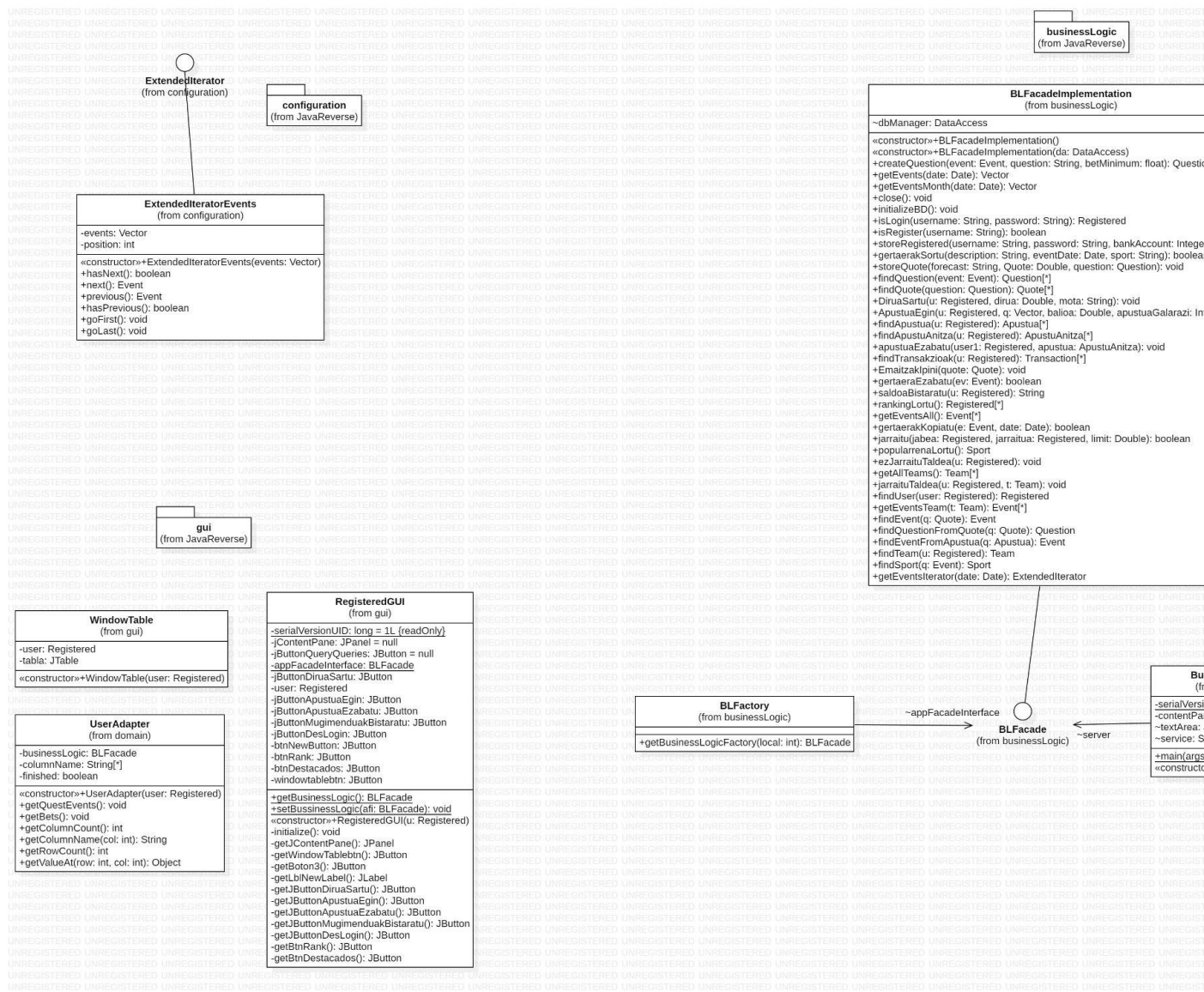


Trabajo Patrones de Diseño

Leire Gesteira y Daria Paslavska



(se puede ver en la carpeta de documentación del proyecto)

1. Patrón Factory Method

- a) Para implementar el patrón de diseño Factory hemos creado una clase llamada BLFactory que devuelve un BLFacade especificando si la conexión es local o no. El creador es el BLFactory, el product es el BLFacade y el ConcreteProduct es el int que especifica si la conexión es local o no.

- b) Hemos creado el método `getBusinessLogicFactory(int local)` que está basado en un trozo de código del método `main` de la clase `ApplicationLauncher` que dado un `BLFactory`, devuelve un `BLFacade` local o externo.

```
package businessLogic;

import java.net.MalformedURLException;

public class BLFactory {

    BLFacade appFacadeInterface;

    public BLFacade getBusinessLogicFactory(int local) {

        ConfigXML c=ConfigXML.getInstance();
        if(local==1) {
            DataAccess da= new DataAccess(c.getDataBaseOpenMode().equals("initialize"));
            appFacadeInterface=new BLFacadeImplementation(da);

        }else { //If remote

            String serviceName= "http://"+c.getBusinessLogicNode()+"."+c.getBusinessLogicPort()+"/ws/"+c.getBusinessLogicName()+"?wsdl";

            //URL url = new URL("http://localhost:9999/ws/ruralHouses?wsdl");
            URL url;
            try {
                url = new URL(serviceName);

                //1st argument refers to wsdl document above
                //2nd argument is service name, refer to wsdl document above
                QName qname = new QName("http://businessLogic/", "FacadeImplementationWSService");
                QName qname = new QName("http://businessLogic/", "BLFacadeImplementationService");

                Service service = Service.create(url, qname);

                appFacadeInterface = service.getPort(BLFacade.class);
            } catch (MalformedURLException e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
        }
        return appFacadeInterface;
    }
}
```

2. Patrón Iterator

- a) Para implementar el patrón de diseño Iterator hemos creado una Interfaz llamada `ExtendedIterator` que declara unos métodos que sirven para iterar de manera contraria a la usual (de fin a principio).

A su vez, también hemos creado la clase `ExtendedIteratorEvents` que implementa los métodos previamente mencionados de la interfaz `ExtendedIterator` y unos cuantos mas para poder iterar de principio a fin.

b)

En la clase `ExtendedIteratorEvents`:

```
public ExtendedIteratorEvents(Vector<Event> events) {
    this.events=events;
    this.position = events.size() - 1;
}
```

Hemos guardado `events` que recibimos del método `getEvents(Date date)` de la base de datos.

```

@Override
    public Event next() {
        if(hasNext()) {
            Event e=events.get(position);
            position++;
            return e;
        }
        return null;
    }

    @Override
    public Event previous() {
        if (hasPrevious()) {
            return events.get(position--);
        } else {
            return null; // Or handle the case when there are no more elements
        }
    }

    @Override
    public boolean hasPrevious() {
        return position >= 0;
    }

    @Override
    public void goFirst() {
        position=0;
    }

    @Override
    public void goLast() {
        position = events.size() - 1;
    }

```

Aquí implementamos métodos para iterar en nuestro ExtendedIterator.

c)

```

DataBase closed
Opening DataAccess instance => isDatabaseLocal: true getDatabBaseOpenMode: initialize
>> DataAccess: getEvents
1;Atletico-Athletic
2;Eibar-Barcelona
3;Getafe-Celta
4;Alaves-Deportivo
5;Espanol-Villareal
6;Las Palmas-Sevilla
7;Malaga-Valencia
8;Girona-Leganes
9;Real Sociedad-Levante
10;Betis-Real Madrid
22;LA Lakers-Phoenix Suns
23;Atlanta Hawks-Houston Rockets
24;Miami Heat-Chicago Bulls
27;Djokovic-Federer
DataBase closed

```

RECORRIDO	HACIA	ATRÁS
27;Djokovic-Federer		
24;Miami Heat-Chicago Bulls		
23;Atlanta Hawks-Houston Rockets		
22;LA Lakers-Phoenix Suns		
10;Betis-Real Madrid		
9;Real Sociedad-Levante		
8;Girona-Leganes		
7;Malaga-Valencia		
6;Las Palmas-Sevilla		
5;Espanol-Villareal		
4;Alaves-Deportivo		
3;Getafe-Celta		
2;Eibar-Barcelona		
1;Atletico-Athletic		

RECORRIDO	HACIA	ADELANTE
1;Atletico-Athletic		
2;Eibar-Barcelona		
3;Getafe-Celta		
4;Alaves-Deportivo		
5;Espanol-Villareal		
6;Las Palmas-Sevilla		
7;Malaga-Valencia		
8;Girona-Leganes		
9;Real Sociedad-Levante		
10;Betis-Real Madrid		
22;LA Lakers-Phoenix Suns		
23;Atlanta Hawks-Houston Rockets		
24;Miami Heat-Chicago Bulls		

3. Patrón Adapter

- a) Hemos creado una clase nueva UserAdapter.java donde guardamos información sobre los bets de usuario registrado y eventos y preguntas sobre estos bets. Lo añadimos en tres listas **private** List <ApustuAnitza> **betslist**, **private** List <Question> **questions** y **private** List <Event> **events**

Hemos creado otro clase WindowTable.java que hace la tabla en la ventana nueva

Y en clase RegisteredGUI.java hemos añadido nuevo método **private** JButton getWindowTablebtn() donde configuramos nueva boton WindowTable.

b)

En la clase UserAdaptor:

```

public UserAdapter(Registered user) {
    this.user = user;
    getBets();
    getQuestEvents();
    // Initialize 'data' with the appropriate data from the user
}

```

Guardamos registered user en el campo privado.

```

public void getBets() {

    for(ApustuAnitza ap : businessLogic.findApustuAnitza(user)){
        finished=false;
        if(ap.getEgoera().equals("jokoan")) {
            for(Apustua a: ap.getApustuak()) {
                if(new
Date().compareTo(businessLogic.findEventFromApustua(a).getEventDate())>=0) {
                    finished=true;
                }
            }
            if(!finished) {
                betslit.add(ap);
            }
        }
    }
}

```

En el metodo getBets() guardamos en betslit todos los bets que el usuario ha hecho, es decir, en cada ApustuAnitza ap si situacions esta igual a “jokoan” miramos todos los Apustua a que ApustuAnitza ap tiene y si la data es correcta (del futuro) y de este registered usuario, lo añadimos en la lista.

```

public void getQuestEvents() {

    for(ApustuAnitza bet : betslit) {
        Vector<Apustua> apustua = bet.getApustuak();
        for(Apustua a : apustua) {
            Quote q = a.getKuota();
            Question quest = q.getQuestion();
            Event e = quest.getEvent();
            questions.add(quest);
            events.add(e);
            System.out.println("Question: " + quest.getQuestion() + " Event: " +
e.getDescription());
            System.out.println("Event date: " + e.getEventDate());
        }
    }
}

```

```

        System.out.println(events.size());
        for(Event ev: events) {
            System.out.println(ev.getDescription());
            System.out.println(ev.getEventDate());
        }
    }
}

```

En el método getQuestEvent() queremos guardar los eventos y cuestionarios de bets del usuario registrado y ponerlo en la tabla. Para hacerlo miramos todos los bets hechos por el usuario ApustuAnitza bet : `betslist` donde cogemos apustua del `bet.getApustuak()` y mirando cada apustua cogo el quote. Solo después puedo ver eventos y questions del cada bet de usuario.

```

public Object getValueAt(int row, int col) {

    System.out.println(col);

    switch (col) {
        case 0:
            return events.get(row).getDescription();
        case 1:
            return questions.get(row).getQuestion();
        case 2:
            return events.get(row).getEventDate();
        case 3:
            return betslist.get(row).getBalioa();
        default:
            return null;
    }


}

```

Y después podemos coger todo para printearlo en la tabla.

c)

Nueva boton WindowTable:

 Registered

Select Option

Query Questions

Enter money

Bet

Remove bet

See movements

Ranking

Window Table

Featured

Close session

Apuestasrealizadaspordasha:			
Event	Cuestion	Event Date	Bet
Atletico-Athletic	Zeinek irabaziko du partidua?	Fri Nov 17 02:00:00 MSK 2023	12.0
Atletico-Athletic	Zeinek irabaziko du partidua?	Fri Nov 17 02:00:00 MSK 2023	20.0