**Project Plan**

***DAF Maintenance Prediction***

| **Date : 09 March 2022** |
| --- |
| **Version : v0.4** |
| **State : Draft** |
| **Author : Group 1 (Alex, Angel, Atanas, Stoyan, Tsanko)** |

**Version history**

| **Version** | **Date** | **Author(s)** | **Changes** | **State** |
| --- | --- | --- | --- | --- |
| v0.1 | 13.02 | Group 1 | Add initial context, project goal, strategy, activities and timeplan, test cases, potential testing strategies, configuration management, risks | Draft |
| v0.2 | 15.02 | Group 1 | Further explain context, strategy, timeplan, inside and outside scope | Draft |
| v0.3 | 16.02 | Group 1 | Elaborate on testing environment, format document | Draft |
| v0.4 | 09.03 | Group 1 | Elaboration on constraints, requirements,end product, and research questions | Draft |

**Distribution**

| **Version** | **Date** | **Receivers** |
| --- | --- | --- |
| v0.3 | 16.02 | [rick.adriaensen@daftrucks.com](mailto:rick.adriaensen@daftrucks.com)  [jeffrey.aben@daftrucks.com](mailto:jeffrey.aben@daftrucks.com)  [arie.naber@daftrucks.com](mailto:arie.naber@daftrucks.com) |

[**1. Project assignment**](#_yaos5oeoy6kr) **4**

[1.1 Context](#_r2bxkvhvqq0y) 4

[1.2 Goal of the project](#_astl7h4z1yu7) 5

[1.3 Scope and preconditions](#_8rpuiqnr3jk2) 5

[1.4 Strategy](#_fw8w39etp8oq) 6

[1.5 Research questions and methodology](#_o1q3aerct37t) 7

[1.6 End products](#_lrjd0lyaz3uc) 7

[**2. Project organization**](#_mylla6cuxc6e) **8**

[2.1 Stakeholders and team members](#_49vek4wzmou1) 8

[2.2 Communication](#_x40fsz7upzco) 11

[**3. Activities and time plan**](#_tss11996lp5w) **12**

[3.1 Phases of the project](#_4xmhxqqshr8u) 12

[3.2 Time plan and milestones](#_pnk0ems8m7nd) 12

[**4. Testing strategy and configuration management**](#_ygca4as6ztiv) **13**

[4.1 Testing strategy](#_a50xeqx48i53) 13

[4.2 Test environment and required resources](#_wx83ru5xt77c) 13

[4.3 Configuration management](#_738h9jdjgrkz) 14

[**5. Risk**](#_lcrl94ja2xlc) **15**

[5.1 Risk and mitigation](#_ny5oh4zcszfz) 15

# **1. Project assignment**

## 1.1 **Context**

In the Engine Factory, DAF is gathering information from multiple CNC machines using various sensors, including temperature and current sensors, at intervals of half or full seconds. The data is then compiled in a central database and accessed through Grafana, a data visualization tool used by employees.

The primary objective of this data acquisition process is to enable the identification of anomalies and to facilitate condition-based or preventative maintenance. This involves constant monitoring of the machines to detect any irregularities just before their components fail.

When data is collected from the CNC machines in the engine factory using sensors, it gets sent to a program called *Grafana*. *Grafana* has a special system that alerts the people who need to know when something is going wrong with the machine. This lets technicians know they need to stop the machine to fix it. Once the problem is fixed, the people in charge can use *Grafana* to look at what happened and figure out what to do next time.



*Figure: Flow of the data from the CNC machines all the way to the employees.*

## 

## 1.2 **Goal of the project**

The goal of the project is to create new intuitive insights in Grafana, in order to enable the identification of anomalies and to facilitate condition-based or preventative maintenance, as well as start data acquisition on other machines in the factory. This involves configuring the server to capture the right data streams, and subsequently translate those to visuals using a variant of MySQL.

A balance needs to be found between creating a general solution applicable to many machines, and allowing each machine to deviate from this general solution.

Goal is to optimize the data stream and data visualization of the current 6 machines and document how to handle and visualize data when adding new machines.

Moreover, any research covering the different aspects of the current solution that DAF has is beneficial to the end goal.

## 1.3 **Constraints**

| Sampling rate data input and output data of the machines | <= 1.4 Hz |
| --- | --- |
| Database | MySQL, InfluxDB, AWS,… |
| Gateway | Siemens Brownfield Connectivity Gateway |
| Visualization | Grafana |
| Security | Data has to be secured from access outside the company |

## 

## 1.4 Scope and preconditions

| **Inside scope:** | **Outside scope:** |
| --- | --- |
| 1 Database expansion | 1 Research into database change |
| 2 Research into another graph frontend | 2 Middleware development |
| 3 Expand GUI with more charts | 3 Backend |
| 4 Test report | 4 On-site testing |
| 5 User requirements specification |  |
| 6 Architecture design document |  |
| 7 Schemes/Diagrams |  |

## 1.5 Functional requirements

| **Data to be logged** | Machine input and output data |
| --- | --- |
| Multiple sensors per machine |
| Position sensors, Temperature sensors, Current sensors, Vibration sensors, etc. |
| **Triggers** | A trigger has to be sent when an anomaly is detected |
| The effort to set a trigger limit to detect anomalies should be minimized |
| The sampling rate for validating the trigger limits has to be set by a Maintenance Engineer with a maximum sampling rate of 0.033 Hz |
| **Visualization** | The machine data has to be visualized live with a maximum sampling rate of 0.2 Hz |
| Historical data has to be visualized for different time ranges (5 min, 1 hr, 12 hr, 1 day, 7 days, etc.) with a minimized load time |
| Long-history fast-refreshing-rate real time dashboards are not a use-case we’re considering – these two are mutually exclusive.  The same is true for frequently-checked triggers – The more often a trigger-query is issued, the smaller the timeframe of the considered data. |
| It has to be possible to visualize data of different machines and/or sensors in one graph (merged) with a minimized load time |
| It should be a reasonable task for a non-IT employee to generate simple graphs |
| **Scalability** | Capable to be scaled up to > 60 machines and <=400 sensors per machine Note that each BFC is limited to 60 machines, so the scaled solution utilizes multiple gateways |
| The effort to add a machine and/or sensor to the solution should be minimized |

## 

## 1.6 **Strategy**

The group will follow an agile way of development with 3-week long sprints.

Sprint retrospectives are conducted every Thursday in the early afternoon in the week before sprint delivery.

Sprint planning is conducted every Monday at the beginning of the sprint.

| **What?** | **When?** |
| --- | --- |
| Sprint 1 Planning | 06.02.2023 |
| Sprint 1 Retrospective | 05.03.2023 |
| Sprint 1 Planning | 06.03.2023 |
| Sprint 1 Retrospective | 26.03.2023 |
| Sprint 2 Planning | 27.03.2023 |
| Sprint 2 Retrospective | 16.04.2023 |
| Sprint 3 Planning | 17.04.2023 |
| Sprint 3 Retrospective | 14.05.2023 |
| Sprint 4 Planning | 15.05.2023 |
| Sprint 4 Retrospective | 04.06.2023 |
| Sprint 5 Planning | 05.06.2023 |
| Sprint 5 Retrospective | 18.06.2023 |

## 

## 1.7 **Research questions and methodology**

**Main research question**

How can we optimize the data visualization techniques currently used by DAF to detect potential component failures in the DAF engine plant?

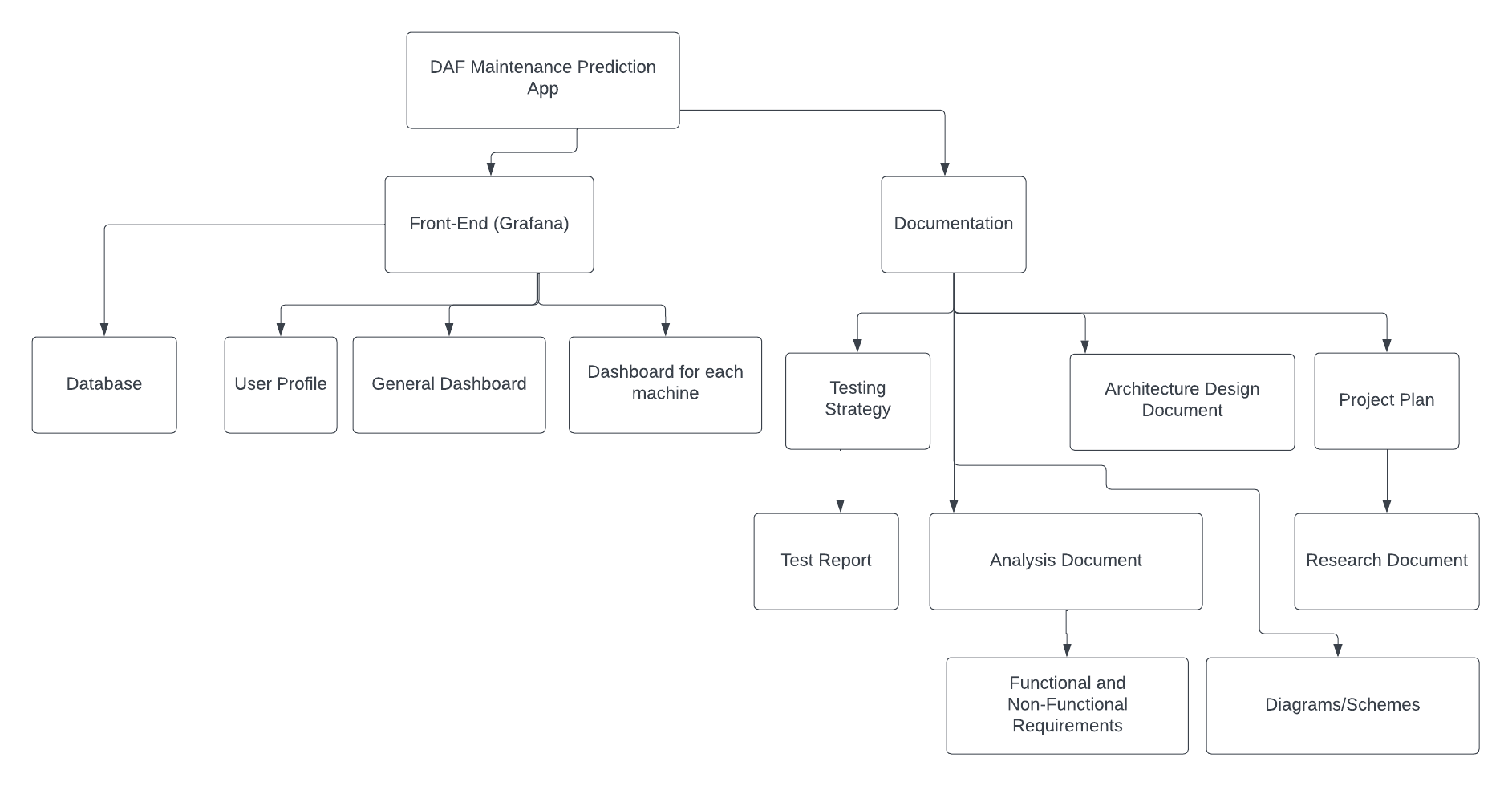
**Sub questions**

* What does the current database schema look like? What are its drawbacks?
* Is there a more optimal solution for data representation than Grafana?
* How can we expand the database to handle new sensor data?
* How can data be queried to ensure load times stay as low as possible?
* How can queries’ complexity be minimized?
* How can we ensure and optimize scalability and usability for additional machines?

## 1.8 **End products**

In order to provide a clear and comprehensive understanding of the final results that will be delivered, a product breakdown structure (PBS) has been developed and is presented below. This tool serves as a visual roadmap, starting with the main application at the highest level, and breaking down into the various sub-applications that are necessary for the successful completion of the project.

With the PBS, stakeholders can easily gain insight into the overall structure of the project, understand the dependencies and relationships between different components, and ensure that all necessary elements are accounted for. By organizing the project in this way, teams can more effectively plan and coordinate their efforts, reducing the risk of oversights or misunderstandings.



# 2. Project organization

## 2.1 **Stakeholders and team members**

| **Name** | **Abbreviation** | **Role and functions** | **Availability** |
| --- | --- | --- | --- |
| *Jeffrey Aben*  [*Jeffrey.Aben@DAFTRUCKS.com*](mailto:Jeffrey.Aben@DAFTRUCKS.com) |  | *DAF Representative* | *Mon-Fri*  *8:00-17:00* |
| *Arie Naber*  [*Arie.Naber@DAFTRUCKS.com*](mailto:Arie.Naber@DAFTRUCKS.com) |  | *DAF Representative* | *Mon-Fri*  *8:00-17:00* |
| *Rick Adriaensen*  [*Rick.Adriaensen@DAFTRUCKS.com*](mailto:Rick.Adriaensen@DAFTRUCKS.com) |  | *DAF Representative* | *Mon-Thu*  *8:00-17:00* |
| *Leon Schrijvers*  [*l.schrijvers@fontys.nl*](mailto:l.schrijvers@fontys.nl) | *Schrijvers, Leon L.* | *Technical teacher* | *Monday and Thursday*  *9:00-16:00* |
| *Bart Rabeling*  [*b.rabeling@fontys.nl*](mailto:b.rabeling@fontys.nl) | *Rabeling, Bart B.T.* | *Technical teacher* | *Monday and Thursday*  *9:00-16:00* |
| *Nicole Zuurbier-Munneke*  [*n.munneke@fontys.nl*](mailto:n.munneke@fontys.nl) | *Zuurbier-Munneke, Nicole N.* | *Semester coach* | *Monday and Thursday*  *9:00-16:00* |
| *Stoyan Kostadinov*  [*s.kostadinov@student.fontys.nl*](mailto:s.kostadinov@student.fontys.nl) | *Kostadinov, Stoyan S.L.* | *Team member* | *Every workday from 09:00-16:00. Occasionally available on weekends and vacations.* |
| *Aleksandar Svetoslavov*  [*a.svetoslavov@student.fontys.nl*](mailto:a.svetoslavov@student.fontys.nl) | *Svetoslavov, Aleksandar A.B* | *Team member* | *Every workday from 09:00-16:00. Occasionally available on weekends and vacations.* |
| *Tsanko Nedelchev*  [*t.nedelchev@student.fontys.nl*](mailto:t.nedelchev@student.fontys.nl) | *Nedelchev, Tsanko T.N* | *Team member* | *Every workday from 09:00-16:00. Occasionally available on weekends and vacations.* |
| *Angel Chernaev*  [*a.chernaev@student.fontys.nl*](mailto:a.chernaev@student.fontys.nl) | *Angel Chernaev A.C.* | *Team Member* | *Every workday.* |
| *Atanas Georgiev*  [*atanas.georgiev@student.fontys.nl*](mailto:atanas.georgiev@student.fontys.nl) | *Georgiev, Atanas A.A* | *Team Member* | *Every workday from 09:00-16:00. Occasionally available on weekends and vacations* |

## 2.2 **Communication**

To communicate effectively and collaborate within the team, as well as with stakeholders and teachers, we rely on various channels. MS Teams is the primary platform used for attending virtual meetings and discussions, while email is used for ongoing updates and other forms of communication. Additionally, in-person meetings are organized whenever needed. Moreover, a WhatsApp group is used in between the group members for faster means of communication. Its purpose is to obtain information quickly or announce that a member will be, for example, late or absent.

Specifically, we aim to meet in-person at Fontys (university building) at least two days per week, supplemented by one or two virtual meetings via MS Teams. At the beginning of each meeting, a general agenda is discussed to ensure all relevant topics are covered for the day. At the conclusion of each meeting, each team member records their actions in a daily journal to maintain a clear record of their own progress and the group fills in the group journal that describes the work that has been done and exactly when it has been done.

Furthermore, having clear documentation of actions and decisions taken ensures that team members are accountable to one another and can remain focused on delivering high-quality results. Overall, our approach to communication and collaboration is a critical element of our success as a team.

Another form of communication is peer review feedback between the group members. By the end of each sprint, every team member will write feedback for each member about their progress during the sprint, including what is done well and what needs to be improved.This way, team members can see the track on which they are with the opportunity to improve in future.

# **3. Activities and time plan**

## 3.1 **Phases of the project**

The beginning of the project during the first 1-2 sprints, the focus will be towards research and prototyping, the next 2 sprints (3 and 4) will be focused more on development and less on research and last sprint (5) will be focused on final touches, finalizing development, handover documentation, (peer) evaluation and group reflection.

## 3.2 **Time plan and milestones**

In the table below, the phases and milestones for the project are displayed. It shows every sprint planned for the project implementation with its start and finish date. In addition, the effort for each sprint is determined with a number from 1 to 10, where 1 is the least effort put and 10 is many nights without proper sleep xd.

| **Phasing** | **Effort**  **1-10** | **Start date** | **Finish date** |
| --- | --- | --- | --- |
| 1 Sprint 0 | 4 | 06.02.2023 | 05.03.2023 |
| 2 Sprint 1 |  | 06.03.2023 | 26.03.2023 |
| 3 Sprint 2 |  | 27.03.2023 | 16.04.2023 |
| 4 Sprint 3 |  | 17.04.2023 | 14.05.2023 |
| 5 Sprint 4 |  | 15.05.2023 | 04.06.2023 |
| 6 Sprint 5 |  | 05.06.2023 | 18.06.2023 |

# **4.** **Testing strategy and configuration management**

## 4.1 **Testing strategy**

The tests that will be performed will be for the sql queries and the alarm functionality. We will use tools to automate the tests for each deployment, so that the project version on production is always stable. We will automate both the sql query validation and the alarm testing (email notification to the client). In addition, we will manually test the features implemented by us before delivering the product.

The Grafana GUI will be tested using Selenium IDE, so that we make sure the data coming from the backend is properly sent and placed on the frontend.

The current strategy of testing the alarm function will be:

* Generating random data. At some point, a value of a sensor will be changed to be above the alarm threshold for a short period of time
* Sending an email from the GUI to the client’s email
* Check if the data sent in the email is correct for the machine
* Return the results

## 4.2 **Test environment and required resources**

As part of our testing process, we will include test reports that document each test case in the test environment. Each test case will be thoroughly documented, including step-by-step instructions for reproducing the test and the expected answer. As outlined in [section 4.1](https://www.ema.europa.eu/en/documents/presentation/presentation-section-41-therapeutic-indications_en.pdf), we will utilize automated testing with Selenium to ensure that the main functionalities of the project are properly tested. We will use continuous integration and continuous deployment (CI/CD) to ensure that every commit to the main branch is stable and tested. By implementing these robust testing procedures, we can minimize errors and ensure the quality of the final product.

## 

## 4.3 **Configuration management**

The project is located in GitLab at: <https://git.fhict.nl/I450851/daf-maintenance-prediction>.

The branching strategy used will be some form of Gitflow. Every team member has their own branch that they work on and open a merge request (MR) towards the developbranch when their work is ready. The members have the option to ask for review by others if they have any uncertainties or straight up merge into *develop*. However, in the latter, the team member is obliged to notify other team members via a message through WhatsApp or MSTeams.

To ensure effective version control and collaboration, we have adopted a Gitflow-based branching strategy. Under this approach, every team member has their own dedicated branch to work on. When a team member completes their assigned work, they open a merge request (MR) to the develop branch. The MR will be reviewed by at least one other team member to ensure quality and consistency before it is merged.

If a team member has any uncertainties about their work, they can ask for additional review and feedback before opening the MR. Alternatively, they can merge their branch directly into the develop branch. In the latter case, however, the team member must notify the rest of the team via a message on either WhatsApp or MS Teams. This ensures that everyone is kept up to date on the state of the project and can adjust their work accordingly.

By adopting this branching strategy, we are able to work efficiently and effectively as a team. Each member has their own space to work on their assigned tasks, while the MR process ensures that changes are carefully reviewed and integrated into the develop branch in a controlled and organized manner. Finally, our use of messaging apps for notifications ensures that everyone is kept up to date with the latest developments, reducing the risk of misunderstandings or conflicts.

# 

# **5.** **Risk**

## 5.1 **Risk and mitigation**

There are several risks that may endanger the project development process which are written on the table below. In order to reduce the risk of failure as much as possible, we wrote prevention and mitigation activities for each risk.

| **Risk** | **Prevention activities** | **Mitigation activities** |
| --- | --- | --- |
| 1 Stakeholders lack communication | Maximize points of contact. We currently have 3 of them. | Contact teachers as soon as possible. |
| 2 Our work may not work on the stakeholder’s environment | Establish agreement with stakeholders on testing on their environment. | Docker. Confirm the version of the different software and types of connections.. |
| 3 Database corruption/Database problem | Every person has a copy of the database/sample data. | Another team member shares the data via WeTransfer/Flash drive/etc.. |
| 4 Team member gets sick | Dress warm. Stay fit. | Communicate with the team immediately and plan others to take over one’s work. |
| 5 A laptop/computer break | Handle laptops with respect. | Acquire a laptop from the ISSD desk or buy another one as soon as possible. |