Decibel Threshold Event Displayer

BTI3031 Project 1 | Final Presentation

January 8, 2025

Dominic Gernert, Lukas von Allmen, Darius Degel

Table of Contents

- ▶ **Problem Description**
- ▶ **Implementation**
- ▶ **Scrum**
- ▶ **Demo**
- ▶ **Conclusion & Future Work**

Initial Situation

Project Goals

- Analyze Audio File

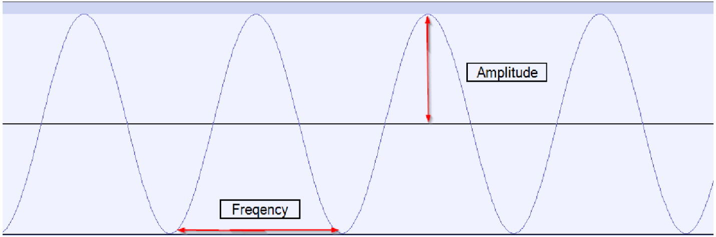
Project Goals

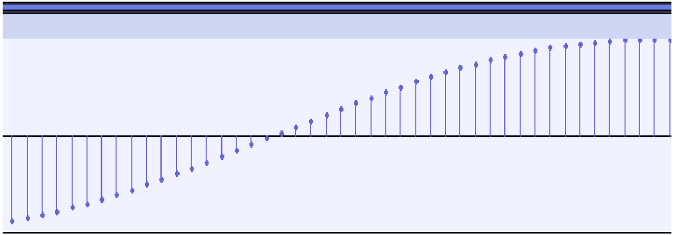
- Analyze Audio File
- Summarize findings in a PDF

Project Goals

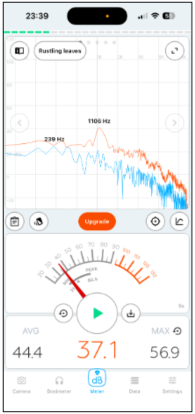
- Analyze Audio File
- Summarize findings in a PDF
- Easy to use

Audio Files

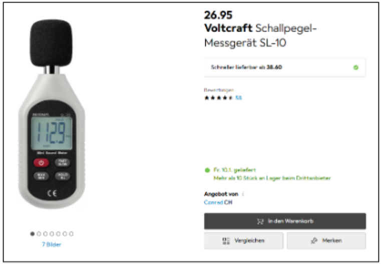




Measuring the Sound Level



DecibelX for IOS



26.95

Voltcraft Schallpegel-Messgerät SL-10

Schneller lieferbar ab 38.60

Bewertungen

★★★★☆ 58

Fr, 10.1. geliefert

Mehr als 10 Stück an Lager beim Drittanbieter

Angebot von

Conrad CH

Sound level measuring device from Galaxus

Requirements

- Take .wav file, threshold and additional reference values as input

Requirements

- Take .wav file, threshold and additional reference values as input
- Analyze and Summarize

Requirements

- Take .wav file, threshold and additional reference values as input
- Analyze and Summarize
    - Metadata

Requirements

- Take .wav file, threshold and additional reference values as input
- Analyze and Summarize
  - Metadata
  - Plot

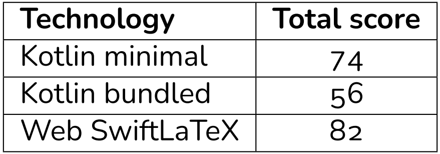
Requirements

- Take .wav file, threshold and additional reference values as input
- Analyze and Summarize
  - Metadata
  - Plot
- User should not need any Technical know-How

Requirements

- Take .wav file, threshold and additional reference values as input
- Analyze and Summarize
    - Metadata
    - Plot
- User should not need any Technical know-How
- Platform independent

Technology evaluation



Technology	Total score
Kotlin minimal	74
Kotlin bundled	56
Web SwiftLaTeX	82

Table of Contents

▶ Problem Description

▶ **Implementation**

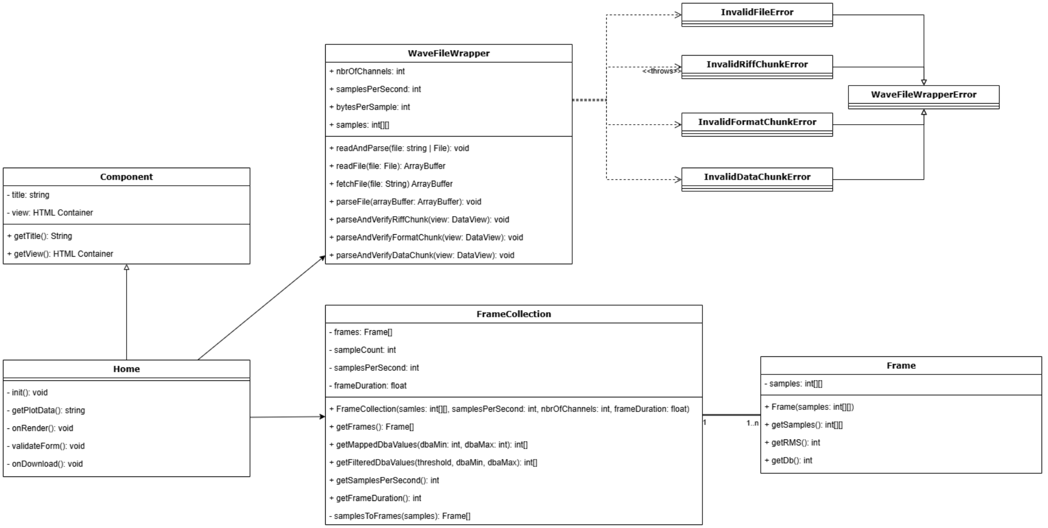
- Architecture and Processes
- Testing
- License and Privacy
- Deployment / Distribution

▶ Scrum

▶ Demo

▶ Conclusion & Future Work

Architecture - Class Diagram



InvalidFileError

InvalidRiffChunkError

InvalidFormatChunkError

InvalidDataChunkError

WaveFileWrapperError

<<throws>>

WaveFileWrapper

+ nbrOfChannels: int
+ samplesPerSecond: int
+ bytesPerSample: int
+ samples: int[][]

+ readAndParse(file: string | File): void
+ readFile(file: File): ArrayBuffer
+ fetchFile(file: String) ArrayBuffer
+ parseFile(arrayBuffer: ArrayBuffer): void
+ parseAndVerifyRiffChunk(view: DataView): void
+ parseAndVerifyFormatChunk(view: DataView): void
+ parseAndVerifyDataChunk(view: DataView): void

Component

- title: string
- view: HTML Container

+ getTitle(): String
+ getView(): HTML Container

Home

- init(): void
- getPlotData(): string
- onRender(): void
- validateForm(): void
- onDownload(): void

FrameCollection

- frames: Frame[]
- sampleCount: int
- samplesPerSecond: int
- frameDuration: float

+ FrameCollection(samles: int[][], samplesPerSecond: int, nbrOfChannels: int, frameDuration: float)
+ getFrames(): Frame[]
+ getMappedDbaValues(dbaMin: int, dbaMax: int): int[]
+ getFilteredDbaValues(threshold, dbaMin, dbaMax): int[]
+ getSamplesPerSecond(): int
+ getFrameDuration(): int
- samplesToFrames(samples): Frame[]

1

1..n

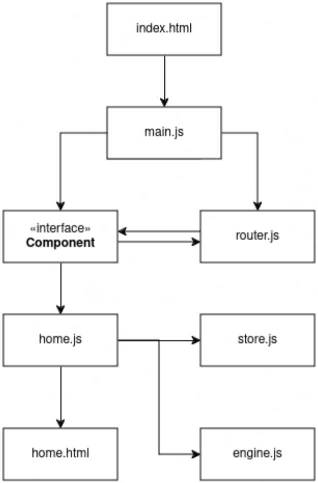
Frame

- samples: int[][]

+ Frame(samples: int[][])
+ getSamples(): int[][]
+ getRMS(): int
+ getDb(): int

Architecture - SPA Techstack

- Vanilla JS SPA Framework (Web Programming Module)
- Bootstrap CSS Framework
- SwiftLaTeX in Browser WASM LaTeX rendering Library

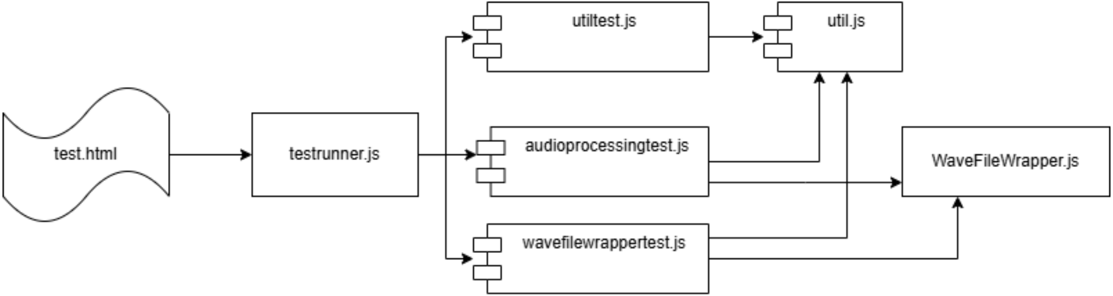


Processes

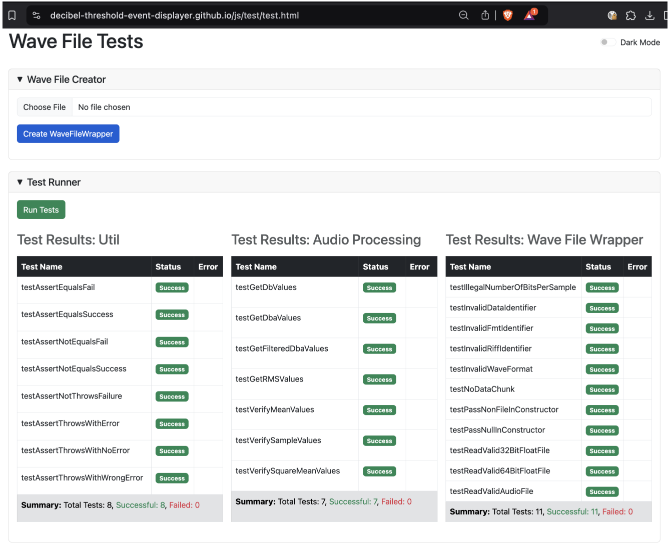
1. Read *.wav File
2. Group samples into frames (duration of 300ms)
3. Calculate root-mean-square (RMS) per frame
4. Convert RMS dB values per frame
5. Map the relative dB to absolute dB(A)
6. Filter the resulting list of dB(A)
7. Render PDF with dB(A) and user data

```javascript
function rms(values){
  const squared = values.map(
      sample => Math.pow(sample, 2)
  );
  const sum = squared.reduce((a, b) => a + b);
  const mean = sum / values.length;
  return Math.sqrt(mean);
}

function rmsToDb(rms) {
  return 20 * Math.log10(rms);
}

function dbToDba(db, dbMin, dbMax, dbaMin, dbaMax) {
  return (db - dbMin) * (dbaMax - dbaMin) /
         (dbMax - dbMin) + dbaMin;
}
```

Testing - Overview



Testing - In action



decibel-threshold-event-displayer.github.io/js/test/test.html

Wave File Tests

▼ Wave File Creator

▼ Test Runner

Test Results: Util

Test Name	Status	Error
testAssertEqualsFail	Success	
testAssertEqualsSuccess	Success	
testAssertNotEqualsFail	Success	
testAssertNotEqualsSuccess	Success	
testAssertNotThrowsFailure	Success	
testAssertThrowsWithError	Success	
testAssertThrowsWithNoError	Success	
testAssertThrowsWithWrongError	Success	

Summary: Total Tests: 8, Successful: 8, Failed: 0

Test Results: Audio Processing

Test Name	Status	Error
testGetDbValues	Success	
testGetDbaValues	Success	
testGetFilteredDbaValues	Success	
testGetRMSValues	Success	
testVerifyMeanValues	Success	
testVerifySampleValues	Success	
testVerifySquareMeanValues	Success	

Summary: Total Tests: 7, Successful: 7, Failed: 0

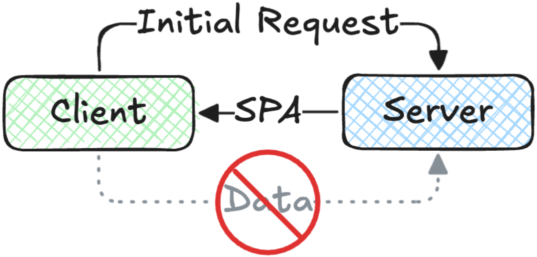
Test Results: Wave File Wrapper

Test Name	Status	Error
testIllegalNumberOfBitsPerSample	Success	
testInvalidDataIdentifier	Success	
testInvalidFmtIdentifier	Success	
testInvalidRiffIdentifier	Success	
testInvalidWaveFormat	Success	
testNoDataChunk	Success	
testPassNonFileInConstructor	Success	
testPassNullInConstructor	Success	
testReadValid32BitFloatFile	Success	
testReadValid64BitFloatFile	Success	
testReadValidAudioFile	Success	

Summary: Total Tests: 11, Successful: 11, Failed: 0

Privacy concerns

- No data is sent to the server, after the initial request
- From the Plot on the PDF the original Audio File cannot be recreated



The user does not get into legal trouble, using the application or the resulting PDF!

License

Dependency Licenses:

- SwiftLaTeX: AGPL-3.0
- pgfplots: GPL-3.0

Resulting License:

- **GPL-3.0 licence (FLOSS)**

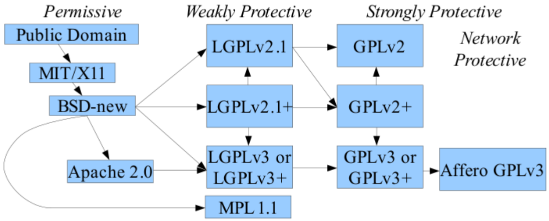


Image Source: https://dwheeler.com/essays/floss-license-slide.html

Deployment / Distribution

1. A dev pushes or merges code to the main branch
2. GitLab automatically mirrors the repository to GitHub
3. GitHub deploys automatically to GitHub Pages
4. The Application is available under: https://decibel-threshold-event-displayer.github.io/

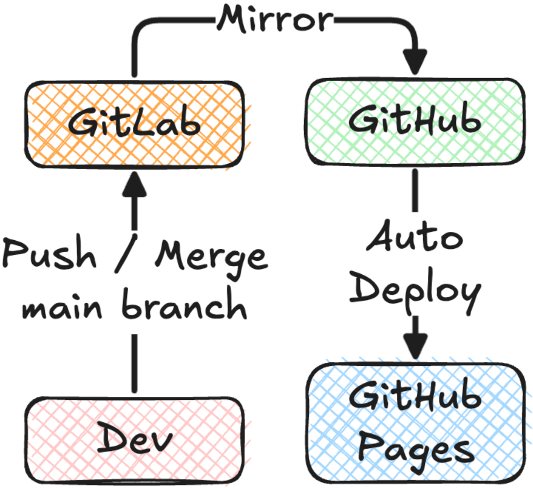


Table of Contents

- Problem Description
- Implementation
- **Scrum**
- Demo
- Conclusion & Future Work

Scrum

- 2 week iterations
- Daily every week
- Review / planning every other week
- Product goals / sprint goals
- GitLab, MS Teams, LaTeX, excalidraw, draw.io

Table of Contents

- Problem Description
- Implementation
- Scrum
- **Demo**
- Conclusion & Future Work

Demo

- Watch the demo on YouTube
- Or better yet: Try it yourself on Github Pages!

Table of Contents

- Problem Description
- Implementation
- Scrum
- Demo
- **Conclusion & Future Work**
  - Conclusion
  - Future Work

Conclusion

- Product goals achieved
- Scrum generally applied
- Minor issues with GitLab
- Great teamwork and team chemistry
- Interesting and well-defined project
- Lots of insights about audio, WAV files, and WebAssembly

Future Work

- Localization (DE, FR, IT)
- Custom thresholds
- Custom form fields
- Support more audio formats
- Dark mode / visual improvements

Questions / Discussion