

# **Proyecto Bets22**

## Verificación del Software 22-23

UPV-EHU Ingeniería del Software II

Anal Lucía Durán Lengo

Jon Ortega Goikoetxea

## Contenido

Integrantes y Horas de Trabajo .....	2
Método 1: gertaerakSortu() .....	2
Código .....	2
Pruebas Unitarias .....	2
Diseño .....	2
Implementación .....	3
Resultados .....	3
Pruebas de Integración .....	4
Diseño .....	4
Implementación .....	4
Resultados .....	4
Método 2: EmaitzakIpin() .....	5
Código .....	5
Pruebas Unitarias .....	6
Diseño .....	6
Implementación .....	9
Resultados .....	9
Pruebas de Integración .....	9
Diseño .....	9
Implementación .....	9
Resultados .....	9
Enlaces .....	10

## Integrantes y Horas de Trabajo

- Ana Lucía Durán Lengo: 15h → Método 1: gertaerakSortu()
- Jon Ortega Goikoetxea: 15h → Método 2: EmaitzakIpini()

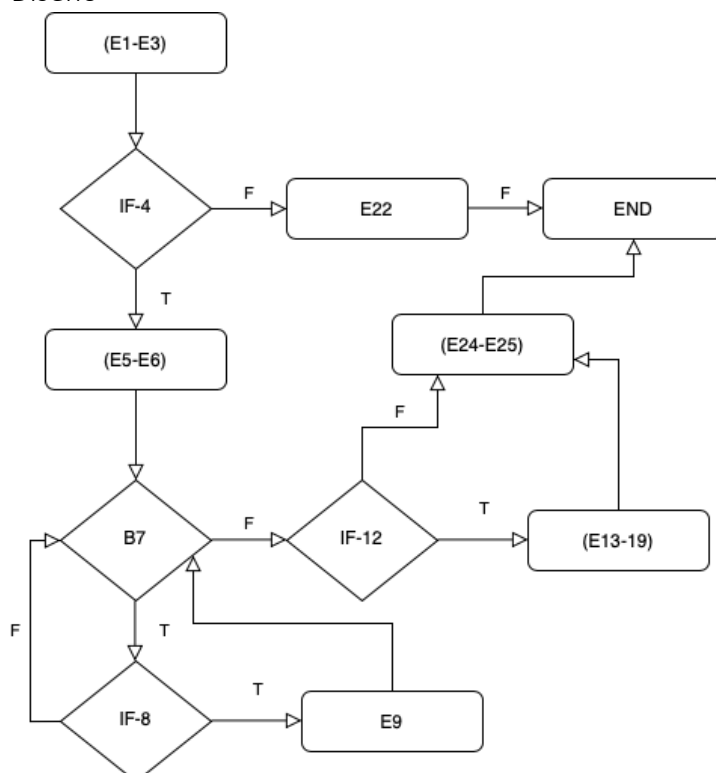
## Método 1: gertaerakSortu()

### Código

```
public boolean gertaerakSortu(String description, Date eventDate, String sport) {
    boolean b = true;
    db.getTransaction().begin();
    Sport spo = db.find(Sport.class, sport);
    if(spo != null) {
        TypedQuery<Event> Equerry = db.createQuery("SELECT e FROM Event e WHERE e.getEventDate() = ?1 ", Event.class);
        Equerry.setParameter(1, eventDate);
        for(Event ev: Equerry.getResultList()) {
            if(ev.getDescription().equals(description)) {
                b = false;
            }
        }
        if(b) {
            String[] taldeak = description.split("-");
            Team lokala = new Team(taldeak[0]);
            Team kanpokoa = new Team(taldeak[1]);
            Event e = new Event(description, eventDate, lokala, kanpokoa);
            e.setSport(spo);
            spo.addEvent(e);
            db.persist(e);
        }
    } else {
        return false;
    }
    db.getTransaction().commit();
    return b;
}
```

## Pruebas Unitarias

### Diseño



## Base de Datos Utilizada

Eventos	
Evento 1	("Sevilla- Betis", 14/03/2023, "Fútbol")
Evento 2	("Eibar-Sevilla", 28/02/2023, "Fútbol")
Evento 3	(null, 14/03/2023, "Fútbol")
Evento 4	("Sevilla- Betis", null, "Fútbol")
Evento 5	("Sevilla- Betis", 14/03/2023, null)
Evento 6	(12345, 14/03/2023, "Fútbol")
Evento 7	("Sevilla- Betis", "14/03/2023", "Fútbol")
Evento 8	("Sevilla- Betis", 14/03/2023, 12345)
Evento 9	("Sevilla- Betis", 14/03/2020, 12345)

## Caja Blanca

Caminos
IF4(F)-END
IF4(T)-B(0)-IF12(T)-END
IF4(T)-B(N)-IF8(T)-IF12(F)-END
IF4(T)-B(N)-IF8(F)-IF12(T)-END

Nº caso	Camino seguido	Condición de entrada	Entrada	Salida
1	IF4(F)-END	Sport no está en la base de datos	sport==null	false
2	IF4(T)-B(0)-IF12(T)-END	No hay evento en esa fecha		True
3	IF4(T)-B(N)-IF8(T)-IF12(F)-END	Hay evento en esa fecha		True
4	IF4(T)-B(N)-IF8(F)-IF12(T)-END	Hay evento para la descripción dada		False

## Caja Negra

Nº caso	Condiciones de entrada	Clases de equivalencia válidas	Clases de equivalencia no válidas
1	El evento está en la BD	Evento en BD(1)	Evento no está en BD(2)
2	Descripción no es null	description!=null(3)	description==null(4)
3	EventDate no es null	eventDate!=null(5)	eventDate==null(6)
4	sport no es null	sport!=null(7)	sport==null(8)
5	Descripción es de tipo String	description es String(9)	description no es String(10)
6	EventDate es de tipo date	eventDate es Date(11)	eventDate no es Date(12)
7	Sport es de tipo String	sport es String(13)	sport no es String(14)
8	Sport pertenece a sport.class	sport pertenece a sport.class(15)	sport no pertenece a sport.class(16)
9	La fecha es posterior a hoy	eventDate es posterior a hoy(17)	eventDate no es posterior a hoy(18)

Nº caso	Clases de equivalencia cubiertas	Entrada	Salida
1	1,3,5,7,9,11,13,15,17	Evento 1	El evento se crea, b=true
2		Evento 2	b=false
3		Evento 3	b=false
4		Evento 4	b=false
5		Evento 5	b=false
6		Evento 6	b=false
7		Evento 7	b=false
8		Evento 8	b=false
9		Evento 9	b=false
		Evento 9	b=false

## Implementación

## Resultados

## Pruebas de Integración

Diseño

Implementación

Resultados

## Método 2: EmaizakIpini()

### Código

```
public void EmaizakIpini(Quote quote) throws EventNotFinished{

    Quote q = db.find(Quote.class, quote);
    String result = q.getForecast();

    if(new Date().compareTo(q.getQuestion().getEvent().getEventDate())<0)
        throw new EventNotFinished();

    Vector<Apustua> listApustuak = q.getApustuak();
    db.getTransaction().begin();
    Question que = q.getQuestion();
    Question question = db.find(Question.class, que);
    question.setResult(result);
    for(Quote quo: question.getQuotes()) {
        for(Apustua apu: quo.getApustuak()) {

            Boolean b=apu.galdutaMarkatu(quo);
            if(b) {
                apu.getApustuAnitza().setEgoera("galduta");
            }else {
                apu.setEgoera("irabazita");
            }
        }
    }
    db.getTransaction().commit();
    for(Apustua a : listApustuak) {
        db.getTransaction().begin();
        Boolean bool=a.getApustuAnitza().irabazitaMarkatu();
        db.getTransaction().commit();
        if(bool) {
            this.ApustuaIrabazi(a.getApustuAnitza());
        }
    }
}
```

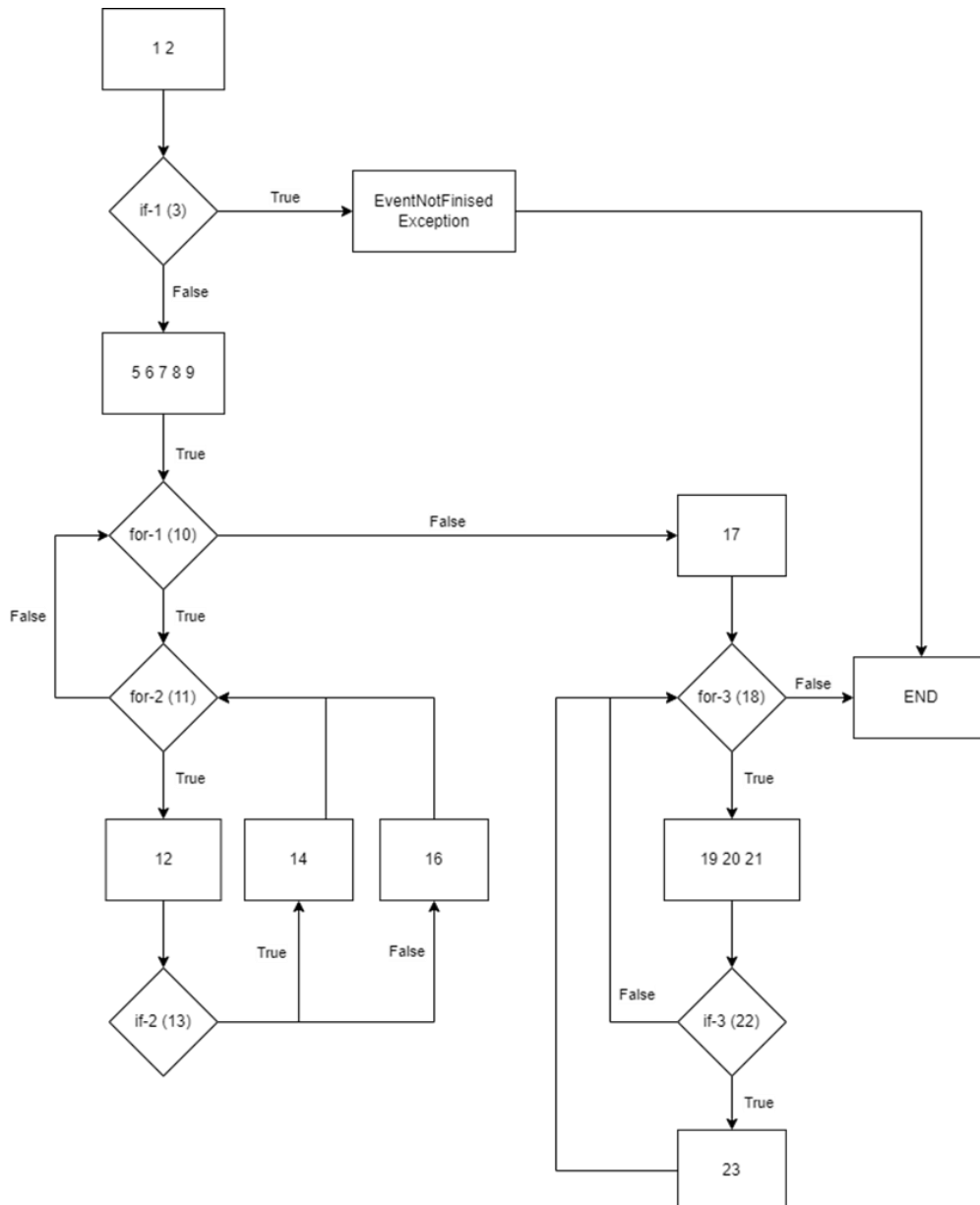
# Pruebas Unitarias

## Diseño

### Base de Datos Utilizada

Base de datos Caja negra					
Team1	Atletico (nombre)				
Team2	Athletic (nombre)				
Event1	1 (numero de evento)	Atletico-Athletic (descripcion)	17/01/2023 (fecha)	team1 (local)	team2 (visitante)
Event2	2 (numero de evento)	Atletico-Athletic (descripcion)	17/01/2021 (fecha)	team1 (local)	team2 (visitante)
Question1	¿Quién ganara el partido? (pregunta)	1.0 (apuesta minima)	Event1 (event)		
Question2	¿Quién marcara primero? (pregunta)	2.0 (apuesta minima)	Event2 (event)		
Quote1	100 (valor)	2 (forecast)	Question1 (pregunta)		
Quote2	20 (valor)	2 (forecast)	Question2 (pregunta)		
Quote3	50 (valor)	1 (forecast)	Question1 (pregunta)		
ApustuAnitza1	registrado (estado)	5.0			
Apuesta1	ApustuAnitza1	Quote3			
ApustuAnitza2	registrado (estado)	3.0			
Apuesta2	ApustuAnitza2	Quote1			
Base de datos Caja blanca					
Team1	Atletico (nombre)				
Team2	Athletic (nombre)				
Event1	3 (numero de evento)	Atletico-Athletic (descripcion)	17/01/2024 (fecha)	team1 (local)	team2 (visitante)
Event2	2 (numero de evento)	Atletico-Athletic (descripcion)	17/01/2021 (fecha)	team1 (local)	team2 (visitante)
Question1	¿Quién ganara? (pregunta)	1.0 (apuesta minima)	Event1 (event)		
Question2	¿Quién marcara primero? (pregunta)	2.0 (apuesta minima)	Event2 (event)		
Question3	¿Quién marcara primero? (pregunta)	2.0 (apuesta minima)	Event1 (event)		
Question4	¿Quién marcara ultimo? (pregunta)	2.0 (apuesta minima)	Event1 (event)		
Quote1	10 (valor)	X (forecast)	Question1 (pregunta)		
Quote2	20 (valor)	2 (forecast)	Question2 (pregunta)		
Quote3	50 (valor)	2 (forecast)	Question1 (pregunta)		
Quote4	20 (valor)	2 (forecast)	Question3 (pregunta)		
Quote5	5 (valor)	2 (forecast)	Question4 (pregunta)		
Quote6	2 (valor)	1 (forecast)	Question4 (pregunta)		
ApustuAnitza1	registrado (estado)	5.0			
Apuesta1	ApustuAnitza1	Quote3			
ApustuAnitza2	registrado (estado)	2.0			
Apuesta2	ApustuAnitza2	Quote4			
ApustuAnitza3	registrado (estado)	40.0			
Apuesta3	ApustuAnitza3	Quote5			
Apuesta4	ApustuAnitza3	Quote6			

Caja Blanca



$V(G) = 7$

Caminos:

1- 1 2 if-1(T) Except. END

2- 1 2 if-1(F) 5 6 7 8 9 for-1(T) for-2(F) for-1(F) 17 for-3(F) END

3- 1 2 if-1(F) 5 6 7 8 9 for-1(T) for-2(T) 12 if-2(T) 14 for-2(F) for-1(T) for-2(F) for-1(F) 17 for-3(F) END

4- 1 2 if-1(F) 5 6 7 8 9 for-1(T) for-2(T) 12 if-2(F) 16 for-2(F) for-1(F) 17 for-3(T) 19 20 21 if-3(T) for-3(F) END

5- 1 2 if-1(F) 5 6 7 8 9 for-1(T) for-2(T) 12 if-2(F) 16 for-2(F) for-1(T) for-2(T) 12 if-2(T) 14 for-2(F) for-1(F) 17 for-3(T) 19 20 21 if-3(F) for-3(F) END



#caso	Condición de Entrada	Valor Entrada	Valor Salida
1	todayDate > eventDate	Quote2	EventNotFinished exception
2	question1.quotes.size() = 1	Quote1	No hay salida
3	question1.quotes.size() = 2 quote3.apustuak.size() = 1 apuesta1.galdutaMarkatu(Quote1) = true	Quote1	No hay salida
4	question3.quotes.size() = 1 quote4.apustuak.size() = 1 apuesta2.galdutaMarkatu(Quote4) = false Quote4.apustuak.size() = 1 apuesta2.getApustuAnitza().irabazitaMarkatu() = true	Quote4	No hay salida
5	question4.quotes.size() = 2 quote5.apustuak.size() = 1 quote6.apustuak.size() = 1 apuesta3.galdutaMarkatu(Quote4) = false apuesta4.galdutaMarkatu(Quote4) = true Quote5.apustuak.size() = 1 apuesta3.getApustuAnitza().irabazitaMarkatu() = false	Quote5	No hay salida

### Caja Negra

Condición de Entrada	Clases de Equivalencia Válidas	Clases de Equivalencia No Válidas
¿La fecha del evento es posterior?	true (1)	false (2)
¿Apuesta perdida?	true (3)	
¿Apuesta ganada?	true (4)	

#	Clases Cubiertas	Contexto de Entrada		Contexto de Salida	
		Estado BD	Parámetros	Estado BD	Salida
1	1, 3, 4	Quote siempre va a estar en BD (porque es un parámetro que se pasa desde la interfaz y no puede ser null i no estar en la BBDD)	Quote3	question1.resultado = 1 apuesta1.egoera = "irabazita" apuesta2.apustuAnitza2.egoera = "galduta" this.Apustualrabazi(apuesta1.getApustuAnitza())	No hay salida
2	2		Quote2	No cambia	EventNotFinished

Implementación

Resultados

### Pruebas de Integración

Diseño

Implementación

Resultados

## Enlaces

GitHub: <https://github.com/anadurlen/IS2Bets>

Sonarcloud: <https://sonarcloud.io/project/overview?id=IS2Bets>

**Nota:** Hemos realizado esta práctica junto con otro grupo, siendo sus integrantes Pablo Martínez Amunarri y Raúl Silva Alonso.