

# Software Evolution and Maintenance

## FCAI-CU-SWE 2022

You are given a Java project University Timetabling (<https://sourceforge.net/projects/unitime/>) as a base version, you are the maintenance team that needs to implement changes to the existing software. We will only focus on the Java backend.

The change request to the project can either be a bug fix, change in an existing feature or addition of a new requirement. Motivate the need for the change that you have proposed.

### Project Implementation

- You should work in teams of no more than 4 and no less than 3 From the same lab.
- **The project is organized into two phases**

Phase 1:

1. Create a GitHub repository and all team members should be contributors. contributors (Note: All of your work should be pushed to GitHub as pull requests and reviewed by another team member.)
2. Provide a description of what the project does.
3. Run static code analysis tool and generate an analysis report (You may use SonarQube as we did in the lab)
4. Provide a description for the feature/bug fix that you will start working on it.
5. Register the change request you defined in step 4 to a free ticketing system or software such as (Bugzilla, Jira, Agiloft, ...),
6. You are requested to update it through the lifetime of the change request to indicate its status, and any changes that are related.

- **Deliverables**

- 1- A link to your Github repo
- 2- A link to your ticketing system
- 3- Static Analysis Report For java backend code
- 4- text file (word or pdf) containing the description of what the project does. Mention what technique(s) you used to obtain this description
- 5- A description for the feature/bug fix that you will start working on it.

- **Notes:**

- All of the required deliverables will be pushed to your Github repository, all team members must use Github to push their tasks.
- You should submit a text or pdf file with your names and github repo (don't push anything after deadline),
- the file should have the following format file name:  
Group\_1stStudentId\_2ndStudentId\_3rdStudentId\_4thStudentId

---

### Team Member:

ID	Name	GitHub Username
20196037	Amr Halaby	AmrHalaby
20196026	Sohaila Gamal	Sohailla
20206023	Dalia Gamal	DaliaGamal11
20196003	Ahmed Gamal	AhmedGamal444

- **GitHub Repository:** <https://github.com/Sohailla/uni-time>
- **Ticketing System [Trello]:** <https://trello.com/b/nGqGdbd8/uni-time>

## • Static Analysis Report For java backend code

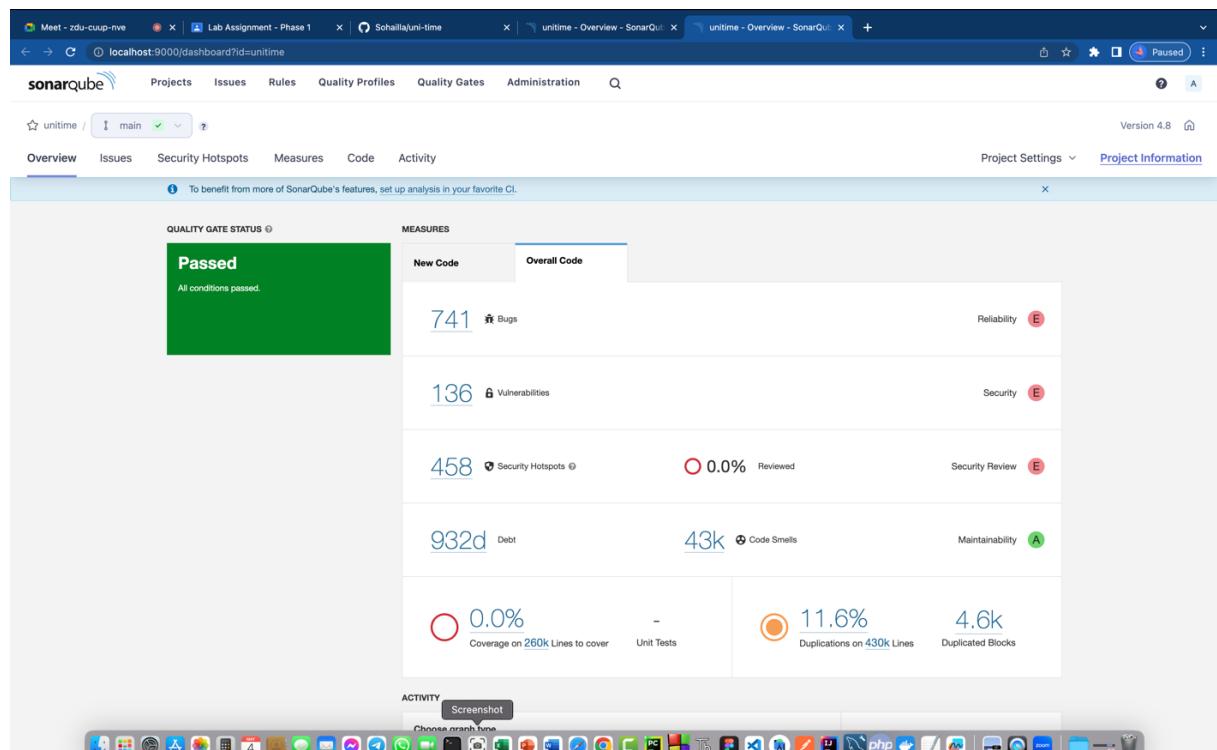
The screenshot shows the JPA Buddy IDE interface. The left sidebar displays the project structure for 'unitime' with files like README.md, build.properties, and JavaSource. The main pane shows the terminal output of the Apero static analysis tool. The output includes various logs such as sensor execution, CPD block indexing, SCM publisher tasks, and finally a 'BUILD SUCCESS' message. The bottom status bar indicates the persistence unit initialized and the current master branch.

```

unitime - build.properties
unitime ~/Desktop/unitime
# The Apero Foundation licenses this file to you under the Apache License,
# Version 2.0 (the "License"); you may not use this file except in
# compliance with the License. You may obtain a copy of the License at:
#
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

[INFO] Sensor Zero Coverage Sensor
[INFO] Sensor Zero Coverage Sensor (done) | time=318ms
[INFO] Sensor Java CPD Block Indexer
[INFO] Sensor Java CPD Block Indexer (done) | time=2008ms
[INFO] SCM Publisher SCM provider for this project is: git
[INFO] SCM Publisher 2381 source files to be analyzed
[INFO]
[INFO] SCM Publisher 2381/2381 source files have been analyzed (done) | time=13602ms
[INFO] CPD Executor 504 files had no CPD blocks
[INFO] CPD Executor Calculating CPD for 1741 files
[WARNING] Too many duplication groups on file JavaSource/org/unitime/timetable/onlinesectioning/custom/purdue/PurdueCourseRequestsValidatorProvider.java. Keep only the first 100 groups.
[INFO] CPD Executor CPD calculation finished (done) | time=444ms
[INFO] Analysis report generated in 593ms, dir size=54.9 MB
[INFO] Analysis report compressed in 6738ms, zip size=19.3 MB
[INFO] Analysis report uploaded in 338ms
[INFO] ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=unitime
[INFO] Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
[INFO] More about the report processing at http://localhost:9000/api/ce/task?id=AyfoP8-j2wEpB7Dcw44q
[INFO] Analysis total time: 5:32.902 s
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 11:05 min
[INFO] Finished at: 2023-05-04T21:31:50+02:00
[INFO] -----
+ unitime git:(master) x

```



SonarQube Issues - Lab Assignment - Phase 1

localhost:9000/project/issues?id=uniTime&resolved=false

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

My Issues All

Filters

Issues in new code

Type: Bug (741), Vulnerability (136), Code Smell (43k)

Severity: Blocker (199), Critical (12k), Major (14k), Minor (17k), Info (96)

Scope, Resolution, Status, Security Category, Creation Date, Language, Rule, Tag, Directory, File, Assignee, Author

JavaSource/CourseCreditUnitConfig.hbm.xml

Remove this commented out code.

Code Smell, Major, Open, Not assigned, 5min effort, Comment

9 years ago L88 % T unused

JavaSource/CourseOffering.hbm.xml

Remove this commented out code.

Code Smell, Major, Open, Not assigned, 5min effort, Comment

11 years ago L75 % T unused

JavaSource/InstructionalOffering.hbm.xml

Remove this commented out code.

Code Smell, Major, Open, Not assigned, 5min effort, Comment

5 years ago L60 % T unused

Remove this commented out code.

Code Smell, Major, Open, Not assigned, 5min effort, Comment

9 years ago L128 % T unused

JavaSource/PitCourseOffering.hbm.xml

Complete the task associated to this "TODO" comment.

Code Smell, Info, Open, Not assigned, 0min effort, Comment

6 years ago L176 % T cwe

JavaSource/PitSchedulingSubpart.hbm.xml

Complete the task associated to this "TODO" comment.

Code Smell, Info, Open, Not assigned, 0min effort, Comment

6 years ago L157 % T cwe

JavaSource/PreferenceGroup.hbm.xml

Remove this commented out code.

Code Smell, Major, Open, Not assigned, 5min effort, Comment

3 years ago L107 % T unused

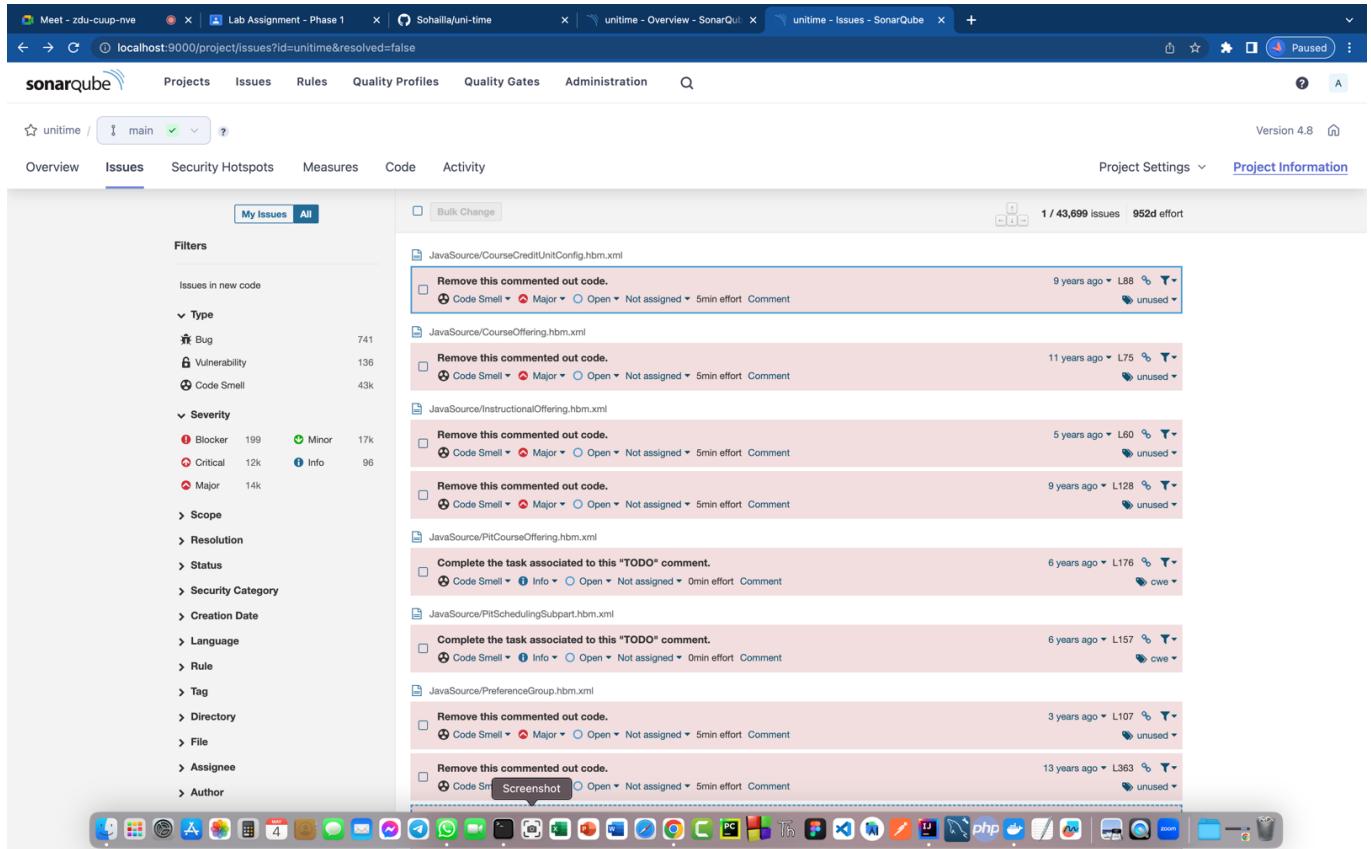
Remove this commented out code.

Code Smell, Major, Open, Not assigned, 5min effort, Comment

13 years ago L363 % T unused

Screenshot

1 / 43,699 issues 952d effort



SonarQube Issues - Lab Assignment - Phase 1

localhost:9000/project/issues?id=uniTime&resolved=false

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

My Issues All

Filters

Issues in new code

Type: Bug (741), Vulnerability (136), Code Smell (43k)

Severity: Blocker (199), Critical (12k), Major (14k), Minor (17k), Info (96)

Scope, Resolution, Status, Security Category, Creation Date, Language, Rule, Tag, Directory, File, Assignee, Author

JavaSource/SolverParameterGroup.hbm.xml

Remove this commented out code.

Code Smell, Major, Open, Not assigned, 5min effort, Comment

9 years ago L826 % T unused

JavaSource/dbupdate.xml

Remove this commented out code.

Code Smell, Major, Open, Not assigned, 5min effort, Comment

15 years ago L57 % T unused

Remove this commented out code.

Code Smell, Major, Open, Not assigned, 5min effort, Comment

11 years ago L3662 % T unused

JavaSource/hibernate-groups-tcp.xml

Remove this commented out code.

Code Smell, Major, Open, Not assigned, 5min effort, Comment

9 years ago L67 % T unused

Remove this commented out code.

Code Smell, Major, Open, Not assigned, 5min effort, Comment

6 years ago L71 % T unused

JavaSource/menu-custom.xml

Remove this commented out code.

Code Smell, Major, Open, Not assigned, 5min effort, Comment

11 years ago L74 % T unused

JavaSource/menu.xml

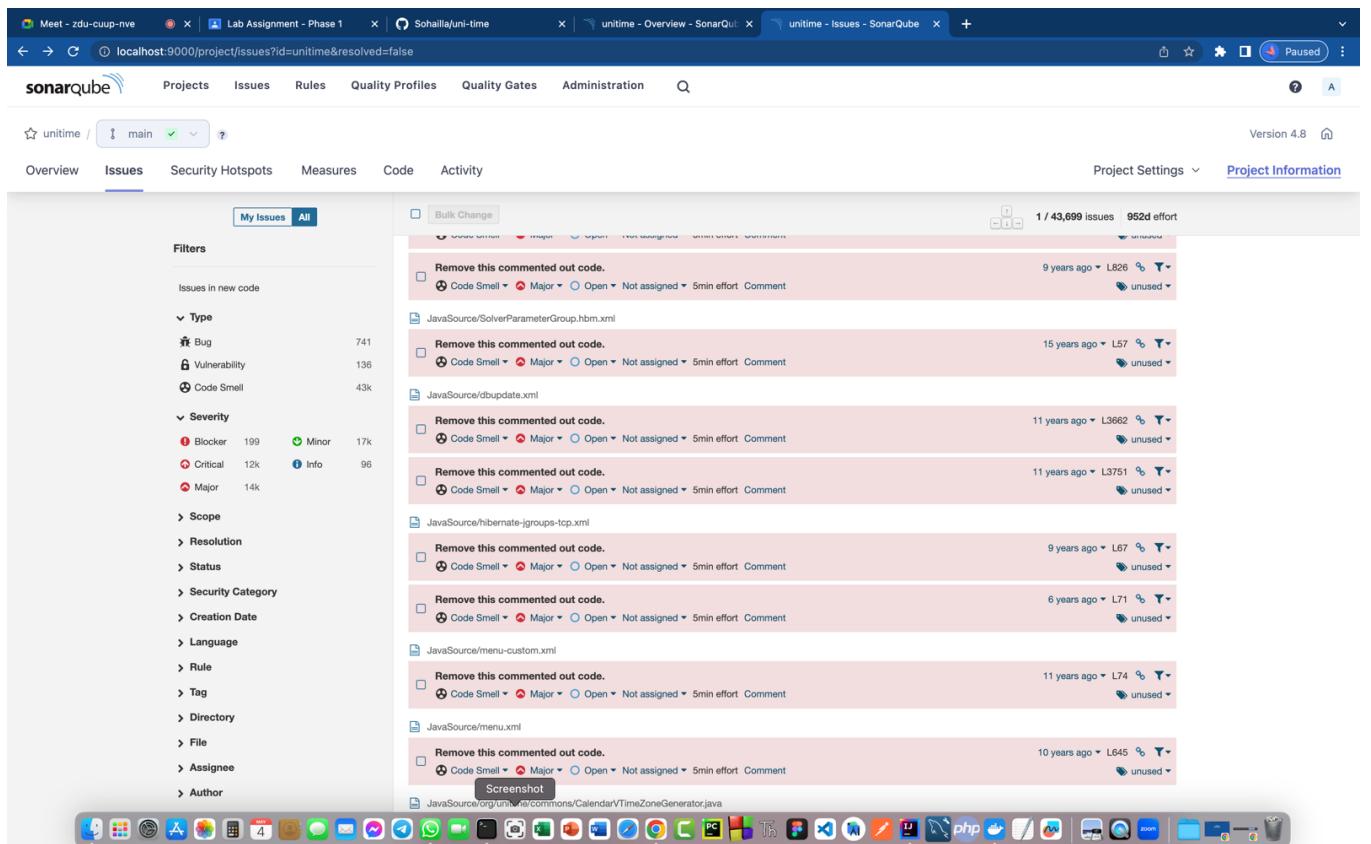
Remove this commented out code.

Code Smell, Major, Open, Not assigned, 5min effort, Comment

10 years ago L645 % T unused

Screenshot

1 / 43,699 issues 952d effort



Screenshot of SonarQube Issues page for the 'unitime' project.

The page shows 1 / 43,699 issues with 952d effort. The main content area displays a list of issues, with the first few entries being:

- JavaSource/org/unitime/commons/Calendar/TimeZoneGenerator.java: Change this "try" to a try-with-resources. (Code Smell, Critical, Open, Not assigned, 15min effort, Comment)
- JavaSource/org/unitime/commons/Debug.java: Define and throw a dedicated exception instead of using a generic one. (Code Smell, Major, Open, Not assigned, 20min effort, Comment)
- JavaSource/org/unitime/commons/Email.java: Add a private constructor to hide the implicit public one. (Code Smell, Major, Open, Not assigned, 5min effort, Comment)

The left sidebar contains a 'Filters' section with various dropdown menus for filtering issues by type, severity, scope, resolution, status, security category, creation date, language, rule, tag, directory, file, assignee, and author. The 'Issues' tab is selected.

At the bottom of the page, there is a toolbar with icons for different applications like Finder, Mail, Safari, etc.

Screenshot of SonarQube Issues page for the 'unitime' project.

The page shows 1 / 43,699 issues with 952d effort. The main content area displays a list of issues, with the first few entries being:

- JavaSource/org/unitime/commons/Email.java: Convert the abstract class "Email" into an interface. (Code Smell, Minor, Open, Not assigned, 10min effort, Comment)
- JavaSource/org/unitime/commons/Email.java: Define and throw a dedicated exception instead of using a generic one. (Code Smell, Major, Open, Not assigned, 20min effort, Comment)
- JavaSource/org/unitime/commons/Email.java: Define and throw a dedicated exception instead of using a generic one. (Code Smell, Major, Open, Not assigned, 20min effort, Comment)

The left sidebar contains a 'Filters' section with various dropdown menus for filtering issues by type, severity, scope, resolution, status, security category, creation date, language, rule, tag, directory, file, assignee, and author. The 'Issues' tab is selected.

At the bottom of the page, there is a toolbar with icons for different applications like Finder, Mail, Safari, etc.

- **The description of what the project does. Mention what technique(s) you used to obtain this description**

## **1. What the project does:**

UniTime is a comprehensive educational scheduling system that supports developing course and exam timetables, managing changes to these timetables, sharing rooms with other events, and scheduling students to individual classes. It is a distributed system that allows multiple university and departmental schedule managers to coordinate efforts to build and modify a schedule that meets their diverse organizational needs while allowing for minimization of student course conflicts. It can be used alone to create and maintain a school's schedule of classes and/or exams, or interfaced with an existing student information system.

### **Features**

- course timetabling
- examination timetabling
- event management
- student scheduling

## **2. Maintenance Technique: "Opportunistic Approach"**

Opportunistic Approach is a hybrid of the two

1. Begin with top-down, gain an overview of the functions of the program
2. Then selectively apply bottom-up strategies when nearing "code level"
3. Presence of beacons can indicate opportunity for change of strategy

- **A description for the feature/bug fix that you will start working on it.**

We will fix the Email.java and ToolBox.java classes.

In Email.java:

We will modify the code into two classes to support solid principles

In ToolBox.java:

Add new features to class to support availability