Expertise
and insight
for the future

Sarita Khanal

Customer Portal: A Project for Digitalization in Mining and Metals

Metropolia University of Applied Sciences

Bachelor of Engineering

Information Technology

Bachelor's Thesis

25 November 2019

Author Title Number of Pages Date	Sarita Khanal Customer Portal: A Project for Digitalization in Mining and Metals 34 pages + 1 appendices 25 November 2019
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Professional Major	Software Engineering
Instructors	Janne Salonen, Head of Department (ICT) at Helsinki Metropolia University of Applied Sciences Kari Saloheimo, Line Manager of Digital Services at Outotec Oyj

The main purpose of this thesis is to demonstrate the digital customer engagement by showing a real project, Customer Portal, which is an ongoing project in the case company, Outotec Oyj. Customer Portal is a web service that acts as a single access point for customers to all Outotec cloud-based digital offerings and engagement contents. It includes user access management, user profiles, assets analytics, notification center, service requests, collaboration space, and personalized support.

The additional purpose of this thesis is to understand the modern technologies that have been used to create the application. Customer Portal is built with ReactJS, TypeScript and Redux in the frontend, TypeScript and Node.js in the backend, MongoDB in database management and Azure in Cloud computing. Outotec React Component library is used for getting the different components of the application. The architecture and designing processes are also included in the thesis. Furthermore, a vivid view of the integration of all modern technologies to make a whole web application is shown.

This thesis can be useful to those people who want to see the impact of digitalization in mining and metals and know how to use it for better user experience. This project can give detailed overview of use of digital data in mining and metals industries to be more productive and sustainable. Furthermore, it will give detail information of development of a web application in the real world.

Keywords	Digitalization, Single Page Application, ReactJS, Typescript, Redux, Node.js, MongoDB, Azure

Contents

Appendices

Appendix 1. Sample Codes of Project

List of Abbreviations

ACT	Advanced Process Control. One of the Outotec platform.
API	Application Program Interface. Set of tools and protocols used for software development.
DOM	Document Object Model. Representor of document in tree model.
DSUI	Design System User Interface. Library to store Outotec designs.
HSC	One of the Outotec platform.
HTML	Hypertext Markup Language. Standard markup language for displayed documents in web browser
HTTP	Hyper Text Transfer Protocol. Protocol used by World Wide Web.
laaS	Infrastructure as a Service. Cloud computing service.
IDE	Integrated Development Environment. Software application for providing software development.
IT	Information Technology.
JSX	JavaScript extension. React extension which allows to write JavaScript looks like HTML
KPI	Key Performance Indicator. Type of performance measurements.
MVC	Model View Controller. Application design model which includes model (data), view (user interface), and controller (processes that handle input).
NPM	Node Package Manager. Package manager.
NPX	NPM Package Runner. Package runner tool.

Oyj	Julkinen osakeyhtiö. Public Stock Company listed on the Helsinki Stock Exchange
PaaS	Platform as a Service. Cloud computing service.
Props	Properties. Data to pass information from parent to child components.
R&D	Research and Development
SaaS	Service as a Service. Cloud computing service.
.TS	TypeScript. Extension for TypeScript.
TSC	TypeScript Compiler. Compiler to run TypeScript code.

1 Introduction

The world has become the global village through the internet connection. Mobile, analytics, social media, sensors, and cloud computing have crossed the limit of imagination and creativity. With the enhancement of the internet, digitalization is also growing at a rapid pace. Digitalization has become a hot topic to discuss and must needed component in the business and technology world, but what is digitalization then? In simple words, digitalization refers to the term “going digital” which means the process of changing data to digital formats, so that the computer can read and understand them.

The common belief in digitalization being just a part of media and entertainment industries has changed drastically. Health, education, transportation and many sectors are also digitalizing. Even, mining and metals industries are already feeling the impact of digitalization. Digital transformations have eased out the workload in mining and metals industries. Sensors, robotics, modeling, automation and operational hardware have improved way of performing work that has been carried out manually in the past centuries. In past days, getting ores from the mining areas, taking the source to the refining areas and disposing the waste were all done by human resources.

Customer Portal is a project carried out in Outotec Oyj to focus on the advantages of digitalization. Outotec is a Finnish company that provides solutions mostly in mining and metals and others in energy companies. Customer Portal is also one of the digital solutions being made in Outotec to provide a platform, where the user can access different digital offerings of the company from a single site. All these digital offerings are designed to facilitate customers in handling equipment, detecting errors, providing safety, accessing documents and helping in many other activities.

The main purpose of this thesis is to dig into the Customer Portal and find out the benefits of digitalization in mining and metals industries. The thesis is divided into five sections. At first, it goes through the digitalization, and then provides a brief introduction of the case company with the information of how digitalization is carried on within the company. Then, the technologies used and their integration in the portal are described in detail. After that, it will go through the whole Customer Portal and the digital offerings in the application. Some parts of the image in thesis have been hidden with gray rectangle in order to protect the confidential information. Then, the thesis will end with the conclusion.

2 Digitalization in Mining and Metals

Most of the raw materials used to build infrastructures and instruments including computers, mobiles, cooking utensils, means of transportation and many other tools that we used in daily life are provided by mining and metals industries. But, to get suitable raw materials, it goes through many processes from exploring mine to distribution. To summarize whole mining and metals practices, its value chain falls within the scope of the analysis including obtaining a license, exploration, setup, mining, ore processing, metals smelting and processing, distribution, and sales and marketing. (Carvalho 2017: 61)

Historically, mining and metals companies have been technology innovators with full of noteworthy examples. Steam engine was designed by Thomas Newcomen in early eighteenth century to draw water from tin mines in Cornwall. In 1784, James Watt and his friend Matthew Boulton patented steam locomotive which was used to move mined ore. Flotation, the most effective method of separation of minerals from gangue, or the barren parts of mined ore was made in Broken Hill, Australia, in 1903. Beside these, different inventions were made. (Minalliance and Michel 2012: 31)

However, in case of digital utilization, they are lagging. Mining and metal industries have very expensive asset operations and machineries, whose maintenance and depreciable life can have huge impact on production output, operating costs and ongoing capital expenditure. These processes include many equipment and manual works in between. Manual works consume a lot of time and cost. Talking about the handling of mining equipment, it is also not an easy task to do. Most of the worker must go to the mining field and even work in harsh temperature. There can be a risk of serious accidents or possibly, death as well. Therefore, it is hard to get enough skilled manpower to work in mining and metals industries. (Mitchell 2019)

Exploring mining resources is becoming more challenging every day. Not surprisingly, mining comes on the top list of factors affecting environment, as large amount of waste and harmful gases is produced. So, mining and metals industries always get pressure to reduce carbon emission and energy use. Market demand, in one side is expanding rapidly, and on the other side profiting with sustainability is becoming more complex. Hence, it is very important for mining and metals industries to bring up new ideas and seize the opportunities provided by digital technologies in order to thrive in this environment.(Mitchell 2019)

Digitalization might be the answer to the challenges of mining to bring vast benefits. It has already changed the nature of industries and their interaction with the employees, communities, government and the environment. From the process of mineral exploration and valuation through mining, ore processing and metals productions to downstream sales and distribution, digitalization is fading the traditional ways and challenging the past business models. Visual, thermal and tactile sensors connected to diggers, trucks, crushers, conveyers, plants and tailing treatment are used to get and share the data about machines, processes and environmental conditions. Then, data can be fetched and analyzed by sitting in one corner of the world and services can be provided accordingly. For most mining and metals processing plants, digitally enabled automation is implemented which means humans are now decision-makers and operation controllers, while machines will do the physical work. (World Economic Forum and Accenture 2017: 5-9)

Furthermore, productivity can also be optimized by implementing Machine learning and Artificial Intelligence. Connected sensors, monitors and alarms are used for reporting potential harmful events and conditions. Warnings about the upcoming risks from these digital offerings can also reduce the risks of accidents. It also helps to improve health, safety and environmental impact by warning beforehand, saving lives, reducing injuries, lowering emissions and waste, and increasing transparency and sustainability.

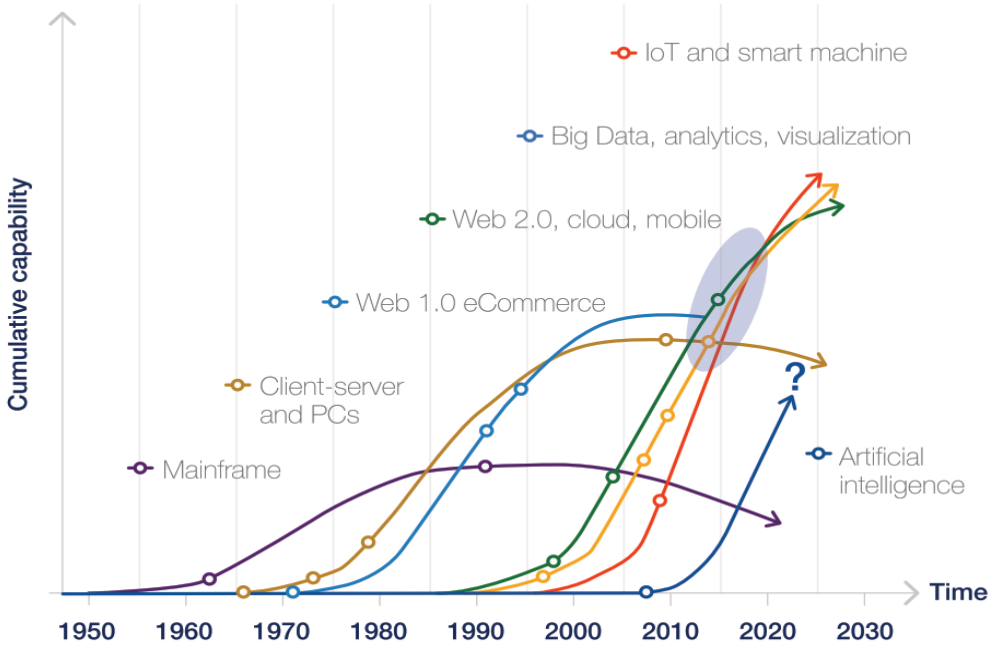

Figure 1. Increasing capabilities of digital technologies (World Economic Forum 2017)

Considering the pros of digital transformation, mining and metals industries are focusing on digitalization and adapting multiple digital technologies. This can also be seen in figure 1. Automation, robots, operational hardware, smart sensors and digital applications are changing the way of performance and how they have changed the work practices in mining and metals industries are described below.

Automation, robots and operational hardware has stepped out from the traditional way of handling machinery in mining and metals industries. Automated exploration drones, robotic trucks, diggers, autonomous stockpile management, autonomous robots for recovery of recycling material are some examples that are being used. Robot or automated machine can get the data about environment and move the machinery without human assistance. Autonomous operations can help in providing worker safety, especially in extreme conditions like in underground mines or hot mills. (World Economic Forum and Accenture 2017: 11-12)

Smart sensors can collect physical, biological and chemical data and convert them into digital format. They can also process the information they get and make decisions accordingly. Smart sensors can be used to get real-time insights of the performance of equipment such as physical condition and performance value. These data can be deployed to support other technologies such as robotics or application to analyze all the data and find the result. These data can also be used in predicting equipment performance, equipment failures, maintenance and asset management. All of these will help mining and metals for decision making and increase productivity ensuring environmental and human safety. (World Economic Forum and Accenture 2017: 13-14)

Beside these, connecting IT devices and systems to operational technology can also bring up significant value in mining and metals industries. They have rich data resources collected from manufacturing operations. Advanced analytical techniques can help them to learn from this data and enable higher levels of real-time control performance. Digital applications can include tools to accelerate and simplify planning, scheduling, enhancing productivity while providing significant health and safety benefits. Workers can also get advantages from real-time access to documentation, decision-support and troubleshooting tools. They can record their activities and observations in digital form, which can be stored and used as references for future analysis and improvements.(Noterdaeme *et al.* 2018)

3 Case Company

Outotec Oyj is the case company of this project, Customer Portal. Outotec is a Finnish process technology company. It develops and delivers leading technologies and services mostly in the mining and metals, and others in energy, environmental and chemical industries, which enables those industries to increase sustainable profit. Outotec build, maintain and run all the operations from root level to ensure the smartest value from natural resources. They also work with customers to find the most sustainable solutions for handling the full value chain from ore to metals including water management, energy recovery as well as waste and side streams. Currently it has more than 4,000 experts working to attend the demand of their customers. (Outotec 2019)

Outotec was listed as the fifth world's most sustainable company, according to a report released in January 2018 by Canadian financial information company and magazine, Corporate Knights.(Menear 2018)

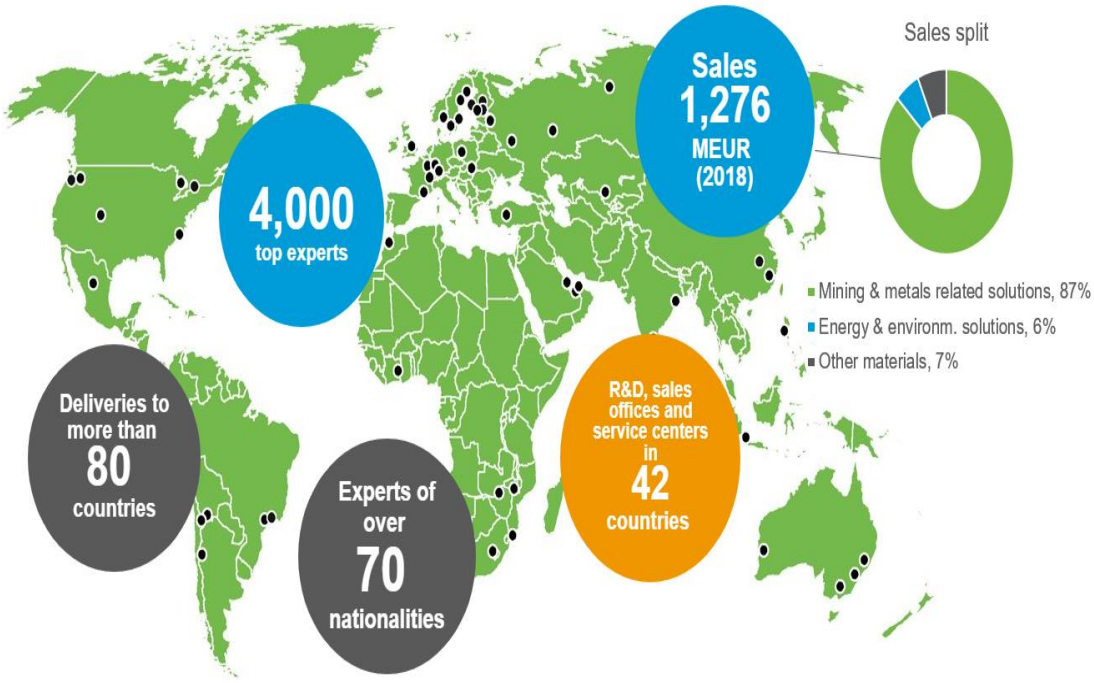

Figure 2. Local operations and global presence of Outotec. (Outotec 2019)

Figure 2 shows that Outotec is providing services all over the world in more than 80 countries. According to figure, total sales of 2018 was from 87% of mining and metals related solutions, 6% of energy and environmental solutions and 7% of other materials. It also has R&D, sales offices and service centers in 42 countries.

Outotec believes digitalization is the main tool to provide effective services to its customer as it says.

> Digitalization is at the heart of our operations. We create plants of the future, working closely with our customers. With intelligent services, applications and equipment we improve safety, predictability and performance of our customers' assets.

Looking through the history of Outotec, digitalization in Outotec started in 1969 with the first computer-controlled plant. Then, it started to build up intelligent equipment, processes, and services. Now, it has been expanding and integrating with many technologies. There are several teams specializing in various fields like design, marketing, development and management, working on different services. Multiple platform elements are being made to help customer in planning, handling, analyzing and providing safety. Some of them are listed below. (Outotec 2019)

- HSC Chemistry is the first Outotec software package designed to combine chemical, thermodynamic, and mineral-processing features. It helps to digitize process design and provides operational support through modeling and simulation.
- ACT is Advanced Process Control solution that builds customized applications. It allows stability and optimization of the plant process from single-unit processes to plant-wide production. It can also be connected to all existing plant control systems for efficient utilization of the process.
- Asset Analytics provides critical event-based equipment condition and operating data of equipment. It includes equipment health and asset monitoring capability to connect equipment and provide health and other condition information.
- Customer Portal is a single access point to all the Outotec digital offerings. It provides multiple services to help the customer in planning, finding or storing documents and requesting for help.

4 Technologies Used in Customer Portal

Customer Portal is made using many modern technologies like ReactJS, Node, TypeScript, MongoDB and Azure. The introduction and importance of each technology are described below.

4.1 ReactJS

ReactJS is an open-source, front-end JavaScript library developed and maintained by Facebook. It is responsible for the view layer of the application. React application is made up of different components, each responsible for outputting a small, reusable piece of code. These components can be nested with other components enabling complex applications to be built with simple building blocks. React components can be written in JSX, which allows us to write components using HTML. Props and states are used to pass information into a React component. They can be of any type including functions and callbacks.(StackOverflow 2018 : 2-3). ReactJS was first used on Facebook but, later, it started to grow rapidly. Many companies started to switch to React. Currently, large companies like Netflix, PayPal, Instagram, Airbnb, Yahoo are using react mainly because of its good features. It is simple, fast and scalable. Developers do not need to spend lots of time learning React as it allows JSX syntax, which is mostly like HTML. As a result, they can easily start working on the application instead of spending more time studying. (Dr. Michael J. Garbade 2016)

In ReactJS, a full view can be divided into different components such as Navbar, Form, cards, etc. These components are small and can be reused, which helps to save time and effort of developers as they don't need to write code again for the same components in the future. It has a one-way flow for data binding and virtual React DOM for just updating the parts that have been changed, providing high-performance client and server-side rendering. Moreover, ReactJS also has libraries and tools which help developers to code and test easily. Some of them are create-react-app, Material UI, Redux and Enzyme.(Dr. Michael J. Garbade 2016).

React works on the main concept of reusing components. This is possible because of props and states. Props stands for properties and are used to pass data and methods

from a parent component to a child component. Props are read-only. So, data coming from parent cannot not be changed.

```
class Parent extends React.Component{
  doSomething(){
     console.log("Parent component");
  }
  render() {
    return <div>
         <Child
           text="This is the child number 1"
           title="Title 1"
           onClick={this.doSomething} />
         <Child
           text="This is the child number 2"
           title="Title 2"
           onClick={this.doSomething} />
      </div>
  }
}

class Child extends React.Component{
  render() {
    return <div>
       <h1>{this.props.title}</h1>
       <h2>{this.props.text}</h2>
      </div>
  }
}
```

Figure 3. Screenshot showing how to use props in React reproduced from Stack Overflow

State is used to manage and communicate data in our application. It is represented as a JavaScript object and has component level scope that means it contains private data of component.

```
class ExampleComponent extends React.Component {
  constructor(props){
    super(props);

    // Set-up our initial state
    this.state = {
      greeting: 'Hiya Buddy!'
    };
  }

  render() {
    // We can access the greeting property through this.state
    return(
      <div>{this.state.greeting}</div>
    );
  }
}
```

Figure 4. Screenshot showing how to use state in React reproduced from Stack Overflow

4.2 Redux

Libraries like React, Angular, etc. use components to build the application. Small application does not need any external library or tool to manage the state of the components used but in large application, this can be complex. So, library called Redux is introduced. Redux is state management tool that allows to keep the state of application in a store. Each component can access any state from store when needed. It enforces the restriction on how and when state update can happen which makes state management easier. Redux also helps to write application that run in various environments like client, server and native. Redux is mostly used with React. However, it can be used with other JavaScript frameworks or libraries. (Copes 2018)

Redux data flow is unidirectional which means application data will flow in one-way binding data flow. First, actions are created from the component which will be later dispatched in the store. Reducer gets the current state and returns new state in store after performing certain action. This updates the component. (Eschweiler 2017)

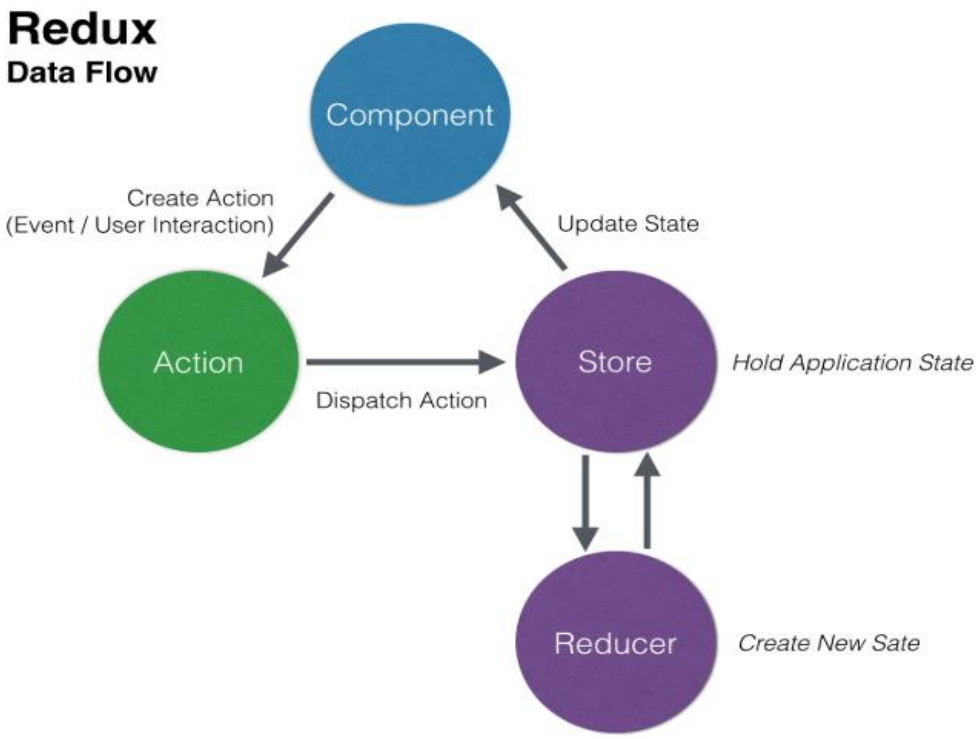

Figure 5. Redux data flow (Eschweiler 2017)

The redux data flow can be seen from figure 5. It is clear from the figure that action, reducer and store are the building blocks of Redux. Actions are events that can send data from application to Redux store by using **store.Dispatch()** method. Action must have a type property to indicate the type of action and have a payload to contain the information about the work done by the action. Actions are created by action creators. Reducers are pure functions which take the current state of an application, performs an action and returns a new state based on action passed. All the application states are stored in Redux store. These states can be accessed and updated by the components using helper methods. Store can be created using **createStore().**

According to Redux official documentation, Redux works on three principle. First one is "Single source of truth", which means states of the application is stored in an object tree in a single store. This helps to make debugging and inspection of application easier resulting to a faster development. Second one is "State is read-only". This means state cannot be changed. The only way to change state is to emit an action and this help to ensure that both views and the network callbacks will never write directly to the state. Last principle is "Changes are made with pure functions". Pure reducers are used to make the changes. Reducers simply means function which take the previous state and action and return the next state.(Abramov 2015)

4.3 TypeScript

Typescript is an open-source programming language which is also known as strict syntactical superset of JavaScript. TypeScript checks static type, which is not available in JavaScript. It was developed by Microsoft and was originally announced by Anders Hejlsberg (the mind behind Turbo Pascal, Delphi and C#) in October 2012. TypeScript makes JavaScript more strong-types, object-oriented language like C# and Java. That also means it is very good for large application. It can be also used in both client and server-side. (StackOverflow 2018)

Static type system feature of TypeScript makes it more suitable language for development of large applications. Code with TypeScript is easier to understand as it defines what kind of arguments are accepted and what kind of values are returned. TypeScript notifies error if there is any typo, value which can be null, or an object passed to a place where it should be an array. Hence, the code will have less bugs. Furthermore, with

TypeScript, writing simple boilerplate unit or integration tests can be ignored so that focus will be more on testing business logic of application. As coding with typescript does not need boilerplate unit or integration test, new features can be developed in shorter time, which is also less complicated, less error-prone and easier to maintain. (StackOverflow 2018: 2-10)

TypeScript has .ts extension and is compiled using tsc compilation command which comes with TypeScript. The command is **tsc my-code.ts**, which will create my-code.js file. (StackOverflow 2018: 3)

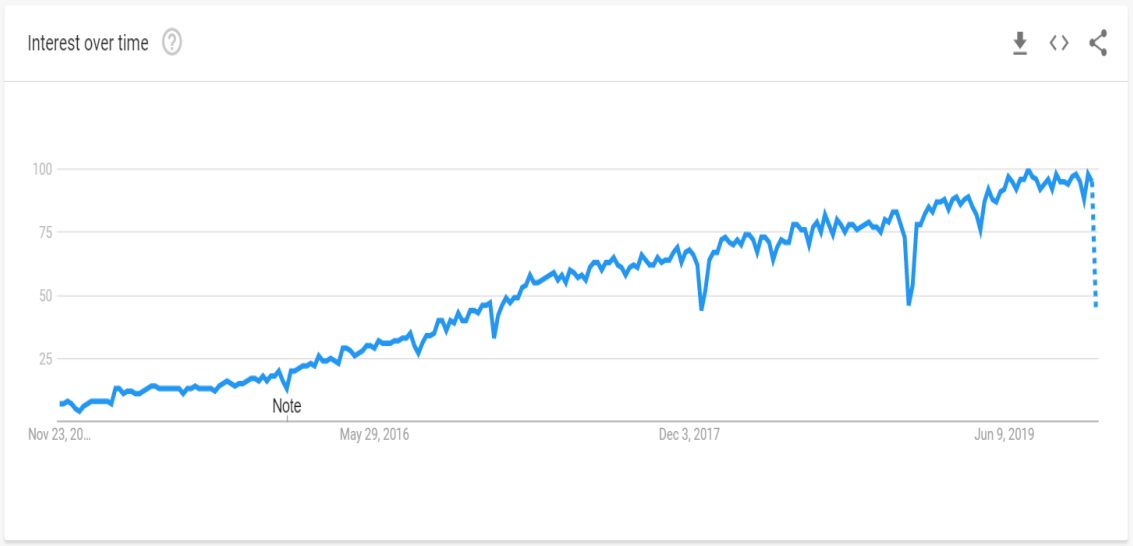

Figure 6. Growing Popularity of Typescript (Google trends 2019)

In Figure 6, search query trends for TypeScript for last five years before May 7, 2019 can be seen. It can be predicted that TypeScript is increasing more and more in web development. It is getting popular in development world.

4.4 Node.js

Node.js is a V8 JavaScript engine runtime created by Google. It is used in Chromium (browser technology powering Google Chrome). It can also be known as open-source server environment which can run in multiple platforms like Windows, Linux, Unix, Mac OS X, etc. In short, it's a tool to write server-side applications using JavaScript. Node.js is designed for making scalable network applications. NodeJS is being used in large amount because of its simplicity. It is very easy to code in NodeJS as it uses JavaScript

for writing code. It allows developers to use JavaScript in both frontend and backend development, so it is also known as full-stack JavaScript serving both client and server-side applications. Furthermore, it is much easier to start Node.js for backend, as JavaScript is one of the most popular programming languages. Developers can access to many libraries and tools in Node which makes the coding large application easier and less time consuming. (Dr. Michael J. Garbade, 2016)

Http is the most important module in Node.js to make it suitable for the foundation of a web library or framework. It helps to transfer data over the Hyper Text Protocol. Http module is used to create http server by using **http.createServer(),** and then, server is listened in port x. So, web server can be run in http://localhost:x using **node app.js**, if file name is app.js. (Node 2019)

```
const http = require('http');

const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World\n');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

Figure 7. code snippet for creating http server with Node.js reproduced from Redux official documentation

4.5 MongoDB

MongoDB is NoSQL document-based database, which means it stores data in JSON format. It is designed to simplify the data management and scaling of databases. Data in JSON is the most natural way to think about data and more powerful than traditional row/column model. MongoDB has wide varieties of data and queries that helps to filter

and sort by any field despite of complex nesting of document. MongoDB is schema-less, which means data structure is not defined before insertion which makes defining data easier. It also supports aggregation and modern use-cases like geo-based search, graph search, and text search. Furthermore, MongoDB also offers both a Community and an Enterprise version of the database. (StackOverflow 2018)

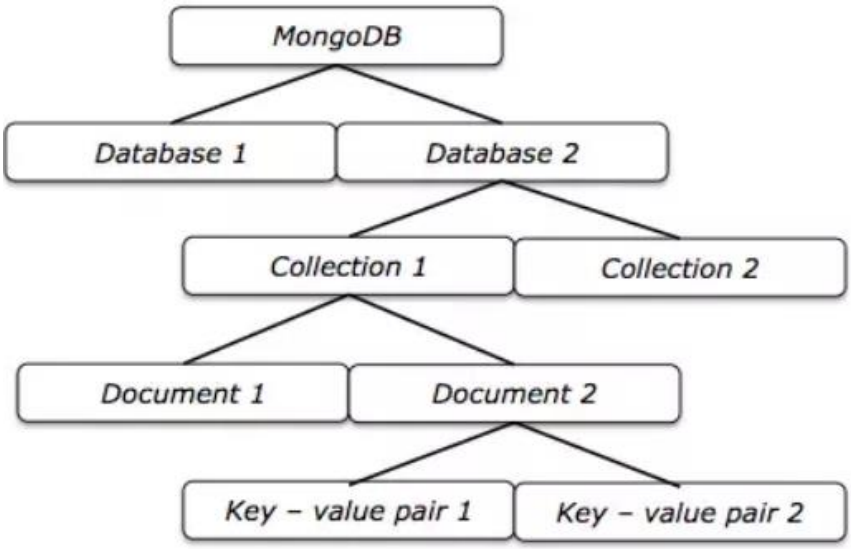

Figure 8. Architecture of MongoDB

Figure 8 shows the architecture of MongoDB. Database in MongoDB is a container that contains collections. Collections is set of documents and document is set of key-value pairs.

MongoDB has four crud operations: Insert operation, Read operation, Update operation and Delete operation. Insert operation adds new document to the collection. MongoDB provides two options to perform this operation. Mongo shell method **db.collection.insertOne( )** is used for inserting one document whereas **db.collection.insertMany( )** is used for inserting many documents. Figure 9 shows insert operation. It will insert a document into the inventory collection.

```
copy
db.inventory.insertOne(
   { item: "canvas", qty: 100, tags: ["cotton"], size: { h: 28, w: 35.5, uom: "cm" } }
)
```

Figure 9. Insert Operation

Read operation finds the document from the collection. MongoDB performs this operation by using mongo shell method **db.collection.find().** Figure 10 shows read operation. It will find the document where item equals “canvas”.

```
db.inventory.find( { item: "canvas" } )
```

Figure 10. Read Operation

Update operation updates documents in the collection. This operation targets on a single collection. MongoDB provides the following mongo shell methods **db.collection.updateOne( )**, **db.collection.updateMany( )** and **db.collection.replaceOne( )** to update documents of a collection. Figure 11 shows update operation. It will update a single document where name is “Central Perk Café” with the violations field.

```
try {
   db.restaurant.updateOne(
      { "name" : "Central Perk Cafe" },
      { $set: { "violations" : 3 } }
   );
} catch (e) {
   print(e);
}
```

Figure 11. Update Operation

Delete operation deletes documents in the collection. MongoDB provides mongo shell method **db.collection.deleteOne( )** to delete one document from database and **db.collecton.deleteMany( )** to delete many documents. Figure 12 operation deletes document where status is “D”.

```
db.inventory.deleteOne( { status: "D" } )
```

Figure 12. Delete Operation

4.6 Azure

Microsoft Azure is a cloud computing service created by Microsoft for building, testing, deploying, and managing applications and services through Microsoft-managed data centers. Azure can handle millions and billions of transactions using hyper-scaling processing. It provides multi-petabytes of storage to data and, also, provides interconnected services that can pass the data among themselves. Hence, the data can be processed properly. (Microsoft 2019)

Azure comes with lots of advantages. Azure provides all the benefits of the cloud while remaining open and flexible. It supports a wide variety of operating systems, languages, tools, platforms, utilities, and frameworks. Azure is very much compatible with the user's choice of technology stack. Azure has data centers across the globe and combines these data centers into regions. Also, it is cheaper in cost as compare to other Cloud service Provider. (Microsoft 2019)

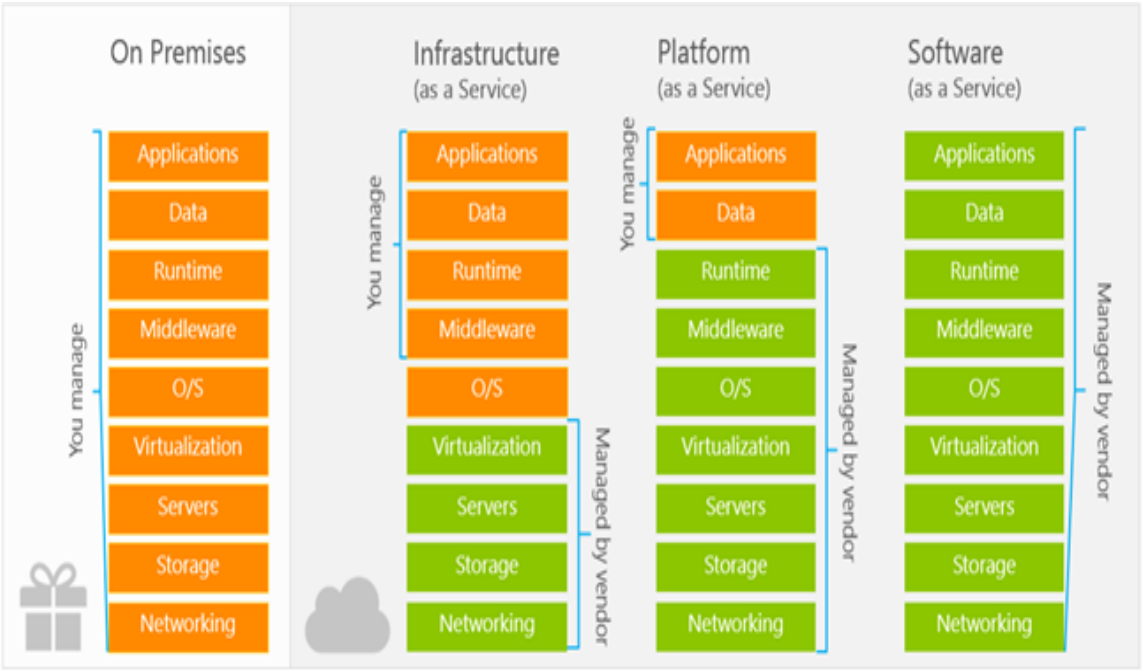

Figure 13. Cloud Service provided by Azure

Figure 13 demonstrates three cloud service models of Azure shortly named as IaaS, PaaS and SaaS and the differences between them. It is very important to have the knowledge about these cloud services before using Azure. They are described in detail in following.

IaaS, Infrastructure as a Service, is a cloud service model that provides maximum control to customer. It allows full control over operating system, middleware, runtime and installation of data as well as applications. Azure will ensure the maintenance of the physical infrastructure on which these virtual infrastructure resources are hosted. Under this approach, customers require active management and operations in the Azure environment. (Microsoft 2019)

PaaS, Platform as a Service, is other cloud service model that provides platform to customer for bringing their own application, data and deploy them. This pattern will take away the infrastructure deployment and control from the customer. Provided platforms are handled by Azure so, customers are only responsible for application deployment. Comparing with IaaS, PaaS is the faster and easier option for the deployment of applications. (Microsoft 2019)

SaaS, Software as a Service, is the third cloud service model which is a higher-level abstraction compared to PaaS. In this model, software and its services are available for end-user consumption and customers can only bring their data into these services. Customers has very less control over infrastructure and application and need to trust the vendor more. (Microsoft 2019)

Azure DevOps Pattern helps to build, test and deploy application that runs on multiple clouds. This pattern has unified the practice of continuous integration and continuous delivery. Continuous Integration build and test the code committed in version control whereas continuous delivery automates each step from a build to production environment. Combination of these processes creates a release process that enables deployment of applications across different environments. With this pattern, code can be drafted and then deployed to different private and public clouds. Instead of changing code, changing configuration file can deploy the application in different environments. With a consistent set of development tools across on-premises, private cloud, and public cloud environments, practice of continuous integration and continuous delivery can be implemented. Apps and services deployed using the DevOps Pattern are interchangeable and can run in multiple locations, taking advantage of on-premises and public cloud features and capabilities. (Microsoft 2019)

Figure 14 shows the DevOps pattern of Azure. First developer writes the code and then go through the build and release pipeline. After releasing, user can get the updated version of the applications.

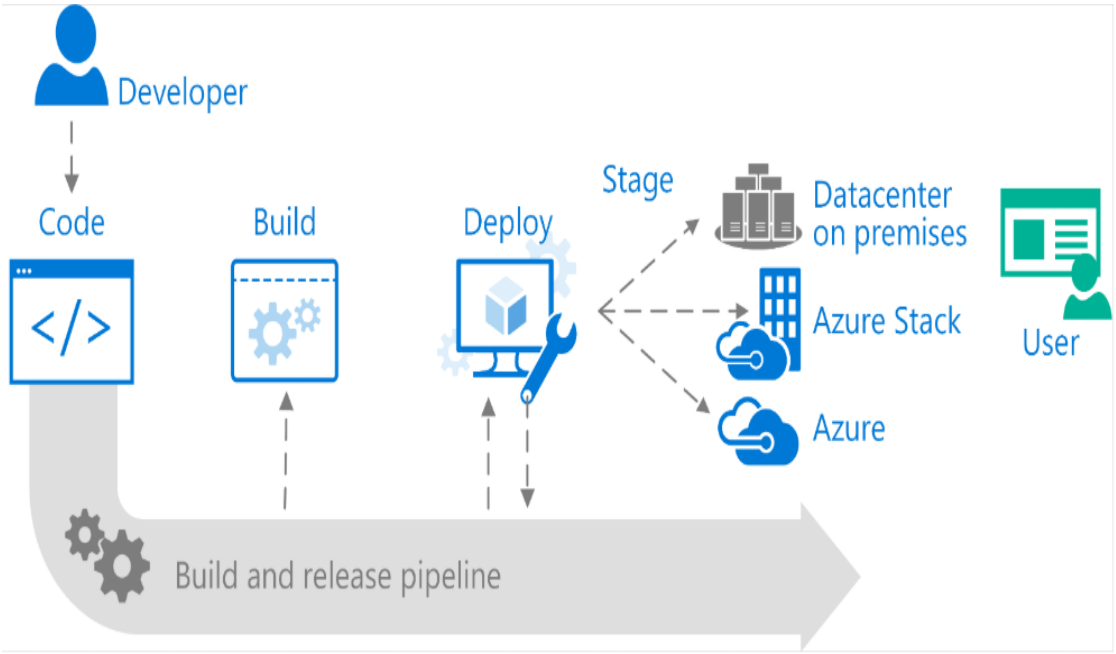

Figure 14. Azure DevOps pattern

5 Customer Portal

In late 2015, Outotec Remote Service Portal was made in order to handle shifts and record the measurements in mining company, but it was local based software. It was very complex and working with software needs training beforehand. In addition, software engineers themselves should go to the site and install the software, which is very expensive. So, with business unit people, it was decided to make new web application using modern technologies to replace the Outotec Remote Service Portal. Hence, Customer Portal was born.

5.1 Planning

In late 2018, Outotec decided to increase focus in developing a Digital Customer Interface (Customer Portal). The target was set to complete the development plan by 2018. The business ownership and a responsible project manager were determined. The real work continued together with a selected partner to create the concept, roadmap and business case for the Portal.

As shown in Figure 15, at first, different kick-offs and meetings were held to get the Portal architecture. Then, visions and roadmaps were made as per portal goals, business plan, and three user personas. The priorities were defined as per the needs. Some parts of the image are hidden with grey rectangle to protect confidential information.

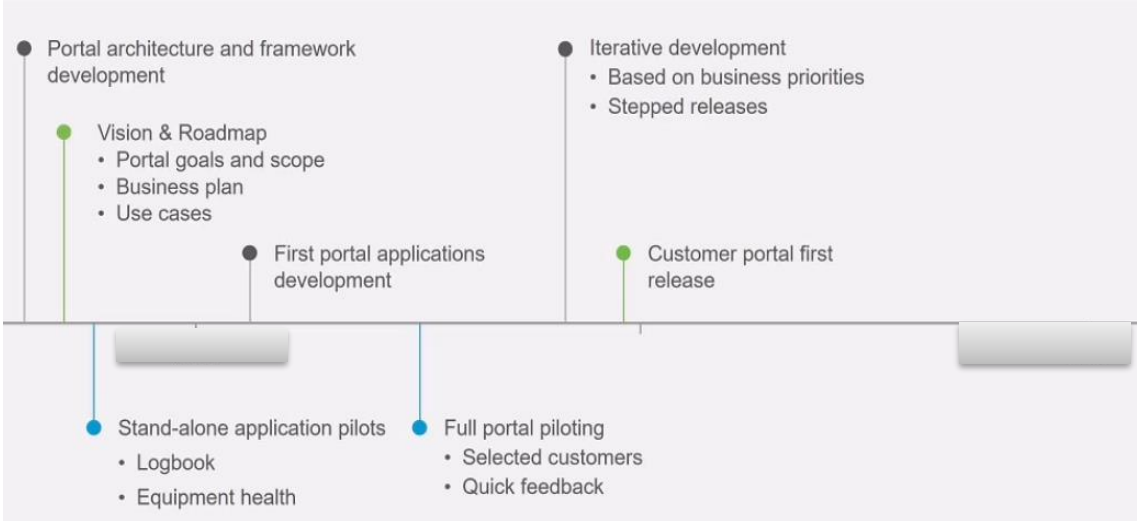

Figure 15. Planning phase of Customer Portal

5.2 Goals

The main goal behind the project is to integrate all the digital services given by Outotec in a web application. Digital services like Home, Asset hub, Service request, Document center and Logbook can be accessed from Customer Portal. User can check the health of assets from Asset Analytics and get the information about setbacks in the equipment.

The warnings provided by Asset analytics can reduce the risks of accidents in the company. Through the Service request, worker can request for the services, and, also, give feedbacks after request is resolved. Spare parts catalogue can provide manuals about different spare parts to the workers when needed.

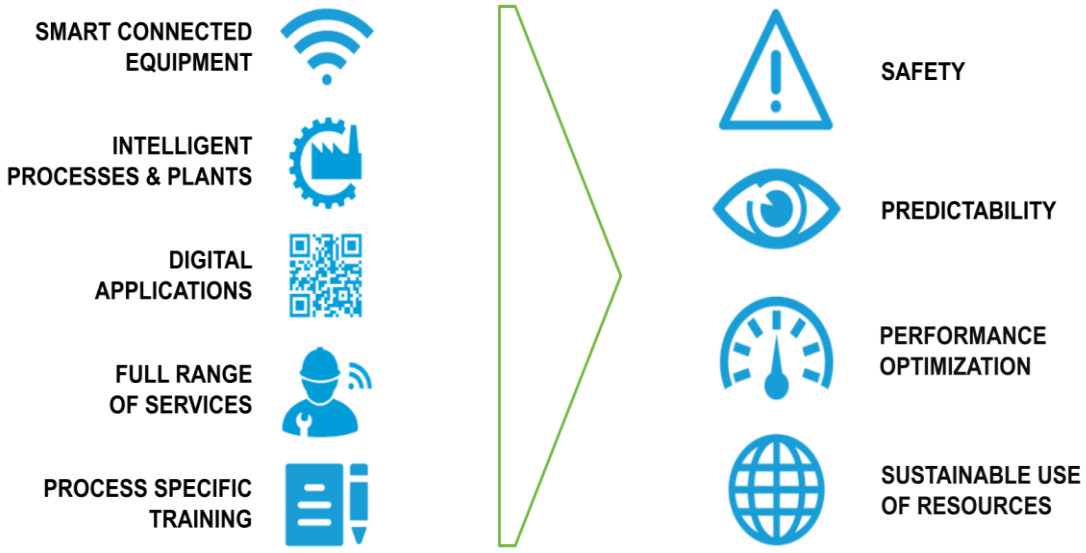

Figure 16. Benefits of Customer Portal (Outotec 2019)

As shown in Figure 16, there are multiple benefits of using Customer portal. Customer will get access to smart connected equipment, intelligent processes and plant, digital applications, services and process-specific training. This will ensure safety, increase predictability, optimize performance and ultimately the customer satisfaction. In the long run, this will also help to save the environment by providing solutions to use the resources effectively.

5.3 Development Process

In early 2019, development process of Portal started. A team was made with a group of developers, platform owners, designers, managers and marketers. All the team members started working in the specialized field. All the technologies and processes for the development were decided. UX designer started to make the design and developers started to code accordingly. Many features have been already implemented and others are still being developed. There will be more features added in future.

5.4 Integration of different technologies in Customer Portal

Customer portal is a gateway to all the digital services provided by the Outotec. To be a complete web application, Customer portal is using different modern technologies. These technologies are combined and used for the development of various parts of the application. It is a single-page application made with React. Node.js is used for the backend and databases are stored in MongoDB. Outotec has its own design system and has DSUI library as a collection of reusable components. Azure has been used for storing, testing, building and deploying the Portal. Integration of these technologies with Customer Portal and management tools are explained below.

5.4.1 Management tools

Platform owner creates multiple user stories as per the business needs and all these user stories and issues are stored in Jira. Jira is used as a management tool in Customer Portal. It comes with multiple features to store and manage user stories. Tasks are divided into two weeks sprint. Developers take task listed in Jira as per the priorities. Each task is done in separate branch. Once task is completed, assigned developer create pull requests in the branch and after approvals of at least two developers, the branch can be merged. Stories are set to be resolved after testing is done by tester. Then, stories are discussed and closed in the meeting by the platform owner. Agile makes the development process much easier by defining the tasks, tracking the progress of the work and reviewing tasks.

5.4.2 DSUI library

DSUI is a library, which includes all the components and the style guide for the different components like navbar, button, table and so on, used in all the application made by Outotec. In other words, DSUI is Outotec React Components. This library provides unified language and consistent look throughout all products.

5.4.3 Integration of Azure

The whole project, Customer Portal is stored in Azure. Each branch and push are tested with unit testing. Integration testing and deployment to production are also handled in Azure. As shown in Figure 17, all the APIs, Outotec React Components and Customer Portal development and production are stored in Azure.

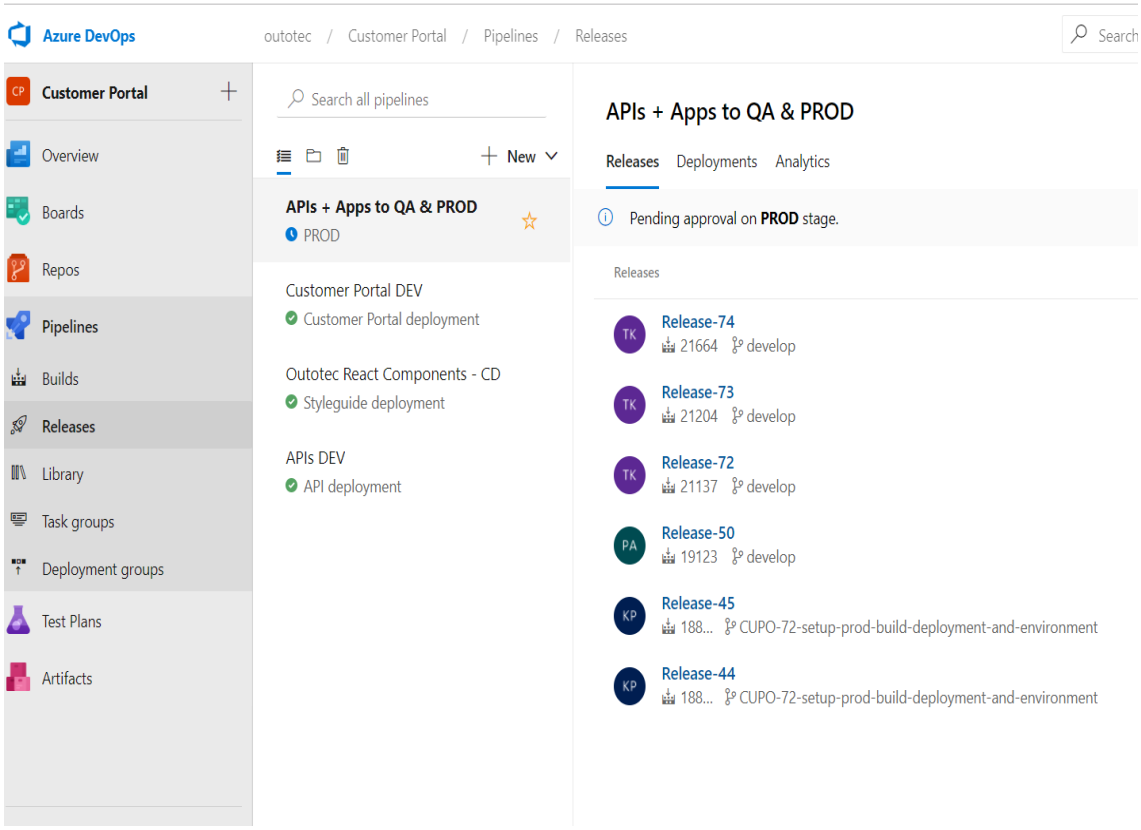

Figure 17. Screenshot of Azure DevOps of Customer Portal

5.4.4 Integration of React, TypeScript and Node

React is used as main frontend tool for the Customer Portal. All the states are stored and managed by redux. Typescript is used in frontend and backend for building user interfaces, and, testing the functionalities. APIs and databases are also handled using node and TypeScript. All the main components are imported from DSUI library. This makes the process of development of user interfaces easier.

Figure 18 shows the Customer Portal view in Visual Studio Code. Different modules are imported from React and other React libraries. Props are defined and used in main view.

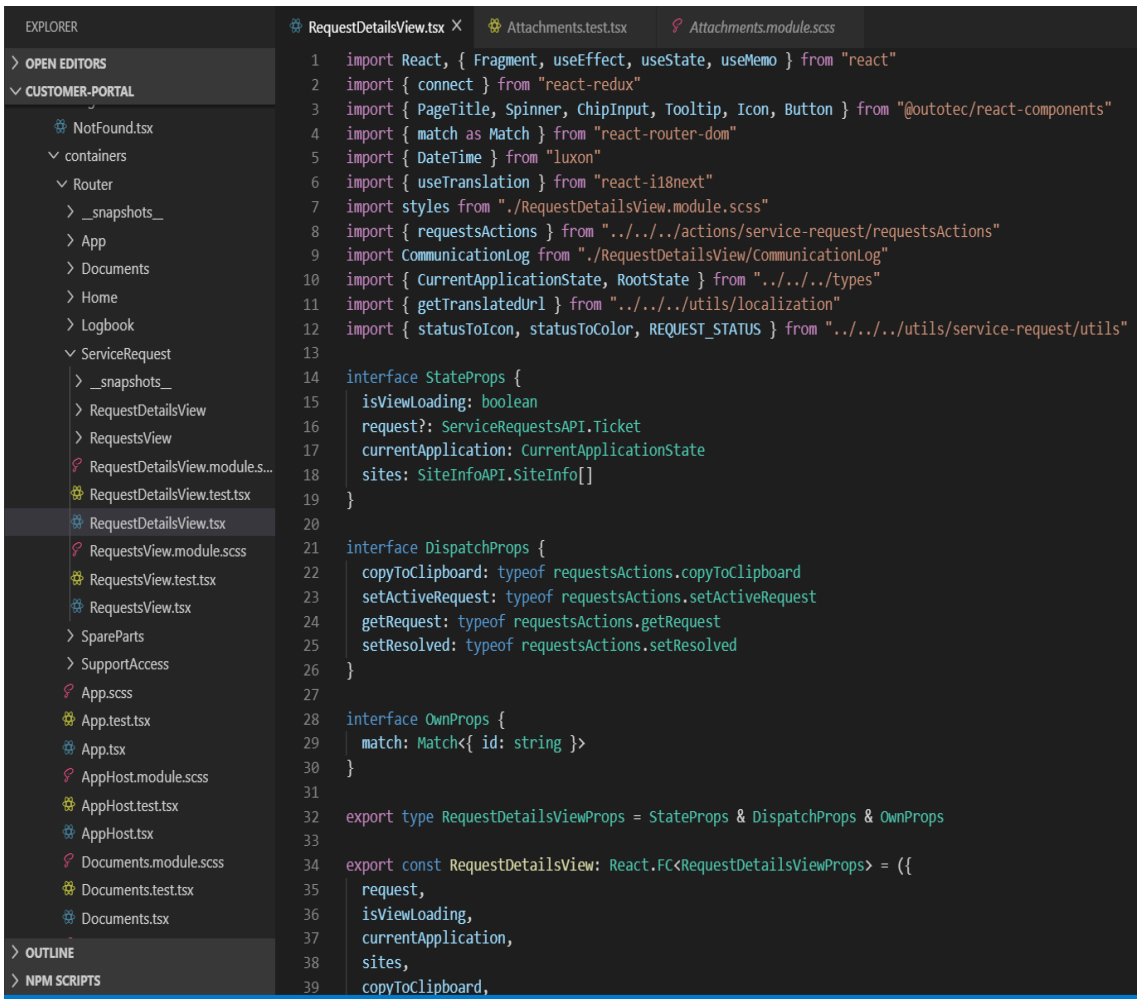

Figure 18. screenshot of Customer Portal project

5.4.5 Integration of MongoDB

All the documents about application are stored in MongoDB. It is used for inserting, updating, reading and deleting documents. Documents related to application registry, cached service requests, data access, events, events.vermongo, localizations, logbooks, service requests, shift groups, site info, and spare parts catalog are stored in MongoDB.

Figure 19 shows the screenshot of MongoDB databases of Customer Portal taken from MongoDB Compass Community.

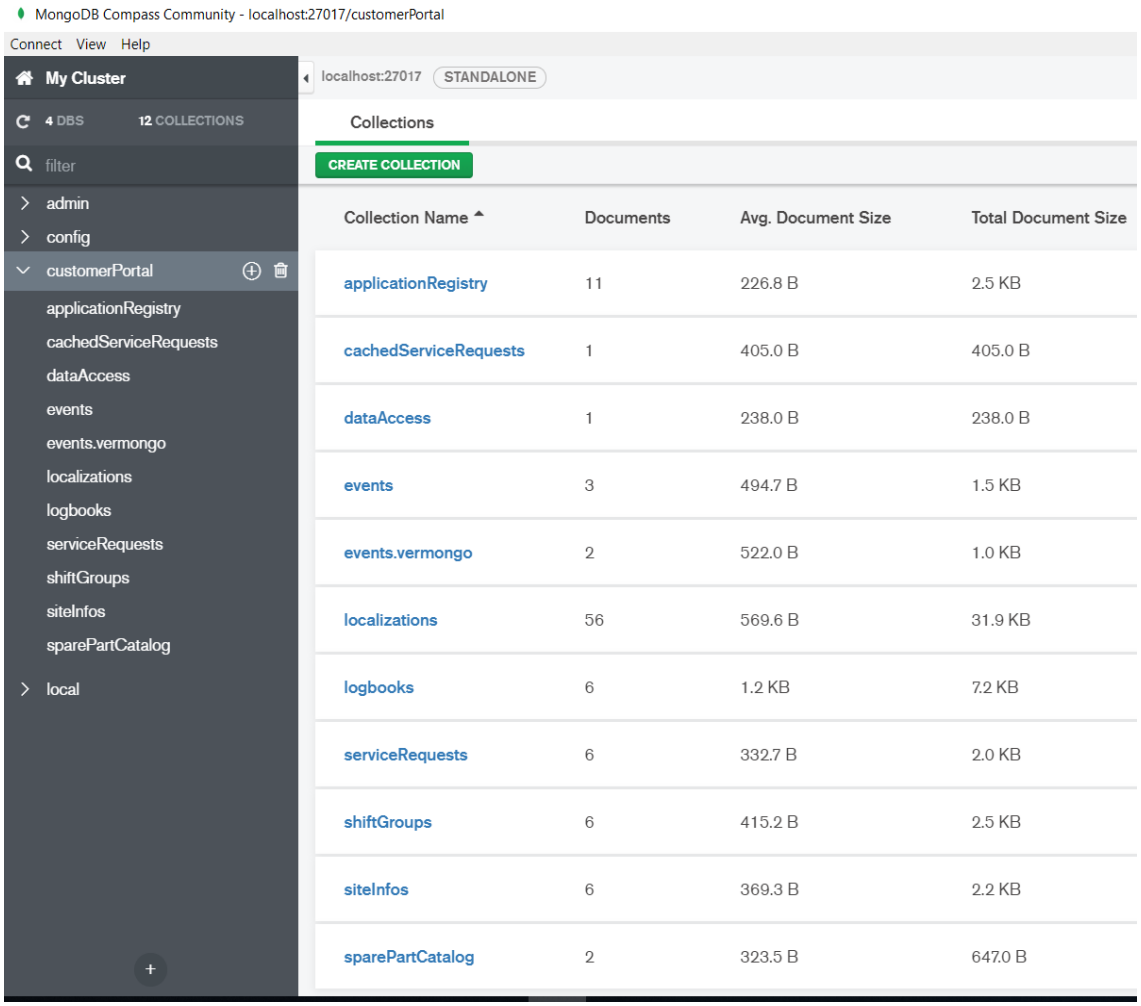

Figure 19. Mongodb database of Customer Portal

5.5 Platforms and applications

Customer Portal provides multiple platforms and applications to the customer. These are mainly made to provide services to customers so that, they can do their work easily and safely. These platforms are designed to ease out the work of customer in handling equipment, storing documents, getting manuals and detecting risks. All the designs used are made after interviewing the real customer. So, they are very simple and easy to understand.

Customer Portal has gone through the multiple changes to provide good user experience. Different platform refactoring has been done in order to be updated with modern web development practices including security. The application changed to single-page application later to lower the loading time of each services. Whole project was moved from Bitbucket to Azure for automated and integration testing. Designs changed as per the user needs.

Services of Customer Portal are Home, Asset hub, Document center, Collaboration space, Logbook and Service request. Landing page which is home, asset health which is Asset hub, service provider which is Service request, collection of documents which is Document center and information exchanging service which is Collaboration space has already been included in the Portal. There are many features coming in future. Detail description of current services are described below.

5.5.1 Home

Home Page is the landing page of the application. After logging into the customer portal with credentials, customers are first directed into the home. Home includes current date, place and weather in the header. There is information about news, insights, webinars and events in the right side of main body and service requests view in the left side of the body. If there are not any service requests then, service requests will be replaced by webinars and events. Furthermore, it has social media links and Outotec Support access information for getting support from Outotec in the bottom.

Figure 20 shows the screenshot of Homepage. Some of the information are hidden using grey rectangle to protect confidential information.

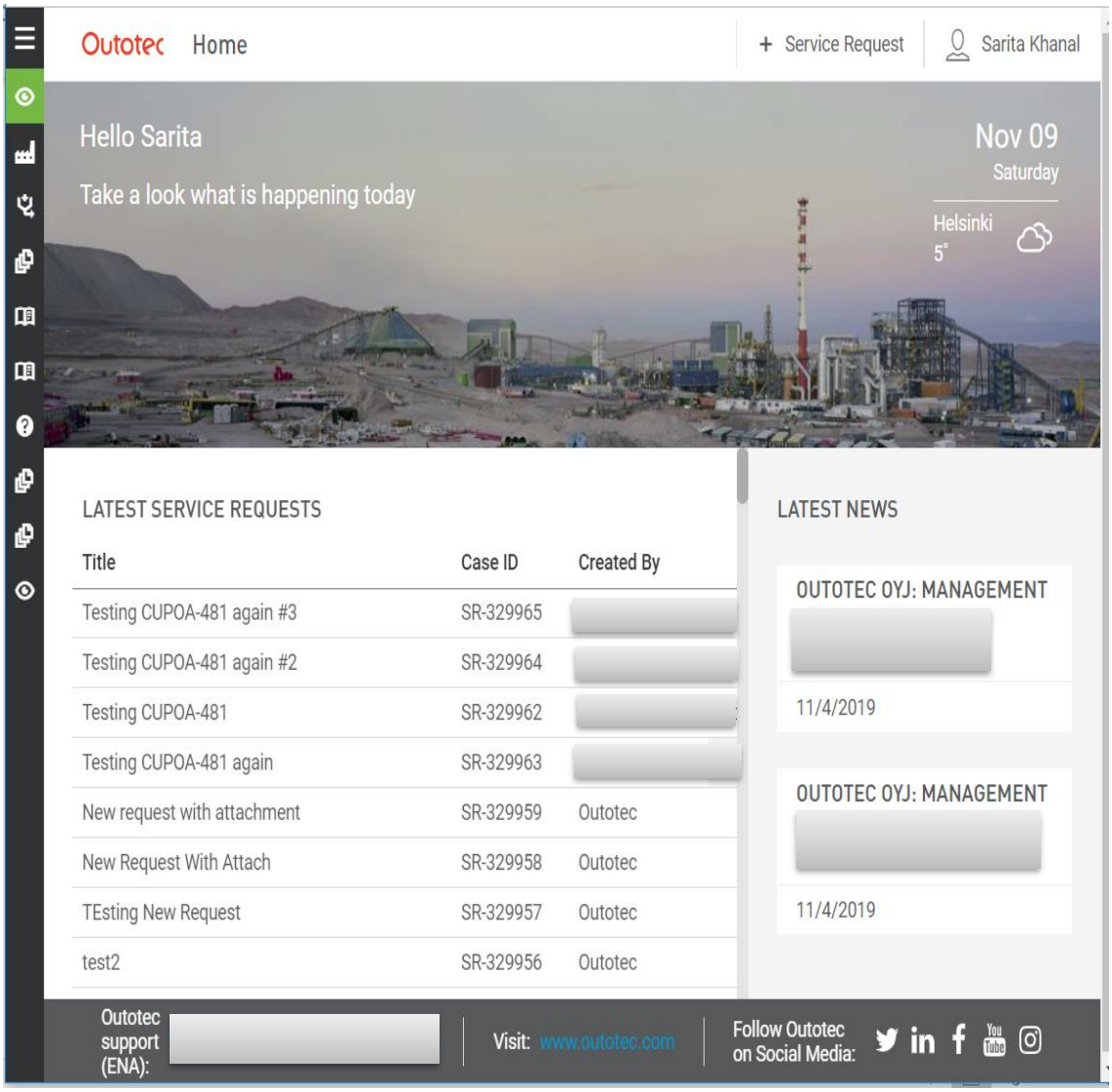

Figure 20. Screenshot of homepage

5.5.2 Asset hub

Asset hub is the collection of Asset analytics. Asset analytics is an analysis tool including information about the equipment used in customer companies. Through this tool, equipment health and detailed KPI information of the equipment are provided to customers in a simple and understandable form. This information alerts users for issues, monitors long-term performance, and enable corrective actions.

It delivers critical event-based equipment condition and operating data of equipment. It provides potential equipment problems before equipment failure minimizing unplanned

downtime and productivity losses. Operating costs can be controlled by streamlining service and maintenance schedules. Asset analytics is functioning across various assets including analyzers, grinding mills, flotation cells, filters and thickeners.

Moreover, with the information provided by Asset analytics, there is a possibility to focus on maintenance actions in problematic areas and systematically improve overall availability of the asset. Asset analytics can also trigger short-term maintenance actions using configurable notifications of equipment health. In short, the main features of Asset analytics are data collection, reporting, KPI calculations, online monitoring, advanced control and remote support.

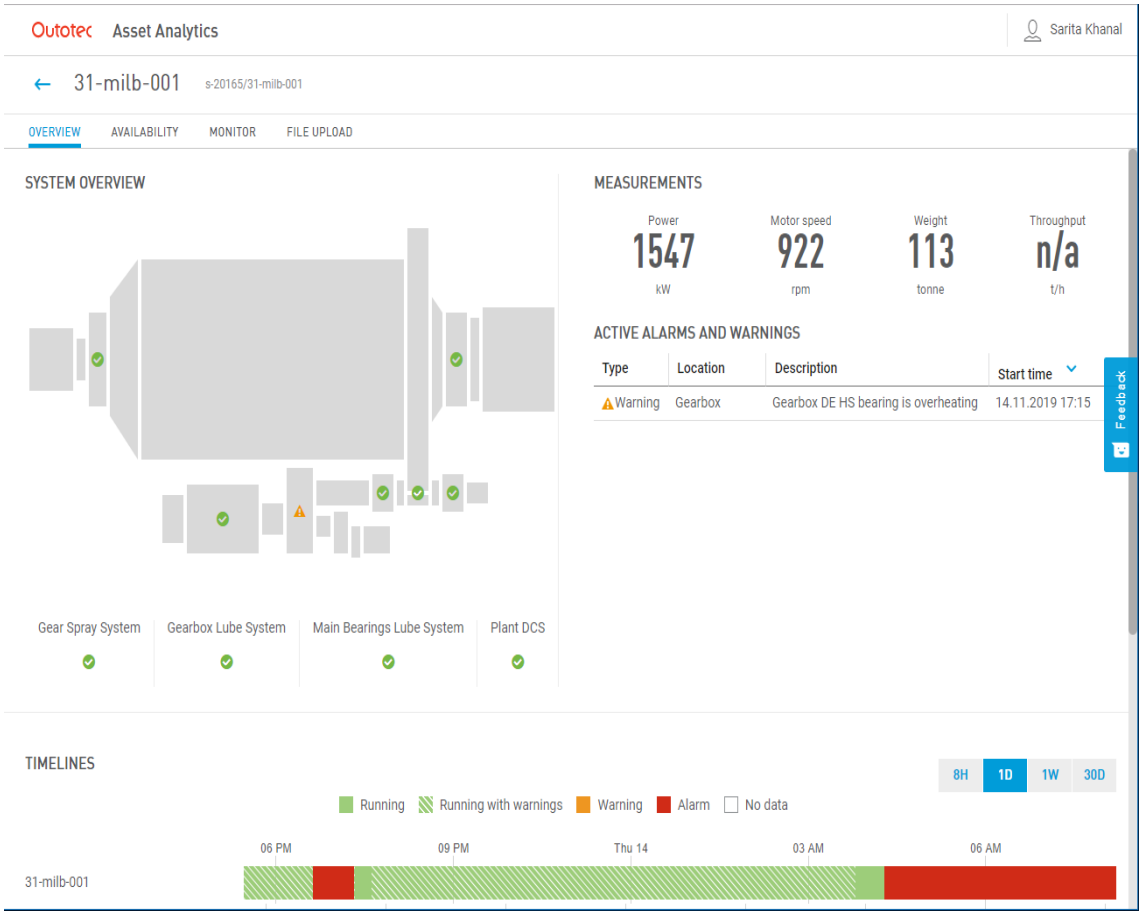

Figure 21. Screenshot of Asset analytics

Figure 21 shows the screenshot of Asset analytics. All the information about equipment power, speed, and weight are displayed. It also gives warnings about upcoming risks with location, description and time.

5.5.3 Document Center

Document Center is the collection of manuals, safety bulletins, documents and information. Customers can get different documents related to different site from document center. Different measurements are done in sites and these kind of information and reports are stored in Document Center. These documents can also be downloaded from Document Center.

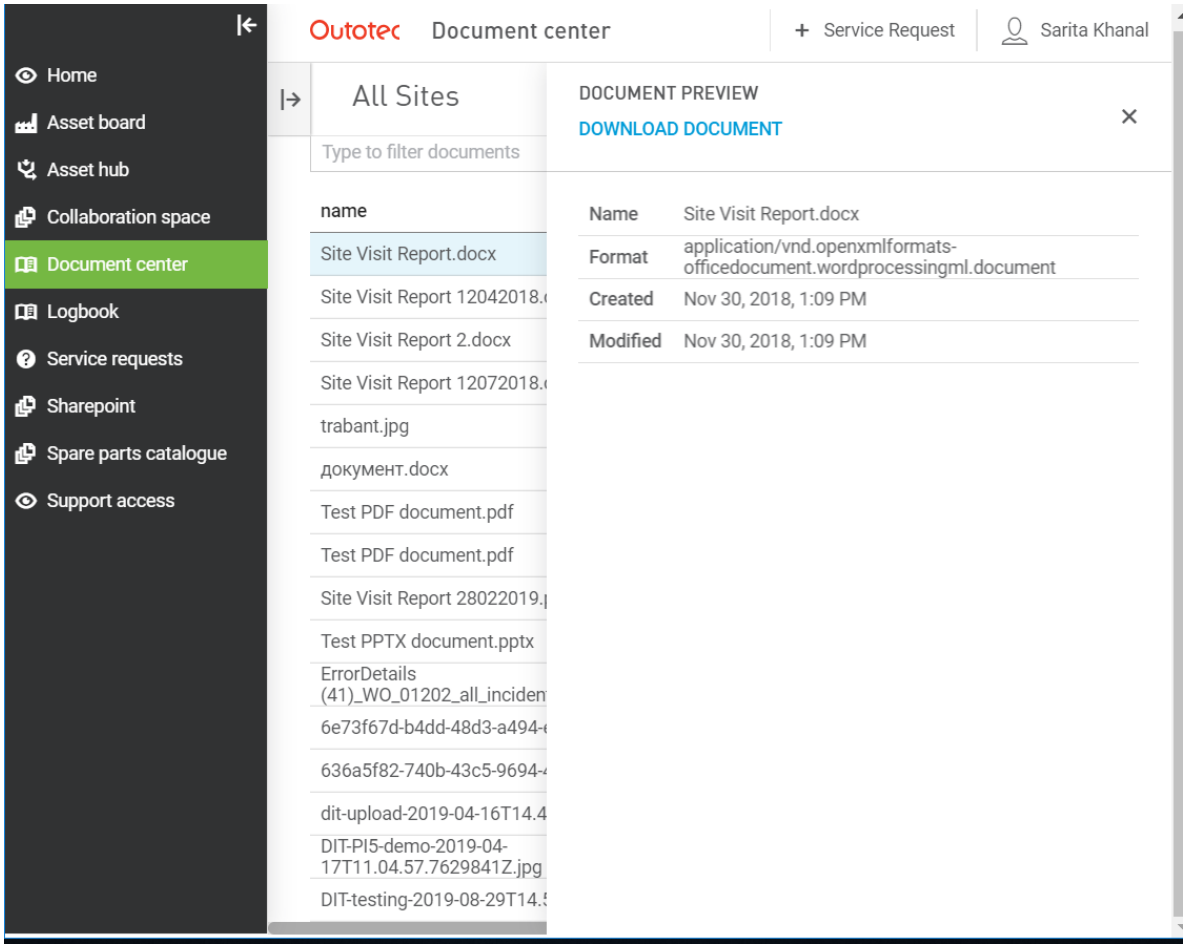

Figure 22. Screenshot of document center

Figure 22 shows the screenshot of Document Center. There are lists of documents of sites. On clicking documents, document preview can be seen and download link, name, format, created and modified date are included in the preview.

5.5.4 Logbook

The logbook addresses the need to share time stamped notes, observations or other textual information among the users, in a more orderly manner than a chat log would do. It can be configured according to the work shifts operating at the site, or by a flow of events. All relevant data for decision-making is made available in one place and logbook entries can be automatically combined in reporting. These entries can be accessed from anywhere and sharing the data is much easier. Typical use cases of Logbook are listed below.

- Communication between shifts with shift handover reporting
- Communication in start-ups as commissioning diary
- Incident reporting for any kind of exceptions in productions like machine failures, environmental issues and production issues.

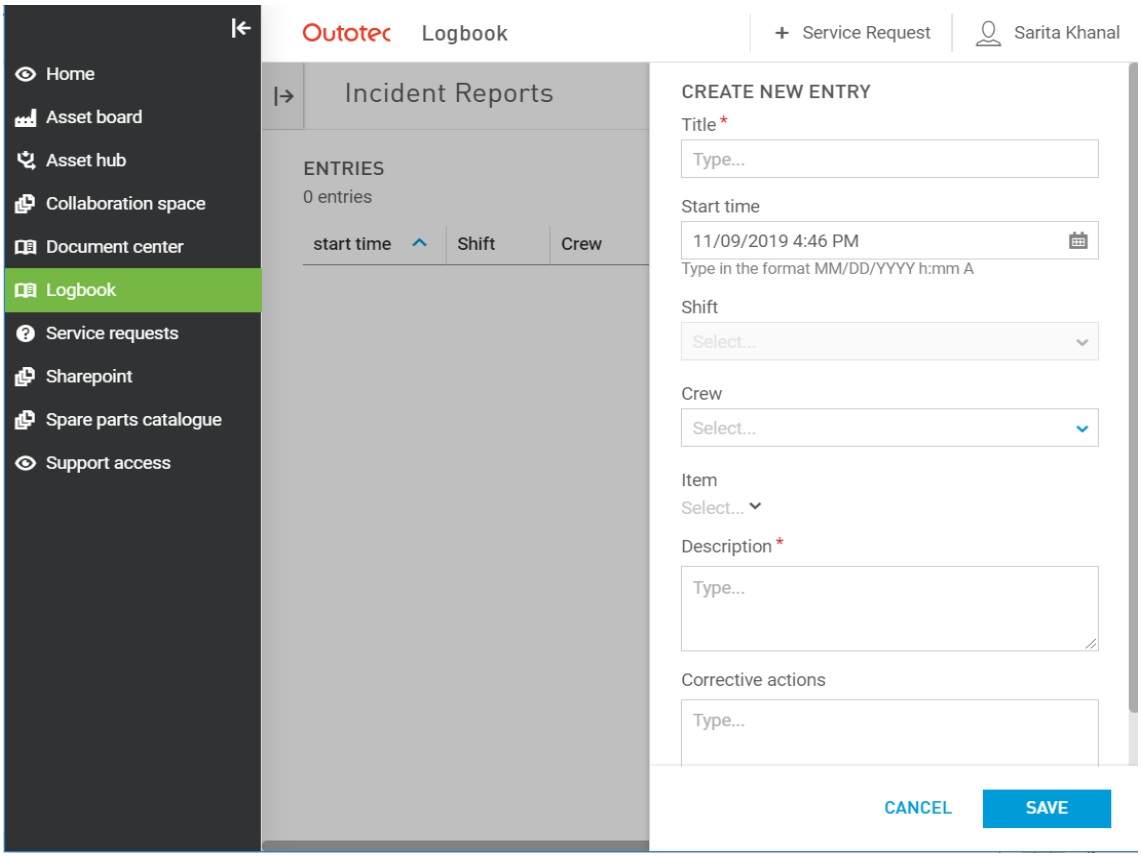

Figure 23. screenshot of logbook

Figure 23 shows the screenshot of logbook. Logbook can be created from create new entry. Entry includes title, start time, shift, crew, description and corrective actions.

5.5.5 Service requests

Service requests is one of the most important features of Customer Portal. Through the service requests, customers can create requests with the form. Requests can be about maintenance, ordering or other requests. Service Request Form is available in every route in the header, so, service requests can be accessible from every place of the application.

Service request form includes title, description, attachments and photos, request type, severity, people involved, equipment identification id and customer as shown in Figure 24. The requests can be seen in the Service requests. These requests can be filtered with statuses.

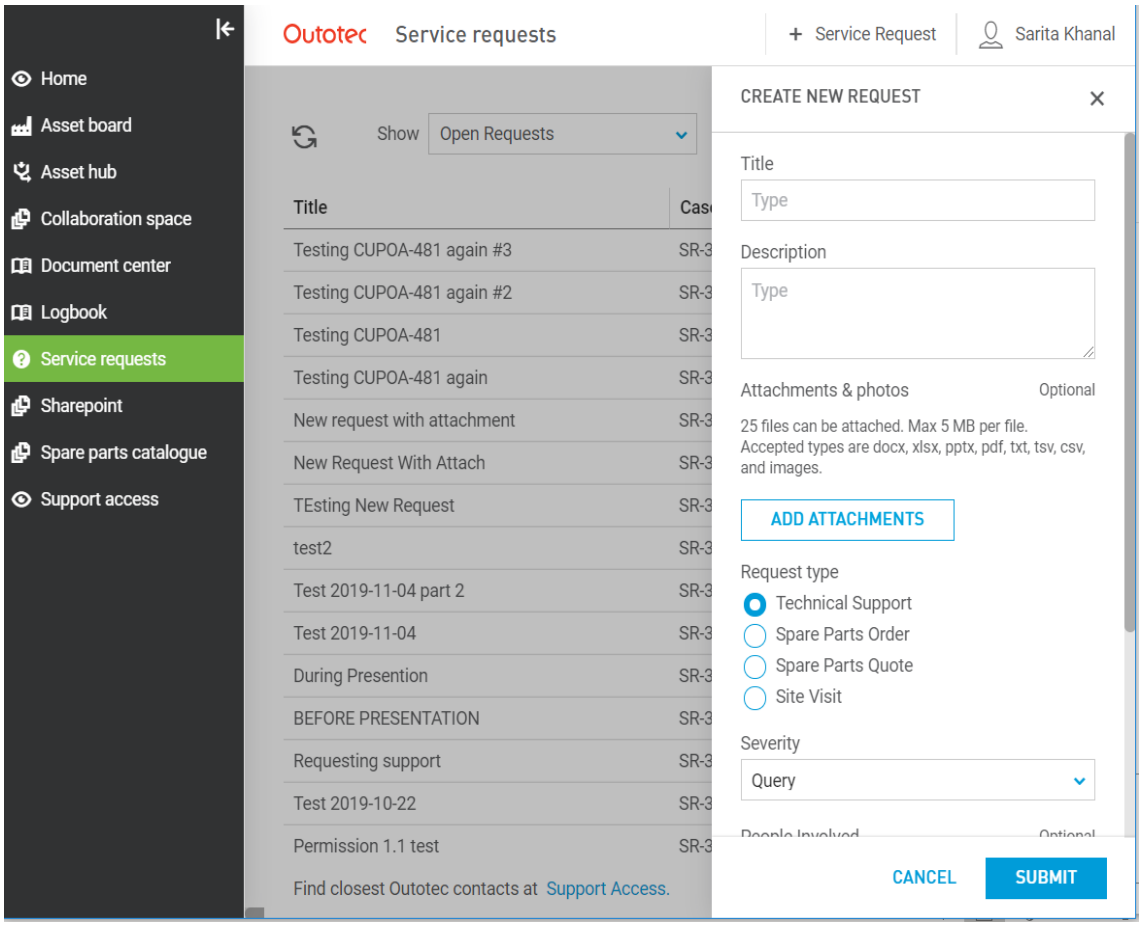

Figure 24. screenshot of service requests

Clicking service request takes to the full view of service request. Information about service requests like title, case id, created by, customer, created on, updated on, status, severity and type are in the left side and the chat box is in the right side.

Previously, all the emails and the replies were handled with emails but through the service requests, requests information can be seen in the same view page. After request is resolved, customer can request to resolve. Figure 25 shows all these details of service request. When resolved, customer can also give feedback.

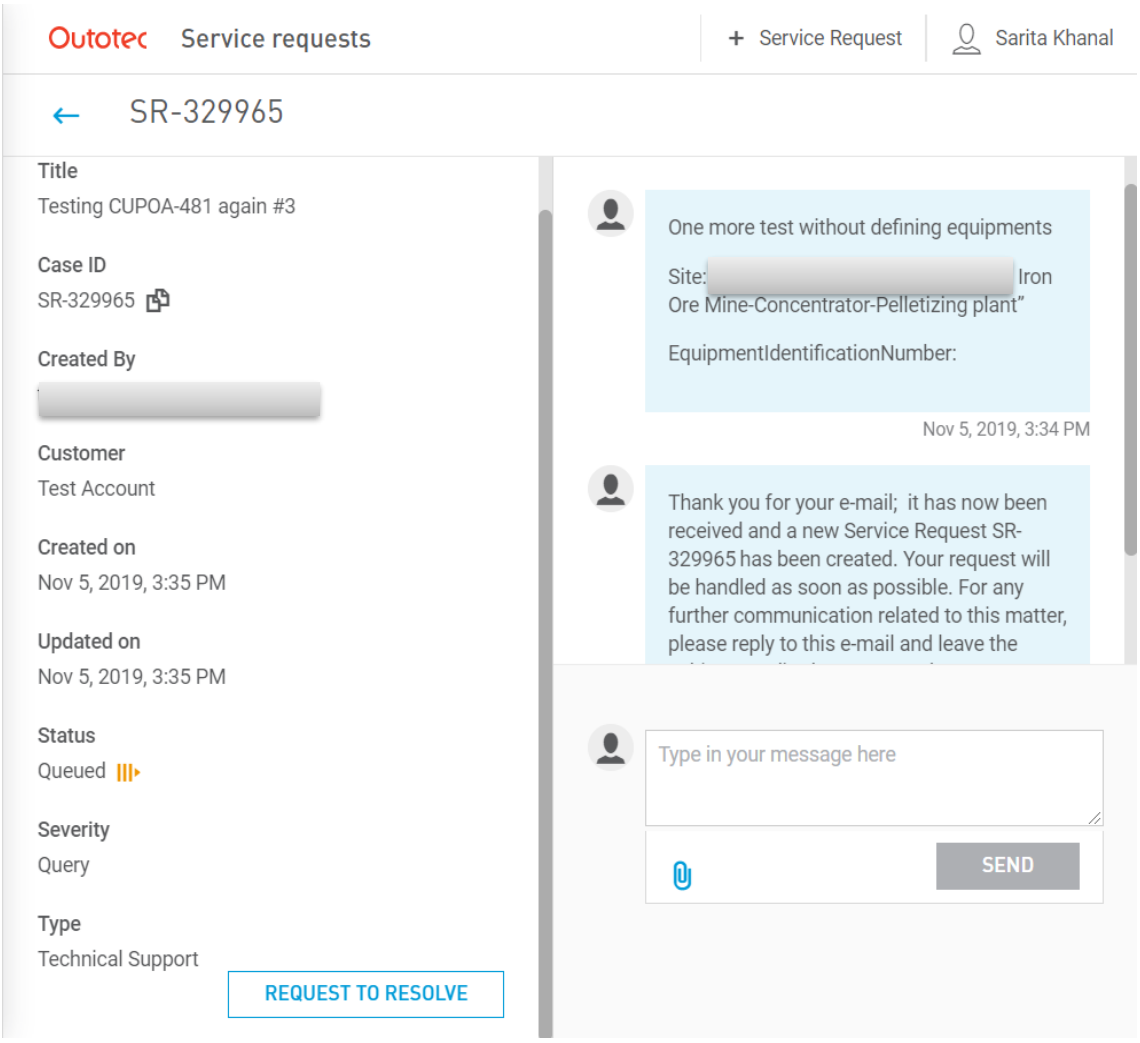

Figure 25. Screenshot of Service Request

5.5.6 Collaboration Space

Microsoft Teams is used as the Collaboration space in Customer Portal. This is used to connect with the other people and share the documents.

5.5.7 Spare parts catalogue

Spare parts catalogue is a customer site-specific collection of books that include lists, and manuals of spare parts for the delivered equipment. It also includes all the information about the spare parts used in sites. This can be accessed by authorized customer persons to view the relevant books but there is no pricing information about books included.

Figure 26 shows the screenshot of the spare parts catalogue. Main body includes equipment name and link to the catalogue.

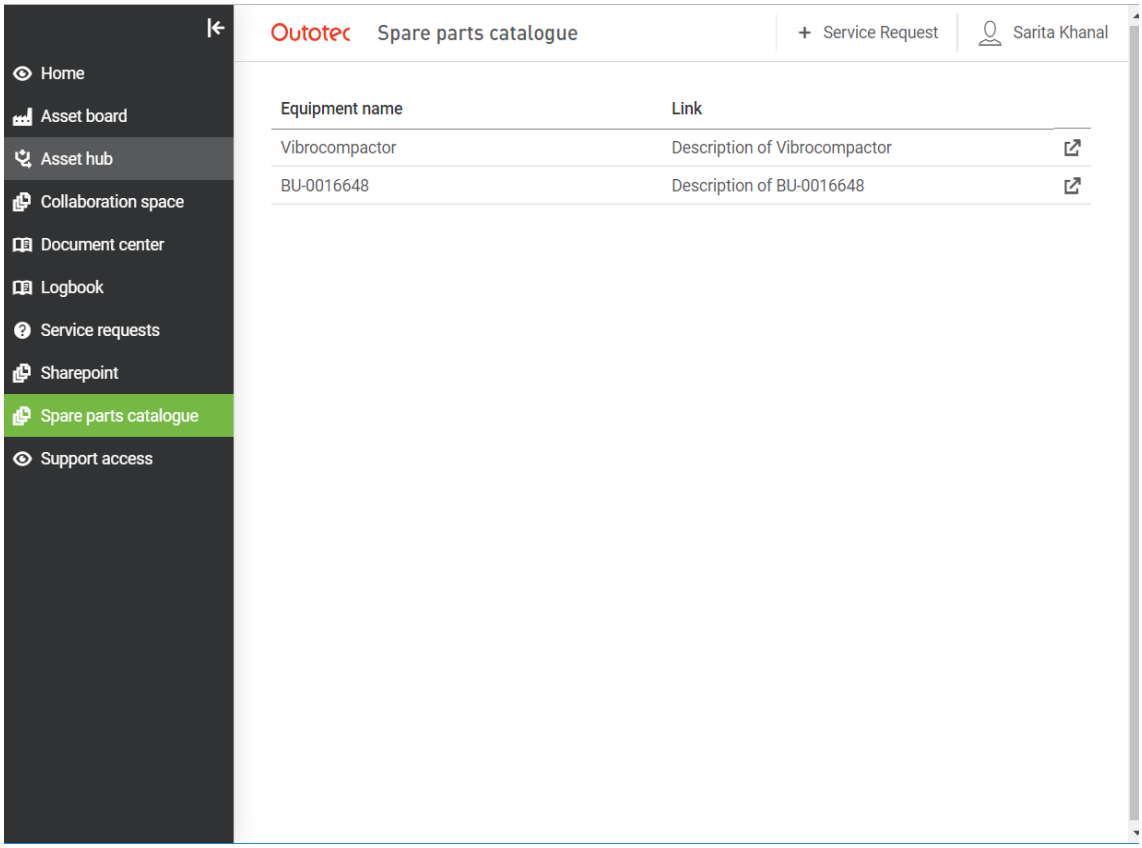

Figure 26. Screenshot of Spare parts catalogues

6 Conclusion

The main goal of this thesis was to dive deep into the digitalization in mining and metals industries with the project, Customer Portal. Customer Portal is a good example to prove the importance of digital applications in mining and metals. The service provided by Portal helps customers in difficult tasks like finding errors or predicting equipment conditions and productivity. It enables customer to analyze the equipment installed in different part of fields and to get the possible risks by staying in one corner of the world with just few clicks. Getting manuals and documents needed on process is also other benefits which can reduce the working time for certain tasks. Similarly, sending requests and getting help as soon as possible will eventually help to enhance productivity.

Looking through the digitalization and understanding the needs of digital transformation, one can easily predict that digitalization plays an important role for transforming these industries to become the environmentally sustainable and highly profitable. There might be many complications and problems in mining and metals industries because of the location, large equipment and even bad weather. So, one should always be careful to get the information and prevent the possible accidents. Technology provider like Outotec is also helping these industries to digitalize by providing different digital technologies and solutions to overcome the challenges.

This thesis is also an example for proving that digitalization is not only changing the world of technology and entertainment, but also other industries like mining and metals. Even mining and metals industries are gradually knowing the importance of digitalization and investing in digital solutions. Customer Portal is also one of the digital solutions being made for facilitating mining and metals industries.

Moreover, this thesis explores the modern technologies like ReactJS, TypeScript, Node.js, MongoDB and Azure used in web application. Integration of these technologies in the project are also presented in the thesis. All these technologies are most used technologies in the modern development practice. The thesis goes through each of the technologies and provides the insights of benefits of using them in web application. There is also the information about the management tool like Jira and agile system. The thesis also provides the information about how projects are handled in company.

References

Abramov, D. the R. documentation (2015). Three Principles : Redux. [online] Available at: https://redux.js.org/introduction/three-principles [Accessed October 20 2019].

Carvalho, F. P. (2017). “Mining industry and sustainable development : time for change,” (Reardon 2011), pp. 61–77. doi: 10.1002/fes3.109. Available at: https://www.researchgate.net/publication/317779578_Mining_industry_and_sustainable_development_Time_for_change [Accessed October 2 2019].

Copes, F. (2018). A quick guide to Redux for beginners - freeCodeCamp.org - Medium. [online] Available at: https://medium.com/free-code-camp/a-quick-guide-to-redux-for-beginners-971d18c0509b [Accessed October 20 2019].

Dr. Michael J. Garbade (2016). Angular vs React vs Node - which framework the best? [online] Available at: https://blog.education-ecosystem.com/angular-js-vs-react-js-vs-node-js-which-is-the-best/ [Accessed October 12 2019].

Eschweiler, S. (2017). Learn Redux — Introduction To State Management With React [online] Available at: https://medium.com/codingthesmartway-com-blog/learn-redux-introduction-to-state-management-with-react-b87bc570b12a [Accessed October 20 2019].

Google trends (2019). TypeScript - Explore - Google Trends. [online] Available at: https://trends.google.com/trends/explore?date=today 5-y&q=%2Fm%2F0n50hxv [Accessed October 12 2019].

Menear, H. (2018). Top 10 sustainable companies | Energy Digital - Energy News, Magazine and Website. [online] Available at: https://www.energydigital.com/top10/top-10-sustainable-companies [Accessed October 1 2019].

Microsoft (2019). Microsoft Azure Documentation | Microsoft Docs. [online] Available at: https://docs.microsoft.com/en-us/azure/ [Accessed October 20 2019).

Minalliance and Michel, J. (2012). Innovations In the Mining Industry. French edi. Mining Association of Canada, The Prospectors and Developers Association of Canada. Available at: http://www.oma.on.ca/en/ontariomining/resources/minalliance_100_innovations_en.pdf [Accessed October 10 2019]

Mitchell, P. (2019). Top 10 business risks and opportunities – 2020. [online]. Available at: https://www.ey.com/en_gl/mining-metals/10-business-risks-facing-mining-and-metals [Accessed October 11 2019].

Node (2019). About | Node.js. [online]. Available at: https://nodejs.org/en/about/ [Accessed October 20 2019].

Noterdaeme, O. et al. (2018). "Mapping heavy industry's digital-manufacturing opportunities | McKinsey." Available at: https://www.mckinsey.com/business-functions/operations/our-insights/mapping-heavy-industrys-digital-manufacturing-opportunities [Accessed October 15 2019]

Outotec (2019). About Outotec - Outotec. [online] Available at: http://www.outotec.com/company/about-outotec [Accessed October 10, 2019].

StackOverflow (2018). MongoDB Notes for Professionals. doi: 10.1016/S0148-0685(81)95186-1. Available at: https://goalkicker.com/MongoDBBook/ [Accessed October 22 2019].

StackOverflow (2018). React JS Notes for Professionals. GoalKicker.com. Available at: https://goalkicker.com/ReactJSBook/ [Accessed October 16 2019].

StackOverflow (2018). TypeScript Notes for Professionals. Available at: https://goalkicker.com/TypeScriptBook2/ [Accessed October 17 2019].

World Economic Forum and Accenture (2017). Digital Transformation Initiative Mining and Metals Industry In collaboration with Accenture-transformation 3 Digital Transformation Initiative: Mining and Metals. Available at: http://www.outotec.com/company/about-outotec [Accessed October 8 2019].

Sample Codes of Project

Snippet of code of Service Request view

```
import React, { Fragment, useEffect, useState, useMemo } from
"react"
import { connect } from "react-redux"
import { useTranslation } from "react-i18next"
import { Link } from "react-router-dom"
import { Spinner, Loader } from "@outotec/react-components"
import ServiceRequestsTable from "../../../components/Service-
RequestsTable/ServiceRequestsTable"
import RequestsFilters from "./RequestsView/RequestsFilters"
import {
  filterByProperties,
  filterByRequestStatusTypes,
  requestStatusTypes,
  RequestStatusTypes,
} from "../../../utils/service-request/utils"
import styles from "./RequestsView.module.scss"
import { RootState, ServiceRequest } from "../../../types"
import { requestsActions } from "../../../actions/service-re-
quest/requestsActions"

interface StateProps {
  requests: ServiceRequestsAPI.Ticket[]
  isViewLoading: boolean
  sites: ServiceRequest.SiteState
}

interface DispatchProps {
  getRequests: typeof requestsActions.getRequests
}

export type RequestsViewProps = StateProps & DispatchProps
export const RequestsView: React.FC<RequestsViewProps> = ({ is-
ViewLoading, requests, sites, getRequests }) => {
  const { t } = useTranslation("serviceRequest-ns", { useSus-
pense: false })
  const [filterText, setFilterText] = useState("")
  const [requestStatus, setRequestStatus] = useState<RequestSta-
tusTypes>(requestStatusTypes.OPEN)
  const filteredRequestsByStatus = useMemo(() => filterByRe-
questStatusTypes(requests, requestStatus), [
    requests,
    requestStatus,
    filterByRequestStatusTypes,
  ])
  const filteredColumns: ServiceRequestsAPI.StringKeys[] = ["ti-
tle", "customerName", "caseNumber", "status", "createdBy"]
```

```
  const filteredRequests = useMemo(
    () => filteredRequestsByStatus.filter(request => filterBy-
Properties(request, filteredColumns, filterText.toLowerCase(),
t)),
    [filteredRequestsByStatus, t, filterText, filteredColumns]
  )

  useEffect(() => {
    getRequests()
  }, [getRequests])

  const noRequestText = useMemo(
    () =>
      requestStatus !== requestStatusTypes.ALL
        ? requestStatus === requestStatusTypes.OPEN
          ? t("thereAreNoOpenRequests")
          : t("thereAreNoResolvedRequests")
        : t("thereAreNoRequests"),
    [requestStatus, t]
  )

  return isViewLoading && !requests.length ? (
    <Spinner />
  ) : (
    <Fragment>
      <div className={styles.container}>
        <Loader loading={isViewLoading} />
        <div className={styles.requestsTableContainer}>
          <RequestsFilters
            filterText={filterText}
            showRequest={requestStatus}
            onChangeFilterInput={(_, { value }) => setFilter-
Text(value)}
            onChangeShow={setRequestStatus}
            onRefresh={() => getRequests()}
          />
          <ServiceRequestsTable loading={isViewLoading} re-
quests={filteredRequests} sites={sites}
getRequests={getRequests} />
          {filteredRequestsByStatus.length === 0 && !isViewLoad-
ing && <div className={styles.noRequestText}>{noRe-
questText}</div>}
        </div>
        <div className={styles.siteManagerInfo}>
          {t("findClosestOutotecContactsAt")} 
          <Link to="/support-access">{t("supportAccess")}</Link>
        </div>
      </div>
    </Fragment>
  )
}
```

```
export default connect<StateProps, DispatchProps, {}, Root-
State>(
  ({ serviceRequest: state }) => ({
    requests: Object.values(state.requests.data),
    isViewLoading: state.requests.isViewLoading,
    sites: state.sites,
  }),
  {
    getRequests: requestsActions.getRequests,
  }
)(RequestsView)
```