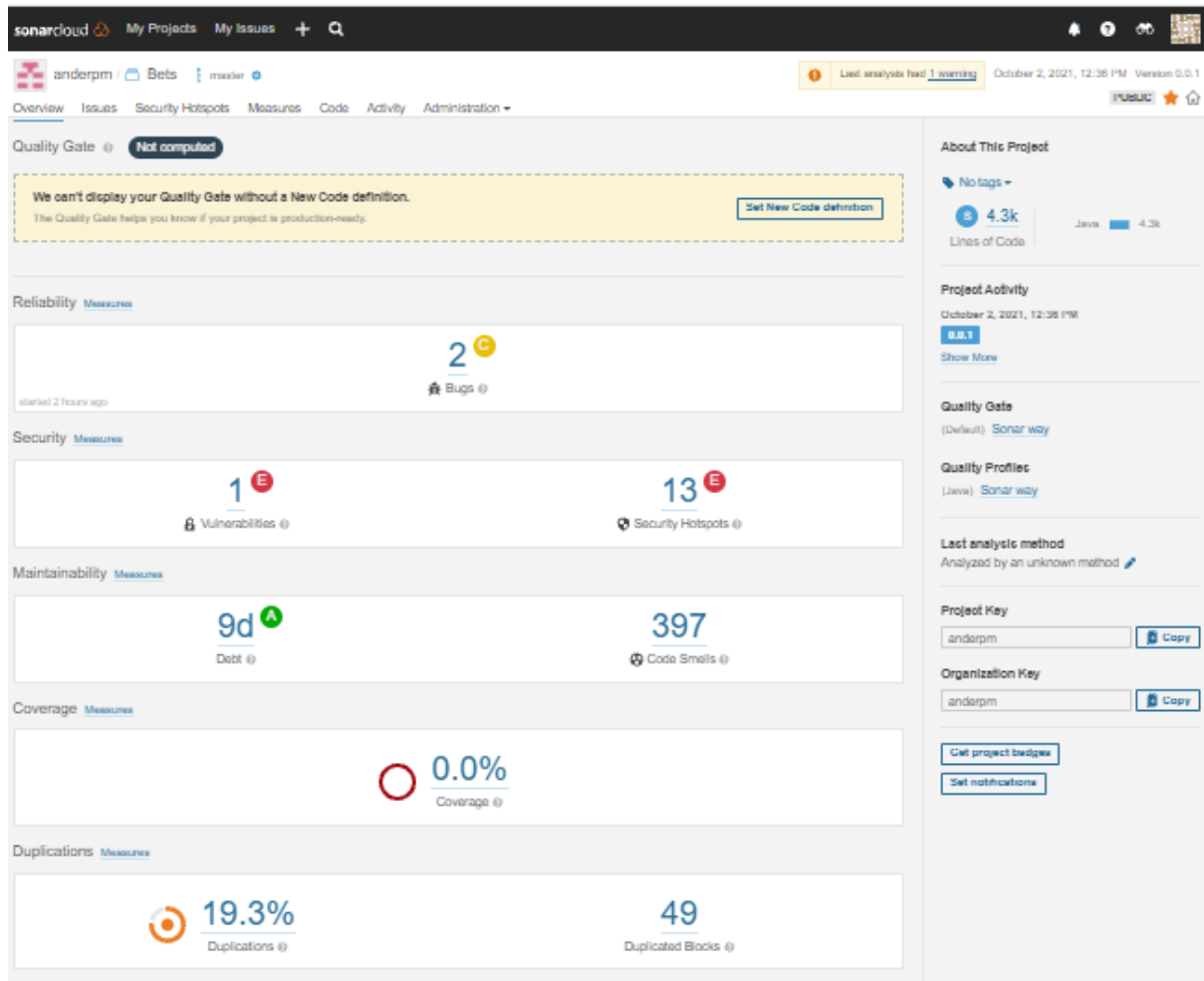


# SONAR CLOUD

Egilea: Ander Pollacino Merino

Lehenik eta behin aurreko urteko Bets aplikazioari sonarLint aplikatu diogu. Sonarcloud-era igotzean hau da sortutako proiektuaren informazioa:



Maintainability atalean ikusten den bezala, 9 egun behar ditugu proiektu guztiaren ohar guztiak egokitzeko.

2 bug ditugu, hau da, kode akatsak, eta berehala konpondu beharko ditugu:

1. Either re-interrupt this method or rethrow the "InterruptedException" that can be caught here.
2. Remove this conditional structure or edit its code blocks so that they're not all the same.

Security ataleko, Vulnerabilities 1 eta 13 Security Hotspots ditugu, hauek segurtasun arazoak dira eta kanpo erasoak saihesteko zuzendu behar ditugu.

Vulnerabilities:

1. Disable access to external entities in XML parsing.

Azkenik 397 Code Smells ditugu, hauek ez dira akatsak, baina zuzendu beharko ditugu etorkizunean bug-ak saihesteko.

Zuzenketak:

Bug:

1. Lehenengo bug-ean, `Thread.sleep(1000)` sortzen duen salbuespena ez zen tratatzen, eta hori konpontzeko catch berria sortu dugu *InterruptedException* salbuespenarekin):

```
        while(System.in.available()==0){
            Thread.sleep(1000);
        }

    } catch (InterruptedException ie) {
        Thread.currentThread().interrupt();

    } catch (Exception e) {
        System.out.println("Something has happened in ObjectdbManagerServer: "+e.toString());
    }
}
```

2. Bigarren bug-eko hasierako egoera:

```
try {
    BLFacade facade = MainGUI.getBusinessLogic();
    Vector<domain.Event> events = facade.getEvents(firstDay);

    if (events.isEmpty())
        jLabelListOfEvents.setText(ResourceBundle.getBundle(language).getString("NoEvents")
            + ": " + dateFormat1.format(calendarAct.getTime()));
    else
        jLabelListOfEvents.setText(ResourceBundle.getBundle(language).getString("Events") + ": "
            + dateFormat1.format(calendarAct.getTime()));
    jComboBoxEvents.removeAllItems();
    System.out.println("Events " + events);

    for (domain.Event ev : events)
        modelEvents.addElement(ev);
    jComboBoxEvents.repaint();

    if (events.size() == 0)
        jButtonCreate.setEnabled(true);
    else
        jButtonCreate.setEnabled(true);
} catch (Exception e1) {
    jLabelError.setText(e1.getMessage());
}
}
```

azkeneko `if(events.size()==0)` ez da beharrezkoa, if-aren barruko kodea aurreko if-ean sartu daitekeelako, beraz hori aldatu dugu eta hau izan emaitza:

```

try {
    BLFacade facade = MainGUI.getBusinessLogic();
    Vector<domain.Event> events = facade.getEvents(firstDay);

    if (events.isEmpty()) {
        jLabelListOfEvents.setText(ResourceBundle.getBundle(language).getString("NoEvents")
            + ": " + dateFormat1.format(calendarAct.getTime()));
        jButtonCreate.setEnabled(true);
    } else {
        jLabelListOfEvents.setText(ResourceBundle.getBundle(language).getString("Events") + ": "
            + dateFormat1.format(calendarAct.getTime()));
        jButtonCreate.setEnabled(true);
    }

    jComboBoxEvents.removeAllItems();
    System.out.println("Events " + events);

    for (domain.Event ev : events)
        modelEvents.addElement(ev);
    jComboBoxEvents.repaint();

} catch (Exception e1) {
    jLabelError.setText(e1.getMessage());
}

```

#### Vulnerabilities:

Proiektu osoan zegoen ahultasun bakarra zuzentzeko, *dbFactory.setFeature()*; jari behar izan dugu.

```

try {
    DocumentBuilderFactory dbFactory = DocumentBuilderFactory.newInstance();
    dbFactory.setFeature("http://apache.org/xml/features/disallow-doctype-decl", true);
    DocumentBuilder dBuilder = dbFactory.newDocumentBuilder();
    Document doc = dBuilder.parse(new File(configFile));
    doc.getDocumentElement().normalize();
}

```

#### Code Smells:

Ez dago Blocker zorrotasunezko Code Smell-ik, beraz besteetako bana aurkeztuko dugu:

#### Critical:

Define a constant instead of duplicating this literal "initialize" 3 times.

Kodean "initialize" 3 aldiz agertzen da, beraz konpontzeko konstante bat definituko dugu balio horrekin:

```

private static final String INITIALIZE = "inialize";

public BLFacadeImplementation() {
    System.out.println("Creating BLFacadeImplementation instance");
    ConfigXML c=ConfigXML.getInstance();

    if (c.getDataBaseOpenMode().equals(INITIALIZE)) {
        dbManager=new DataAccess(c.getDataBaseOpenMode().equals(INITIALIZE));
        dbManager.initializeDB();
        dbManager.close();
    }
}

public BLFacadeImplementation(DataAccess da) {
    System.out.println("Creating BLFacadeImplementation instance with DataAccess parameter");
    ConfigXML c=ConfigXML.getInstance();

    if (c.getDataBaseOpenMode().equals(INITIALIZE)) {
        da.open(false);
        da.initializeDB();
        da.close();
    }
    dbManager=da;
}
}

```

Major:

This block of commented-out lines of code should be removed.

Hemen bakarrik, import bat comentatuta dago eta lerroa kendu behar dugu zuzentzeko:

```

9
10 //import domain.Booking;
11 import domain.Question;
12 import domain.TMBet;
13 import domain.Transaction;

```

Replace the synchronized class "Vector" by an unsynchronized one such as "ArrayList" or "LinkedList".

Arazo honetan metodoak bueltatzen zuena Verctor izatetik List izatera aldatu beharra zegoen, interface bat zenez suposatu du beste klaseetan ere aldatu behar izatea:

```

/**
 * This method retrieves the events of a given date
 *
 * @param date in which events are retrieved
 * @return collection of events
 */
@WebMethod public List<Event> getEvents(Date date);

```

Minor:

Remove this unnecessary cast to "Node".

Hemen configXML klasera joango gara eta getTagValue klaseko (Node) castea kenduko dugu:

```

private static String getTagValue(String sTag, Element eElement)
{
    NodeList n1List= eElement.getElementsByTagName(sTag).item(0).getChildNodes();
    Node nValue = (Node) n1List.item(0);

    return nValue.getNodeValue();

}

```

Info:

Complete the task associated to this TODO comment.

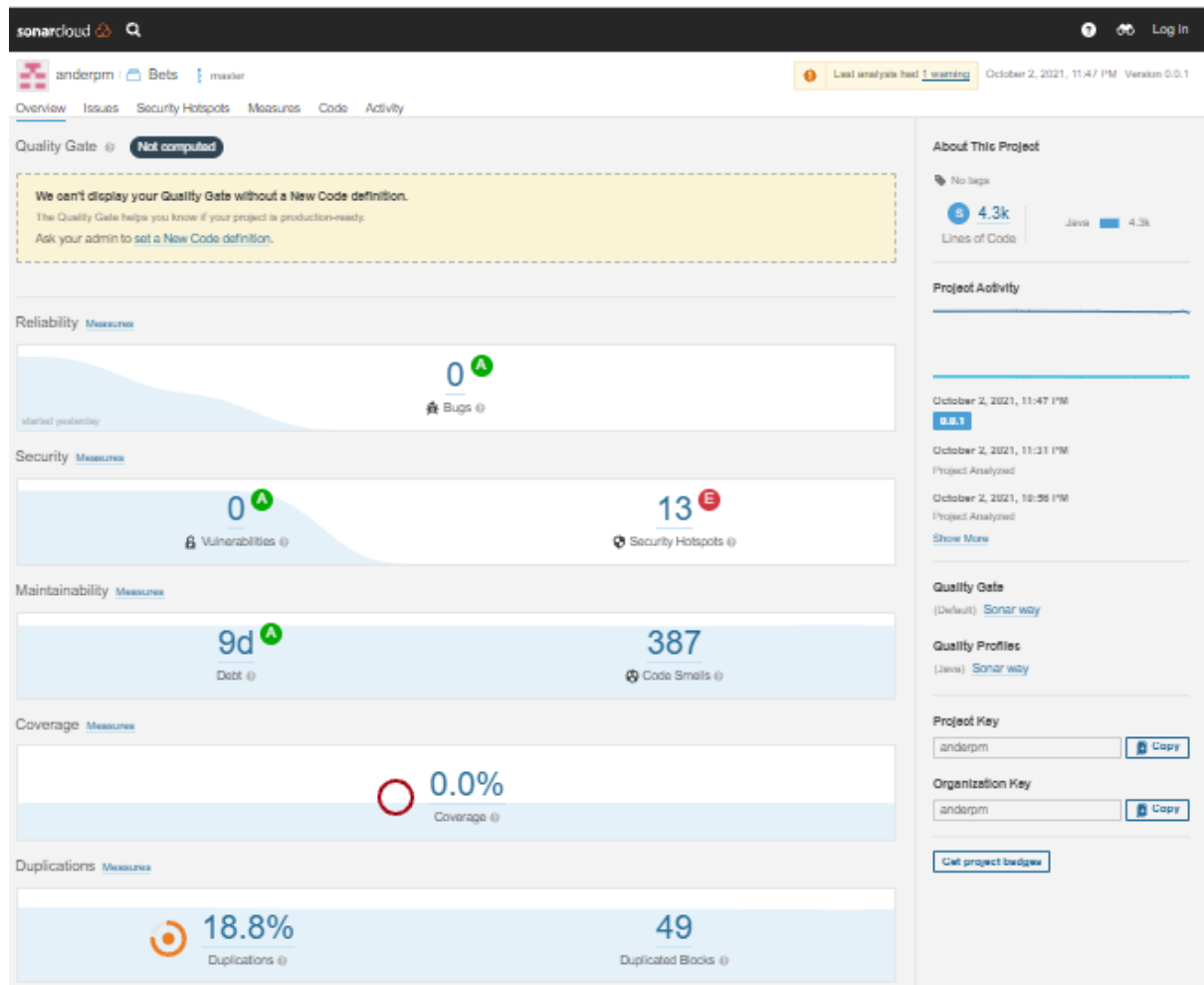
Proiektuko info bakarra da, TODO-k zeregin bat markatzen zuen, baina eginda dagoenez borratu behar da.

```

public void windowClosing(WindowEvent e) {
    try {
        //if (ConfigXML.getInstance().isBusinessLogicLocal()) facade.close();
    } catch (Exception e1) {
        // TODO Auto-generated catch block
        System.out.println("Error: "+e1.toString()+" , probably problems with Business Logic or Database");
    }
    mainFrame.setVisible(true);
    //System.exit(1);
}

```

Bug, vulnerabilities eta Code Smell batzuk zuzendu ondoren, hau da sornarcloud-en agertzen den proiektuko informazioa:



Hasierako Bets proiektuaren github helbidea: <https://github.com/anderpm/hasierakoBets>

Sonarcloud-era igo den Bets proiektuaren github helbidea: <https://github.com/anderpm/Bets>

Sonarcloud-eko esteka: <https://sonarcloud.io/dashboard?id=anderpm>