

Overview:

The client wanted to build a social networking platform to assist travelers. The idea was to highlight tourist places in Myanmar and let people share their experiences, by way of reviews and ratings, with their loved ones & others. If successful, it would lead to the development of a community of travelers, with valuable information available to assist them to explore exciting tourist places, hotels, restaurants etc.

The platform has a role-based admin portal where office staff can manage attractions, articles, hotels, restaurants etc. Business owners can also use the platform to create their business page.

The backend application exposes a REST API to the client device and browser. Backend system is setup in AWS and uses AWS's various services for hosting, media storage, database etc. We have used AWS elastic search as a search and analytics engine. There are features to enable "free-text search" and "search as you type" for users. A queue has been used for faster API response. This makes it possible for the application to run the heavy processing tasks, like image/video processing, compression, and data sync with elastic in the background. The application also exposes various third-party authentication mechanisms like Google, Facebook, and Twitter.

Client details:

Name: Confidential | **Industry:** Travel | **Location:** Myanmar

Technologies:

LAMP, Laravel, AWS, REST, Elasticsearch, GIT, Queue, Google API, Facebook API, Twitter API, Firebase, React Native, FFmpeg

Project Description:

The client sensed an opportunity to create an ecosystem amidst the tourism industry. They wanted to forge a connect between fellow tourists, the businesses that exist around popular tourist places and those that facilitated important activities pertaining to travel and tourism. They felt that all of these experiences could be seamless and a synergy could arise out of these activities; booking, traveling, getting the most out of tourist destinations

They felt that the solution lay in having a single platform that provided a one-stop solution to all the stakeholders. So the demand was to build a unified platform that had a custom search engine where businesses and users could interact with each another, and get access to relevant and up-to-date information.

The Mindfire Solution:

Mindfire Solutions developed a comprehensive platform that had 3 distinct layers.

1. Mobile Applications where users can interact
 2. Web Admin portal where business can registered themselves
 3. Backend which connected everything seamlessly
- **Mobile Application:** Our client needed the solution for Android and iOS devices. After reviewing all the available native and cross-platform technologies, we decided to proceed with making front-end applications in React Native. Since the majority of the application was based on fetching up-to-date data and rendering the UI, we were able to re-use 90% of the code base to develop the application on both platforms using RN. This gave enough time and flexibility to focus on UI/UX aspect of the application and providing and delivering a complete native-level performance while doing the majority of the code in React Native.
 - **Web Admin Portal:** This is used to manage the complete application.
 - **Backend:** The backend is completely built on LAMP stack. Laravel is used as the PHP framework for building the application. AWS EC2 instance is used for hosting the PHP server. AWS RDS-MySQL is used for data storage. The backend search engine is built on AWS Elasticsearch. The request from the client comes to AWS Elastic Load Balancer (ELB). The ELB divides the request to an EC2 instance based on the traffic at that time. We have used different DBs for the admin portal and customer portal so that the main database (for the customer portal) is not over loaded.

Various queues are leverage to different operations. In order to optimally maintain the queues, some queue workers and supervisors are also used at the backend. Whenever a user creates a post/article, the server sends a successful response to the device. After that, data is also sent to the elastic search for searching purpose. The sending of data from EC2 to the Elastic server is done with the help of a queue.

All the media (audio and video) information is stored on AWS S3 so that the main EC2 server is not overloaded. Whenever the end user uploads any file, the request goes directly to S3 for file uploading. After uploading, the AWS S3 server calls an API of the EC2 server and we maintain the file uploaded record on our DB. The client device uploads the file to S3 via the AWS post-policy mechanism. Post uploading success, a response is sent to the device, and then in the background the EC2 server compresses and generates the thumb for both audio and video files with the help of the FFmpeg package. This process is done with the help of a queue.

When a user creates a post the server sends real-time notifications to all the users in the network with the help of Firebase Cloud Messaging(FCM).

The logger is implemented so that when the application encounters any exception, the server logs the exception to the log files and triggers a mail to the developer regarding the exception details. The application also uses various third-party APIs (like Google, Facebook, Twitter etc.) for social login.

All the security related information like credentials etc. are kept on .env file and it is not pushed to Git. Various types of security attacks like (SQL Injection, XSS, and Phishing etc.) are handled properly. Rate limiting is also used on the API's request to prevent excessive requests.

Key Features

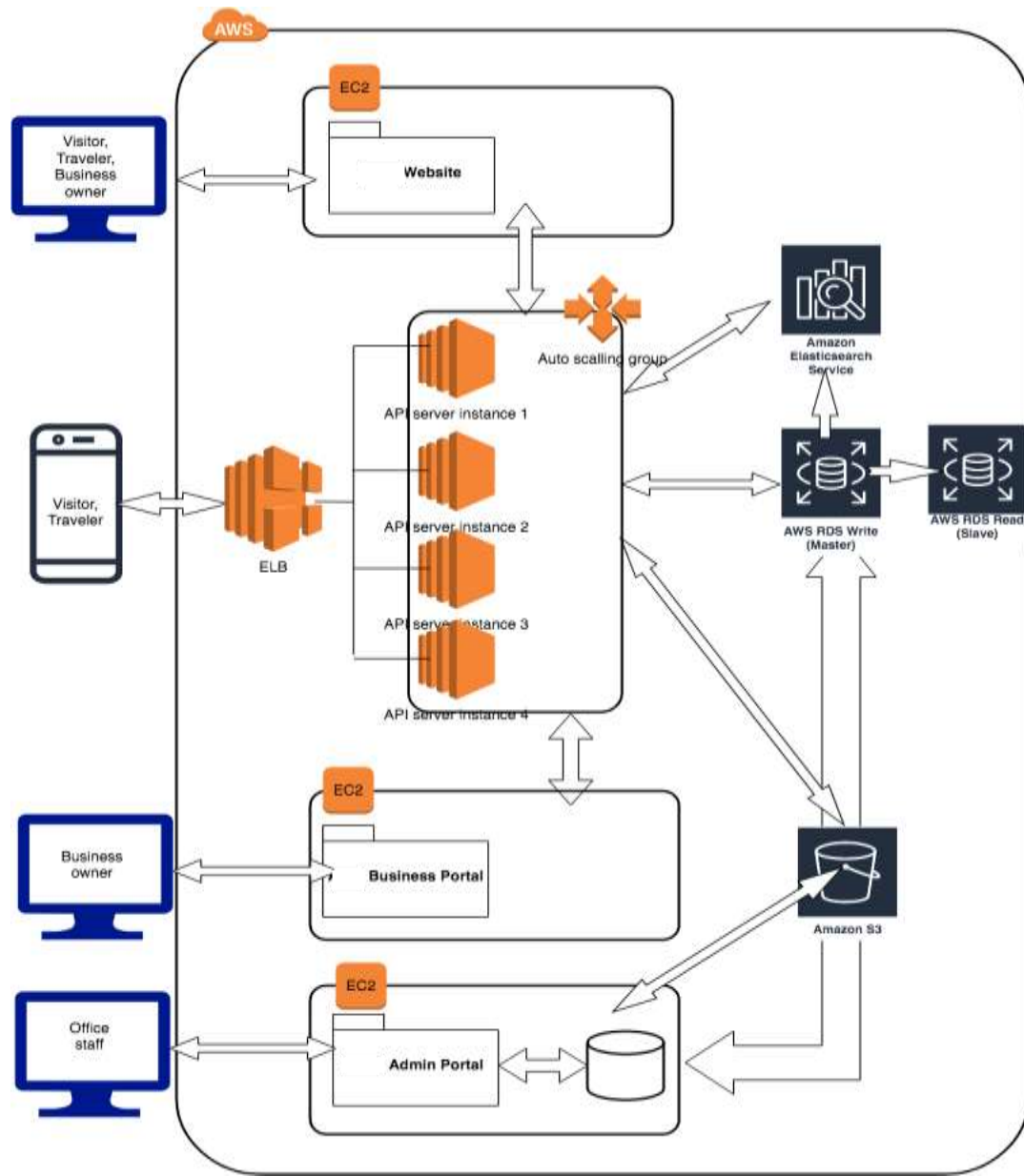
- Allow authentication of user with email as well as phone number
- Continuous update of Home feed to allow user to explore their surrounding
- A search engine to search posts/user/businesses/articles
- Find other users within certain radius using their geo location
- Full range of social networking features like create post, share, interact, comment
- Allow clients to create articles and integrate in app
- Push Notification to notify users of new changes

Collaborative Platform for Travelers

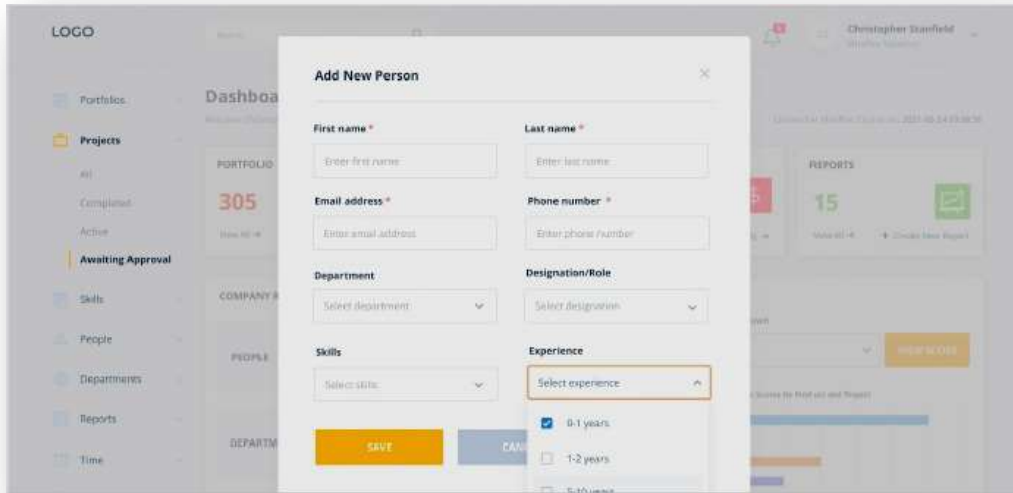


Mindfire Solutions delivered for the client the exact platform they had visualized within the scheduled time frame. All the components (i.e. Android and iOS applications, backend, and the admin portal) were developed in-house. The product is expected to be available in the market in the near future. Besides offering our ongoing support we look forward to future enhancements that may spring based on the responses that come.

Architecture:



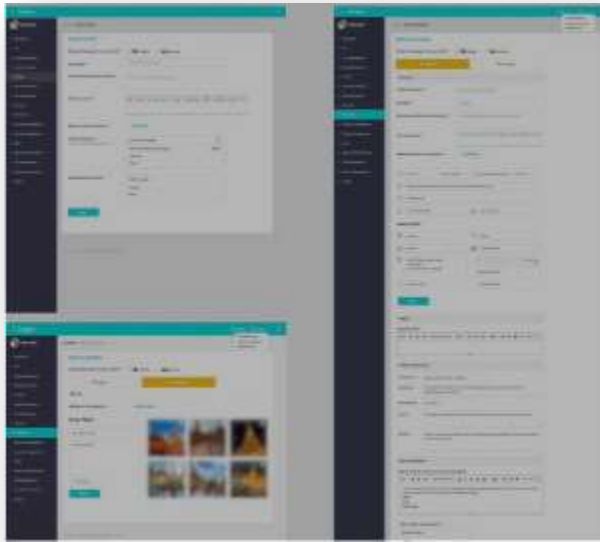
Screenshots:



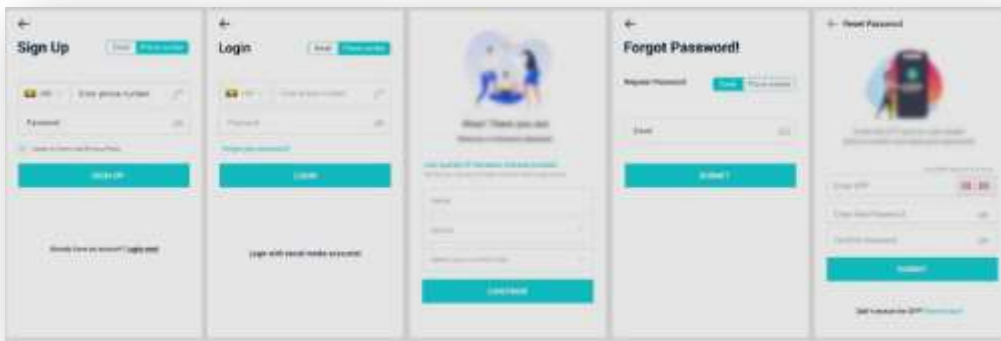
Screenshot 1: Adding of Member



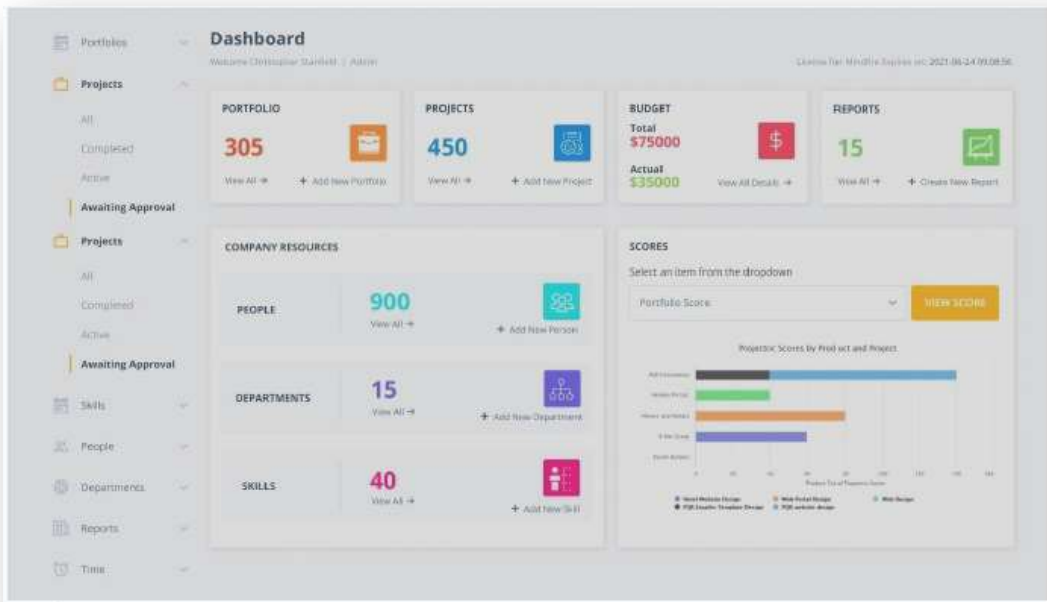
Screenshot 2: Adding of Business Information



Screenshot 3: Adding Attraction & Article Info



Screenshot 4: Mobile Screens



Screenshot 5: Dashboard