Decibel Threshold Event Displayer

Scrum & Project Management Report

vonal3, gernd1, deged2

January 1, 2025

Contents

List of Tables

List of Figures

1 Introduction

1.1 Initial Situation

According to the Federal Office for the Environment (DE: BAFU) one in seven people in Switzerland is affected by noise pollution [**?**]. The pollution comes primarily from road traffic, followed by railways and then air traffic. In addition to these noise sources, construction sites, nightclubs and public facilities also produce noise. Because of this, Switzerland has set up upper noise limits that must be respected. However, it is of course the case that these limits are not always adhered to. Affected people must then either accept this or take action against it. For the latter, they must gather evidence to prove their noise disturbance to the police and the courts. This evidence then comes from an audio recording which the affected person made them self.

1.2 Project Goal

To help the people affected by noise pollution, we want to create an application which processes a given sound file (.wav) and analyzes it. It should detect when a specified threshold has been exceeded and then summarizes the result in a PDF document. The document should then contain all necessary information for filing a complaint. As our application will have a wide range of end users, two of our main design goals are to make it as user-friendly as possible and to make it platform independent, so it can be used with any PC operating system and ideally mobile device.

1.3 Problems with Audio Files

A wave file (.wav) contains samples of the recorded audio, where each sample represents the amplitude at a given moment. Those amplitude values are relative to each other and not absolute. It is therefore impossible to determine the actual dB(A) (loudness relative to the human ear) someone would perceive without any

further information [?, ?, ?]. Because of this, we require the user to also give information about the minimal and maximal dB(A) measured in the given audio file. To do this, we recommend using a smartphone app like DecibelX for IOS, which allows the user to record the audio and also conveniently inspect the minimal and maximal dB measured in that recording. With those two values, we can then map the relative values from the wave file to its db(A) values.

2 Scrum Roles

The project team has, in coordination with the tutor, determined that the association Lärmliga (`https://laermliga.ch/`) will not be directly involved in the project. Instead, the tutor Dr. Simon Kramer, will take on the role of the stakeholder and render the decisions to be made by the customer.
The other scrum roles were determined internally by the project team. At first, the decision was made for Dr. Simon Kramer to also be the Product Owner. However, the project team realized it would be impractical to include Dr. Simon Kramer in every product decision and especially in the prioritization of the backlog. The project team felt that, due to their relative inexperience, it would be best to have the Product Owner in the team, so decisions could be made quickly and validated with the stakeholders afterward, if necessary.

Name	**Role(s)**
Dr. Simon Kramer	Stakeholder, Tutor
Dominic Gernert	Product Owner, Developer
Lukas von Allmen	Scrum Master, Developer
Darius Degel	Developer

Table 1: Scrum Roles

3 Sprint Goals

The sprint goals are defined by the project team at the Sprint Planning right after the conclusion of the previous sprint. They are formulated to be compliant with the S.M.A.R.T principles. The project team decides the sprint goals based on the backlog prioritization and the issues assigned to the next sprint. The sprint goals are defined and reviewed in a Markdown file in the repository (Gitlab).

3.1 Sprint 1

For the first sprint, the project team decided to focus on research and prototyping. This can be understood as a feasibility study. Due to the team's relative inexperience with LaTeX and due to the project scope, they felt that having a

working prototype had to be made before a design decision could realistically be made.

Goal	Reached
Prototypes with two different technologies are implemented and their pros and cons are evaluated	Yes
Tech stack for the project has been chosen	Yes
Licence is defined	No
Git repository and documentation skeleton are created	No

Table 2: Sprint goals of sprint 1

In sprint 1, the project team managed to complete the first two goals (prototyping and tech stack evaluation). However, they were unable to evaluate the licensing because the decision about which tech stack should be used was made only at the end of the sprint. The project team underestimated the issue weights, leading to goals that were not reached.

3.2 Sprint 2

Sprint 2 has not ended, as of the completion date of this document. Nonetheless, four of the seven defined sprint goals have been reached at this point. Although the project team defined more goals for the second sprint than the first, it is important to note that the total weight of tasks assigned to this sprint is equal to the first sprint and estimates are more conservative.

Goal	Reached
Intermediate presentation is prepared and presented	n.a.
Requirements are specified	n.a.
UX-Prototype is defined	Yes
System delimitation is specified	n.a.
Decibel values can be calculated	Yes
Licence is defined	Yes
Git repository and documentation skeleton where created	Yes

Table 3: Sprint Goals of Sprint 2

4 Requirements

4.1 Epics

Epics were defined at the very beginning of the project and are largely product focussed. Epics are defined in the Epics section of the Planning tab in Gitlab (Gitlab). Epics are subject to change but for the sake of completeness, they are included in this document anyway (see Figure 1).

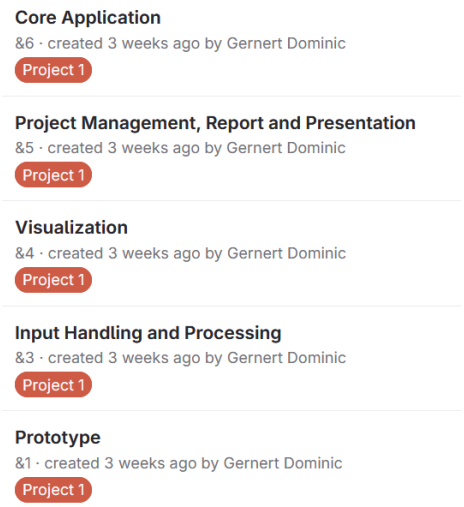

Figure 1: Epics

4.2 User Stories

User Stories are created as Issues in Gitlab. They are associated with an Epic and are prioritised. Issues contain a *Definition of Ready* (4.2.1), Acceptance Criteria, as well as a *Definition of Done* (4.2.2). When an issue is selected for a sprint, sub-tasks are created, estimated and assigned. The issue itself is assigned to the person with the most assigned sub-tasks. The full list of issues is available in Gitlab's issue list or the Development Board.

4.2.1 Definition of Ready

The *Definition of Ready* is defined as a checklist on every User Story. In every Sprint Planning the issues selected for the next Sprint are validated by crossing off the checklist. All necessary adjustments are made together by the project team before the issue is estimated and planned. The *Definition of Ready* is part of the Issue Template in the repository (see Gitlab).

Definition of Ready:

- Requirements and Acceptance Criteria Defined
- Acceptance Criteria must be testable
- Understood by the Team
- Sized and estimated
- Prioritised in the Backlog
- No major Impediments

4.2.2 Definition of Done

Similarly to the *Definition of Ready* (4.2.1), the *Definition of Done* is a static checklist that is part of the issue template. The tasks are checked off by the reviewer, which is the Product Owner unless specified otherwise.

Definition of Done:

- Acceptance Criteria met
- Tested and no critical bugs
- Documentation updated
- Reviewed and approved

4.3 Product Backlog

Issues are prioritised first and foremost by the Product Owner. This prioritisation is implemented as tags in Gitlab and is can be viewed on the Development Board (see Figure 2)

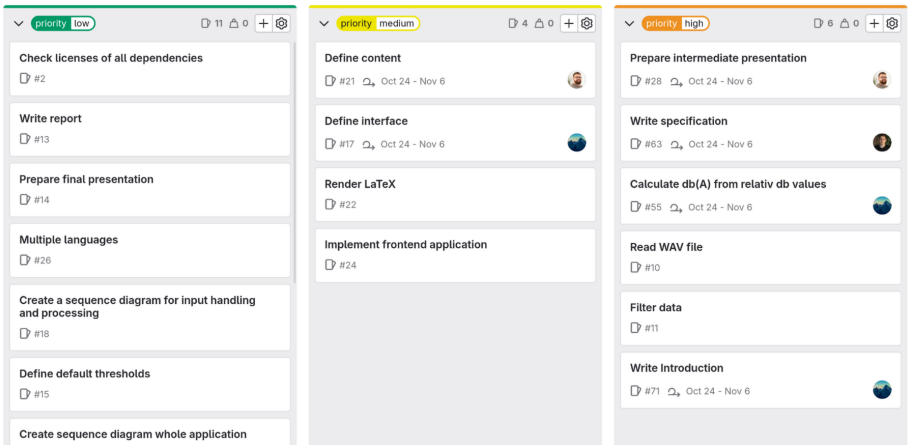

Figure 2: Product Backlog

4.4 Sprint Backlog

The Sprint Backlog of the currently running sprint is displayed as a column on the Development Board. In combination with the prioritisation (see 4.3), this makes for a convenient way for the project team to select issues for the next sprint based on priority (see Figure 3).

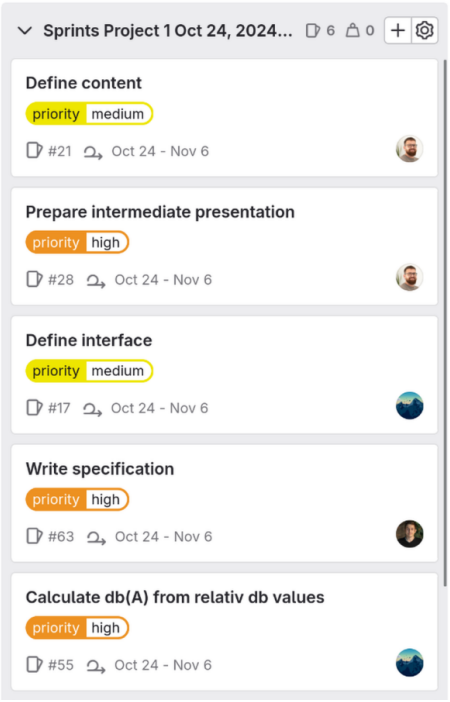

Figure 3: Sprint Backlog

4.5 Impediment Backlog

Impediments are also created as Issues in Gitlab. They are however, assigned the label *Impediment* and are displayed on a separate Impediment Board (see Figure 4 and Gitlab). Impediments are created either ad hoc or during the Daily Standup meeting. There is a template for Impediments analogous to the Issue Template (see Gitlab).

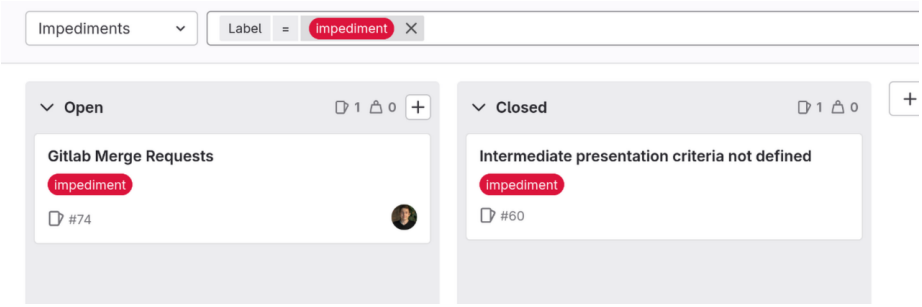

Figure 4: Impediment Board

5 Scrum Adaptations

5.1 Gitlab

Mr. Frank Helbling was added to the gitlab "Decibel Threshold Event Displayer" group of our project as a reporter (Gitlab Group). He has access to inspect Epics on group level. Issues and Issue Boards can be accessed on project level. The Daily Journals and Sprint Notes can be accessed in the doc/scrum folder (Scrum Notes).

5.2 Product Owner

As described in section 2, the Product Owner role has gone from our tutor, Dr. Simon Kramer, to Dominic Gernert. Although not specifically mentioned in the Scrum Guide [**?**], in the experience of the project team and an article on applied frameworks [**?**], the Product Owner is usually someone with a business background and not a developer. This makes sense, because they need to be able to make decisions about the product that only a business expert could make. Since this is not the real world but a school project, the project team had to reduce the coordination effort. Thus, the team decided to make Dominic Gernert the Product Owner. Since he is not a business expert for our project, decisions are still made as a collective. If however, there is a disagreement on a decision, the Product Owner's suggestion is followed. For bigger decisions,the Stakeholder Dr. Simon Kramer is involved.

5.3 Daily Scrum

As suggested by Mr. Frank Helbling in the Scrum presentation [**?**], the project team has decided to confine their Daily Scrum to 15 minutes. They are scheduled weekly on Wednesday at 18:15. The Scrum meetings that took place until the completion date of this document took about 30 minutes, however. The

likely reason for this is that the project team meets only weekly, but each person actually works on the project at multiple days of the week. This leads to more information, questions, and impediments being generated and having to be discussed during the Daily Scrum. The project team is still trying to keep their Daily Scrum short (less than 20min.) in the future.

5.4 Release Plan

The project team has decided against implementing a Release Plan. Mainly because it was never explicitly demanded by the tutor, Dr. Simon Kramer. The other reason for forgoing the Release Plan is that there is no real customer awaiting product updates and Dr. Kramer is capable of running the application himself should he want to do so. Product demonstrations by the team are not on a fixed scheduled and are done if necessary or requested by the stakeholder.

5.5 Retro

Instead of following the Keep-Try-Drop model for the Sprint Retrospective, the project team decided to focus on successes, problems, and improvements. A reason for that is the focus on the individual by the team due to the low number of members. This focus on successes and problems provides a way for each team member to explain their frustrations and discuss possible improvements. This adaptation is comparatively small and mostly one of wording. The project team feared however, that using the Keep-Try-Drop model, they would have to justify using certain tools when running into minor problems. Instead, the team hopes to focus more on the product than the tool chains.

6 Project Setup Review

6.1 Identifying the initial situation

For identifying the initial situation we analyzed 11.1 Project Description and used multiple online resources such as the websites of Lärmliga Schweiz and Federal Office for the Environment FOEN. Based on those findings we designed our application, which now provides a convenient way for affected people to document noise pollution.

6.2 Topic analysis

The issue at hand needs some in depth knowledge about how audio recoding and measurement works. Looking back, we should have invested time earlier to research and learn the basics of audio recoding and measurements.

6.3 Stakeholder / Stakeholder Management

Initially we identified a list of possible stakeholders:

- Tutor: Dr. Simon Kramer
- People affected by noise pollution
- Noise producers (Construction Sites, Night Clubs, Highways)
- Lärmliga Schweiz
- Federal Office for the Environment FOEN

For the context of this module we decided to only consider our tutor as the sole stakeholder, because otherwise the stakeholder management would cost much more time without much benefit. This way the stakeholder management turned out to be straight forward, as Dr. Kramer provided his expectations and the scope of the project beforehand, only impediments and key decisions had to be discussed together.

6.4 Organisation

TODO: Ask what is wanted here

6.5 Installations

TODO: Ask what is wanted here

7 Review

7.1 Product goals

7.2 Sprint goals

7.3 Product delimitation

7.4 Deliverables

7.5 Product backlog / sprint backlogs

8 Retrospective

8.1 Retrospective I: Scrum roles and stakeholders

8.2 Retrospective II: Scrum Events and Artifacts

8.3 Retrospective III: Tools/Instruments

9 Lessons learned

9.1 Insights into framework conditions

9.2 Cooperation

10 Bibliography

11 Appendix

11.1 Project description

The **goal (what)** of this project is to deliver a FLOSS-licensed, platform-independent piece of software (computer program), called the *Decibel Threshold Event Displayer*, that

1. takes as inputs a WAV-file and a list of sound level thresholds in decibels (e.g., legal day and nighttime noise maxima above which your health deteriorates);

2. filters out all data points in the file that correspond to sound events below the lowest of the above thresholds; and

3. displays the remaining data points (as a blue vertical comb plot) on a horizontal time axis (with the dates and times corresponding to the data points) as well as the thresholds (as horizontal red lines) in decibel, and statistically summarises the data set with the help of the LaTeX-package pgfplots.

The **purpose (why)** of this project is to empower poor folks who suffer from insomnia due to ambient noise (`https://laermliga.ch/`) by arming them with the (peaceful) means of proving their noise hell (a smart phone app such as `https://apps.apple.com/ch/app/dezibel-x-pro-l%C3%A4rm-messger%C3%A4t/id1257651611` together with your software) to the police and the courts of law.

The code should be minimal, modular, and self-explaining.

The project report should be concise (maximally informative, minimally long). It must contain this project description as a quotation.

11.2 Declaration of Authorship

The project team, namely Dominic Gernert, Lukas von Allmen, and Darius Degel, hereby declare that the report submitted is our own unaided work. All direct or indirect sources used are acknowledged as references.