

# TFC – Aplicació web per gestionar tornejos d'escacs, amb tecnologia J2EE

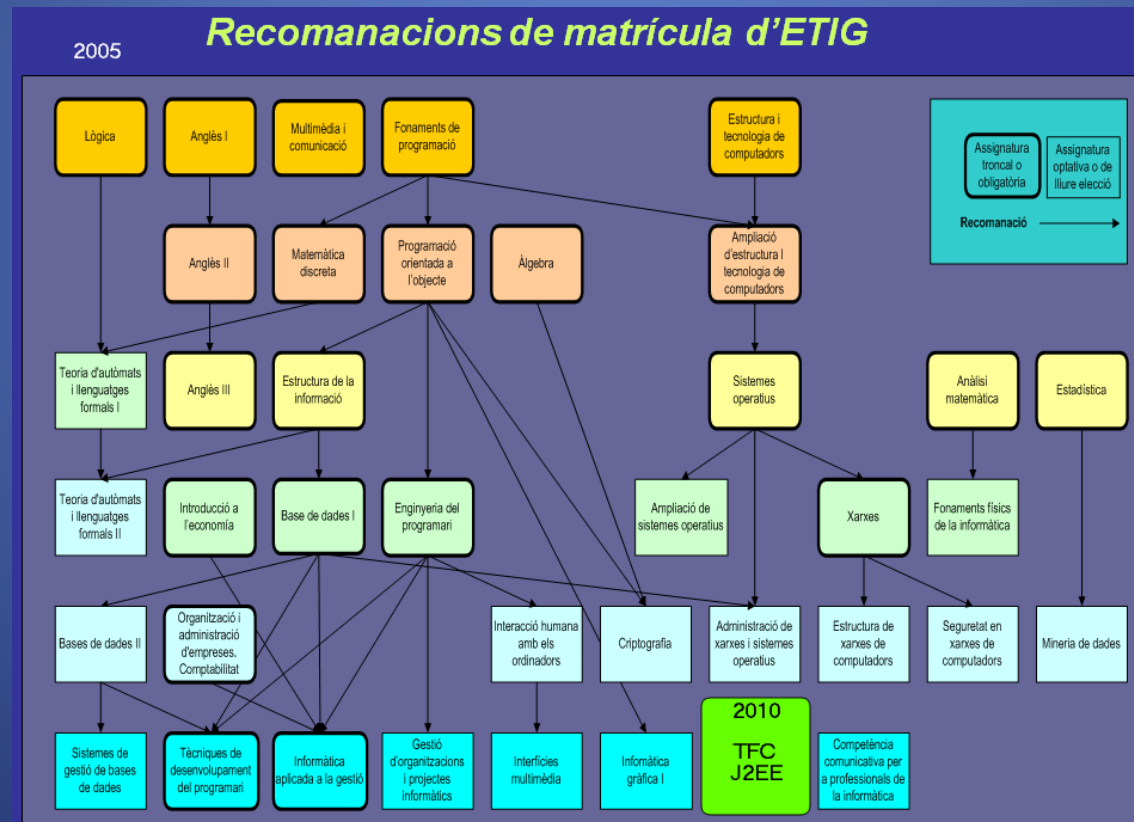
Alumne: Xavier Benet ETIG

Consultor: Jose Juan Rodriguez

Curs 2010-11/1

# 2005 - 2010

Després de 6 anys dedicat a anar avançant amb els estudis d'ETIG, arriba el moment de fer el Treball Final de Carrera i posat a fer-lo, amb què millor que en JAVA?



# Perquè JAVA to Enterprise Edition?

L' àrea escollida per realitzar el TFC ha estat J2EE perquè a part de ser una tecnologia que està competint amb molta força contra tecnologies de propietari, com per exemple .NET de Microsoft, ofereix una potència, flexibilitat i escalabilitat molt altes.

A banda, es tracta d'una tecnologia molt oberta i flexible, que fa que la corba d'aprenentatge hagi estat molt forta i més tard s'ha pogut constatar el risc d'haver escollit J2EE, quan m'he topat amb algunes dificultats a l'hora d'escollir els diferents *Frameworks* amb què treballar.

Però el resultat final és el que compta i pel sol fet d'haver après tantes coses sobre aquest entor, el camí recorregut fins aquí ha valgut la pena.

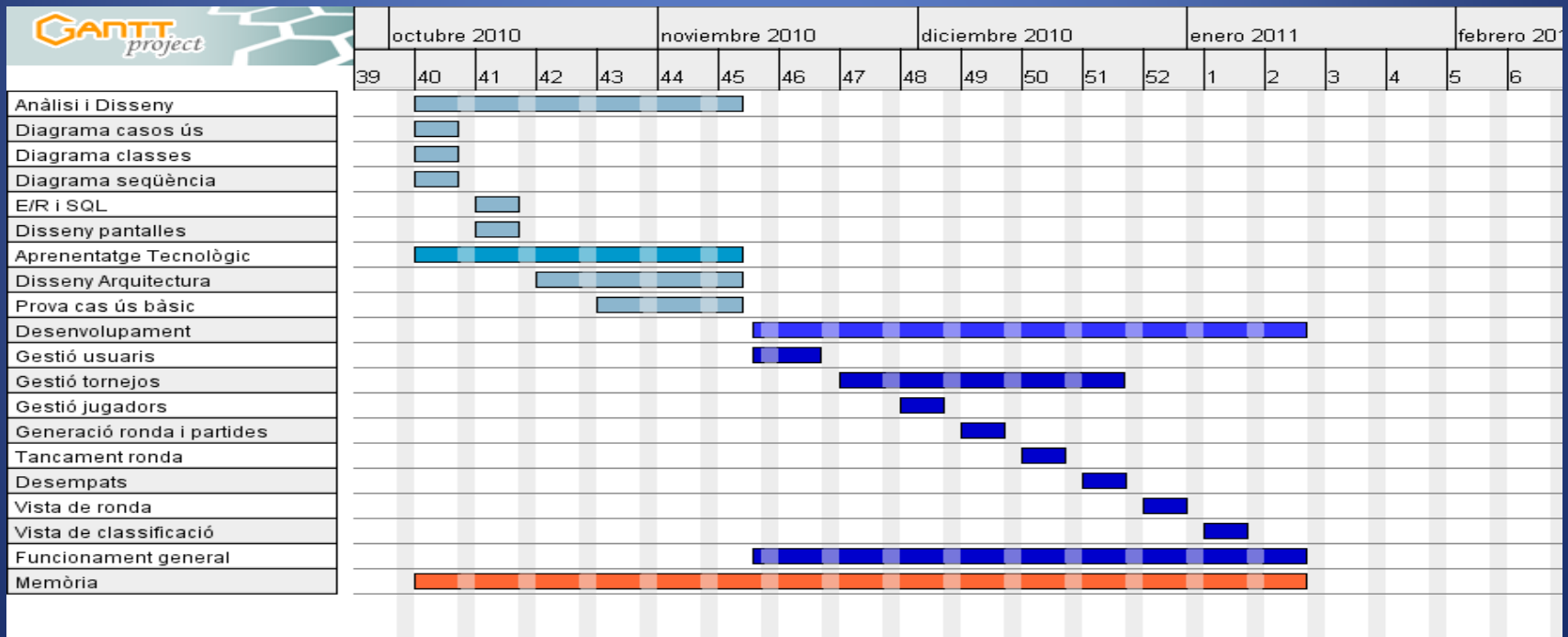
# El Tema del Treball

Vaig escollir un tema prou conegut i familiar per estalviar-me una fase de documentació massa feixuga.

Es tracta d'una aplicació web que servirà per dur a terme la gestió de tornejos d'escacs amb tres rols ben diferenciats. El rol més bàsica és el de qualsevol usuari d'Internet i aquest podrà consultar l'estat dels diferents tornejos, els aparellaments de les rondes i els resultats. Els usuaris gestors s'encarregaran de donar d'alta tornejos, assignar-hi jugadors i generar noves rondes amb les seves partides. Finalment, hi haurà un Administrador que s'encarregarà de donar d'alta als usuaris gestors (àrbitres dels tornejos) i als jugadors a dins del sistema, a més, també tindrà els privilegis per poder fer les tasques de qualsevol àrbitre.

# 1ª fase: la planificació

En aquesta fase es va fer un llistat de les tasques que caldria fer durant tot el procés i es va fer una primera aproximació del temps destinat a desenvolupar cada tasca.



# 2ª fase: anàlisi i disseny

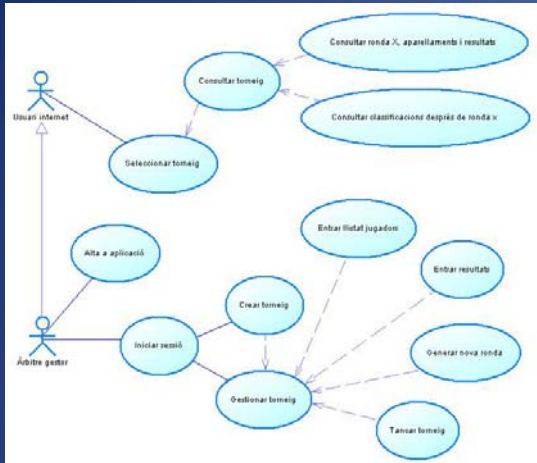


Diagrama de Casos d'ús

Diagrama de classes

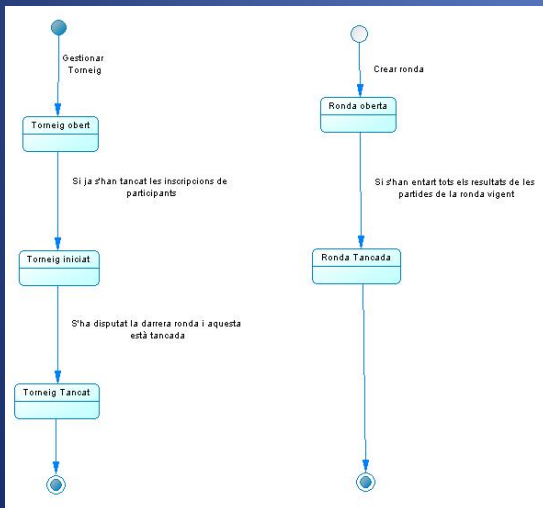
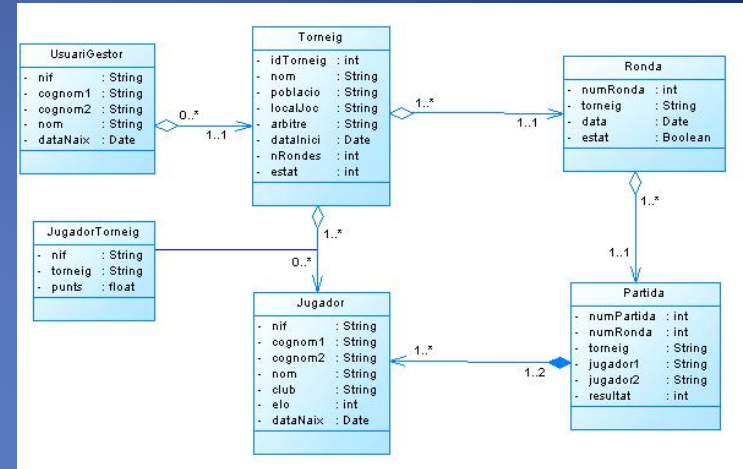


Diagrama d'Estats

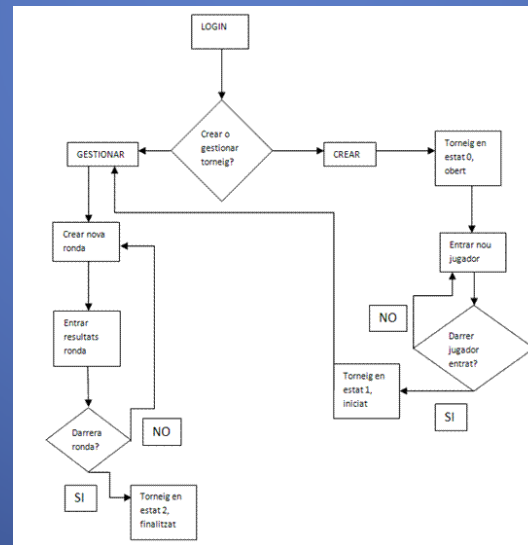


Diagrama Funcional

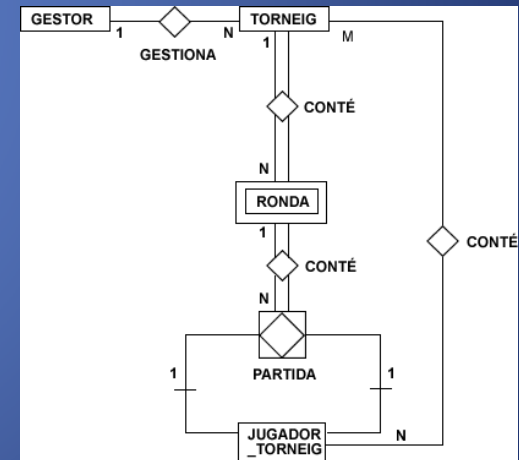
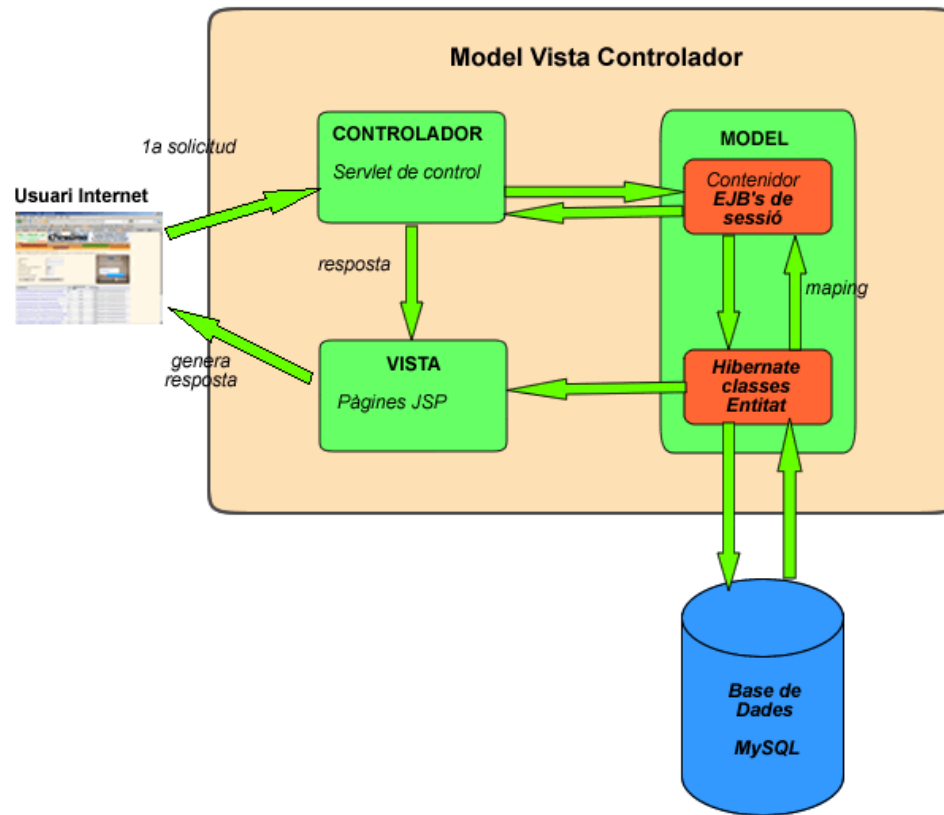


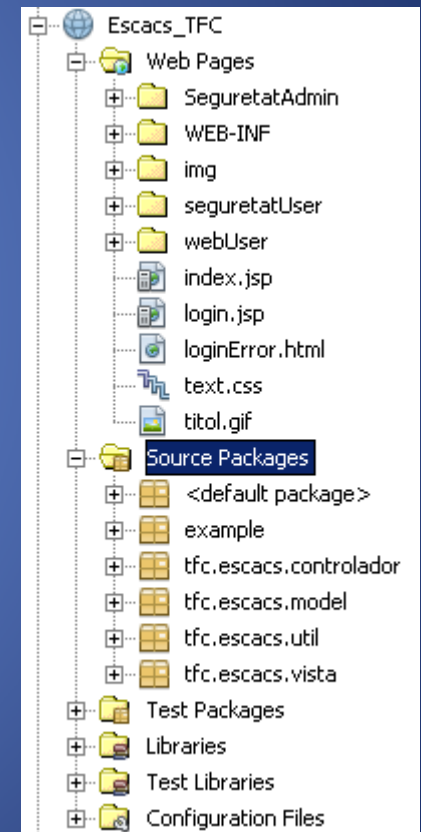
Diagrama ER

# El patró MVC (Model Vista Controlador)



# Caracteristiques MVC

- El Model Vista Controlador és un model molt utilitzat en l'actualitat i es caracteritza per la divisió entre tres capes, de l'arquitectura de l'entorn.
- En la capa Model hi tindrem les classes java
- A la capa de vista les accions que farem sobre les entitats i els fitxers amb els que l'usuari ha d'interaccionar: *.html*, *.jsp*...
- Finalment a la capa de Control hi tindrem el negoci entre les capes model i vista, per obtenir els resultats de les dades i passar-los a la capa de vista.
- Per tal d'aplicar aquest model es creen diferents paquets (packages), un per cada capa i a més en farà falta un altre anomenat *util* on hi configurarem la sessió de persistència per *hibernate*.
- A més d'aquestes capes, també haurem de tenir en compte altres factors de configuració que solen anar en fitxers *.xml*





# Frameworks i Tecnologies

- El llenguatge, J2EE
- El servidor web, GlassFish
- L'entorn de Programació, NetBeans
- El Sistema Gestor de Bases Dades, MySql
- La Vista, pàgines jsp i html
- L'enllaç entre classes i dades i navegador, Struts2 + Hibernate

# Codi Java d'una classe entitat

Una de les classes de java que trobarem a la capa Model és la classe JugadorTorneig, que serveix per emmagatzemar els jugadors que contindrà un determinat torneig. Aquí el codi:

```
package tfc.escacs.model;

import java.io.Serializable;
import java.sql.Date;

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Table;

// Classe entitat JugadorTorneig intermediària amb la Base de Dades
@Entity
@Table(name="jugador_torneig")
public class JugadorTorneig implements Serializable{

    private static final long serialVersionUID = -8767337896773261247L;

    private Long id;
    private int torneig;
    private int jugador;
    private float punts;

    //private Date created;

    @Id
    @GeneratedValue( strategy = GenerationType.AUTO)
    @Column(name="id")
    public Long getId() {
        return id;
    }

    @Column(name="jugador")
    public int getJugador() {
        return jugador;
    }

    @Column(name="torneig")
    public int getTorneig() {
        return torneig;
    }

    @Column(name="punts")
    public float getPunts() {
        return punts;
    }

    public void setId(Long id) {
        this.id = id;
    }

    public void setJugador(int jugador) {
        this.jugador = jugador;
    }

    public void setTorneig(int torneig) {
        this.torneig = torneig;
    }

    public void setPunts(float punts) {
        this.punts = punts;
    }
}
```

# Codi Java a la capa Vista

Pel que fa a la capa de vista, aquí tenim el que hauran de fer els mètodes sobre la classe PartidaJugador, que seran els que es lligaran amb Struts2

```
package tfc.escacs.vista;

import java.util.List;

import tfc.escacs.controlador.JugadorTorneigManager;
import tfc.escacs.model.JugadorTorneig;

import com.opensymphony.xwork2.ActionSupport;

//Classe amb accions per sobre Torneig
public class JugadorTorneigAction extends ActionSupport {

    private static final long serialVersionUID = 9149826260758390091L;
    private JugadorTorneig jugadorTorneig;
    private List<JugadorTorneig> jugadorTorneigList;
    private Long id;

    private JugadorTorneigManager linkController;

    public JugadorTorneigAction() {
        linkController = new JugadorTorneigManager();
    }

    public String add() {
        System.out.println(getJugadorTorneig());
        try {
            linkController.add(getJugadorTorneig());
        } catch (Exception e) {
        }

        return SUCCESS;
    }

    public String delete() {
        linkController.delete(getId());
        return SUCCESS;
    }

    public JugadorTorneig getJugadorTorneig() {
        return jugadorTorneig;
    }

    public void setJugadorTorneig(JugadorTorneig jugadorTorneig) {
        this.jugadorTorneig = jugadorTorneig;
    }

    public Long getId() {
        return id;
    }

    public void setId(Long id) {
        this.id = id;
    }
}
```

# Codi Java a la capa Controlador

A la capa Controladora i tenim el negoci entre les capes Vista i Model. Aquí el codi font per a la classe *JugadorTorneigManager.java*

```
package tfc.escacs.controlador;

import java.util.List;

import org.hibernate.HibernateException;
import org.hibernate.classic.Session;

import tfc.escacs.model.JugadorTorneig;
import tfc.escacs.util.HibernateUtil;

// Classe que gestionarà les accions per l'entitat JugadorTorneig
public class JugadorTorneigManager extends HibernateUtil {

    public JugadorTorneig add(JugadorTorneig jugadorTorneig) {
        Session session = HibernateUtil.getSessionFactory().getCurrentSession();
        session.beginTransaction();
        session.save(jugadorTorneig);
        session.getTransaction().commit();
        return jugadorTorneig;
    }

    public JugadorTorneig delete(Long id) {
        Session session = HibernateUtil.getSessionFactory().getCurrentSession();
        session.beginTransaction();
        JugadorTorneig jugadorTorneig = (JugadorTorneig) session.load(JugadorTorneig.class, id);
        if(null != jugadorTorneig) {
            session.delete(jugadorTorneig);
        }
        session.getTransaction().commit();
        return jugadorTorneig;
    }
}
```

```
    public List<JugadorTorneig> list() {

        Session session =
            HibernateUtil.getSessionFactory().getCurrentSession();

        session.beginTransaction();

        List<JugadorTorneig>
            jugadorsTorneig = null;

        try {

            jugadorsTorneig =
                (List<JugadorTorneig>)session.createQuery("from
                JugadorTorneig").list();

        } catch
            (HibernateException e) {

                e.printStackTrace();

                session.getTransaction().rollback();
            }

        session.getTransaction().commit();

        return
            jugadorsTorneig;
    }
}
```

# Pantalla inicial


Entrar com a visitant [aquí!](#)

Entrar com a àrbitre [aquí!](#)

Entrar com a Administrador [aquí!](#)

En aquesta pantalla inicial, haurem d'escollir amb quin rol volem entrar a l'aplicació. Als usuaris d'internet se'ls pot facilitar la URL directa a la carpeta per aquest rol.

# Pantalla d'autenticació



**Autenticació**

**Username:**

**Password:**

En el cas que el nom d'usuari i/o contrasenya siguin incorrectes, s'enviarà a l'usuari a una URL amb el següent missatge:

**Error d'autenticació!!, si us plau comprova que l'usuari i contrasenya siguin correctes...**

# Pantalla inici usuari Internet

## GESTIÓ DE TORNEJOS D'ESCACS



### Opcions

Veure tornejos



**Entrant a l'opció de "veure tornejos" podreu consultar els tornejos en joc, i d'aquests, les partides i rondes disputades.**

# Pantalla inici usuari Gestor

## GESTIÓ DE TORNEJOS D'ESCACS



### Opcions

- Gestionar tornejos
- Gestionar jugadors per torneig
- Gestionar rondes
- Gestionar partides



**Un usuari gestor podrà llistar, afegir o eliminar registres de tornejos, jugadors a tornejos, rondes i partides.**



# Pantalla inici Administrador

## GESTIÓ DE TORNEJOS D'ESCACS



### Opcions

- Gestionar àrbitres
- Gestionar tornejos
- Gestionar jugadors del sistema
- Gestionar jugadors per torneig
- Gestionar rondes
- Gestionar partides



**Amb aquesta aplicació web podreu gestionar tornejos d'escacs. Es podran donar d'alta els àrbitres. Seguidament es podran crear un torneig, que haurà de tenir un àrbitre associat. El següent pas serà entrar els jugadors al sistema i associar-los a tornejos existents. Es podran gestionar les rondes una a una i dins d'aquestes les partides jugades.**

# Opcions Usuari Internet

## GESTIÓ DE TORNEJOS D'ESCACS

[Inici](#) | [Tornejos](#) | [Jugadors Torneig](#) | [Rondes](#) | [Partides](#)

[Listar Tornejos](#)

Torneig	Àrbitre	Població	Adreça	Data inici	Data fi	Rondes	Estat
Open Calldetenes	1	Calldetenes	Plaça Major	15/07/10	15/10/10	8	0
Open de Gurb	1	Gurb	Plaça Menor	11/11/10	11/12/11	9	0
Open de Vic	2	Vic	Casino	15/11/10	23/11/10	8	0
Open de Taradell	4	Taradell	H.Rosa	15/01/11	23/01/11	8	0
Open Sants	1	Andorra	ggg	15/07/73	20/04/11	10	0
www	3	Llagosta	La vella	20/02/10	20/07/73	8	0
u	1	u	u	20/02/10	20/07/73	8	0

Aquest tipus d'usuari no podrà introduir ni eliminar cap dada del sistema, únicament podrà consultar les que hi hagi, segons les opcions que tingui visibles.

# Opcions Gestor

## GESTIÓ DE TORNEJOS D'ESCACS

Inici | Tornejos | Jugadors Torneig | Rondes | Partides |

Llistar Tornejos

Torneig	Població	Adreça	Àrbitre	Data inici	Data fi	Rondes	Estat	Eborra
Open Calldetenes	1	Calldetenes	Plaça Major	15/07/10	15/10/10	8	0	<a href="#">Eborra</a>
Open de Gurb	1	Gurb	Plaça Menor	11/11/10	11/12/11	9	0	<a href="#">Eborra</a>
Open de Vic	2	Vic	Casino	15/11/10	23/11/10	8	0	<a href="#">Eborra</a>
Open de Taradell	4	Taradell	H.Rosa	15/01/11	23/01/11	8	0	<a href="#">Eborra</a>
www	17	654	ggg	15/07/73	20/04/11	10	0	<a href="#">Eborra</a>
POPOPOPO	1	ggg	yyy	20/02/11	20/04/11	10	0	<a href="#">Eborra</a>

Torneig:   
Població:   
Adreça:   
Àrbitre:   
Data inici:   
Data fi:   
Rondes:

Afegir torneig

Aquest tipus d'usuari podrà introduir i eliminar les dades del sistema referents a la gestió de tornejos i partides, en cap cas podrà entrar nous àrbitres o jugadors, però sí que podrà assignar jugadors a un torneig.

# Opcions Administrador

**GESTIÓ DE TORNEJOS D'ESCACS**

[Inici](#) | [Àrbitres](#) | [Tornejos](#) | [Jugadors](#) | [Jugadors Torneig](#) | [Rondes](#) | [Partides](#)

Torneig	Població	Adreça	Àrbitre	Data inici	Data fi	Rondes	Estat	Eborra
Open Calldetenes	Calldetenes	Plaça Major	1	15/07/10	15/10/10	8	0	<a href="#">Eborra</a>
Open de Gurb	Gurb	Plaça Menor	1	11/11/10	11/12/11	9	0	<a href="#">Eborra</a>
Open de Vic	Vic	Casino	2	15/11/10	23/11/10	8	0	<a href="#">Eborra</a>
Open de Taradell	Taradell	H.Rosa	4	15/01/11	23/01/11	8	0	<a href="#">Eborra</a>
www	654	ggg	17	15/07/73	20/04/11	10	0	<a href="#">Eborra</a>
POPOPOPO	ggg	yyy	1	20/02/11	20/04/11	10	0	<a href="#">Eborra</a>

Torneig:

Població:

Adreça:

Àrbitre:

Data inici:

Data fi:

Rondes:

Aquest tipus d'usuari podrà introduir i eliminar tot tipus de dades del sistema, i serà qui decidirà quins àrbitres i jugadors hi ha d'haver a l'aplicació.