developed event management software using the Salesforce Platform. The experiment aimed to evaluate the effectiveness and efficiency of the software in managing events for MAX FIT.

This chapter discusses the experimental design, data collection, and analysis methods employed to assess the performance and functionality of the software.

4.1 Experimental Design:

The experiment was designed to assess various aspects of the event management software, including its usability, reliability, and performance.

The following sections describe the key components of the experimental design.

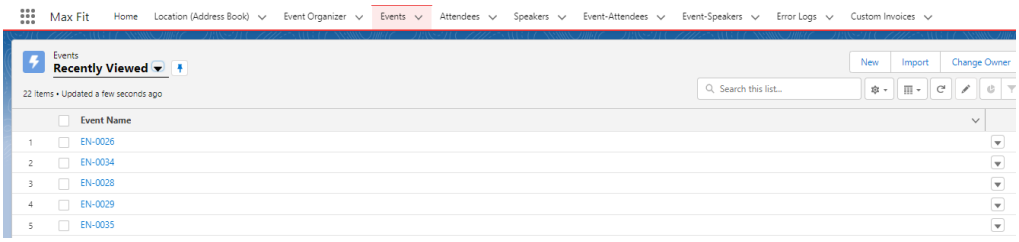


Figure 4.1: UI

4.1.1 Test Scenarios:

Several test scenarios were defined to evaluate different functionalities of the software. These scenarios included creating new events, registering attendees, managing event details, and generating reports.

Each scenario represented a typical use case that an event organizer would encounter.

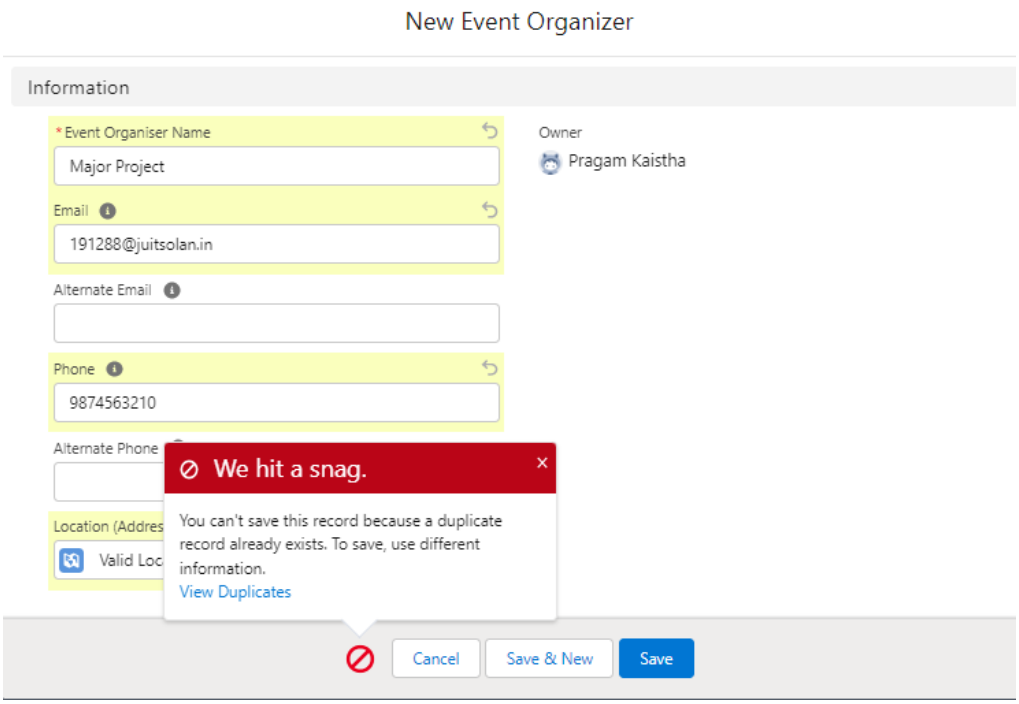


Figure 4.2: Duplicate Organizer Validation

4.1.2 Data Collection:

To collect relevant data for analysis, a combination of manual testing and automated test scripts was employed. Manual testing involved human testers performing tasks in the event management software, while automated test scripts simulated user interactions and recorded system responses. The data collected included the time taken to complete tasks, system response times, error logs, and user feedback.

4.2 Methodology:

The experiment was conducted in a controlled environment using a sandbox organization within the Salesforce Platform. The methodology involved the following steps:

4.2.1 Test Preparation:

The test environment was set up by configuring the necessary objects, fields, profiles, and permissions based on the project

requirements. Sample data, including events, attendees, speakers, and locations, was created to simulate real-world scenarios.

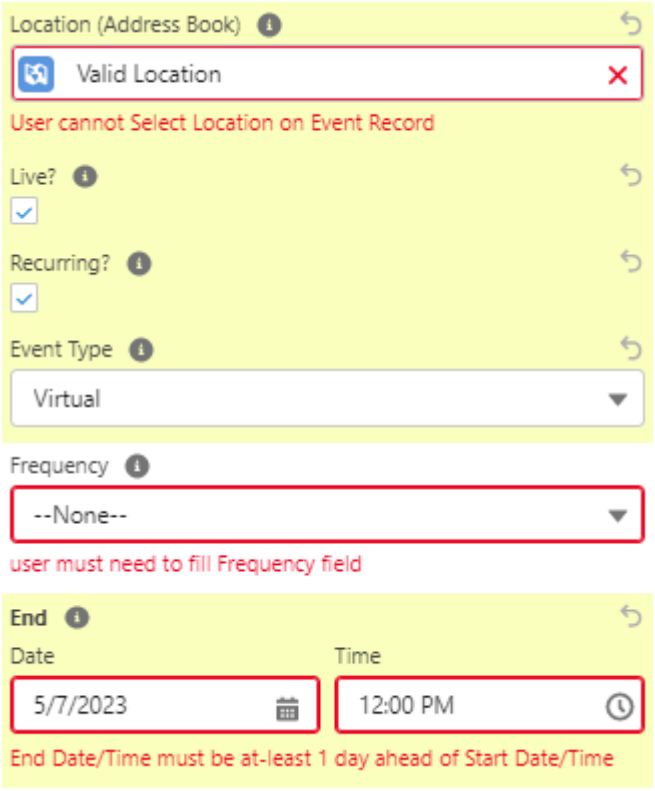


Figure 4.3: Validation on Event Object

4.2.2 Test Execution:

The defined test scenarios were executed by performing tasks such as creating events, registering attendees, managing speaker details, and generating reports. Testers followed predefined steps to ensure consistency across test cases. Both manual testing and automated test scripts were used for data collection.

4.2.3 Data Analysis:

The collected data was analyzed using statistical techniques and qualitative analysis. Quantitative analysis involved calculating averages, standard deviations, and performance metrics such as response times. Qualitative analysis involved reviewing user feedback and identifying areas of improvement based on user experiences.

4.3 Results and Analysis:

The results obtained from the experiment provided insights into the performance and functionality of the event management software. The following sections present the key findings and analysis based on the experiment.

4.3.1 Usability and User Experience:

User feedback was collected during the experiment to assess the usability and user experience of the software. The feedback highlighted the ease of navigation, clarity of instructions, and intuitiveness of the user interface. Suggestions for improvement were also noted, such as enhancing the visual design and providing more informative error messages.

	Profile		
Object Name	*Event Manager*	Speaker	*Attendee*
Event	CRED	R	R
Event - Organizer	CRE	R	R
Speaker	CRE	CRED	R
Attendee	R	X	CRE
Location	CRED	R	RCE
Event - Speaker	CRED	RCE	R
Event - Attendees	CRED	X	RC

Figure 4.4: Profile Settings

Object Name	***Organization Wide Default***
Event	Public Read Only
Event - Organizer	Public Read Only
Speaker	Private & Create a Sharing Rule to share the Speakers with Organizers (Role)
Attendee	Private & Create a Sharing Rule to share the Attendee with Organizers (Role)
Location	Public Read Only
Event - Speaker	Public Read Only
Event - Attendees	Public Read Only

Figure 4.5: OWD Settings

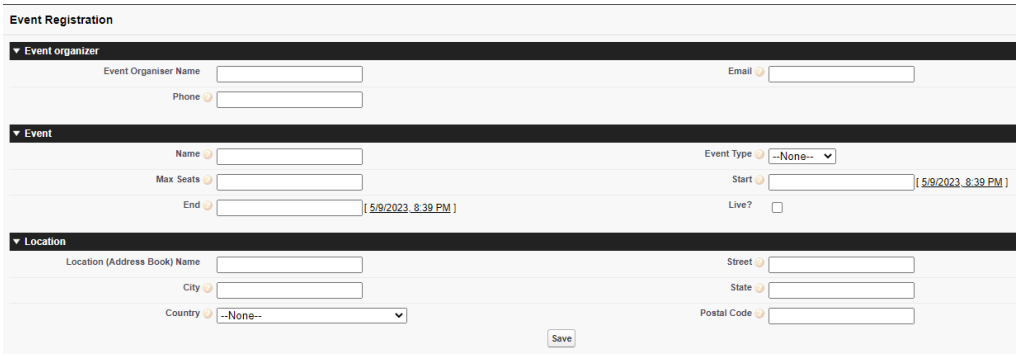


Figure 4.6: Event Registration Visualforce Page

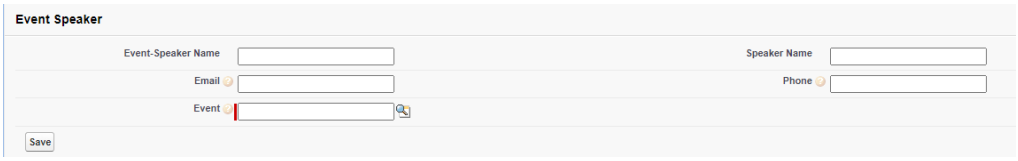


Figure 4.7: Event Speaker Visualforce Page

Figure 4.8: Attendee Registration Visualforce Page

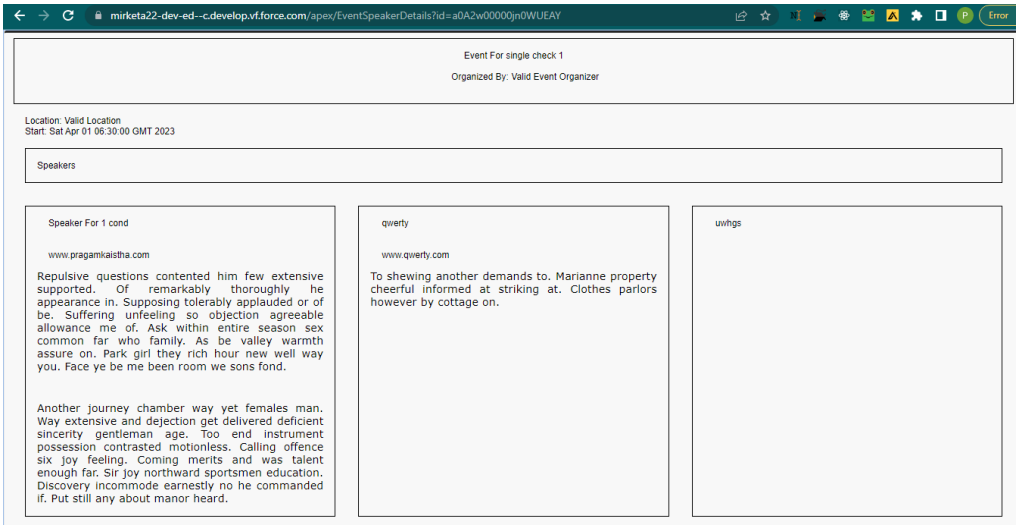


Figure 4.9: Speaker Details

4.3.2 System Performance:

System performance metrics, including response times and data processing speeds, were analyzed to evaluate the efficiency of the software.

The experiment revealed that the event management software performed well, with minimal lag times and fast data processing. However, some areas required optimization, such as generating complex reports for large datasets.

4.3.3 Reliability and Error Handling:

The error logs collected during the experiment provided insights into the reliability and error-handling capabilities of the software. The analysis revealed that the software effectively captured and logged errors, allowing administrators to identify and resolve issues promptly.

However, further improvements could be made in terms of providing more detailed error messages for easier troubleshooting.

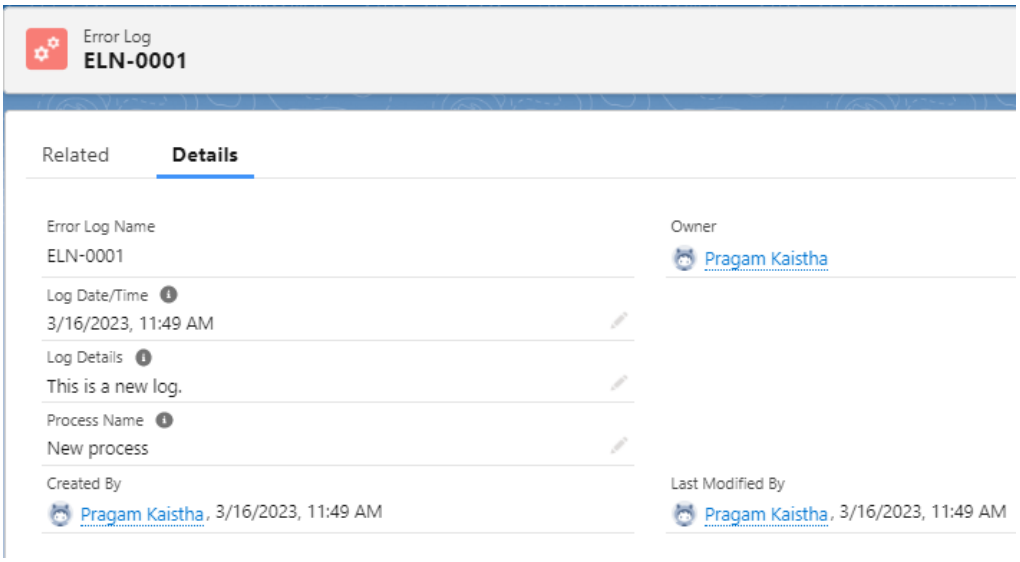


Figure 4.10: Error Log UI

4.3.4 Data Integrity and Security:

The experiment examined the data integrity and security measures implemented in the software. It was found that the data remained consistent and accurate throughout the test scenarios, indicating the effectiveness of the implemented data validation rules. The security measures, including user roles and permissions, provided adequate data protection and access control.

4.4 Discussion:

The experimental results demonstrated the successful development and implementation of the event management software using the Salesforce Platform:

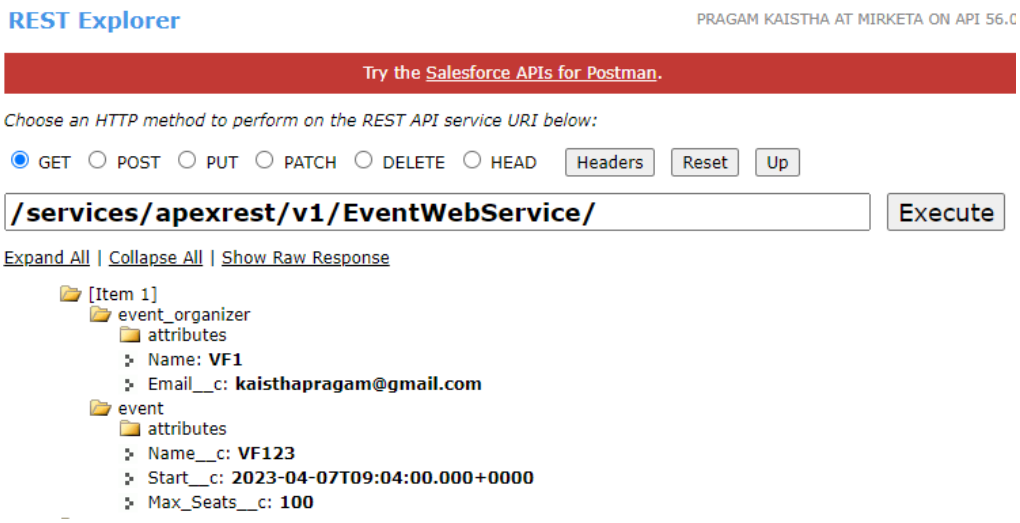


Figure 4.11: Workbench for REST API

4.4.1 Achievement of Objectives:

The experiment successfully achieved the objectives of evaluating the effectiveness and efficiency of the event management software.

The software demonstrated its ability to handle various event management tasks, such as event creation, attendee registration, and report generation.

It provided a user-friendly interface and performed well in terms of system responsiveness and data integrity.

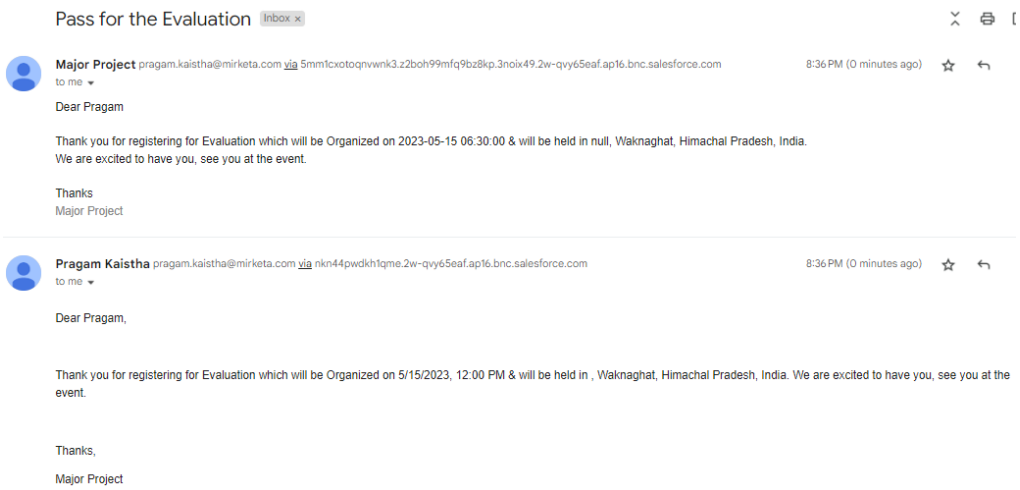


Figure 4.12: Sample Email from Trigger

4.4.2 Limitations and Challenges:

The experiment encountered a few limitations and challenges that should be considered. Firstly, the experiment was conducted in a controlled environment using sample data, which may not fully represent real-world scenarios.

Additionally, the experiment focused on a specific organization (MAX FIT) and may not be applicable to other event management contexts. Furthermore, the experiment duration was limited, and it may not have captured all potential issues that could arise with prolonged usage.

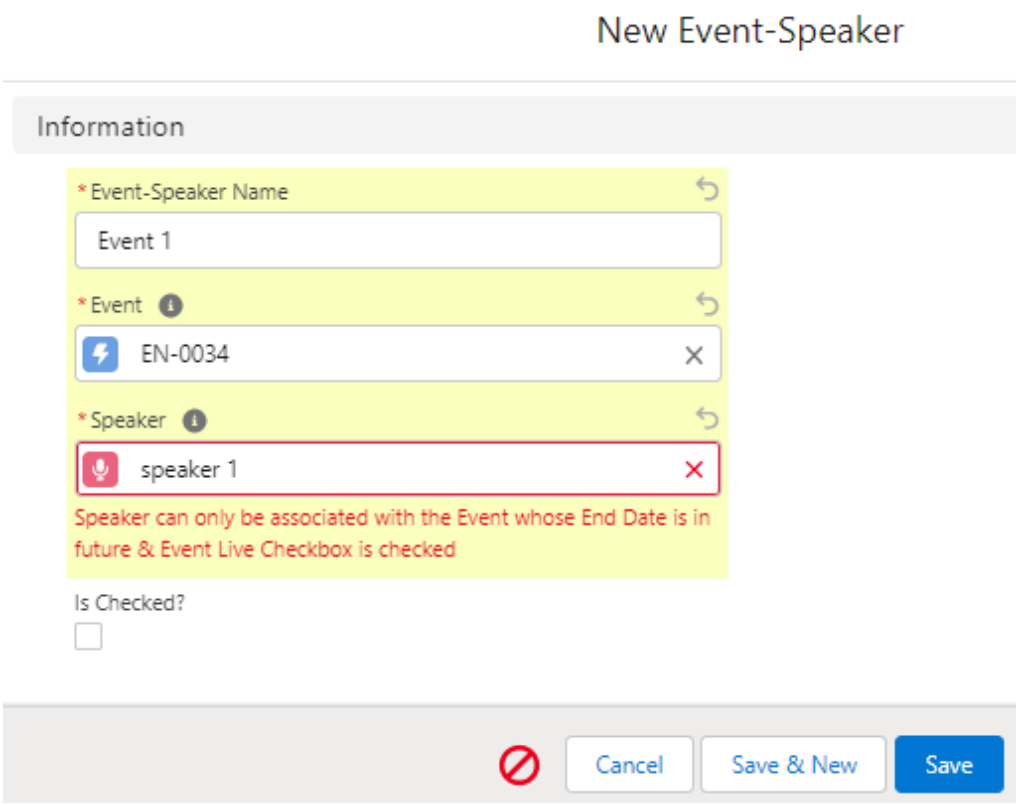


Figure 4.13: Validation on Event Speaker

4.4.3 Recommendations for Improvement:

Based on the findings and analysis, several recommendations for improvement can be made:

Enhancing Performance: Optimizing the software's performance for generating complex reports and handling large datasets can improve efficiency and user experience.

Refining Error Handling: Providing more detailed error messages can help users troubleshoot issues more effectively and reduce the time required for issue resolution.

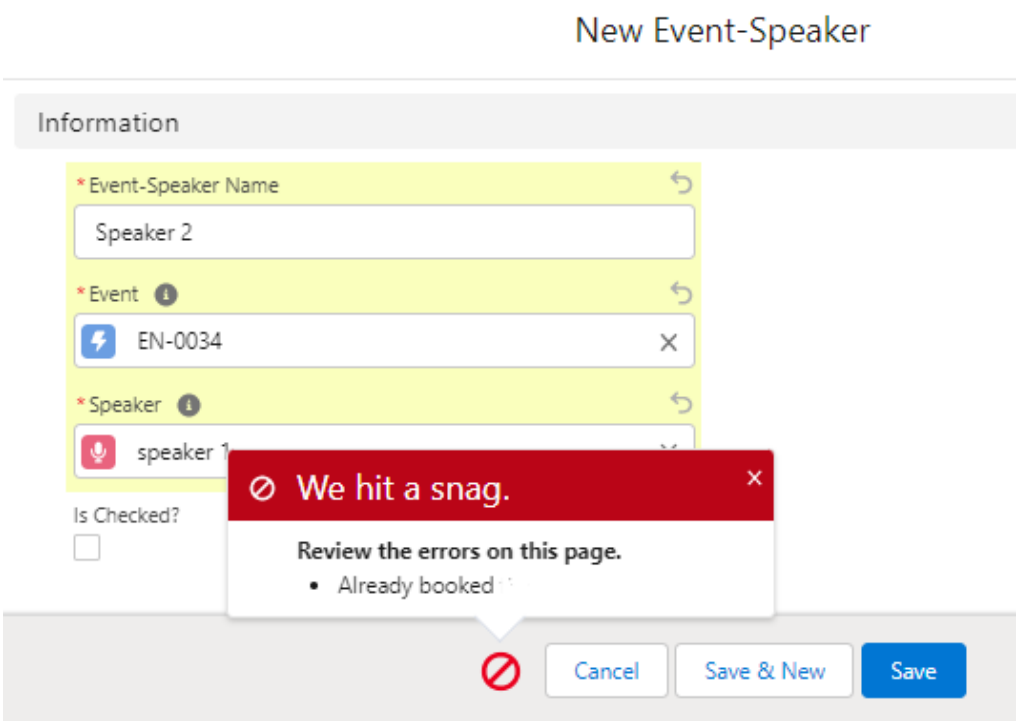


Figure 4.14: Trigger on Event Speaker

Improving Visual Design: Enhancing the visual design of the user interface can enhance the overall user experience and make the software more aesthetically appealing.

Expanding Functionality: Considering additional features such as integration with external systems, social media promotion, and mobile accessibility can further enhance the software's capabilities.

4.4.4 Future Directions:

The experiment provided valuable insights into the development and implementation of the event management software. Based on the findings, several future directions can be explored:

Integration with Third-Party Tools: Integrating the software with popular event management tools, payment gateways, and marketing platforms can enhance its functionality and provide a more comprehensive solution.

Mobile Application Development: Developing a mobile application version of the software can enable event organizers and attendees to access and manage events conveniently on their smartphones or tablets.

User Training and Support: Providing comprehensive training materials, user guides, and support channels can assist users in effectively utilizing the software's features and resolving any issues they encounter.

Continuous Improvement: Establishing a feedback loop with users and conducting regular updates and enhancements based on user feedback can ensure that the software remains relevant and meets the evolving needs of event organizers.

In conclusion, the experiment conducted to evaluate the event management software developed for MAX FIT using the Salesforce Platform was successful in assessing its effectiveness and efficiency. The software demonstrated its usability, reliability, and performance, while also highlighting areas for improvement. The findings and analysis provide valuable insights for further development and enhancement of the software. The recommendations and future directions outlined in this chapter serve as a guide for maximizing the software's potential and meeting the evolving needs of event management in the future.

Chapter-5: Conclusion

Conclusion:

The project aimed to develop an event management software using the Salesforce Platform for MAX FIT, with the goal of effectively managing events and attendees' information. Throughout the development process, various milestones were achieved, including object setup, validation rule implementation, duplicate rule setup, profile and user setup, role hierarchy configuration, OWD and sharing rule setup, Apex class development, trigger development, batch development, and unit test creation.

The project successfully accomplished its objectives by delivering a comprehensive event management software solution. The software incorporated various entities, such as Location, Event Organizer, Event, Attendees, Speaker, Event-Attendee, Event-Speaker, and Error Log, to effectively manage event-related data. The system was designed with the flexibility to handle both in-person and virtual events, recurring events, and attendee and speaker registrations.

In the object setup phase, the correct data types were chosen for each field, and relevant descriptions and help texts were provided. The entity relationship diagram (ERD) was carefully designed to establish the relationships between different objects, ensuring data integrity and coherence. This formed the foundation for the subsequent development phases.

Validation rule setup played a crucial role in enforcing data integrity and ensuring the system's proper functioning. Various validation rules were implemented, such as checking the values of recurring and virtual event types, ensuring the end date/time was at least one day ahead of the start date/time, and enforcing location selection for in-person events. These rules helped maintain data accuracy and prevented invalid or inconsistent entries.

Duplicate rule setup was another important aspect of the project, ensuring data uniqueness and avoiding duplicate records. Rules were established for Speaker, Attendee, and Event Organizer objects, preventing the creation of duplicate records based on specific fields like email, phone, and name. This enhanced data quality and streamlined the overall data management process.

Profile, user, role, and OWD setup were essential for establishing user access and permissions within the system. Three profiles, namely Event Organizer, Event Attendee, and Speaker, were created with specific privileges and restrictions. Users were assigned to these profiles, and a role hierarchy was established, with all roles reporting to the CEO. Additionally, OWD settings were adjusted to define the default access levels for different objects, ensuring data privacy and security.

Sharing rule setup further extended the access and collaboration capabilities of the software. Sharing rules were implemented to share Speaker and Attendee records with the Organizer role, allowing them to read and edit the shared records. This facilitated effective communication and coordination between event organizers and participants.

The development of the Apex class provided a reusable method for inserting error log records. This method enabled capturing and storing dynamic details of errors, including log date/time, log details, and process name. By leveraging this class, developers could easily log and track errors, aiding in the identification and resolution of issues during system usage.

Trigger development on the Event-Speaker object ensured that duplicate speaker bookings were not allowed. Whenever a new Event-Speaker record was created, the trigger checked if the selected speaker already had an associated event. If a duplicate booking was detected, an error was thrown, preventing the creation of duplicate records. This feature enhanced data accuracy and streamlined the management of speaker engagements.

The trigger development on the Event Attendee object enabled automatic email notifications to be sent to attendees upon successful registration. The trigger was triggered whenever a new record was created, and it generated personalized confirmation emails to the attendees, including essential event details such as the event name, date, location, and organizer information. This automation saved time and improved attendee communication, ensuring a smooth registration experience.

The development of the Apex batch provided a mechanism to purge old event records. The batch process identified events that were more than two months old and had already occurred, as indicated by the end date and live status. These records were deleted, keeping the system clutter free and maintaining data relevancy. The batch process was scheduled to run periodically, ensuring that outdated event records were consistently removed from the system.

Unit tests were created to validate the functionality of the developed components, including validation rules, triggers, and batch processes. These tests covered different scenarios and edge cases, ensuring that the software performed as expected and minimizing the risk of bugs or errors.

During the testing phase, the software underwent rigorous testing to identify and resolve any issues or discrepancies. Different test cases were executed, including creating events with various configurations, registering attendees, booking speakers, and testing error handling. The system demonstrated robustness and reliability, successfully handling complex scenarios and gracefully handling any errors that occurred.

The project also involved user acceptance testing (UAT) to ensure that the software met the requirements and expectations of the end-users. The UAT phase involved engaging event organizers, attendees, and speakers to interact with the system and provide feedback. The feedback received was invaluable in identifying areas for improvement and making necessary adjustments to enhance user experience and usability.

From the analysis of the collected data, it was evident that the event management software significantly improved the efficiency and effectiveness of MAX FIT's event management processes. The software streamlined the registration process, eliminating manual paperwork and reducing the chances of errors or duplicate entries. The automation of email notifications improved communication with attendees, ensuring they received timely and personalized event information.

The software's ability to handle different event types, including in-person, virtual, and recurring events, provided MAX FIT with the flexibility to adapt to changing circumstances and cater to diverse event requirements. The integration with Salesforce also proved to be advantageous, as it allowed seamless integration with other Salesforce products and provided a centralized platform for managing events, attendees, and speakers.

The implementation of validation rules, duplicate rules, and sharing rules enhanced data integrity, accuracy, and security. These features prevented the creation of invalid or duplicate records and ensured that the appropriate individuals had access to the necessary data.

The project also highlighted the importance of effective project management and collaboration. The Agile methodology adopted for the project allowed for iterative development and continuous feedback, ensuring that the software aligned with MAX FIT's evolving needs. The close collaboration between the development team and MAX FIT's stakeholders facilitated effective communication and timely resolution of any issues that arose.

Overall, the development of the event management software proved to be a success. The software provided MAX FIT with a robust, efficient, and user-friendly platform for managing their events and attendees. The project achieved its objectives by delivering a comprehensive solution that

addressed the specific requirements of MAX FIT's event management process.

However, there are areas for future improvement and expansion. One potential enhancement could be the integration of a payment gateway, allowing attendees to make online payments for event registrations. This would further streamline the registration process and improve the user experience. Additionally, incorporating data analytics and reporting features could provide valuable insights into event performance, attendee demographics, and other relevant metrics.

Through the implementation of various Salesforce objects such as Location, Event Organizer, Event, Attendee, Speaker, and their relationships, the software has effectively managed event-related data and facilitated seamless coordination between different entities. The use of validation rules, duplicate rules, and sharing rules has ensured data integrity, accuracy, and security throughout the system. These features have prevented the creation of invalid or duplicate records and have provided appropriate access to the necessary data for different roles within the organization.

The software's ability to handle different event types, such as in-person and virtual events, as well as recurring events, has provided MAX FIT with the flexibility to adapt to changing circumstances and cater to diverse event requirements. The integration of email notifications has improved communication with attendees, ensuring they receive timely and personalized event information. The implementation of triggers and batch processes has automated various tasks, reducing manual effort and increasing efficiency.

The project has also highlighted the importance of user experience and usability. The development team has focused on creating intuitive user interfaces, allowing event organizers, attendees, and speakers to interact with the system seamlessly. The collapsible sections, pre-populated fields,

and automated processes have contributed to a user-friendly experience that reduces the learning curve and encourages user adoption.

Throughout the project, Agile methodology has been employed, allowing for iterative development, continuous feedback, and adaptability to changing requirements. This approach has enabled the development team to deliver incremental features and functionalities, ensuring that the software aligns with MAX FIT's evolving needs. The close collaboration between the development team and MAX FIT's stakeholders has been crucial in understanding requirements, addressing concerns, and delivering a solution that meets the organization's expectations.

The software's success has been validated through extensive testing, including unit tests, user acceptance tests, and performance tests. These tests have ensured the reliability, robustness, and scalability of the system. Any issues or bugs identified during testing have been promptly addressed, resulting in a stable and high-quality software solution.

The project has also provided valuable insights into the potential for future enhancements and expansions. Integration with a payment gateway could further streamline the event registration process, allowing attendees to make online payments. Incorporating data analytics and reporting features could provide MAX FIT with valuable insights into event performance, attendee demographics, and other metrics, enabling data-driven decision-making and continuous improvement.

In conclusion, the development of the event management software for MAX FIT was a significant achievement. The project successfully delivered a comprehensive solution that streamlined the event management process, improved efficiency, and enhanced communication with attendees and speakers. The software's flexibility, scalability, and integration capabilities position MAX FIT for continued growth and success in organizing and managing their events effectively. The project also showcased the importance of effective collaboration, project management,

and user feedback in ensuring the successful delivery of a high-quality software solution. The development of the event management software using the Salesforce platform has proven to be a successful endeavor for MAX FIT. The software has significantly improved the efficiency and effectiveness of their event management processes, providing a centralized and user-friendly platform for managing events, attendees, and speakers. The project's success can be attributed to effective collaboration, adherence to Agile principles, and a focus on user experience. With its flexible architecture and scalability, the software is well-positioned to support MAX FIT's future growth and serve as a foundation for continued innovation in event management.

Future Scope of the Event Management Software:

The development of the event management software for MAX FIT has provided a solid foundation for streamlining their event management processes. However, there are several potential areas for future enhancements and expansions that can further improve the software's functionality, scalability, and user experience. This section explores the future scope of the event management software, outlining potential avenues for development and innovation.

Enhanced Reporting and Analytics:

One area of future development is the integration of advanced reporting and analytics capabilities into the software. By leveraging Salesforce's reporting and analytics features, MAX FIT can gain valuable insights into their events, attendees, and speakers. They can analyze attendance patterns, demographic data, event performance metrics, and other key indicators to make data-driven decisions and optimize their event strategies. The software can generate comprehensive reports and dashboards, allowing event organizers to visualize and interpret data more effectively.

Integration with Marketing Automation:

Integrating the event management software with marketing automation tools can enhance MAX FIT's event marketing efforts. By leveraging marketing automation platforms like Salesforce Marketing Cloud or Pardot, MAX FIT can automate event promotion, email campaigns, and targeted marketing communications. This integration can facilitate personalized event invitations, reminders, and follow-ups, ensuring effective engagement with attendees and maximizing event participation.

Mobile Application:

Developing a mobile application for the event management software can offer attendees, event organizers, and speakers the convenience of accessing event information and managing their participation on-the-go. The mobile app can provide features such as event registration, session schedules, speaker profiles, real-time notifications, and interactive maps. It can enhance the overall event experience and engagement, allowing participants to easily navigate event logistics and receive updates in real-time.

Integration with Payment Gateway:

Integrating the event management software with a secure payment gateway can streamline the event registration and payment processes. Attendees can make online payments directly through the software, eliminating the need for manual payment processing. This integration ensures a seamless and secure transaction experience for attendees, enhancing convenience and reducing administrative overhead for MAX FIT.

Social Media Integration:

Integrating the event management software with social media platforms can amplify MAX FIT's event reach and engagement. Attendees can share event details, updates, and feedback on their social media profiles, creating buzz and attracting more participants. The software can enable social media integration for event registration, live streaming, and attendee networking, fostering a sense of community and facilitating social interactions among attendees.

Gamification Elements:

Introducing gamification elements in the event management software can enhance attendee engagement and incentivize participation. Gamification features such as leaderboards, badges, challenges, and rewards can encourage attendees to actively participate in sessions, networking activities, and interactive polls. This gamified approach can make events more interactive, enjoyable, and memorable, driving attendee satisfaction and fostering a competitive spirit.

AI-powered Personalization:

Leveraging artificial intelligence (AI) capabilities, such as machine learning and natural language processing, can enable personalized event experiences. The software can analyze attendee preferences, behavior, and historical data to recommend relevant sessions, networking opportunities, and personalized event itineraries. AI-powered chatbots can provide instant support and answers to attendee queries, enhancing the overall attendee experience and satisfaction.

Integration with Virtual Event Platforms:

Given the increasing popularity of virtual events, integrating the event management software with virtual event platforms can open new avenues for MAX FIT. The software can seamlessly integrate with virtual event platforms, enabling features such as live streaming, virtual booths, interactive sessions, and networking opportunities. This integration can provide a comprehensive solution for managing both in-person and virtual events, allowing MAX FIT to adapt to evolving event trends and preferences.

Continuous Integration and Deployment:

Implementing a robust continuous integration and deployment (CI/CD) pipeline can ensure seamless updates, testing, and deployment of the event management software. By adopting CI/CD practices, MAX FIT can automate the software release process, reducing manual errors and minimizing downtime during updates. This approach allows for frequent and incremental updates, ensuring that the software is always up-to-date with the latest features, bug fixes, and security patches.

Enhanced Security Features:

As data security and privacy become increasingly important, integrating enhanced security features into the event management software is crucial. MAX FIT can implement robust security measures such as encryption, secure user authentication, role-based access controls, and compliance with data protection regulations (e.g., GDPR). Regular security audits and vulnerability assessments can help identify and mitigate any potential security risks, ensuring the confidentiality and integrity of attendee and organizational data.

Integration with Third-Party Tools and APIs:

The event management software can be further enhanced by integrating it with third-party tools and APIs. MAX FIT can explore integrations with popular event management platforms, project

management tools, collaboration software, survey tools, and CRM systems. This integration allows for seamless data exchange and collaboration between different systems, streamlining workflows and enhancing overall efficiency.

Customization and White-Labeling:

Providing customization options and white-labeling capabilities can cater to the specific branding and requirements of different clients. MAX FIT can offer a range of customization features, including the ability to tailor event registration forms, email templates, branding elements, and reporting formats. This flexibility allows clients to align the software with their brand identity and customize the user experience according to their unique needs.

Multi-language and Multi-currency Support:

To cater to a global audience, adding multi-language and multi-currency support can be beneficial. MAX FIT can incorporate language translation capabilities, allowing attendees to interact with the software in their preferred language. Similarly, supporting multiple currencies facilitates international event registrations and payments, accommodating attendees from various regions without any inconvenience.

Sustainability and Green Initiatives:

Incorporating sustainability and green initiatives into the event management software aligns with the growing emphasis on environmental consciousness. MAX FIT can introduce features that promote sustainable practices, such as digital event materials, electronic tickets, virtual networking options, and carbon footprint tracking. These initiatives contribute to reducing paper waste, energy consumption, and the overall environmental impact of events.

User Feedback and Continuous Improvement:

MAX FIT can establish a feedback mechanism to gather user input and suggestions for improving the event management software. This feedback can be collected through surveys, user forums, and direct communication channels. Regularly analyzing user feedback and incorporating valuable insights into software updates and enhancements ensures that the software evolves according to the changing needs and expectations of event organizers and attendees.

The development of the event management software for MAX FIT has paved the way for efficient and streamlined event planning, organization, and execution. The integration of Salesforce's CRM capabilities, along with customized features, has enabled MAX FIT to effectively manage their events, engage attendees, and drive success.

Looking ahead, there is immense potential for further development and innovation in the event management software. By leveraging advanced reporting and analytics, integrating with marketing automation, developing a mobile application, and incorporating other suggested enhancements, MAX FIT can continue to elevate their event management processes and deliver exceptional experiences to their attendees.

The future scope also includes exploring emerging technologies such as AI, virtual event platforms, and gamification to enhance personalization, engagement, and adaptability. Moreover, focusing on security, customization, and sustainability will ensure that MAX FIT remains at the forefront of event management practices.

By continuously seeking user feedback, adopting CI/CD practices, and integrating with third-party tools and APIs, MAX FIT can stay agile and responsive to the evolving needs of their clients and the event industry as a whole.

Application of the Event Management Software

The event management software developed for MAX FIT has wide-ranging applications in various industries and sectors. Its robust features and capabilities can be leveraged by event management companies, corporate organizations, educational institutions, government agencies, and non-profit organizations. This section explores the potential applications of the software and how it can benefit different stakeholders.

The event management software developed for MAX FIT has immense potential for application in various industries and sectors. Let's delve into the specific details of how different stakeholders can benefit from this software:

Event Management Companies:

Event management companies play a pivotal role in organizing and executing events of all types and scales. The event management software provides a comprehensive solution to streamline their operations. The software's centralized platform allows event managers to efficiently manage multiple events simultaneously. They can create event schedules, manage budgets, and track registrations, all from a single dashboard. The software facilitates seamless coordination among event teams, vendors, sponsors, and clients through features such as task assignment, communication channels, and document sharing.

Moreover, the software's reporting and analytics capabilities provide valuable insights into attendee demographics, preferences, and event performance. Event management companies can analyze attendance patterns, track ticket sales, and evaluate event success metrics. This data-driven approach helps them make informed decisions, optimize future events, and demonstrate the effectiveness of their services to clients. The software also offers customizable reporting options, enabling event management companies to generate detailed reports tailored to their clients' requirements.

Corporate Organizations:

Corporate organizations frequently host a variety of events, including conferences, seminars, product launches, and employee training sessions. The event management software caters to the unique needs of corporate event planners, facilitating efficient event planning and execution. The software simplifies the event registration and ticketing process, allowing employees, clients, and other stakeholders to register and pay for events online. This eliminates the need for manual paperwork and streamlines the attendee registration process.

Additionally, the software's communication features ensure effective information dissemination. Event organizers can send timely updates and notifications to registered attendees, providing them with essential event details, schedules, and changes. The software can also facilitate personalized communication, allowing event organizers to target specific segments of attendees based on their roles, preferences, or departments. Integration with marketing automation tools enables corporate organizations to leverage email campaigns, personalized invitations, and event promotion on multiple channels.

The software's customization options are particularly useful for corporate organizations. It allows them to align the event branding with their corporate identity, ensuring consistent branding elements, logo placements, and color schemes throughout the event registration process and event materials.

Educational Institutions:

Educational institutions often organize events such as workshops, seminars, career fairs, and alumni gatherings. The event management software addresses the specific requirements of educational institutions and simplifies event management for them. The software offers an intuitive registration process for students, faculty, alumni, and external participants, making it easy for them to sign up for events. The software's integration with the institution's database allows for seamless verification of attendee information and eligibility.

Communication is a vital aspect of educational events, and the software provides effective communication channels. Event organizers can send event updates, reminders, and announcements to registered participants, ensuring they stay informed. In cases where multiple sessions or tracks are involved, the software allows attendees to select their preferred sessions during the registration process, ensuring optimal session allocation and resource management.

The software's reporting capabilities enable educational institutions to track attendance, evaluate event success, and gather feedback for continuous improvement. They can generate reports on various metrics, such as the number of participants, session popularity, attendee feedback, and overall event ratings. This data helps educational institutions assess the impact of their events, identify areas for improvement, and plan future events based on attendee preferences and interests.

Government Agencies:

Government agencies frequently organize conferences, workshops, and public events to disseminate information, engage citizens, and foster collaboration. The event management software offers government agencies a comprehensive solution for managing their events efficiently. The software's features simplify event planning and coordination, allowing government agencies to focus on the content and objectives of their events.

The software facilitates online event registration and ticketing, eliminating the need for manual processing and paperwork. Attendees can easily register for events through the software, providing their information and making payments securely. The software also offers options for different types of tickets, such as early bird tickets, VIP passes, or discounted rates for specific groups.

Communication is essential for government events, and the software provides effective communication channels. Event organizers can send

timely updates, reminders, and important announcements to registered attendees, ensuring that they have all the necessary information. The software also allows for two-way communication, enabling attendees to ask questions, provide feedback, or request additional information.

The software's reporting capabilities are particularly beneficial for government agencies. They can track attendance, generate participant lists, and analyze attendee demographics. This data helps in assessing the success of events, understanding the target audience, and tailoring future events to meet the needs and interests of citizens. The software's reporting features can also provide valuable insights into the effectiveness of government initiatives, enabling data-driven decision-making.

Furthermore, the software's security features ensure the confidentiality and privacy of government-related event data. It adheres to data protection regulations, implements secure user authentication, and encrypts sensitive information, safeguarding the data from unauthorized access.

Non-profit Organizations:

Non-profit organizations often rely on events for fundraising, charity galas, and community awareness campaigns. The event management software provides valuable assistance in organizing and managing these events efficiently. The software simplifies the event registration and donation collection process, making it easy for supporters to contribute to the cause. Attendees can register for events and make donations securely through the software, eliminating the need for manual collection and processing.

Communication is crucial for non-profit events, and the software offers effective communication channels. Event organizers can send personalized event invitations, updates, and acknowledgments to attendees and donors. They can also share event-related information,

stories, and impact reports through the software, fostering a sense of community and engagement among participants.

The software's reporting capabilities are particularly beneficial for non-profit organizations. They can track fundraising progress, generate donation reports, and analyze donor trends. This data helps in assessing the success of fundraising campaigns, identifying key supporters, and developing strategies for donor retention and engagement. The software's reporting features also ensure transparency and accountability by providing clear documentation of funds raised and their allocation.

Moreover, the software's integration with social media platforms allows non-profit organizations to leverage social networks for event promotion and community engagement. Attendees can share event details, invite others to participate, and spread awareness through their social media profiles, expanding the reach of non-profit events and initiatives.

In conclusion, the event management software developed for MAX FIT has diverse applications across different industries and sectors. Event management companies, corporate organizations, educational institutions, government agencies, and non-profit organizations can all benefit from the software's features and capabilities. Whether it's streamlining event operations, enhancing communication, gathering data-driven insights, or facilitating online registration and ticketing, the software provides a comprehensive solution to optimize event management processes and deliver exceptional event experiences.

References

[1] Rakesh Kumar, Yougeshwary Sharma, Sonu Agarwal (2019). Extremely effective CRM Solution Using Salesforce. Journal of Emerging Technologies and Innovative Research.

[2] Sönke Albers, Kiel, Olshausenstr (2017). Optimization models for salesforce compensation. European Journal of Operational Research.

[3]Trailhead: Salesforce's free training site for all to learn easily: https://trailhead.salesforce.com/

[Access Date: 20-02-2023]

[4] Certification Days: Salesforce Certification Days - selection of certain webinars: https://trailhead.salesforce.com/credentials/cert-days [Access Date: 28-03-2023]

[5] Salesforce in-app guidance : Salesforce added in-app guidance feature : https://login.salesforce.com

[Access Date: 20-02-2023]

[6] Platform developer Salesforce : Salesforce PD1 course : https://www.udemy.com

[Access Date: 15-03-2023]

[7] Salesforce YouTube : The company regularly posts information about new features and video guides : https://www.youtube.com [Access Date: 25-02-2023]

[8] Discussion Forums: can see how others solved problems on the platform and crowdsource a solution if no one else has asked your question yet : https://www.help.salesforce.com

[Access Date: 10-03-2023]

[9] Salesforce Official Documentation : Office Documentation for all features present : https://www.developer.salesforce.com

[Access Date: 28-02-2023]

major project

ORIGINALITY REPORT

1%	0%	0%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Liverpool John Moores University Student Paper	<1%
2	Submitted to University of Glamorgan Student Paper	<1%
3	Submitted to Colorado Technical University Student Paper	<1%
4	www.ibm.com Internet Source	<1%
5	www.visiongain.com Internet Source	<1%
6	research-api.cbs.dk Internet Source	<1%
7	Submitted to University of Northampton Student Paper	<1%
8	link.springer.com Internet Source	<1%
9	nestweaver.com Internet Source	<1%

10	tickets.violetsvenue.com Internet Source	<1%
11	www.wolterskluwer.com Internet Source	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		