

The logo for GadgEon, with 'Gadg' in blue and 'Eon' in orange.

**GadgEon**

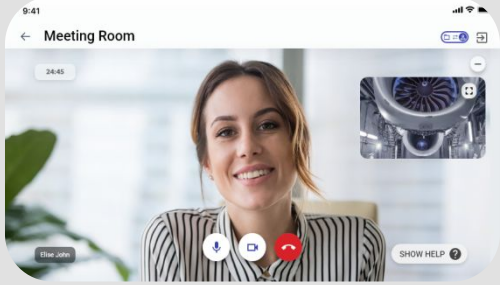
**Engineering  
Smartness**

# **MAINTENANCE PLATFORM DEVELOPMENT ON AZURE & ON-PREMISE SERVER**

June 2020  
Version 1.0



# Maintenance Platform on Azure & On-Premise Server



A startup company, specializing on products that operate on and around people wants to develop a server agnostics software platform that provides hands-free access to technical manuals and instructions for Maintenance of aircrafts and heavy equipment in the mining industry

## Solution Description

- Gadgeon developed an AR based solution with Industrial grade smart glasses that renders the maintenance steps to be followed line by line.
- The mechanic can use voice to navigate the steps, makes the SOP compliance as well the maintenance error free.
- The solution has the following major functional components:
  - **Server Application** – manages the assignment of work to each technician and downloading of required set of manuals, images and videos to the AR headset.
  - **Glass Application** - Android based application supports secure login, voice activated navigation through the maintenance manuals. The amount of time spent on each step is available for the supervisor to review.

## Outcome and Benefits Delivered

- Voice activated navigation, head movements and gesture controls allows the mechanic to be hands-free, improves efficiency and reduces errors
- By leveraging images, videos and the AR content, makes the maintenance of even the most complex parts efficient irrespective of the experience level of the technician.
- Two-way video call with remote technicians, in case any issues during the maintenance session, further enhanced the efficiency, accuracy, and faster ramping up of maintenance staff.
- Supervisors are able to review each mechanic’s performance and make informed decision on their training needs as well performance reviews.



# Problem Statement

A startup company, specializing on products that operate on and around people wants to develop a server agnostics software platform that provides hands-free access to technical manuals and instructions for Maintenance of aircrafts and heavy equipment in the mining industry.

- This platform to have a server application, telepresence backend, progressive web application and apps that run on hardware like Realwear headgear to support day to day activities of maintenance professionals.
- The app will help maintenance professionals to navigate their maintenance manual/instructions in a hands free manner and aid in telepresence support from remote specialists.
- The server will handle document uploads/ updates, daily assignment of tasks and related documents to maintenance personnel.
- The server application can be deployed on a cloud (Azure) or on-premise server depending on the needs of the end customer.



# Technical Architecture Considerations

- The requirements of this platform was evolving parallel to the development. We understood that such a system cannot be fully specified upfront and by design the system should be continuously upgradeable and scalable
  - The development process and the architecture need to enable close collaboration between Business, Engineering, and Operations to respond to evolving requirements and meet business goals
  - Need to add new features, use cases and workflows incorporating learning and feedback from the market all the while ensuring the availability and quality of services
  - We expect to support differentiated service models as we expand with different type of SLAs and flavors of regulatory compliance tailored for different industries
- The application architecture should support easy migration from one platform to another with minimum changes.
- The system should be scalable, with high availability and secure
- The cloud deployment had to handle Multi-Tenant access

# Technical & Architecture Decisions

- Basic guidelines we followed
  - Be ready for change and evolve: Microservices, Kubernetes, Dockers, Agile and Devops,
  - Docker containers will help to isolate and pack software with all its dependencies and Kubernetes will help to deploy and orchestrate the containers. Kubernetes and Dockers based development will support platform independent deployment
  - Don't reinvent the wheel: Leverage third party platform like Azure, open source technologies, etc. wherever feasible
- Micro services
  - Natural choice for developing a large system expected to evolve over time
  - Independent development of new use cases and services as and when needed
  - Easy support for differentiated service
  - Future readiness
  - Enables Easier adoption of DevOps
  - Enables scalability, Easy to maintain, Easy to re-factor

# Technical & Architecture Decisions

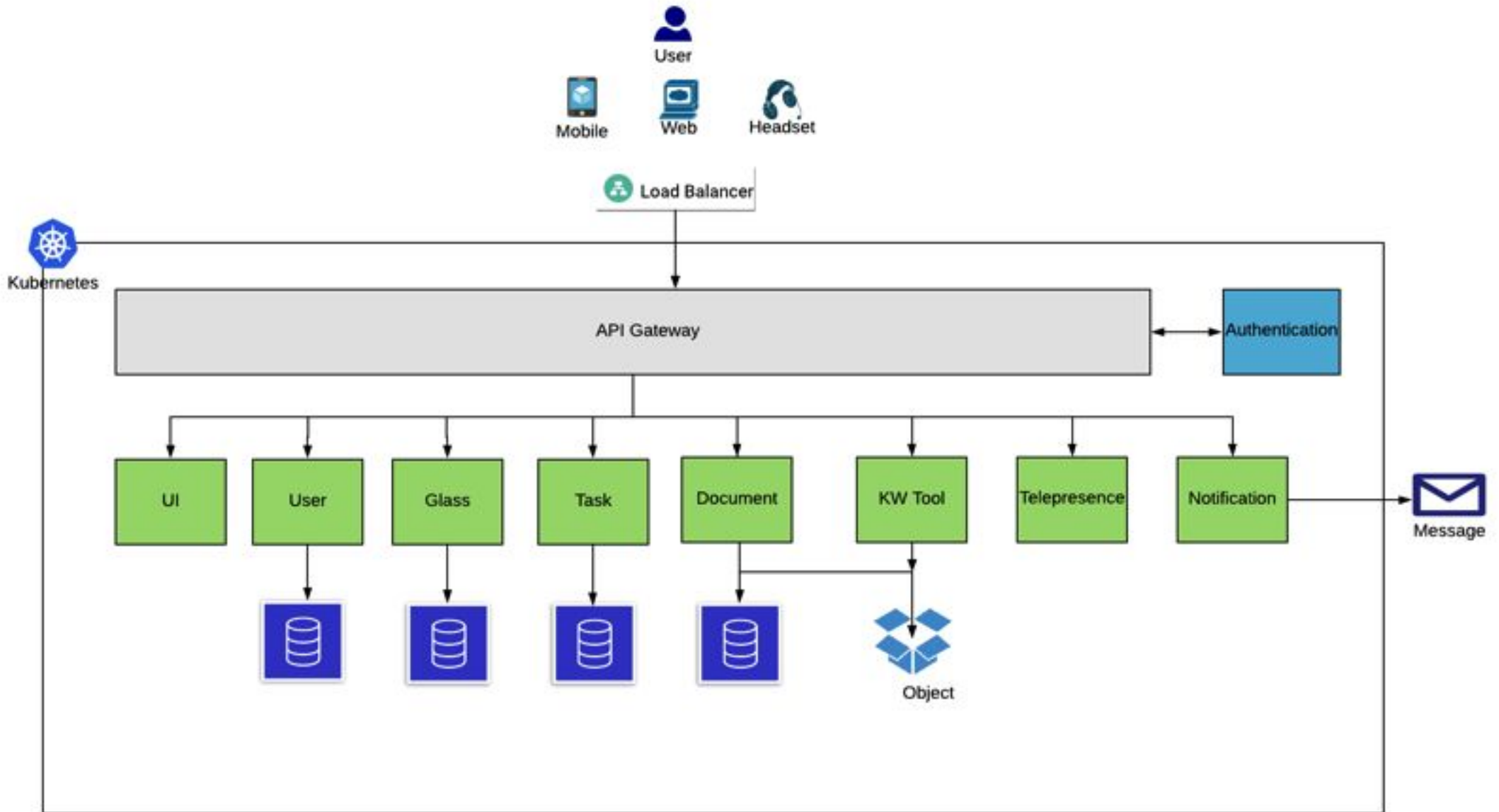
- Agile and DevOps practices for evolutionary development with frequent changes
  - Scrum with two week sprints
  - Full Test automation and Continuous integration
  - Continuous delivery capability and policy based gating for deployment
  - Fully automated Single click environment creation, deletion, rollback to ensure parity of development, staging production environments
  - Blue/Green deployment A/B testing and Canary release to enable feedback, support experimentation and learning
  - Monitoring with log collection, analysis, metrics, alerts
  - Cross functional team doing development with TDD
  - Appropriate tools - Jira, Confluence, Jenkins, Testrail, Bitbucket, Robot framework ,Selenium, Appium



# Technical & Architecture Decisions

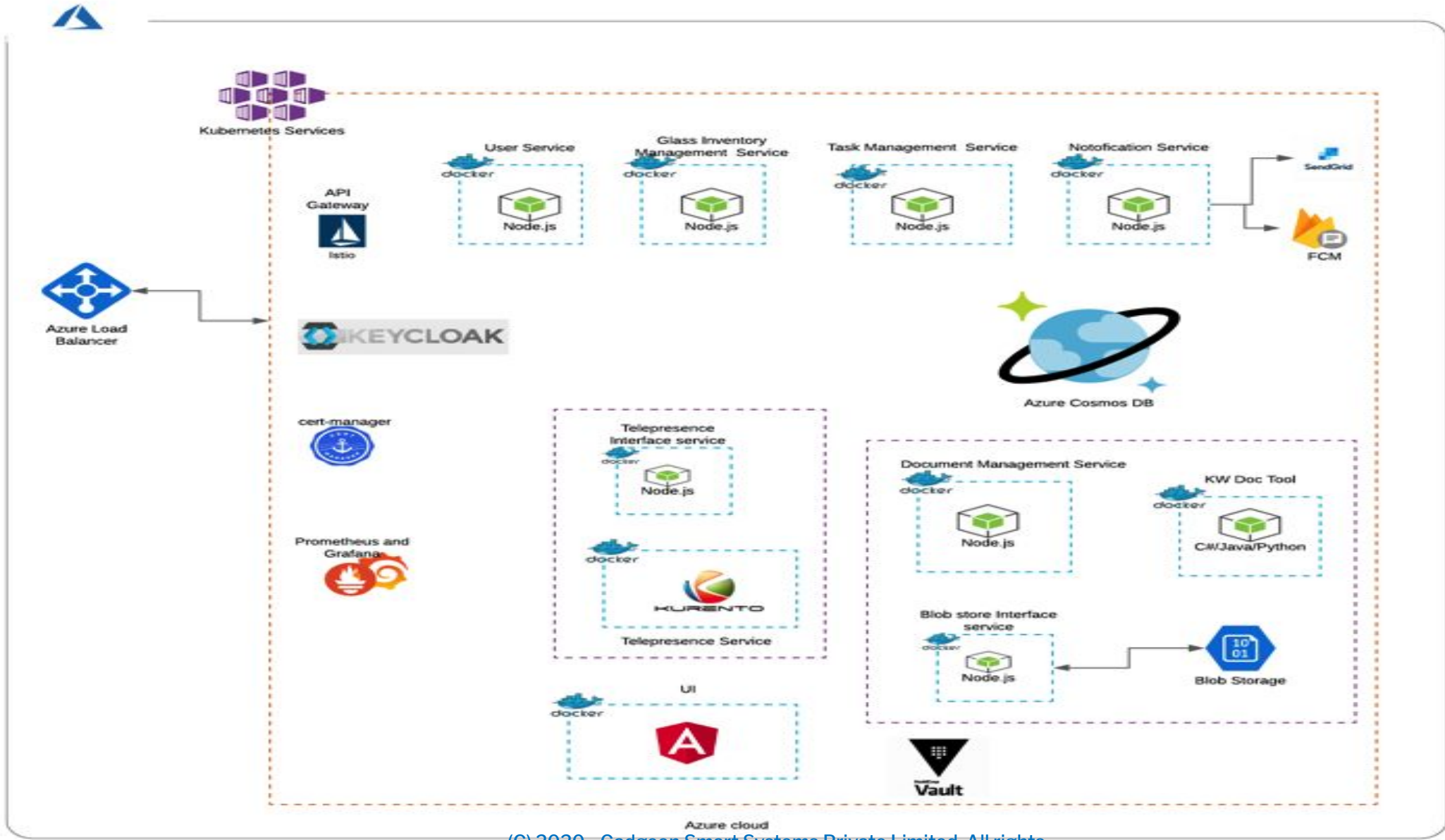
- Leverage Azure services and open source technologies as much as possible
  - The development team could focus on the business logic and not on the infrastructure for building underlying services
  - Time Advantages outweighs the drawback of vendor locking.
  - Core logic remains vendor neutral and in case vendor change is needed can be done with some porting effort
  - Integrate with other enterprise tools becomes easy

# Application Architecture



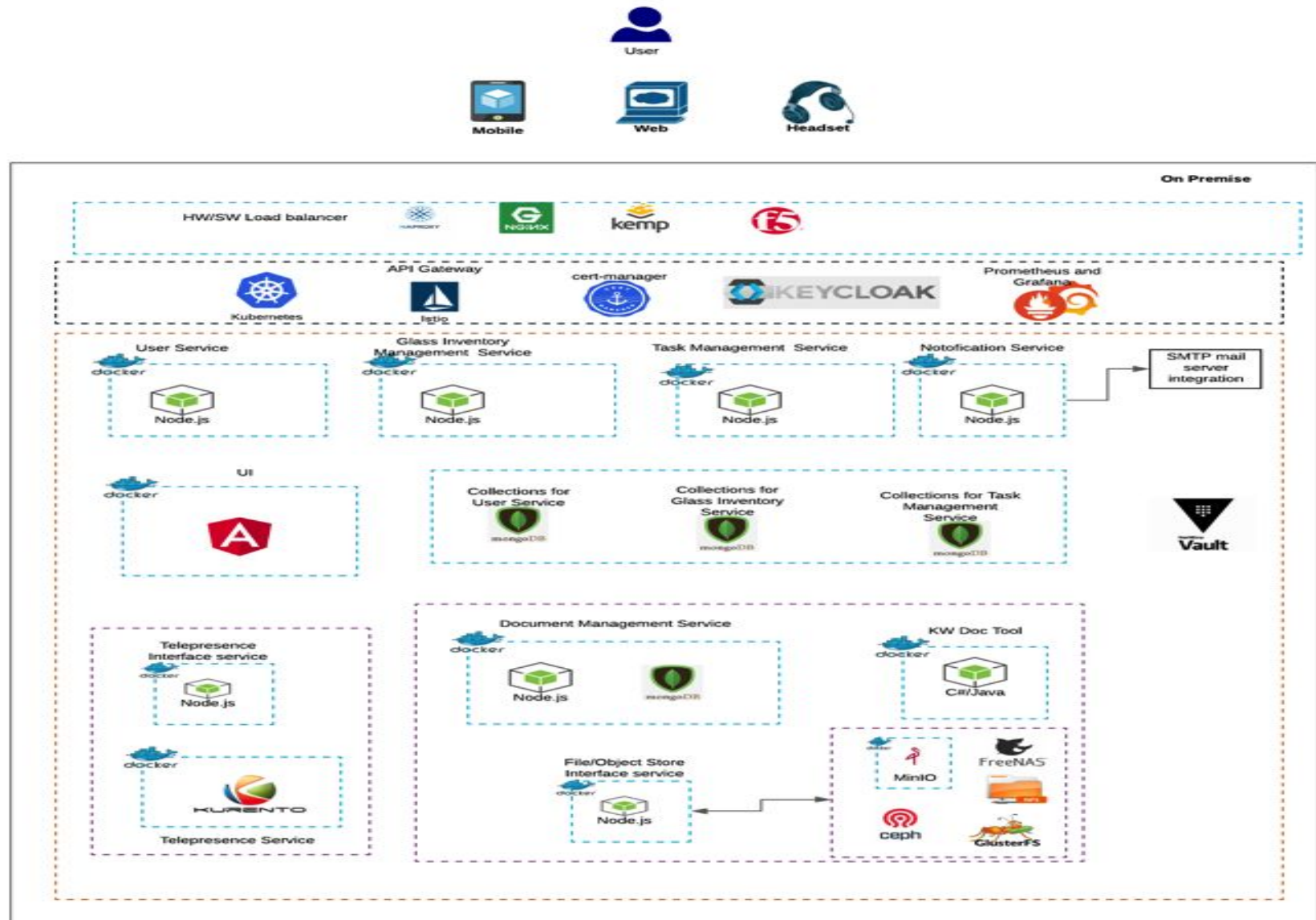


# Azure based System Architecture

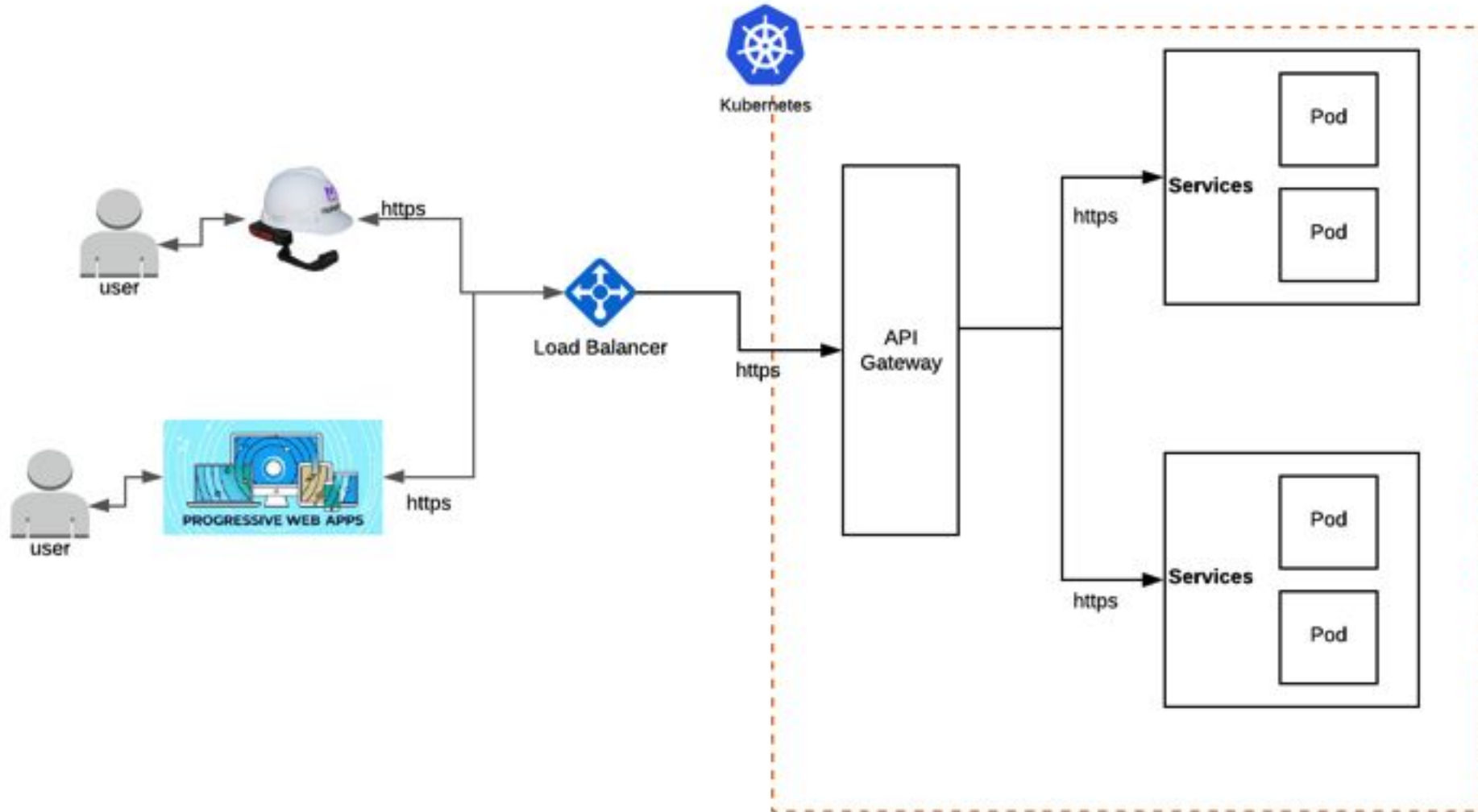




# On-Premise Server Application Architecture

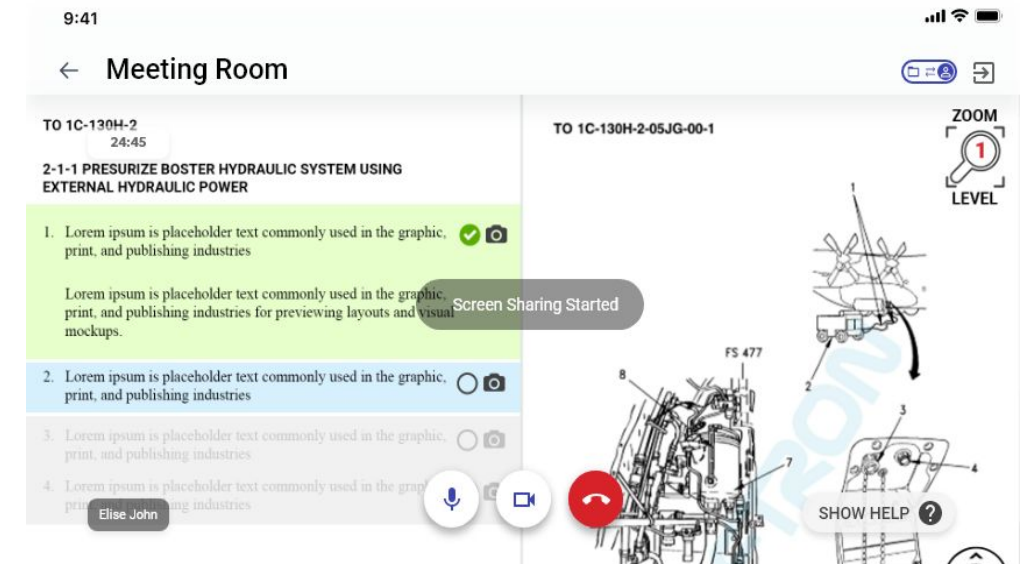
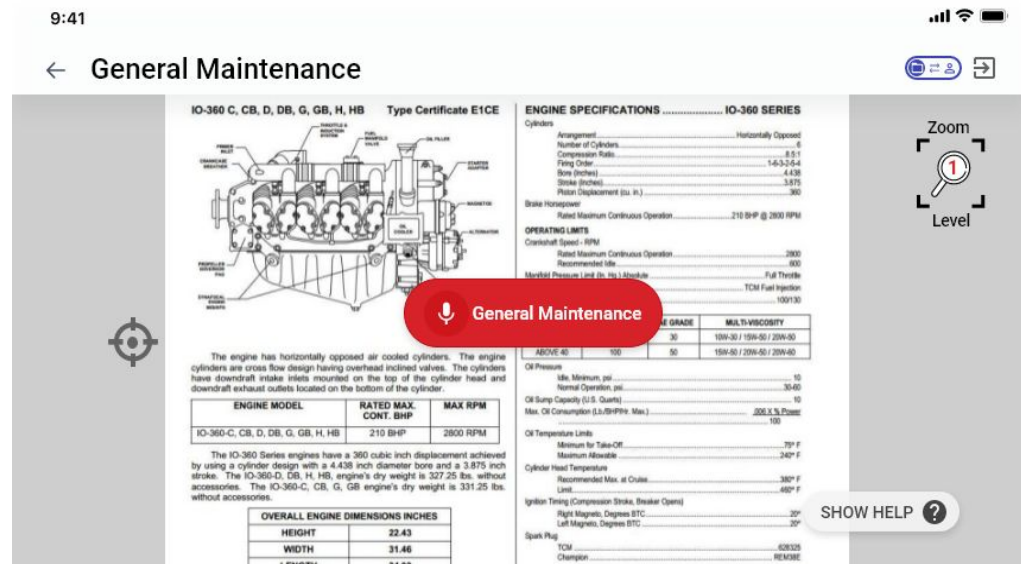
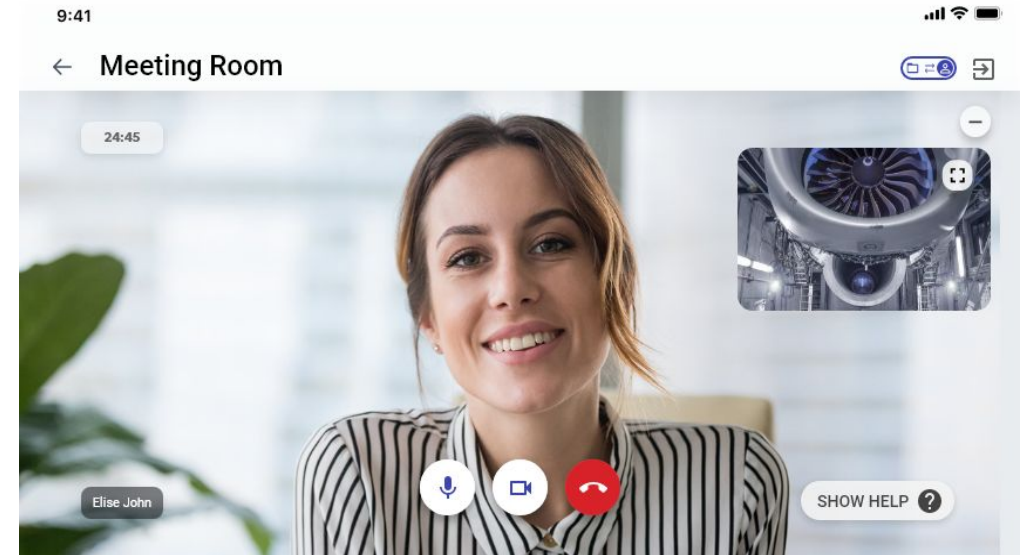
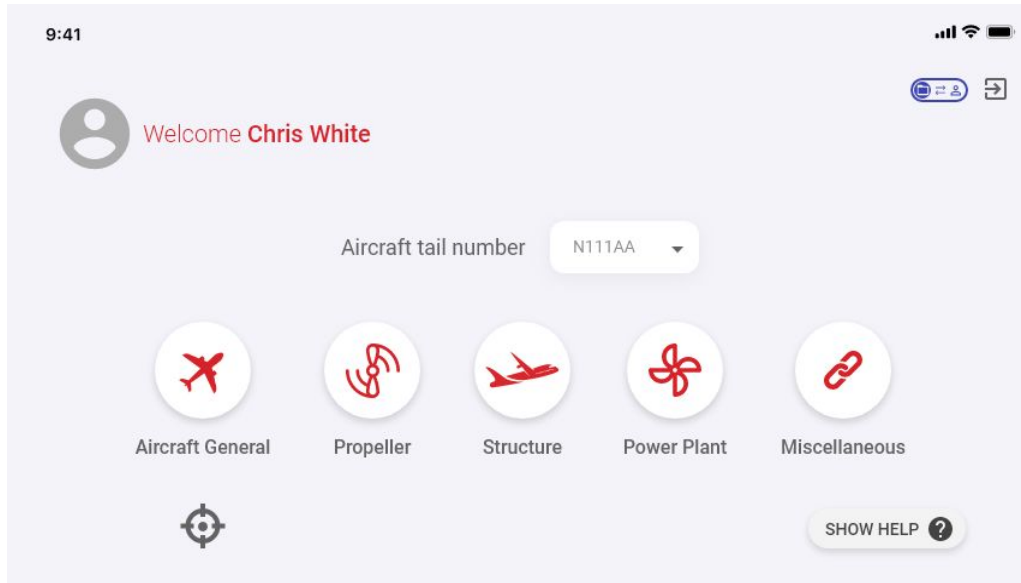


# ▶ Rest API Integrating Glass, Web and Mobile App





# What Users See – Maintenance Technicians





# Technologies/ Tools/Platforms Leveraged

## Cloud Services

- MS Azure VM, AKS, Docker, Azure Load Balancer, SendGrid

## Gateway Technologies

- Yocto Linux, C/C++, MQTT(Mosquitto), SSL/TLS, SQLite3

## Automated Testing

- Python, Selenium, Appium, Robot Framework, Postman

## Security Testing

- OWASP Zap

## Frontend

### Technologies

- Angular, HTML/CSS, MicroFrontend – Single-spa, Progressive Web UI

## Database / Storage

- Azure Cosmos DB, Azure Blob storage, Mongo DB

## Automated Deployment

- Terraform, Jenkins

## Scalability Testing

- Jmeter

## Server Side Technologies

- Node.js, Java, Springboot, Istio API Gateway, KeyCloak, HTTPS, WebSocket, WebRTC, Kurento media Server

## Mobile App

- RealWear glass (Android based) - Kotlin

## Continuous Integration

- Bitbucket, Jenkins, Azure docker registry, AKS

## Performance and Load Testing

- Jmeter

**THANK YOU**



**For More Details, Let's Connect**



**Gadgeon Systems Inc.**

881 Yosemite Way, Milpitas, CA 95035, USA

CONTACT - USA

Mani Ram - Vice President - Solutions and Technology

☎ +1-678-900-0874 | ✉ mani.ram@gadgeon.com

**Gadgeon Smart Systems Pvt Ltd.**

VI 405/E1, Fathima Tower, Maleppally Road, Thrikkakara PO,  
Kochi, Kerala, PIN: 682021, India

CONTACT - INDIA

Hari Nair - CEO & Co-Founder

☎ +91 989-501-5880 | ✉ hari.nair@gadgeon.com

**Gadgeon Europe**

Antwerpsesteenweg 124/54, 2630

Aartselaar, Belgium

☎ +32 475 23 39 46 | ✉ europe@gadgeon.com

✉ [sales@gadgeon.com](mailto:sales@gadgeon.com)