SAALASTI WEB UI FOR MONITORING LIVE DATA FROM BIOMASS EQUIPMENT

NAZIA HASAN – UI / UX DESIGNER
SAMPO SOFTWARE OY
Tampere, Finland

*"It is always filled with excitement and anticipations when a newbie UX Designer gets her first work to design a service from a scratch. This is the story behind **Saalasti Oy's web service** to read and monitor live data and statuses of their bioenergy producing machineries. My very first work it might be, but I still consider this as one of my best works.*
Enjoy!"

Saalasti Oy is a Finnish company expertized in creating and manufacturing heavy duty industrial machineries since 1945. Being a pioneer in bioenergy products, Saalasti designs and produces equipment to process forest-based biomass like stumps, loggings, residues and tree barks into boiler fuels. Their product range includes stationary forest-based **biomass chipper**, **crushers**, **cutters**, **bark dewatering presses** and the entire biomass **processing stations**.

Picture courtesy: https://saalasti.fi/

The Design Problem

Maintenance and servicing of such machineries and associated components are mammoth tasks since they are **prone** to **environmental conditions** (e.g. heat, moisture, temperature, etc.) and **factors** (e.g. amount of fuel, wet/dry condition, size of the chunks, etc.) associated with the **biomass** that needs to be processed. Every day, from a combined processing station, live data on statuses of different machines and their components are fed into a central database server. If an alarm associated with an equipment or component goes off, service engineers are sent on spot for repair works. Often the machineries and their component require periodic servicing and maintain the service logs is essential.

The limitations across the workflow were the following:

No Online Service for remotely monitoring the bioenergy equipment in real-time

With biomass equipment located at various locations in Europe, it was difficult to **remotely keep an eye** on the machineries and **identify**

potential emergency situations only with information from Database and without any visual cues.

No means of storing historical data

For a complex equipment system, **historical data**, presented with appropriate visualization helps determining **equipment performance**, **forecasting emergency service needs** and **identifying potential service breakdown**. Saalasti's bioengineering equipment and components are comprised of different **alarms** and **sensors** indicating their **status** (online, working mode, emergency breakdown, etc.). The lack of a remote monitoring service resulted with **no means of displaying information** acquired from these alarms and sensors.

No means of recording maintenance information

It is important to keep **record of maintenance services** (performed periodically or in emergency basis) for the equipment and their components. There needs to be provision of **accessing** and **updating** this information when needed. With the lack of a remote monitoring system, such information only existed in paperwork and was accessible to only a few people.

No availability of equipment information to stakeholders.

Saalasti's bioenergy equipment can be customized with additional components. So it is important for an **owner** to know what his/her **machinery is comprised of**, along with their **performance information** and **maintenance works** conducted. This is also important for **service engineers** in charge of maintaining respective equipment units. For **Saalasti as a company**, this information becomes helpful while demonstrating available products to potential customers. However, lack of remote service with equipment profiles created barriers for stakeholders to acquire such information on relevant equipment.

Our Design Challenge

*Saalasti Oy required a **web service** which could be used to **remotely monitor** their biomass processing equipment with **live data feeds**. The service would comprise of appropriate **visualizations** to showcase the **machineries** and associated **components**, their **status, usage trends, maintenance log** and other information. The web service should facilitate **adding new equipment** and maintain separate **profile** for them. Equipment owner, service engineers in charge of maintaining the machinery should be able to access related information and to determine how the equipment are performing. In case of* *__emergency__ or service request, __notification__ should be sent to assigned maintenance engineer.*

Barriers We Faced

In the beginning of the implementation work for Saalasti's web portal, out team had a client meeting where we acquired details on what features and functionalities are expected out of the new service. As the **UX designer** for the project, I asked **questions** to the client on **how the machineries are monitored** and **maintained** at that time, how owner of an equipment gets **updates** or **reports** about its **performance** and **status**, how people in charge of service maintenance **assess the situation** of **regular** or **emergency maintenance works**, etc. After the meeting, I along with my team-lead assessed the overall situation and identified the following impediments:

Difficulties in conducting proper user research

Due the **outlying locations** of the actual equipment, it was not possible for us talk with people in charge of the maintenance of the machineries. Because of the **privacy policy**, it was difficult for us to contact Saalasti's customers (owning biomass equipment) to get their opinions and expectations. We had to rely on Saalasti's representative to understand about customer profile and the nature of the maintenance engineers' works. Time to time, I reached out in social media to look for people from known associates to find out if anyone had prior experience in using similar solutions. The purpose was to gain some insights on users' monitoring work of industrial machinery using some existing services, barriers they face in their works and what they expect out of a new solution.

Understanding data feed obtained from Saalasti's Database

We acquired a set of information from Saalasti's database which included possible attributes, alarms associated with biomass machinery units and their components. We had to figure out which **attributes and alarms** are **crucial to determine status of an entire unit** or **individual components**. Also we needed to **determine measurement** units associated with these values.

Separate Locations of Design and Development Team

Our design team was located in Tampere Finland and the development team was located in Russia. On premise discussions about designing ideas with development team was seldomly possible. Delivering the wireframes to the remote team, having discussions with them on the design choices required additional effort.

Our Process

In order to develop the design concept for the web portal and overcome the challenges mentioned above, we decided to have **separate 1-week long design** and **1-week long development sprints** in the overall implementation works. The 1st design sprint took place immediately after the Saalasti client meeting and it was ahead of the 1st development sprint. Some works of the **1st design sprint** included **brainstorming** among my team lead and I. from the ideas emerged, I developed **two primary personas**: **Maintenance Engineer** and **Equipment Owner**. My team lead helped me shape up the **use-case-scenarios** and **conceptualize** the **possible features** within the web service. **Each feature** was added as **backlog item** in Jira with associated **use-cases**. According to the priority of the backlog items, I produced an initial set of **low-fidelity prototypes** using **Balsamiq** mimicking the use-cases. The client reviewed the personas, use-case scenarios and wireframes and gave his feedbacks, which in turned helped me to revise the related materials. At the end of 1st spring I created **high-fidelity wireframes** using **Adobe Photoshop** describing **different views** of the web service. Afterwards, my team lead and I had a meeting with the remote development team in Russia where we discussed about the use-case scenarios and how the compiled design accommodated them. With a few technical modifications from their side, the development team started implementing the web-portal.

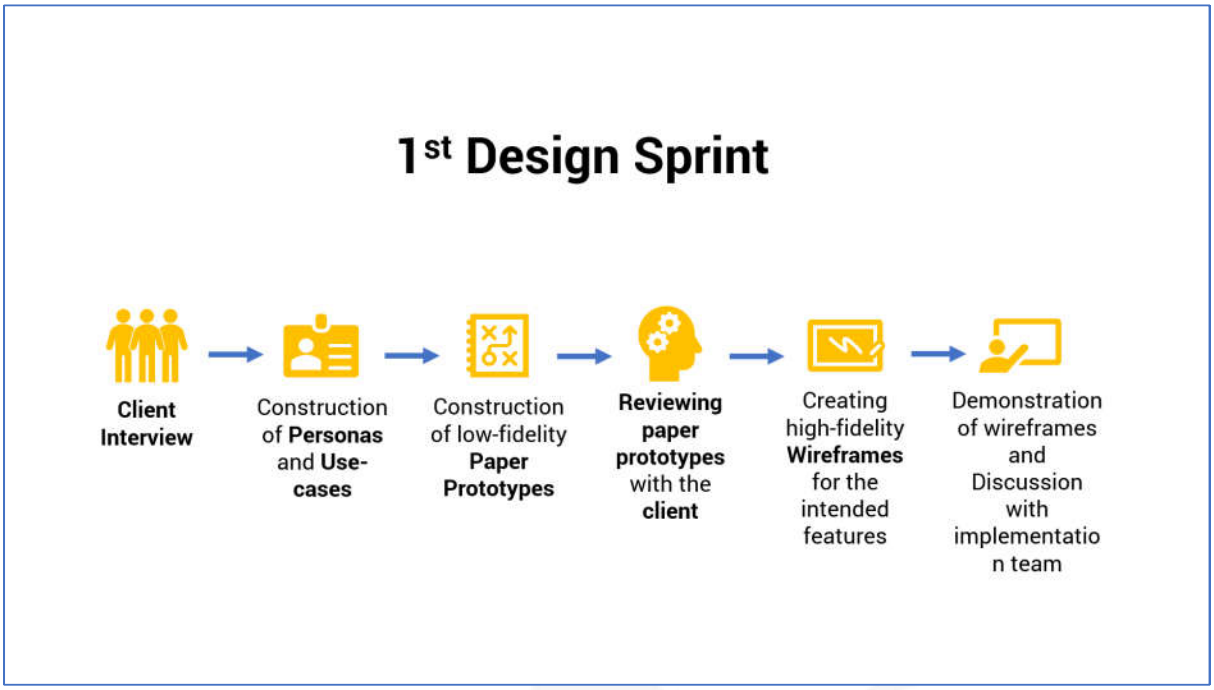

Figure 1 1st Design Sprint of Saalasti Web Service. Persona, use-cases, low fidelity prototypes and initial wireframes were created in this sprint

After the 1st design sprint, I followed the workflow below in the consecutive ones:

1. Preparing **high fidelity wireframes** for the **UI of new features** to be developed.
2. **Reviewing** the **wireframes** with our team lead to verify if the UI and actions described are complying with the use-case scenarios.
3. **Making changes** to the **UI and actions** in the wireframes according to the feedback and delivering them to the development team.
4. Along with my team-lead **attending demo** given by the development team on **implemented features and functionality**. Also providing feedback about their works.
5. **Discussing the designs** for new features and providing **improvement suggestions** to implemented features.
6. Doing quick **heuristic evaluation** along with my team-lead to identify potential issues on implemented features.

Figure 2 One of the personas were a maintenance engineer in charge of taking care of the biomass equipment.

We maintained an **IM channel** in RocketChat with the development for **communication** and quick feedbacks. Also we held **online scrum meetings** with the entire team in **every 2 days** to convey our work **progresses** and **problems** we faced.

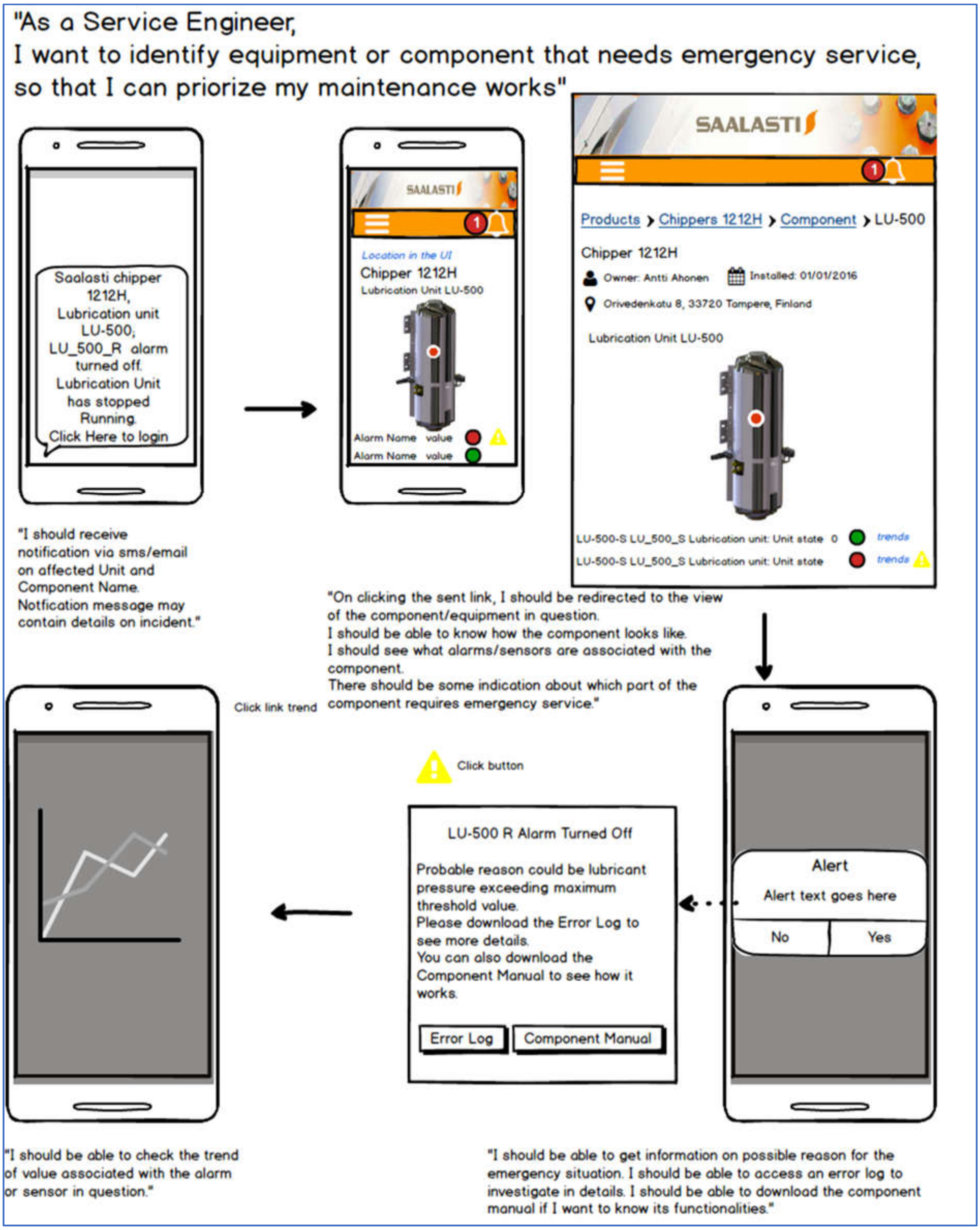

Figure 3 One use-case for service engineer persona was to get notification for emergency maintenance work. The prototype mock-up shows possible user-flow in this scenario and how the intended UI accommodates this.

The design team had a **monthly meeting with Saalasti's client** to show the progress of the implemented work. In each meeting, our team lead demonstrated the implemented functionality alongside their use-cases to the client. The client reviewed the work and provided his opinion on what was implemented accordingly and what needed to be improved or modified. He also prioritized the backlogs and talked about new requirements resulting with new sets of features. In addition, He reviewed wireframes in the backlog and gave his approval or improvement suggestions. The **feedback** from client helped me to **revise the personas** and **use-cases**. These in turn helped me **make necessary changes** to the **UI wireframes**.

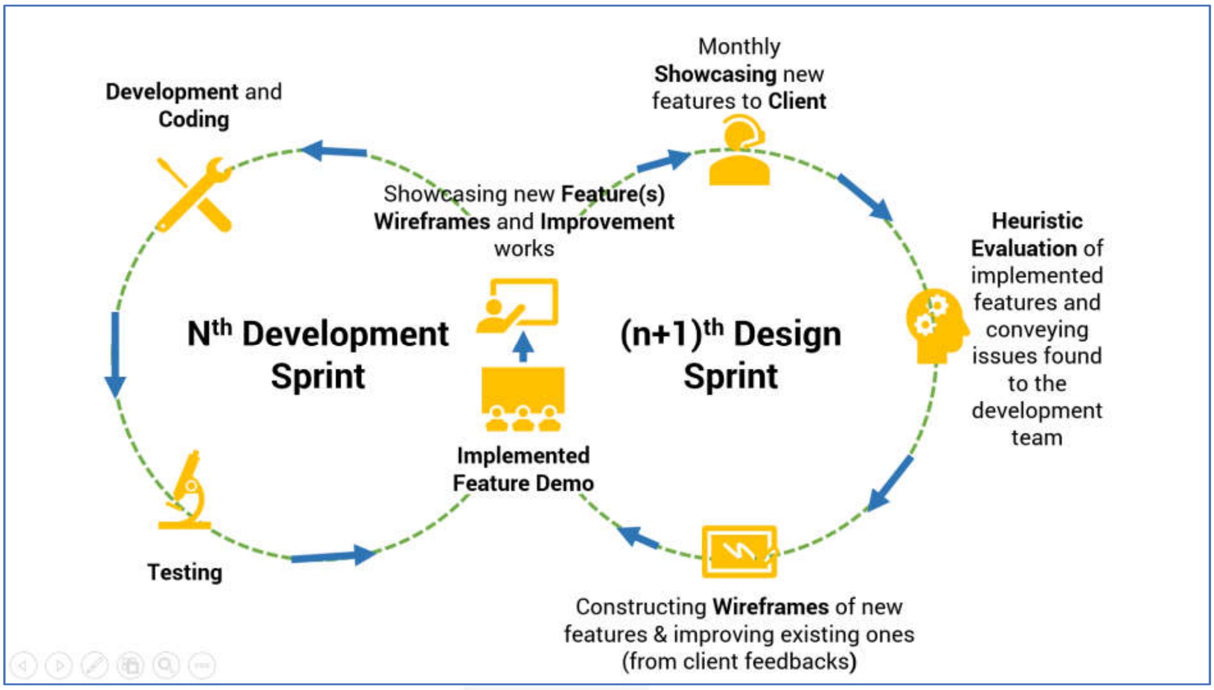

Figure 4 Parallel progress of Design sprint and Development sprint during the implementation of Saalasti's web service. Except the client showcase, all other phases occurred weekly.

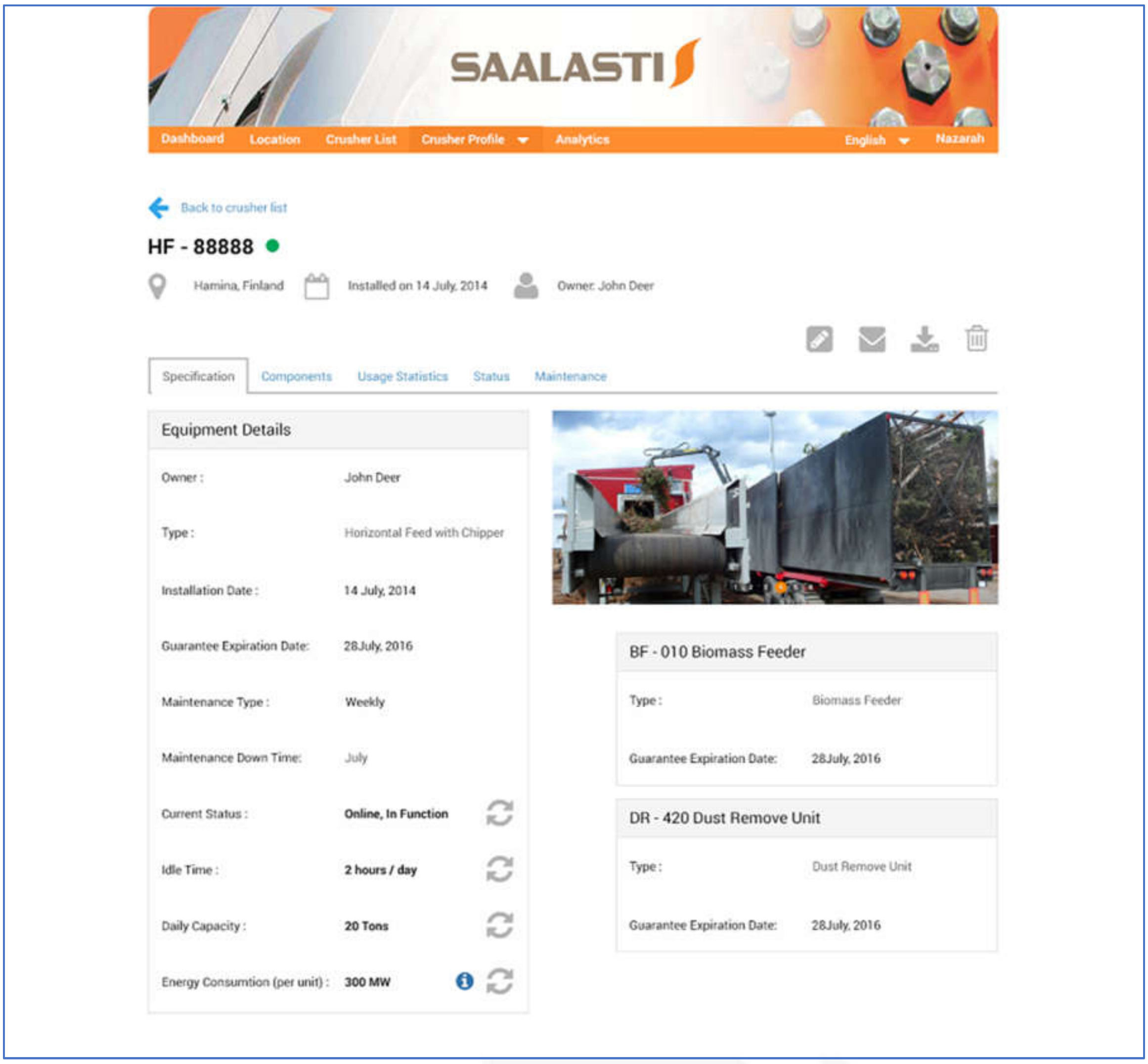

Figure 5 High fidelity wireframe showcasing profile page for a biomass equipment

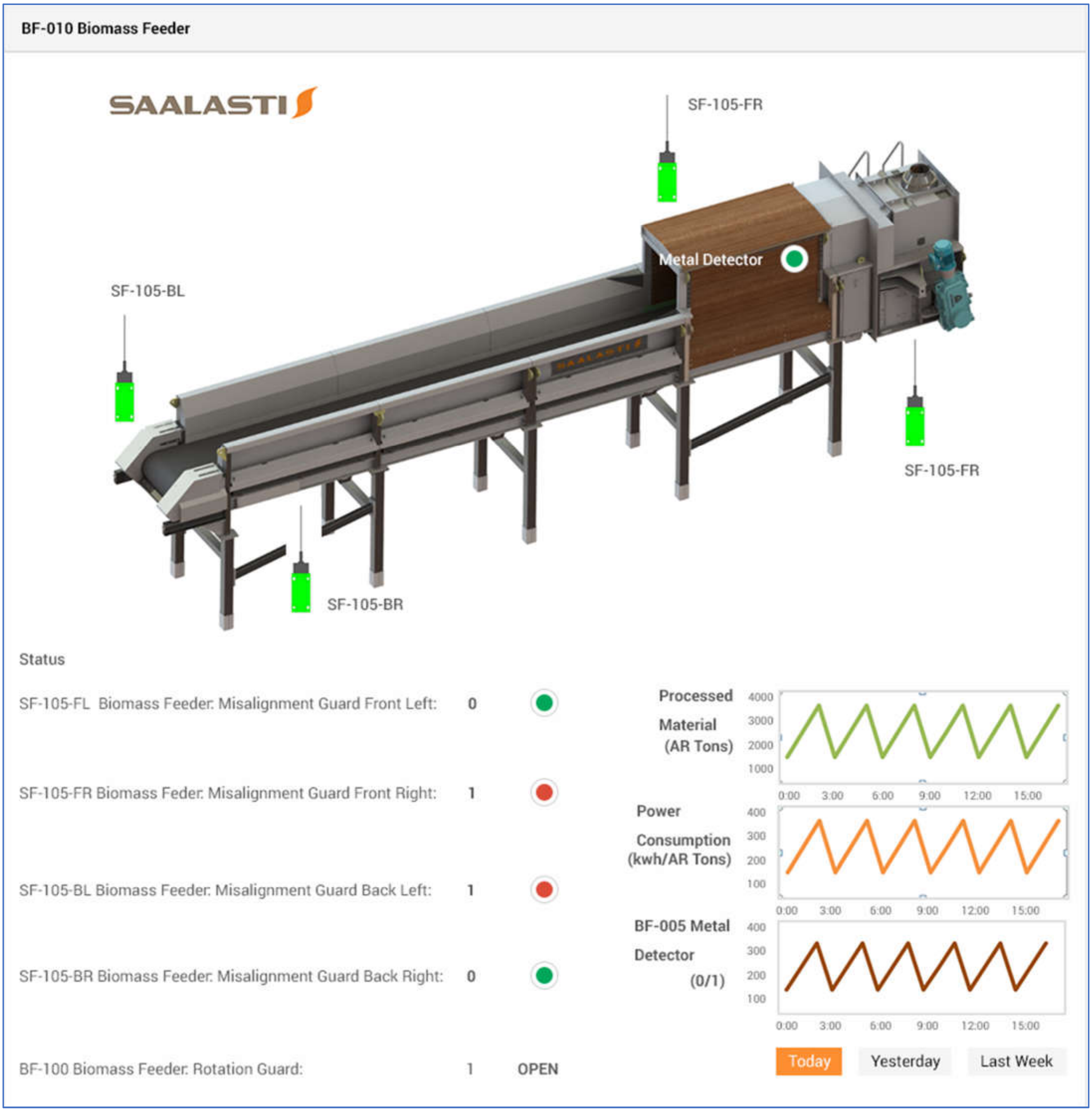

Figure 6 Equipment profile in the web-service includes a section named Components. Here chipper and associated components are displayed with their alarm states and trend charts.

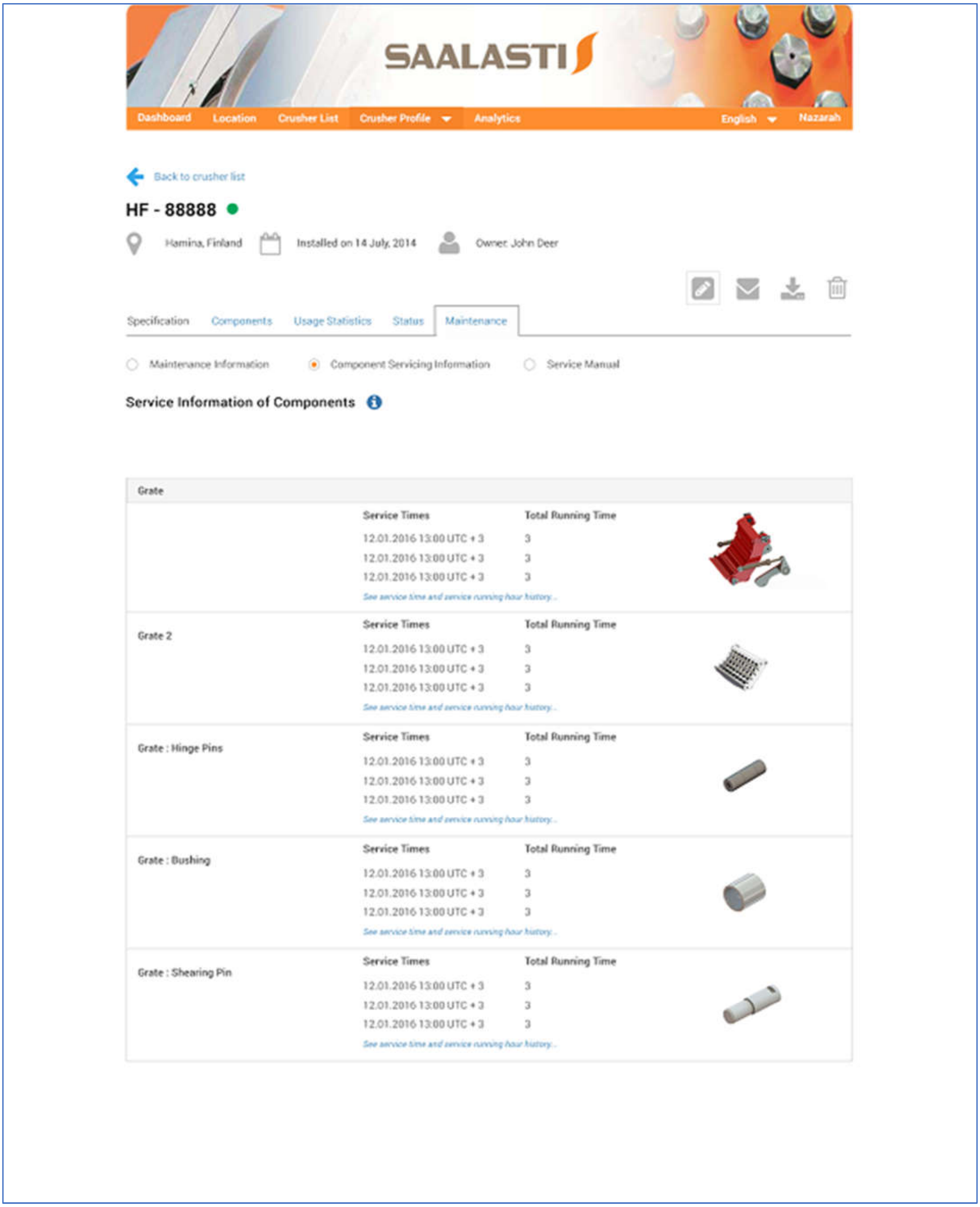

Figure 7 Equipment profile also contained service and maintenance information of biomass equipment and associated components. It was also possible to download individual manuals of the components.

Resulting Product

Saalasti’s web service for monitoring Biomass machinery contains a **Homepage** displaying their existing equipment units across the world along with their statuses. Depending on the access privilege, user can select from a **list of available chippers** to get more information. An equipment **profile** contains **General Information** (owner, location, installation period, service period, etc.), **Components** along with their status and attribute values, **Usage Trends** and information on **Service and Maintenance** works. There is refreshing option to fetch the most recent value of different attributes from the database. We used gif images to make the equipment unit look alive and often to indicate their actual states (online, in function, etc.). **Historical data** on different attribute value and machinery usage can be seen using line charts. If any equipment or component requires maintenance work, or has broken down, or some attribute has unusual values (i.e. value has crossed threshold amount) – this can be identified with appropriate **indicators**. User has the option to see live data feed or historical data filtered as **last 24 hours**, **last 7 days**, **last month** or **date selection**. Under an equipment profile, maintenance works for both scheduled ones and emergency repairs can be seen with **downloadable logs**.

The web service is for private use of Saalasti and its customers. Access to the service can be obtained provided that system admin has registered a user within it. System Admin has the privilege to access all equipment profiles, add new equipment and set initial threshold values for different attributes across all existing and to be added machinery. A maintenance engineer has access to profiles of only the biomass equipment s/he is in charge of. S/he can also access and edit general and maintenance related information of those profiles. An equipment owner can only access the profile of the biomass machinery s/he owns.

Updates on the Web Service

Right now, the web service is in maintenance mode under Sampo Software Oy. The development team has recently made the UI responsive for both smartphones and tablets. A heuristic evaluation was conducted on the newly improved UI and some issues related to the graphs and their interactions were identified. We are awaiting on Saalasti’s agreement on continuing further development. The next feature that is to be implemented is the notification system for alerting on emergency and service requirement situations.