CASE STUDY

IoT framework for Real Time monitoring of Fleet

Introduction

EFMFM is an enterprise telematics end-to-end ERP product by NGFV to manage complete employee transportation process for large enterprise clients encompassing fleet management, intelligent routing, live tracking, compliance and security.

Shell Inc., a customer of NGFV user of EFMFM, have multiple locations have the need to safely move material and workforce on a daily basis. They use a fleet of owned and contracted vehicles for this purpose round the clock and hence tracking them for safe passage and timeliness of delivery is critical. With the advent of IOT devices, mobility based tracking solutions are elegant and cost effective. With growing number of enterprise customers, the team needed a scalable solution with disaster recovery option.

The application needs to be accessed any time, any where and hosting the same in a public cloud infra like AWS makes eminent business sense.

Current System overview:

It is a three-way information sharing and collaboration tool that works on real-time information and addresses key concerns of transport management i.e. employee safety, employee experience, operational efficiency, vendor data management, governance and compliance reporting plus accurate invoicing. The main constituents of the solution are administrative portal, employee mobile app and driver mobile app.

Key Business/Systems requirements:

- To Detect and provide five major parameters namely
    - Exceeding Driving Hours
    - Harsh acceleration
    - Harsh breaking
    - Sharp turn
    - Speeding
- These parameters are detected with the help of atic device that is integrated in each vehicle.
- This atic device gives the above reading every 60 seconds to the EFMFM System.
- EFMFM Sytem has scheduler which runs every 60 seconds to get the above events happened in each and every vehicle in the atic system and shows the result in Dashboard and reports.

- EFMFM system captures these readings and displays in Dashboard namely the IVMS Dashboard and as well as in reports form namely IVMS Reports.
- These Atic Systems can be installed and detected location wise, as of now it is installed in Shell Chennai and Shell Bangalore. Future planned to Shell manila as well.
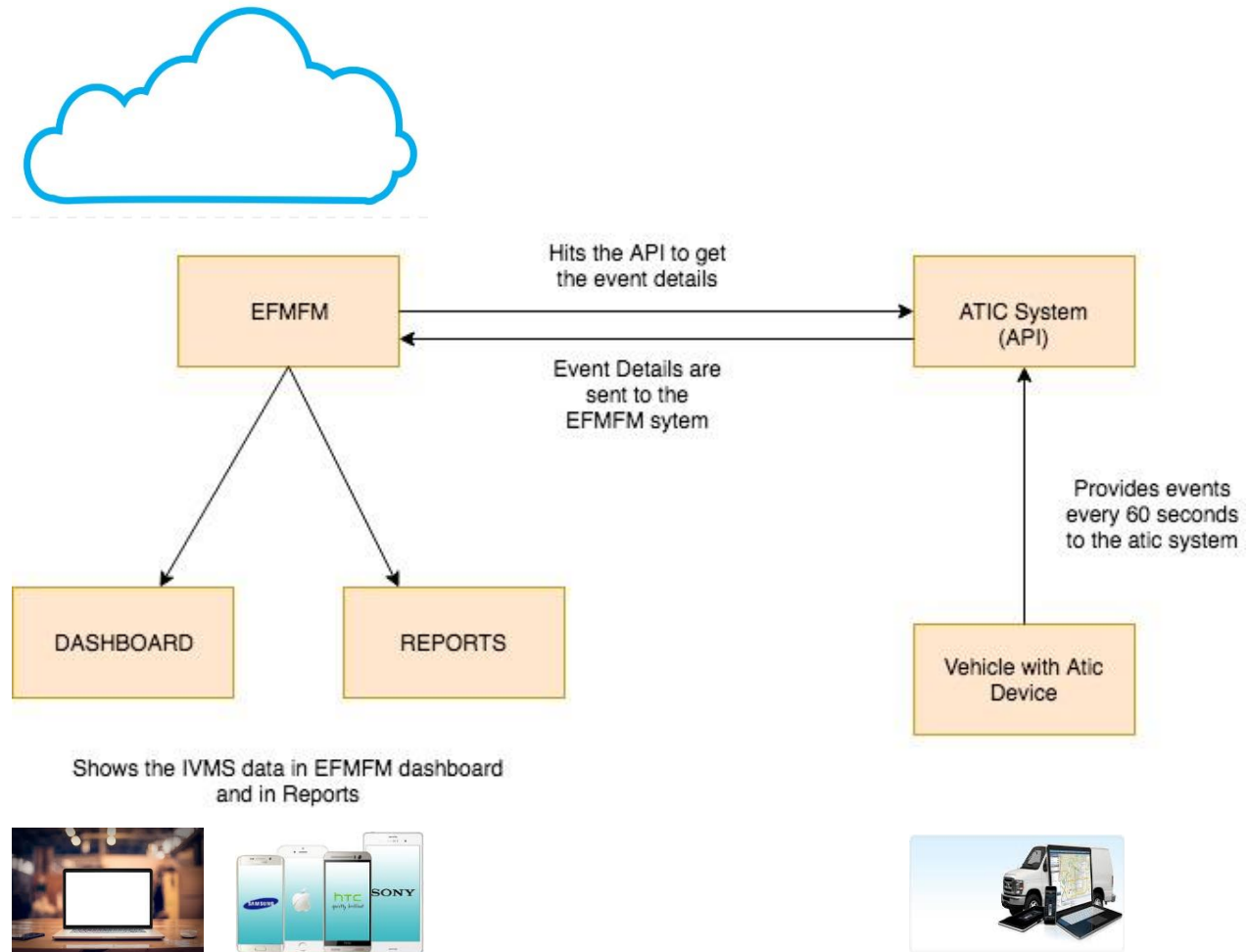
Solution Considerations

Primary objective was to integrate fleet management application to vehicle dashboard. Some of the key technical considerations were related to use a IoT framework to monitoring vehicle real time fashion.

Design Considerations:

Deployement Architecture

To build the confidence of the customer, the application was hosted in a staging environment to validate all the functionalities are fully tested with the client data set. For a period of 6 weeks this environment was used to make changes as required and get them validated with live data sets for performance.

Deployement Architecture

Hits the API to get the event details

EFMFM

Event Details are sent to the EFMFM sytem

ATIC System (API)

Provides events every 60 seconds to the atic system

DASHBOARD

REPORTS

Vehicle with Atic Device

Shows the IVMS data in EFMFM dashboard and in Reports

Implementation Phase:

Since the application is a SaaS product the initial set-up and provisioning of services is quite fast and follows a templated approach. The configuration of DB and porting of data was done in 2 weeks' time. Additional customization of UI and enhancements tool 5 weeks.

Start of the project: JAN 2018 – Completion of all sites: April 2018.

Timelines

The customer had planned a progressive roll-out plan for 2 sites with 5,000 users spread across. A round robin approach was taken to complete the different phases of the implementation.

- Freezing the customization requirements took 3 weeks.
- Setting-up the environment took 1 week
- Developing enhancements took 2 weeks
- UAT with actual data took 2 weeks
- Configuring and launching the production environment took 1 week
- Roll-out to subsequent sites took 1 week each

Team structure

The team size for the first site was 7 people (3 developers, 1 architect, 1 QA, 1 DBA and 1 implementation engineer) all of them relevant AWS certification.
The implementation team for each of the subsequent sites has two-member teams to complete the tasks.

Use of AWS tools

Git Hub, Jira, Confluence, Ansible and Jenkins were used in the process of migration extensively.
AWS services used are: EC2, S3, ELB, Cloud Watch and RDS.

Operations and Optimization
Some of the key areas for focus in the operations phase are:
- ✓ Reserved Instance usage
- ✓ Horizontal scaling based on OS resources
- ✓ AWS health-checkup to clean-up instances, AMIs, volumes, snapshots
- ✓ Schedule stop/start instances based on triggers

Security considerations and implementation
The focus on security was present right through the process of migration and the enterprise has clear guidelines around this and the project team implemented them.
- ✓ Hardened Image provided by Enterprise AWS AMI Team
- ✓ Enabled with SSO login
- ✓ VPC design, security groups, network access control were implemented with the help of IT team at the client place.
- ✓ Application specific IAM Policy which applies in EC2, S3, RDS in all resources
- ✓ TLS Enabled from application end
- ✓ Encryption of data using the proprietary tool of the client

Business benefits of the migration:
The migration project is envisaged to give the following benefits to the client in a virtual data center environment as and when the application is on AWS.
Availability: 99.99 % availability for business continuity and user satisfaction
Autoscaling: Application able to handle 5000 users at peak time
Disaster Recovery: System can be replicated and recovered quickly into AWS without disruption or data loss
Pro-active Monitoring and remediation
Centralized Login mechanism