



Parte 0 - Laboratorio Sonar

Proyecto BETS - Ingeniería del Software II

16 de Octubre de 2023

Nicolás Aguado

Xiomara Cáceres

Eider Fernández

UPV/EHU

Índice

Índice.....	1
Abstracto.....	3
Informe Antes.....	3
Code Smells Arreglados.....	4
Nico - #1,2,3,4.....	4
Descripción del fallo.....	4
Solución.....	4
Nico - #5.....	5
Descripción del Fallo.....	5
Solución.....	5
Eider - #6, 7, 8.....	5
Descripción del Fallo.....	5
Solución.....	5
Eider - #9.....	6
Descripción del Fallo.....	6
Solución.....	6
Eider - #10.....	7
Descripción del Fallo.....	7
Solución.....	7
Xiomara - #11, 12, 13, 14, 15.....	8
Descripción del Fallo.....	8
Solución.....	8
Xiomara - #16.....	9
Descripción del Fallo.....	9
Solución.....	9
Bugs Arreglados.....	10
Nico - #1.....	10
Descripción del fallo.....	10
Solución.....	10
Nico - #2,3,4,5.....	11
Descripción de los fallos y solución.....	11
Solución.....	11
Xiomara - #6, 7, 8.....	12
Descripción de los fallos y solución.....	12
Solución.....	12

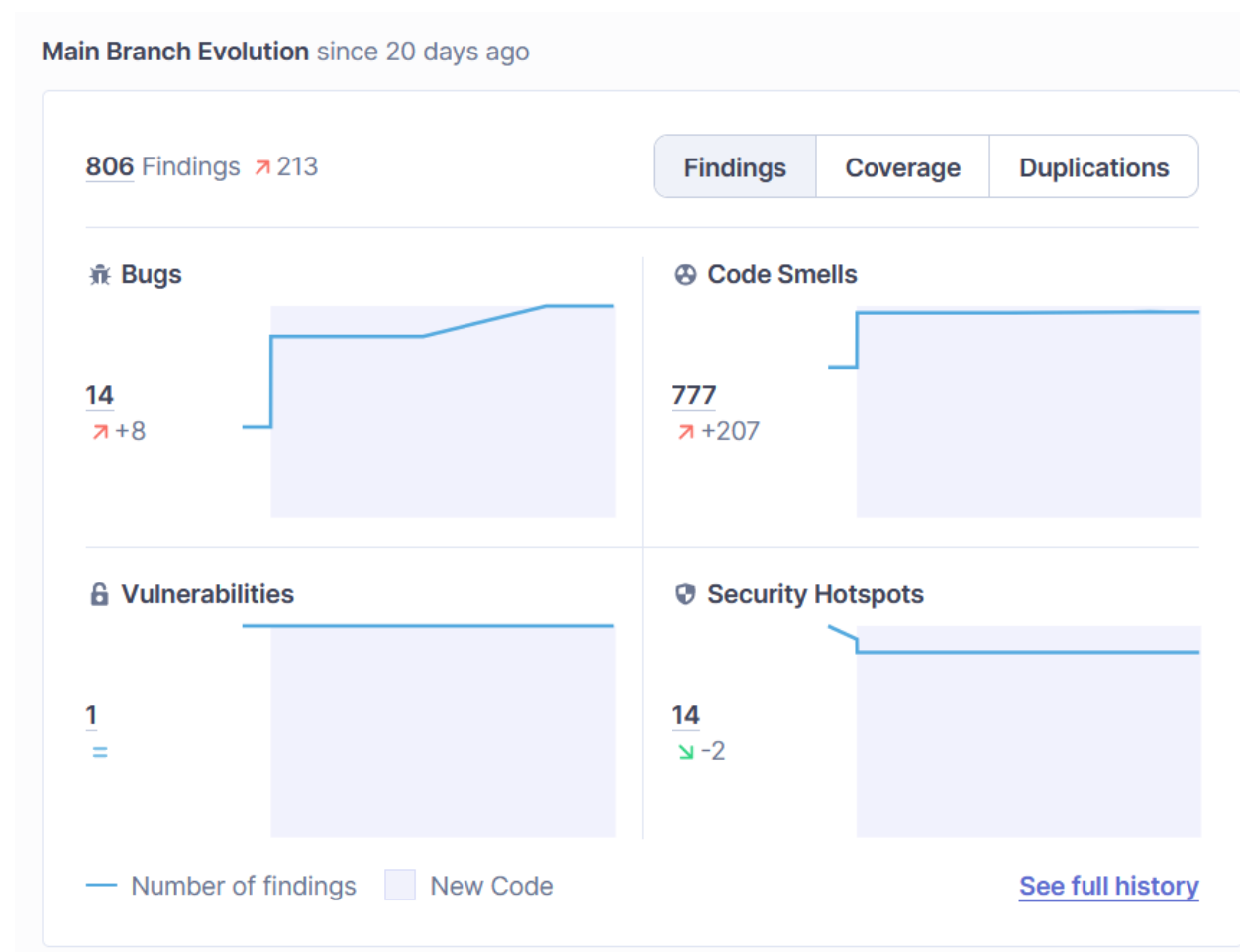
Xiomara - #9.....	13
Descripción de los fallos y solución.....	13
Solución.....	13
Eider - #10, 11, 12.....	13
Descripción del fallo y solución.....	13
Solución.....	14
Eider - #13, 14.....	15
Descripción del fallo y solución.....	15
Solución.....	15
Informe Después.....	17
Enlace.....	17

Abstracto

En este laboratorio se trabaja con la herramienta SonarCloud. Bajo un entorno de análisis, se pueden comprobar diversos fallos y/o defectos leves en un código con la finalidad de mejorar la eficiencia y lectura y reducir la posibilidad de error.

Informe Antes

Este es el estado del proyecto en la herramienta SonarCloud antes de hacer ningún cambio relacionado con este laboratorio.



Code Smells Arreglados

Nico - #1,2,3,4

Estos code smells han sido trabajados por **Nicolás Aguado**. El enlace en sonarcloud para los code smells [uno](#), [dos](#), [tres](#) y [cuatro](#).

Descripción del fallo

src/main/java/businessLogic/BusinessLogicServer.java

En esta clase se están usando 4 bloques de código entre corchetes sin ningún motivo aparente (he ahí 4 code smells). Son lo que se conocen como *code blocks*. Este es un fallo de tipo de intencionalidad que complica la lectura.

```
contentPanel.setLayout(new BorderLayout(0, 0));
{
    textArea = new JTextArea();
    contentPanel.add(textArea);
}
{
    JPanel buttonPane = new JPanel();
    buttonPane.setLayout(new FlowLayout(FlowLayout.RIGHT));
    getContentPane().add(buttonPane, BorderLayout.SOUTH);
    {
        JButton okButton = new JButton("OK");
        okButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                textArea.append("\n\n\nClosing the server... ");

                //server.close();

                System.exit(1);
            }
        });
        okButton.setActionCommand("OK");
        buttonPane.add(okButton);
        getRootPane().setDefaultButton(okButton);
    }
    {
        JButton cancelButton = new JButton("Cancel");
        cancelButton.setActionCommand("Cancel");
        buttonPane.add(cancelButton);
    }
}
}
```

Solución

La solución es simplemente eliminar los *bloques de código* (eliminando los corchetes) y dejar el código en el método sin encapsularlo. [Enlace al commit](#)

```
textArea = new JTextArea();
contentPanel.add(textArea);

JPanel buttonPane = new JPanel();
buttonPane.setLayout(new FlowLayout(FlowLayout.RIGHT));
getContentPane().add(buttonPane, BorderLayout.SOUTH);

JButton okButton = new JButton("OK");
okButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        textArea.append("\n\n\nClosing the server... ");

        //server.close();

        System.exit(1);
    }
});
okButton.setActionCommand("OK");
buttonPane.add(okButton);
getRootPane().setDefaultButton(okButton);

JButton cancelButton = new JButton("Cancel");
cancelButton.setActionCommand("Cancel");
buttonPane.add(cancelButton);
```

Nico - #5

Este code smell ha sido trabajado por **Nicolás Aguado**. Enlace a [sonar](#).

Descripción del Fallo

src/main/java/businessLogic/BusinessLogicServer.java

Es mala práctica dejar líneas comentadas en el código porque perjudican la posterior lectura y análisis del código

```
okButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        textArea.append("\n\n\nClosing the se
        //server.close();
        System.exit(1);
    }
});
```

Solución

La línea comentada ha sido eliminada. [Enlace al commit](#)

```
okButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        textArea.append("\n\n\nClosing the server..
        System.exit(1);
    }
});
okButton.setActionCommand("OK");
```

Eider - #6, 7, 8

Este code smell ha sido trabajado por **Eider Fernández**.

Descripción del Fallo

src/main/java/gui/ApustuaEginGUI.java

No debería haber imports innecesarios (que no se usen), ya que esto puede ser confuso.

```
34 import domain.Question;
35 import domain.Quote;
36 import domain.Transaction;
37 import domain.Registered;
38 import exceptions.ApustuaAlreadyExist;
39 import exceptions.EventFinished;
40 import java.awt.Font;
```

Solución

Las líneas han sido eliminadas. [Enlace al commit](#).

```

34 import domain.Question;
35 import domain.Quote;
36 import domain.Registered;
37 import java.awt.Font;

```

Eider - #9

Este code smell ha sido trabajado por **Eider Fernández**.

Descripción del Fallo

src/main/java/gui/ApustuBerdinakGUI.java

Tener Strings literales duplicados puede llevar a errores a la hora de necesitar cambiar ese valor, pues habría que cambiar todas las apariciones de eseString literal.

```

this.setTitle(ResourceBundle.getBundle(1 "Etiquetas").getString("ApustuBerdinak"));

Duplication
lblApustuak = new JLabel(ResourceBundle.getBundle(2 "Etiquetas").getString("ErabApustuak"));
lblApustuak.setBackground(Color.PINK);
lblApustuak.setOpaque(true);
lblApustuak.setHorizontalAlignment(SwingConstants.CENTER);
lblApustuak.setBounds(10, 10, 366, 13);
getContentPane().add(lblApustuak);

list = new JList();
list.setModel(apustuLista);
list.setBounds(138, 70, 1, 1);
getContentPane().add(list);

scrollBar = new JScrollPane(list);
scrollBar.setBounds(10, 34, 366, 166);
getContentPane().add(scrollBar);
apustuLista.addAll(businessLogic.findApustuAnitza(copia));

Duplication
btnClose = new JButton(ResourceBundle.getBundle(3 "Etiquetas").getString("Close"));

```

Solución

Se ha creado una constante llamada ETIQUETAS que guarda dicho String. [Enlace al commit](#).

Pasaba exactamente lo mismo en la clase AdminGUI.java tras 10 veces, así que también se ha cambiado eso. [Enlace al commit](#).

```

private static final String ETIQUETAS = "Etiquetas";

public ApustuBerdinakGUI(Registered copia, Registered me) {
    this.getContentPane().setLayout(null);
    this.setSize(new Dimension(400, 300));
    this.setTitle(ResourceBundle.getBundle(ETIQUETAS).getString("ApustuBerdinak"));

    lblApustuak = new JLabel(ResourceBundle.getBundle(ETIQUETAS).getString("ErabApustuak"));
    lblApustuak.setBackground(Color.PINK);
    lblApustuak.setOpaque(true);
    lblApustuak.setHorizontalAlignment(SwingConstants.CENTER);
    lblApustuak.setBounds(10, 10, 366, 13);
    getContentPane().add(lblApustuak);

    list = new JList();
    list.setModel(apustuLista);
    list.setBounds(138, 70, 1, 1);
    getContentPane().add(list);

    scrollBar = new JScrollPane(list);
    scrollBar.setBounds(10, 34, 366, 166);
    getContentPane().add(scrollBar);
    apustuLista.addAll(businessLogic.findApustuAnitza(copia));

    btnClose = new JButton(ResourceBundle.getBundle(ETIQUETAS).getString("Close"));

```

Eider - #10

Descripción del Fallo

src/main/java/gui/AdminGUI.java

No debería haber atributos privados que no se usen.

```

private JButton btnNewButton_1;

private Registered user;
private JButton jButtonEmaitzaIpini;
private JButton jButtonDesLogin;
private JFrame thisw;
private JButton btnNewButton_2;
private JButton btnGertaerakKopiatu;
private static final String ETIQUETAS = "Etiquetas";

```

Solución

Se han eliminado esa línea de código. [Enlace al commit](#).


```

private Registered user;
private JButton jButtonEmaitzalipini;
private JButton jButtonDesLogin;
private JFrame thisw;
private JButton btnGertaerakKopiatu;
private static final String ETIQUETAS = "Etiquetas";

```

Xiomara - #11, 12, 13, 14, 15

Estos code smells han sido trabajados por **Xiomara G. Cáceres**. El enlace de sonarcloud para los code smells [uno](#), [dos](#), [tres](#), [cuatro](#) y [cinco](#).

Descripción del Fallo

```

src/main/java/dataAccess/DataAccess.java
src/main/java/domain/ApustuAnitza.java
src/main/java/domain/Event.java
src/main/java/domain/Question.java

```

Solución

Se ha eliminado el tipo especificado entre los "<>". El enlace al commit [uno](#), [dos](#) y [tres](#).

694	-	Map<String, String> properties = new HashMap<String, String>();
694	+	Map<String, String> properties = new HashMap<>();
695	695	properties.put("javax.persistence.jdbc.user", c.getUser());
696	696	properties.put("javax.persistence.jdbc.password", c.getPassword());
---	---	
1039	1039	List<Registered> listR = Rquery.getResultList();
1040	-	List<Registered> ema= new ArrayList<Registered>();
1040	+	List<Registered> ema= new ArrayList<>();
1041	1041	int i;
29	-	private Vector<Apustua> apustuak = new Vector<Apustua>();
29	+	private Vector<Apustua> apustuak = new Vector<>();
30	30	@XmlIDREF
--	--	
26	-	private Vector<Question> questions=new Vector<Question>();
26	+	private Vector<Question> questions=new Vector<>();
27	27	@XmlIDREF

34	34		@OneToMany(fetch=FetchType.EAGER, cascade=CascadeType.A
35		-	private Vector<Quote> quotes=new Vector<Quote>();
	35	+	private Vector<Quote> quotes=new Vector<>();
36	36		

Xiomara - #16

Este code smell ha sido trabajado por **Xiomara G. Cáceres**. [El enlace.](#)

Descripción del Fallo

src/main/java/gui/EmaitzakIpiniGUI.java

Solución

Se ha eliminado la asignación a la variable *que* ya que era innecesaria. [El commit.](#)

154	154		jButtonEmaitzaIpini.addActionListener(new ActionListener() {
155	155		public void actionPerformed(ActionEvent e) {
156		-	Question que= (Question) jComboBoxQuestions.getSelectedItem();
157	156		Quote q = (Quote)jComboBoxQuotes.getSelectedItem();
158	157		try {
159	158		businessLogic.EmaitzakIpini(q);

Bugs Arreglados

Nico - #1

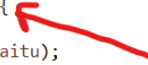
Este bug ha sido trabajado por **Nicolás Aguado**. Enlace en [sonar](#).

Descripción del fallo

src/main/java/dataAccess/DataAccess.java

Encontramos este bug en el método “seguir”, de la clase DataAccess. El problema viene cuando el autor del método desea comprobar si a la hora de una persona seguir a otra, esta acción ya se está realizando. Se está obteniendo la lista de seguidores, pero se está pasando como parámetro un “usuario”, no una instancia de la clase “seguidor”. Entonces, como en la lista no hay ninguna instancia de “usuario” ni la va a haber, la condición siempre se va a evaluar false.

```
public boolean jarraitu(Registered jabea, Registered jarraitua, Double limit) {
    Boolean b=false;
    Registered jarraitu = (Registered) db.find(Registered.class, jarraitua.getUsername());
    Registered harpideduna = (Registered) db.find(Registered.class, jabea.getUsername());
    if(!harpideduna.getJarraitutakolista().contains(jarraitu)) {
        db.getTransaction().begin();
        Jarraitzailea jar = new Jarraitzailea(harpideduna, jarraitu);
        harpideduna.addJarraitutako(jar);
        jarraitu.addJarraitzailea(jar);
        b=true;
        db.persist(jar);
        harpideduna.setDirulimitea(limit);
        db.getTransaction().commit();
    }
    return b;
}
```



Solución

Queremos hacer la comprobación correcta. Entonces, crearemos una clase de tipo “Jarraitzailea” y esa será la que comprobaremos si está en la lista. Además, podremos reutilizar código ya usado. [Enlace al commit](#)

```
public boolean jarraitu(Registered jabea, Registered jarraitua, Double limit) {
    Boolean b=false;
    Registered jarraitu = (Registered) db.find(Registered.class, jarraitua.getUsername());
    Registered harpideduna = (Registered) db.find(Registered.class, jabea.getUsername());
    Jarraitzailea jar = new Jarraitzailea(harpideduna, jarraitu);
    if(!harpideduna.getJarraitutakolista().contains(jar)) {
        db.getTransaction().begin();
        harpideduna.addJarraitutako(jar);
        jarraitu.addJarraitzailea(jar);
        b=true;
        db.persist(jar);
        harpideduna.setDirulimitea(limit);
        db.getTransaction().commit();
    }
    return b;
}
```

Nico - #2,3,4,5

Estos bugs han sido trabajados por Nicolás Aguado. En sonarcloud, contienen los enlaces [uno](#), [dos](#), [tres](#) y [cuatro](#).

Descripción de los fallos y solución

src/main/java/dataAccess/DataAccess.java

src/main/java/domain/Apustua.java

src/main/java/domain/Quote.java

Solución

A la hora de comparar dos instancias, éstas deben ser comparadas utilizando el operador .equals(...) en lugar de el operador ==. [Enlace al commit](#).

```

src/main/java/dataAccess/DataAccess.java
@@ -872,7 +872,7 @@ public boolean ApustuaEgin(Registered u, Vector<Quote> quote, Double balioa, Int
872 872         Registered nork = erab.getNork();
873 873         Boolean b=true;
874 874         for(ApustuAnitza apu: nork.getApustuAnitzak()) {
875 875 -             if(apu.getApustuKopia()==apustuAnitza.getApustuKopia()) {
876 876 +             if(apu.getApustuKopia().equals(apustuAnitza.getApustuKopia())) {
877 877                 b=false;
878 878             }
}

```

```

src/main/java/domain/Apustua.java
@@ -74,7 +74,7 @@ public void setApustuAnitza(ApustuAnitza apustuAnitza) {
74 74     }
75 75
76 76     public boolean galdutaMarkatu(Quote quo) {
77 77 -         if(kuota.getQuestion().getQuestionNumber()==quo.getQuestion().getQuestionNumber() && quo.getQuoteNumber()!=kuota.getQuoteNumber()) {
78 78 +         if(kuota.getQuestion().getQuestionNumber().equals(kuota.getQuestion().getQuestionNumber()) &&
79 79 +         !quo.getQuoteNumber().equals(kuota.getQuoteNumber())) {
80 80             this.egoera="galduta";
            return true;
}

```

```

src/main/java/domain/Quote.java
@@ -98,7 +98,7 @@ public void removeApustua(Apustua a) {
98 98     boolean aurkitua = false;
99 99     int i = 0;
100 100     while( i<apustuak.size() && !aurkitua) {
101 101 -         if(apustuak.get(i).getApostuaNumber()==a.getApostuaNumber()) {
102 102 +         if(apustuak.get(i).getApostuaNumber().equals(a.getApostuaNumber())) {
103 103             apustuak.remove(i);
104 104             aurkitua = true;
}
}

```

Xiomara - #6, 7, 8

Estos bugs han sido trabajados por **Xiomara G. Cáceres**. En Sonarcloud, contienen los enlaces [uno](#), [dos](#) y [tres](#).

Descripción de los fallos y solución

src/main/java/domain/Apostua.java

src/main/java/domain/Quote.java

src/main/java/domain/Sport.java

Solución

A la hora de tratar el equals(...) entre objetos se debe tratar todo tipo de condiciones. Es por ello que se ha generado el método equals(...) automáticamente. [Enlace al commit.](#)

```

88      89      @Override
89      -      public boolean equals(Object o) {
90      -          Apostua a = (Apostua) o;
91      -          if(a==null) {
90      +      public boolean equals(Object obj) {
91      +          if (this == obj)
92      +              return true;
93      +          if (obj == null)
92      94              return false;
93      -      }
94      -      return this.getApostuaNumber().equals(a.getApostuaNumber());
95      +          if (getClass() != obj.getClass())
96      +              return false;
97      +          Apostua other = (Apostua) obj;
98      +          return Objects.equals(apustuaNumber, other.apustuaNumber);
95      99      }

```

```

109     110     @Override
110     -      public boolean equals(Object o) {
111     -          if(o != null) {
112     -              Quote q = (Quote)o;
113     -              return this.getQuoteNumber()==q.getQuoteNumber();
114     -          } else {
111     +      public boolean equals(Object obj) {
112     +          if (this == obj)
113     +              return true;
114     +          if (obj == null)
115     115              return false;
116     -      }
116     +          if (getClass() != obj.getClass())
117     +              return false;
118     +          Quote other = (Quote) obj;
119     +          return Objects.equals(quoteNumber, other.quoteNumber);
117     120      }

```

```

73 74      @Override
74 -      public boolean equals(Object o) {
75 -          Sport sp = (Sport) o;
76 -          if(sp==null) {
75 +      public boolean equals(Object obj) {
76 +          if (this == obj)
77 +              return true;
78 +          if (obj == null)
79 +              return false;
80 +          if (getClass() != obj.getClass())
77 81              return false;
78 -      }
79 -      return this.izena.equals(sp.getIzena());
82 +      Sport other = (Sport) obj;
83 +      return Objects.equals(izena, other.izena);
80 84  }

```

Xiomara - #9

Este bug ha sido trabajado por **Xiomara G. Cáceres**. El enlace a Sonarcloud es [este](#).

Descripción de los fallos y solución

src/main/java/domain/Registered.java

Solución

A la hora de comparar dos Strings se debe usar `.equals(...)` en vez de `==`. [Enlace al commit](#).

```

src/main/java/domain/Registered.java
@@ -167,7 +167,7 @@ public void addJarraitutako(Jarraitzailea reg) {
167 167
168 168      @Override
169 169      public String toString() {
170 -          if(this.mode=="RankingGUI")
170 +          if(this.mode.equals("RankingGUI"))
171 171              return this.getUsername() + " " + this.zenbat + ": " + this.irabazitakoa + "€";
172 172      else
173 173          return this.getUsername();

```

Eider - #10, 11, 12

Estos bugs han sido trabajados por **Eider Fernández**. Enlaces en sonar: [uno](#), [dos](#), [tres](#).

Descripción del fallo y solución

src/main/java/domain/Apustua.java

src/main/java/domain/Quote.java

src/main/java/domain/Sport.java

Solución

Si dos objetos son iguales según una función equals(), llamar al hashCode en cualquiera de los objetos debería devolver el mismo valor integer. Así que se ha creado el método hashCode().

```
@Override
public int hashCode() {
    return Objects.hash(apustuAnitza, apustuaNumber, egoera, kuota);
}

@Override
public boolean equals(Object obj) {
    if (this == obj)
        return true;
    if (obj == null)
        return false;
    if (getClass() != obj.getClass())
        return false;
    Apustua other = (Apustua) obj;
    return Objects.equals(apustuAnitza, other.apustuAnitza) && Objects.equals(apustuaNumber, other.apustuaNumber)
        && Objects.equals(egoera, other.egoera) && Objects.equals(kuota, other.kuota);
}

@Override
public int hashCode() {
    return Objects.hash(apustuak, balio, forecast, question, quoteNumber);
}

@Override
public boolean equals(Object obj) {
    if (this == obj)
        return true;
    if (obj == null)
        return false;
    if (getClass() != obj.getClass())
        return false;
    Quote other = (Quote) obj;
    return Objects.equals(apustuak, other.apustuak) && Objects.equals(balio, other.balio)
        && Objects.equals(forecast, other.forecast) && Objects.equals(question, other.question)
        && Objects.equals(quoteNumber, other.quoteNumber);
}

@Override
public int hashCode() {
    return Objects.hash(apustuKantitatea, events, izena);
}

@Override
public boolean equals(Object obj) {
    if (this == obj)
        return true;
    if (obj == null)
        return false;
    if (getClass() != obj.getClass())
        return false;
    Sport other = (Sport) obj;
    return Objects.equals(apustuKantitatea, other.apustuKantitatea) && Objects.equals(events, other.events)
        && Objects.equals(izena, other.izena);
}
```

Eider - #13, 14

Estos bugs han sido trabajados por **Eider Fernández**. Enlaces en sonar: [uno](#), [dos](#).

Descripción del fallo y solución

src/test/java/(default package)/EmaitzakIpinibLBMTTest.java

No debería llamarse al mismo método que lanza excepciones dentro del mismo try.

```
public void test1() {
    //La cuota no existe en la BD
    q = new Quote(245.0, "2sedrf");
    try {
        Mockito.doThrow(new NullPointerException()).when(da).EmaitzakIpinib(q);
        da.EmaitzakIpinib(q);
        fail("No deberia de ir bien");
    } catch (EventNotFinished | NullPointerException e) {
        assertTrue(true);
    }
}

@Test
public void test2() {
    //La fecha de ese evento es posterior a la fecha actual
    calendar.set(2023, Calendar.NOVEMBER, 01, 0, 0, 0);
    q = new Quote(100.0, "2", new Question(15, "Who will win the match?", 1.0, new Event(21, "
    try {
        Mockito.doThrow(new EventNotFinished()).when(da).EmaitzakIpinib(q);
        da.EmaitzakIpinib(q);
        fail("No deberia de ir bien");
    } catch (EventNotFinished e) {
        assertTrue(true);
    }
}
```

Solución

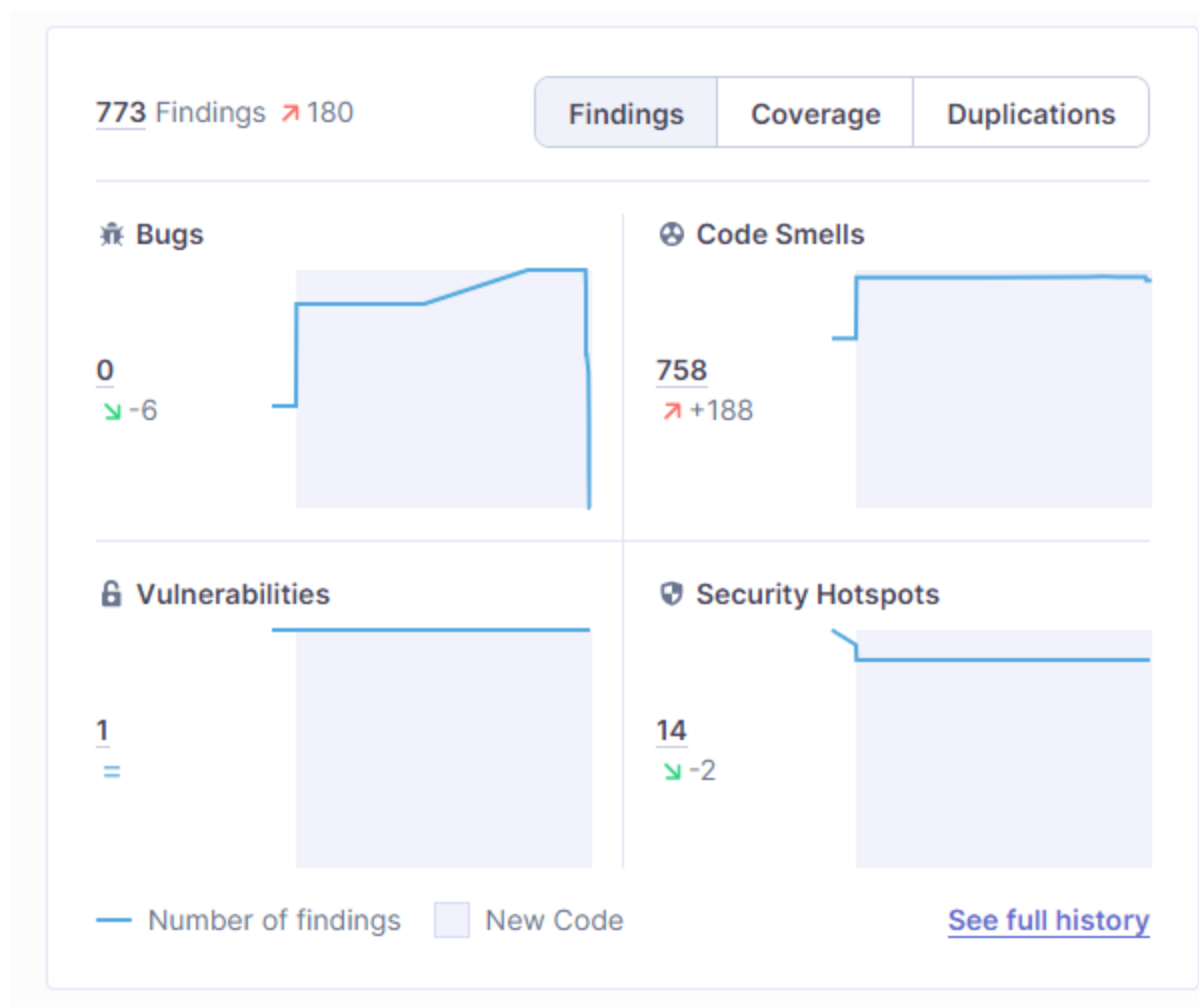
Sacar fuera del try una de las dos llamadas al método que lanza excepción.


```
@Test
public void test1() throws EventNotFinished {
    //La cuota no existe en la BD
    q = new Quote(245.0, "2sedrf");
    Mockito.doThrow(new NullPointerException()).when(da).EmaitzakIpini(q);
    try {
        da.EmaitzakIpini(q);
        fail("No deberia de ir bien");
    } catch (EventNotFinished | NullPointerException e) {
        assertTrue(true);
    }
}

@Test
public void test2() throws EventNotFinished {
    //La fecha de ese evento es posterior a la fecha actual
    calendar.set(2023, Calendar.NOVEMBER, 01, 0, 0, 0);
    q = new Quote(100.0, "2", new Question(15, "Who will win the match?", 1.0, new E
    Mockito.doThrow(new EventNotFinished()).when(da).EmaitzakIpini(q);
    try {
        da.EmaitzakIpini(q);
        fail("No deberia de ir bien");
    } catch (EventNotFinished e) {
        assertTrue(true);
    }
}
```

Informe Después

Este es el estado del proyecto en la herramienta SonarCloud después de hacer todos los cambios de todos los miembros del grupo relacionados con este laboratorio.



Enlace

Para una verificación y una mejor lectura del informe de sonarcloud, se puede visitar el enlace público del proyecto:

https://sonarcloud.io/project/overview?id=por0tos_Bet22porotos