

TASK - 2A

package com.sismics.music.core.event.async;

- **Duplicate code smell** - DirectoryCreatedAsyncEvent and DirectoryDeletedAsyncEvent were two classes that contained the same code. Just the name of the class was different.
- **Duplicate code smell** - LastFmUpdateLovedTrackAsyncEvent.java and LastFmUpdateTrackPlayCountAsyncEvent.java are very similar classes.
- **Duplicate code smell** - PlayCompletedEvent and PlayStartedEvent are very similar classes.
- **Duplicate code smell** - TrackLikedAsyncEvent and TrackUnlikedAsyncEvent are very similar classes.

```
public class DirectoryDeletedAsyncEvent {
    private Directory directory;

    public Directory getDirectory() {
        return directory;
    }

    public void setDirectory(Directory directory) {
        this.directory = directory;
    }

    @Override
    public String toString() {
        return Objects.toStringHelper(this)
            .add("directory", directory)
            .toString();
    }
}
```

```
public class DirectoryCreatedAsyncEvent {
    private Directory directory;

    public Directory getDirectory() {
        return directory;
    }

    public void setDirectory(Directory directory) {
        this.directory = directory;
    }

    @Override
    public String toString() {
        return Objects.toStringHelper(this)
            .add("directory", directory)
            .toString();
    }
}
```

package com.sismics.music.core.listener.async;

- **Missing Abstraction -**
 - CollectionReindexAsyncListener, DirectoryCreatedAsyncListener, DirectoryDeletedAsyncListener, LastFmUpdateLovedTrackAsyncListener, LastFmUpdateTrackPlayCountAsyncListener, TrackLikedAsyncListener, TrackUnlikedAsyncListener.
 - The classes above implement the similar kind of methods. Therefore a class called AbstractAsyncListener should be created and the above classes should extend this class.
 - Explanation for this is provided in Task 3A.

package com.sismics.music.core.service.collection;

CollectionService

- **Long method smell** in addDirectoryToIndex().
- **Long method smell** in readTrackMetaData(). Then method is around 84 lines long.

CollectionWatchService

- Modularize startup()
- Long method smell in run()

package com.sismics.music.core.service.importaudio;

ImportAudioService

- **Long method smell** in run()
- **Long method smell** in importFile().

package com.sismics.music.core.model.dbi;

Track

- **Long parameter** list in constructor ↓
public Track(String id, String albumId, String artistId, String fileName, String title, String titleCorrected, Integer year, String genre, Integer length, Integer bitrate, Integer order, boolean vbr, String format, Date createDate, Date deleteDate)

How the code smell analysis software helped?

- The software used for code smell and code metrics was Jarchitect.
- It pointed some smells like long methods, too many parameters etc.
- This helped us in identifying such classes like Track.java etc.

Task 2B

Logical lines of Code
5 699
0 (NotMyCode)
Estimated Dev Effort 142d

Types
275
3 Projects
60 Packages
1 365 Methods
479 Fields
200 Source Files
1 727 Third-Party Elements

Comment
15.81%
1 070 Lines of Comment

Debt
9.63%
Rating **B** 6d 4h effort to reach **A**
Debt 13d 5h
The technical-debt is incomplete because no coverage data specified.
Explore Debt

Coverage
N/A because no coverage data specified
[Import Code Coverage Data](#)

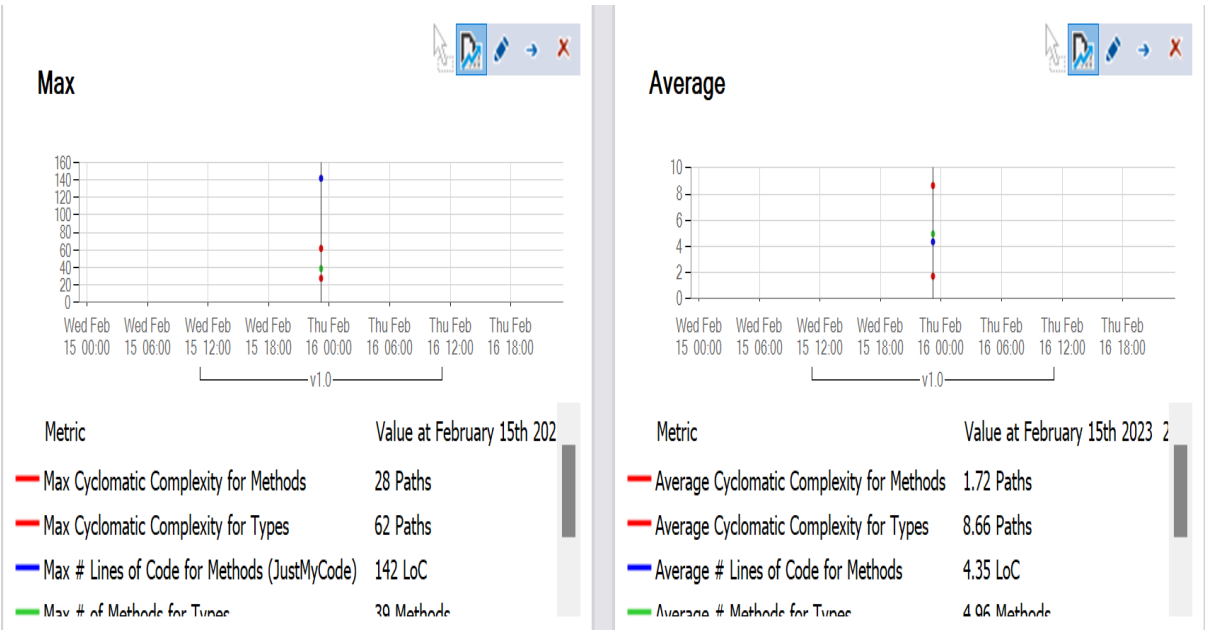
Method Complexity
28 Max
1.69 Average

Quality Gates
Fail 2
Warn 0
Pass 3

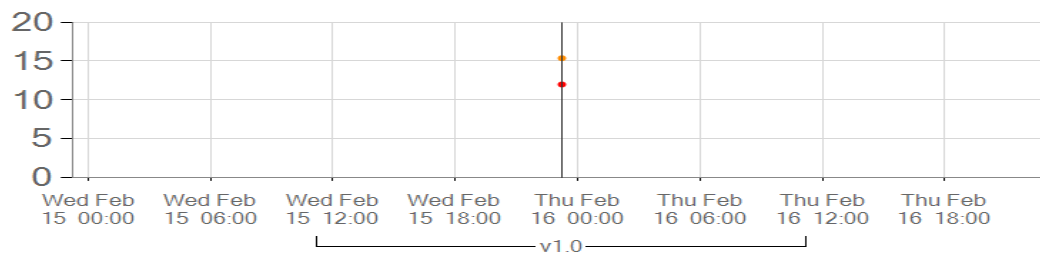
Rules
Critical 2
Violated 61
Ok 145

Issues
All 1 209
Blocker 0
Critical 0
High 325
Medium 130
Low 754

Group issues by elements

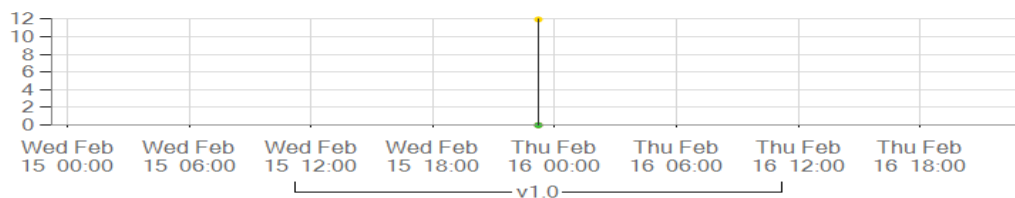


Debt



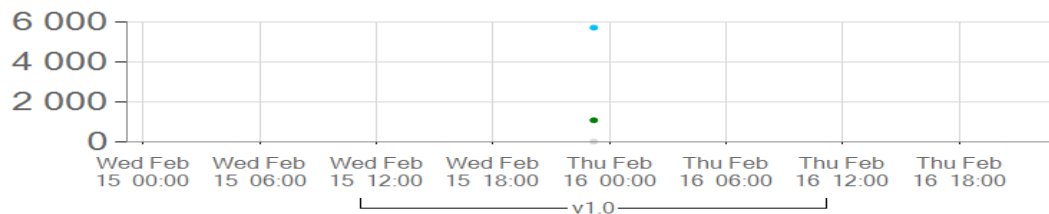
Metric	Value at February 15th 2023 23:11 (v1.0)
Debt (Metric)	11.99 man-days
Annual Interest (Metric)	15.37 man-days

Issues



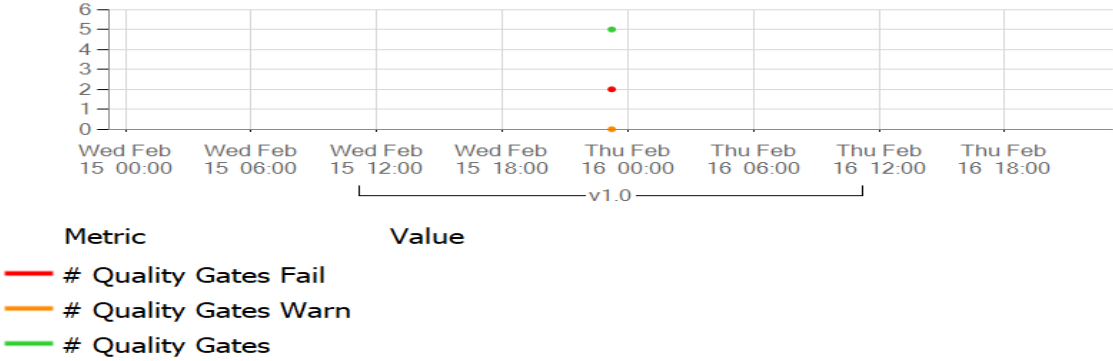
Metric	Value at February 15th 2023 23:11 (v1.0)
# Issues Fixed since Baseline	0 issues
# Blocker/Critical/High Issues	N/A
# Issues	1 198 issues x 0.01

Lines of Code

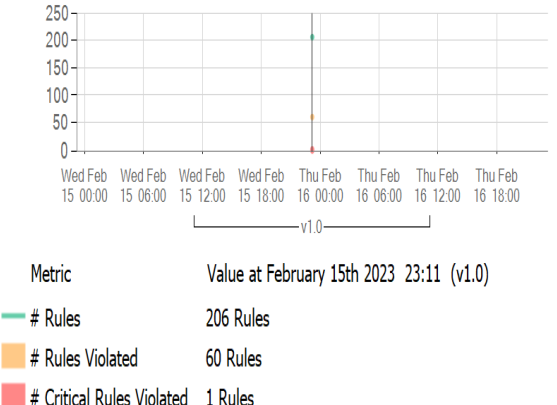


Metric	Value
# Lines of Code	5 000
# Lines of Code (NotMyCode)	1 000
# Lines of Comments	100

Quality Gates



Rules Violated



Rules Violations

