

# Treball de fi de Grau

**Memòria**

CharacterBuilder

**Tutor:** Albert Grau Perisé

**Autor:** Joan Bauzà Soler

## Índex de continguts

Avís legal.....	6
Introducció.....	7
Objectius.....	7
Anàlisi de l'aplicació.....	7
Descripció dels actors.....	7
Casos d'ús.....	8
Administració d'usuaris.....	8
CU1_Login.....	8
CU2_Logout.....	9
CU3_Crear un jugador.....	9
CU4_Esborrar un jugador.....	9
CU5_Modificar un jugador.....	10
CU6_Actualitzar el propi compte.....	10
Manteniments.....	11
CU7_Crear un quirk.....	11
CU8_Esborrar un Quirk.....	12
CU9_Modificar un quirk.....	12
CU10_Crear una habilitat.....	13
CU11_Esborrar una habilitat.....	13
CU12_Modificar una habilitat.....	14
CU13_Crear una categoria d'habilitats.....	14
CU14_Esborrar una categoria d'habilitats.....	15
CU15_Modificar una categoria d'habilitats.....	15
CU16_Crear una categoria de quirks.....	16
CU17_Esborrar una categoria de quirks.....	16
CU18_Modificar una categoria de quirks.....	16
Administració de personatges.....	18
CU18_Crear un personatge.....	18
CU19_Esborrar un personatge.....	19
CU20_Modificar un personatge.....	19
CU21_Canviar el valor d'una característica.....	20
CU22_Seleccionar un quirk.....	20
CU23_Deseleccionar un Quirk.....	20
CU24_Canviar el punts d'experiència assignats a una habilitat a la infantesa.....	21
CU25_Canviar el punts d'experiència assignats a una habilitat a l'edat adulta.....	21
CU26_Validar un personatge.....	22
CU27_Modificar els permisos d'un personatge.....	22
CU28_Transferir la propietat d'un personatge.....	23
CU29_Accedir al log.....	24
CU30_Visualitzar un personatge.....	24
Disseny.....	25
Diagrama de classes.....	25
Usuaris amb múltiples rols.....	25
Multipropietat dels personatges.....	25
Punts d'experiència a les habilitats.....	26
Diferències entre Quirks i virtuts/defectes.....	26
La interfície "Selectable".....	26
Base de dades.....	26

Tecnologies.....	28
Implementació.....	29
Calendari final del projecte.....	29
Anàlisi del procés d'implementació i de les tecnologies emprades.....	30
HTML5+CSS3.....	30
Javascript, JQuery i JQuery UI.....	31
Spring 4.1.2.....	32
Spring Security 3.2.5.....	34
EJB3.....	36
JPA 2 sobre Hibernate 4.0.1.....	36
PostgreSql 9.3.....	37
Jboss 7.1.1 as Final.....	37
Eclipse Luna (4.4.0).....	37
Kubuntu 14.04.....	37
MagicDraw.....	38
Control de versions.....	38
Manual d'usuari.....	39
Concepte: propietat d'un personatge.....	39
Autenticar-se a l'aplicació.....	39
Estructura bàsica de la interfície.....	40
Capçalera.....	40
Barra de navegació.....	40
Cos.....	41
Administració de personatges.....	41
Creació de personatges.....	43
Validació de personatges.....	47
Visualització de personatges.....	48
Edició de personatges.....	48
Esborrat de personatges.....	49
Edició de permisos d'accés.....	49
Manteniments.....	50
Creació d'usuaris.....	52
Edició d'usuaris.....	52
Esborrat d'usuaris.....	52
Creació de categories de quirks.....	54
Edició de categories de quirks.....	54
Esborrat de categories de quirks.....	54
Creació de quirks.....	56
Edició de quirks.....	58
Esborrat de quirks.....	58
Creació de categories d'habilitats.....	60
Edició de categories d'habilitats.....	62
Esborrat de categories d'habilitats.....	62
Creació d'habilitats.....	64
Edició d'habilitats.....	66
Esborrat d'habilitats.....	66
Pantalla de log.....	67
Millores.....	68
Bibliografia.....	69
Apèndix A: Resum del sistema de creació de personatges d'Ars Magica.....	71

Característiques físiques.....	71
Habilitats.....	71
Virtuts i defectes.....	72

## Índex d'il·lustracions

Il·lustració 1: Administració d'usuaris.....	8
Il·lustració 2: Manteniments.....	11
Il·lustració 3: Administració de personatges.....	18
Il·lustració 4: Diagrama de classes.....	25
Il·lustració 5: Esquema ER.....	27
Il·lustració 6: Esquema de tecnologies.....	28
Il·lustració 7: Pantalla de login.....	39
Il·lustració 8: Error d'autenticació.....	39
Il·lustració 9: Pantalla principal.....	40
Il·lustració 10: Capçalera.....	40
Il·lustració 11: Barra de navegació.....	40
Il·lustració 12: Cos principal de la pantalla.....	41
Il·lustració 13: Pantalla d'administració de personatges.....	42
Il·lustració 14: Línia de personatge.....	42
Il·lustració 15: Dades generals.....	43
Il·lustració 16: Pantalla de quirks.....	44
Il·lustració 17: Pantalla d'habilitats.....	45
Il·lustració 18: Assignant punts d'experiència.....	46
Il·lustració 19: Error de validació.....	47
Il·lustració 20: Validació correcta.....	48
Il·lustració 21: Visualització de personatges.....	48
Il·lustració 22: Pantalla d'edició de permisos de personatges.....	49
Il·lustració 23: Pantalla d'usuaris.....	51
Il·lustració 24: Creació d'usuaris.....	52
Il·lustració 25: Pantalla de categories de quirks.....	53
Il·lustració 26: Pantalla de creació de categories de quirks.....	54
Il·lustració 27: Pantalla de quirks.....	55
Il·lustració 28: Pantalla de creació de quirks.....	56
Il·lustració 29: Dades generals del quirk.....	56
Il·lustració 30: Pantalla de categories d'habilitats.....	59
Il·lustració 31: Creació de categories.....	60
Il·lustració 32: Dades generals d'una categoria d'habilitats.....	60
Il·lustració 33: Pantalla de categories d'habilitats.....	63
Il·lustració 34: Creació d'habilitats.....	64
Il·lustració 35: Dades generals d'una habilitat.....	64
Il·lustració 36: Pantalla de log.....	67

## Índex de taules

Taula 1: Requisits d'un quirk.....	58
Taula 2: Requisits d'una categoria d'habilitats.....	61
Taula 3: Requisits d'una habilitat.....	66
Taula 4: Valor de les característiques.....	71
Taula 5: Cost en xps dels nivells d'habilitat.....	72
Taula 6: Equivalència en punts dels quirks.....	72

# Avís legal

Ars Magica i el sistema de joc en el que s'inspira aquest projecte són propietat de Atlas Games<sup>1</sup>. L'ús que se'n fa en aquest projecte no representa cap intent de reclamar-ne la propietat ni posar en dubte els drets de Atlas Games sobre les respectives marques.

L'ús que se'n fa en aquest projecte és estrictament per motius educatius i sense cap ànim de lucre.

---

<sup>1</sup> <http://blog.atlas-games.com/>

# Introducció

La tria d'aquest treball ve donada per la meva afició als jocs de rol així com la intenció d'aprofundir en l'ús de les tecnologies emprades.

## Descripció

El treball consta d'una aplicació Web on tots els usuaris, una vegada autenticats, poden fer les següents tasques:

- Crear, editar i esborrar personatges.
- Donar permisos a altres usuaris per a que puguin veure i editar els personatges creats per ells.

Addicionalment hi ha una part exclusiva de l'administrador on aquest pot:

- Crear, editar i esborrar usuaris.
- Crear, editar i esborrar qualsevol personatge.
- Crear, editar i esborrar virtuts i defectes.
- Crear, editar i esborrar habilitats.
- Crear, editar i esborrar categories d'habilitats.
- Crear, editar i esborrar categories de virtuts i defectes.

## Objectius

Amb el desenvolupament d'aquest treball cerco assolir els següents objectius:

- Desenvolupar una aplicació que pugui servir de base sobre la que construir un entorn d'administració més complet, no només de personatges sinó també de partides i campanyes.
- Aprofundir en el coneixement de Spring i Hibernate.
- Reciclar coneixements de JQuery, HTML5 i CSS3.
- Provar alguns dels plugins de JQuery-UI i la seva integració.

## Anàlisi de l'aplicació

### Descripció dels actors

Dins l'aplicació hi ha dos rols concrets, cadascun amb els seus propis permisos:

**Jugador("ROLE\_PLAYER")** : pot crear, editar i esborrar els personatges creats per ell o aquells personatges que li hagin estat cedits.

**Administrador("ROLE\_ADMIN")**: exerceix d'administrador del sistema, a més de poder

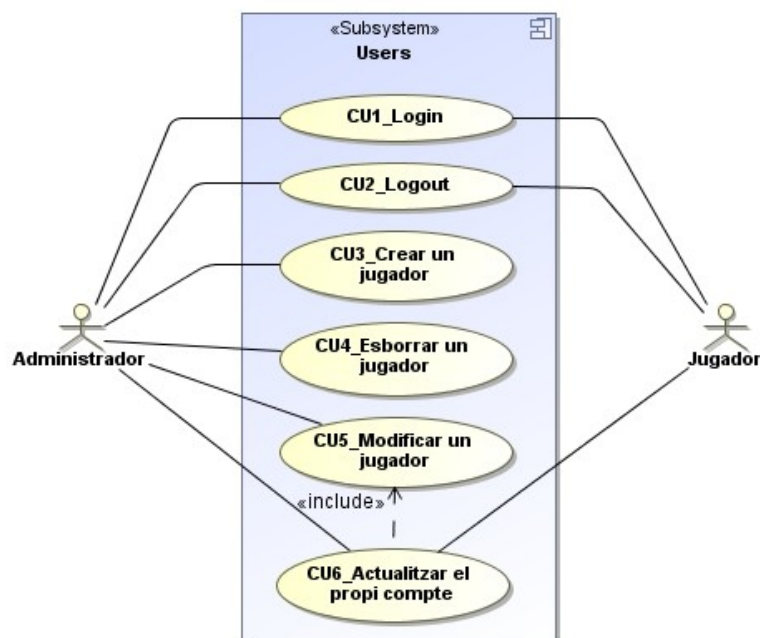
administrar els seus personatges.

## Casos d'ús

A partir de l'anàlisi dels requeriments s'han desenvolupat els següents casos d'ús, provant de definir totes les circumstàncies que poden donar-se durant l'ús del projecte. Aquest casos poden agrupar-se en tres subsistemes en funció del seu àmbit: l'administració de fitxes, l'administració d'usuaris i els manteniments.

### Administració d'usuaris

Aquest subsistema engloba tots els casos d'us referents a les tasques associades als usuaris, ja siguin jugadors o l'administrador.



Il·lustració 1: Administració d'usuaris

#### CU1\_Login

**Actor principal:** usuari

**Precondició:** l'usuari no està loguejat

**Postcondició:** l'usuari se logueja correctament a l'aplicació

**Escenari principal d'èxit:**

1. L'usuari accedeix a la pantalla de login
2. L'usuari insereix el seu identificador i password.

3. El sistema verifica les dades introduïdes.
4. Les dades són correctes, i l'aplicació redirecciona l'usuari a la pantalla inicial.
5. El sistema logueja l'usuari dins el seu perfil.

**Extensions:**

3a. Les dades introduïdes no se corresponen amb cap usuari, l'aplicació dona un missatge d'error i continua a la pantalla de logueig.

## CU2\_Logout

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat

**Postcondició:** l'usuari fa logout de l'aplicació

**Escenari principal d'èxit:**

1. L'usuari prem sobre el botó de logout.
2. L'aplicació esborra la sessió i tota la informació no guardada i redirecciona l'usuari a la pantalla de login.

## CU3\_Crear un jugador

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** se crea un nou usuari de perfil Jugador

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració de jugadors.
2. L'administrador accedeix al formulari de creació de jugadors.
3. L'administrador emplena el formulari i el guarda.
4. L'aplicació guarda les dades en base de dades i mostra un missatge d'èxit
5. L'aplicació retorna l'administrador a la pantalla d'administració de jugadors.

**Extensions:**

3a. L'administrador cancel·la l'edició, l'aplicació descarta tota la informació i torna a la pantalla d'administració de jugadors.

4a. El jugador ja existeix a la base de dades, l'aplicació mostra un missatge d'error i continua a la pantalla d'edició.

## CU4\_Esborrar un jugador

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** s'esborren l'usuari de perfil jugador seleccionat.

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració de jugadors.
2. L'administrador tria el jugador que vol esborrar.
3. L'aplicació demana confirmació de la operació.
4. Si l'administrador accepta, l'aplicació esborra el jugador del sistema.

**Extensions:**

4a. L'administrador no accepta, l'aplicació no fa res.

## CU5\_Modificar un jugador

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** se modifiquen les dades d'un usuari de perfil jugador.

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració de jugadors.
2. L'administrador tria el jugador que desitja editar.
3. L'aplicació mostra la pantalla d'edició d'usuaris amb les dades del jugador seleccionat.
4. L'administrador modifica les dades que desitja i les guarda.
5. L'aplicació demana confirmació de la operació.
6. Si l'administrador accepta, l'aplicació guarda les dades actualitzades.

**Extensions:**

4a. L'administrador cancel·la l'edició: l'aplicació descarta tota la informació i torna a la pantalla d'administració de jugadors.

5a. L'administrador no accepta, l'aplicació no fa res i roman a la pantalla d'edició.

## CU6\_Actualitzar el propi compte

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat

**Postcondició:** se modifiquen les dades de l'usuari loguejat

**Escenari principal d'èxit:**

1. L'usuari prem sobre el seu nom d'usuari a la capçalera de l'aplicació.
2. L'aplicació mostra la pantalla d'edició d'usuaris amb les dades de l'usuari.
3. L'usuari modifica les dades que desitja i les guarda.
4. L'aplicació demana confirmació de la operació.
5. Si l'usuari accepta, l'aplicació guarda les dades actualitzades.

**Extensions:**

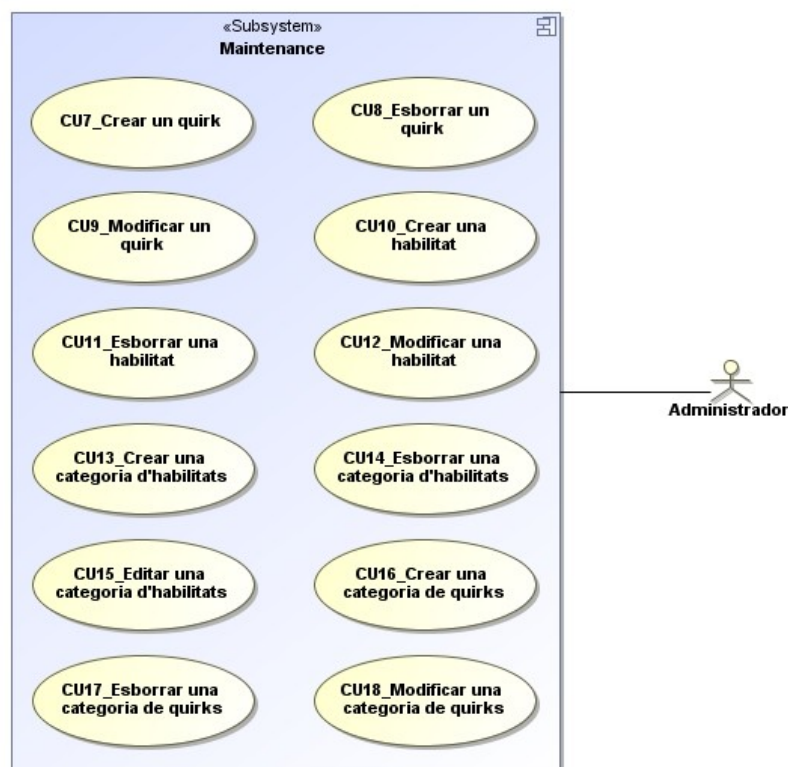
3a. L'usuari cancel·la l'edició: l'aplicació descarta tota la informació i redirigeix l'usuari a la pantalla principal.

4a. L'usuari no accepta, l'aplicació no fa res i roman a la pantalla d'edició.

## Manteniments

Aquesta secció inclou tots els casos relacionats amb el manteniment de les dades de l'aplicació que se fan servir en la creació de personatges.

Només l'administrador té accés a aquests casos d'ús.



*Il·lustració 2: Manteniments*

### CU7\_Crear un quirk

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** se crea el nou quirk

**Escenari principal d'èxit:**

1. L'administrador selecciona accedeix a la pantalla d'administració de quirks.
2. L'administrador accedeix al formulari de creació de quirks.

3. L'administrador introdueix les dades, incloent la categoria, el tipus i la divisió(Virtut / Defecte).
4. L'aplicació guarda el nou quirk i mostra un missatge d'èxit
5. L'aplicació redirecciona l'usuari a la pantalla de quirks.

**Extensions:**

- 3a. L'administrador cancel·la el procés:l'aplicació descarta la informació i torna a la pantalla d'administració de quirks.
- 4a. Ja existeix un nou quirk amb el mateix nom, tipus(Major/Minor) i divisió(Virtut/Defecte): l'aplicació retorna un missatge d'error i continua al formulari.

## CU8\_Esborrar un Quirk

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** s'esborra el quirk i se tornen a validar tots els personatges que el tenien.

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració de quirks.
2. L'administrador selecciona el quirk a esborrar.
3. L'aplicació demana confirmació de la operació.
4. Si l'administrador accepta, l'aplicació esborra el quirk i torna a validar tots els personatges que el tenien.
5. Se guarda al log una descripció del canvi incloent la data.

**Extensions:**

- 4a. L'administrador no accepta, l'aplicació no fa res.
- 4b. El quirk és requisit d'algun element: el requisit desapareix amb el quirk.

## CU9\_Modificar un quirk

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** se modifica el quirk i se tornen a validar tots els personatges que la tenen.

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració de quirks.
2. L'administrador selecciona el quirk a modificar.
3. L'aplicació mostra el formulari d'edició de quirks amb les dades del quirk seleccionat.
4. L'administrador modifica les dades que desitja i les guarda.
5. L'aplicació demana confirmació de la operació.
6. Si l'usuari accepta, l'aplicació actualitza el quirk i valida tots els personatges on està present.
7. Se guarda al log una descripció del canvi incloent la data.

**Extensions:**

4a. L'administrador cancel·la el procés:l'aplicació descarta la informació i torna a la pantalla d'administració de quirks.

6a. L'administrador no accepta, l'aplicació no fa res.

## CU10\_Crear una habilitat

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** se crea la nova habilitat

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració d'habilitats.
2. L'administrador accedeix al formulari de creació d'habilitats.
3. L'administrador introdueix les dades, incloent sota quines condicions és l'habilitat seleccionable i les guarda.
4. L'aplicació afegeix als requisits de l'habilitat els de la categoria a la que pertany(si els tingués) .
5. L'aplicació mostra un missatge d'èxit i guarda la nova habilitat.

**Extensions:**

3a. L'administrador cancel·la el procés:l'aplicació descarta la informació i torna a la pantalla d'administració d'habilitats.

5a. Ja existeix una habilitat amb el mateix nom: l'aplicació retorna un missatge d'error i continua al formulari.

## CU11\_Esborrar una habilitat

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** s'esborra l'habilitat.

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració d'habilitats.
2. L'administrador selecciona l'habilitat a esborrar.
3. L'aplicació demana confirmació de la operació.
4. Si l'administrador accepta, l'aplicació esborra l'habilitat.
5. S'esborra l'habilitat de totes les fixes de personatges que la tinguessin.
6. Se guarda al log una descripció del canvi incloent la data.

**Extensions:**

4a. L'administrador no accepta, l'aplicació no fa res.

## CU12\_Modificar una habilitat

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** se modifica l'habilitat.

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració d'habilitats.
2. L'administrador selecciona l'habilitat a modificar.
3. L'aplicació mostra el formulari de creació d'habilitats amb les dades de l'habilitat seleccionada.
4. L'administrador modifica les dades que desitja i les guarda.
5. L'aplicació demana confirmació de la operació.
6. Si l'administrador accepta, l'aplicació actualitza l'habilitat.
7. Se guarda al log una descripció del canvi incloent la data.

**Extensions:**

4a. L'administrador cancel·la el procés: l'aplicació descarta la informació i torna a la pantalla d'administració d'habilitats.

6a. L'administrador no accepta, l'aplicació no fa res.

6b. Si les modificacions impliquen canvis als requisits se revisa tots els personatges que tenen aquesta habilitat: en cas que hagi deixat de ser accessible, se l'esborra del personatge i se marca el personatge afectats com "Pending revision".

6c. Si les modificacions inclouen canvi de categoria, se perden els requisits de la categoria anterior(si els tingués) i se guanyen els de la nova(si els tingués). A continuació se revisen tots els personatges que tenen aquesta habilitat: en cas que hagi deixat de ser accessible, se l'esborra del personatge i se marca el personatge afectats com "Pending revision".

6d. Ja existeix una habilitat amb el mateix nom: l'aplicació retorna un missatge d'error i continua al formulari.

## CU13\_Crear una categoria d'habilitats

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** se crea la nova categoria

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració de categories.
2. L'administrador accedeix al formulari de creació.
3. L'administrador introdueix les dades, incloent sota quines condicions és la categoria accessible.
4. L'aplicació mostra un missatge d'èxit i guarda la nova categoria d'habilitats.

**Extensions:**

- 3a. L'administrador cancel·la el procés :l'aplicació descarta la informació i torna a la pantalla d'administració de categories.
- 4b. Ja existeix una categoria amb el mateix nom: l'aplicació retorna un missatge d'error i continua al formulari.

## CU14\_Esborrar una categoria d'habilitats

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** s'esborra la categoria.

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració de categories.
2. L'administrador selecciona la categoria d'habilitats a esborrar.
3. L'aplicació demana confirmació de la operació.
4. Si l'administrador accepta, l'aplicació esborra la categoria.
5. Se guarda al log una descripció del canvi incloent la data.

**Extensions:**

- 4a. L'administrador no accepta, l'aplicació no fa res.
- 4b. La categoria encara té habilitats associades: l'aplicació torna un missatge d'error i no fa res més.

## CU15\_Modificar una categoria d'habilitats

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** se modifica la categoria.

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració de categories.
2. L'administrador selecciona la categoria d'habilitats a modificar.
3. L'aplicació mostra el formulari d'edició amb les dades de la categoria seleccionada.
4. L'administrador modifica les dades que desitja i les guarda.
5. L'aplicació demana confirmació de la operació.
6. Si l'administrador accepta, l'aplicació actualitza la categoria.
7. Se guarda al log una descripció del canvi incloent la data.

**Extensions:**

- 4a. L'administrador cancel·la el procés:l'aplicació descarta la informació i torna a la pantalla d'administració.
- 6a. L'usuari no accepta, l'aplicació no fa res.
- 6b. Si les modificacions impliquen canvis als requisits aquests s'estenen a les habilitats de la categoria(si n'hi ha). Finalment, se revisen tots els personatges amb alguna d'aquestes habilitats i, si per algun d'ells, aquesta ha deixat de ser accessible per ell, s'esborra l'habilitat de la fitxa i se marca

el personatge com "Pending Revision".

6c. Ja existeix una categoria amb el mateix nom: l'aplicació retorna un missatge d'error i continua al formulari.

## CU16\_Crear una categoria de quirks

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** se crea la nova categoria

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració de categories.
2. L'administrador accedeix al formulari de creació.
3. L'administrador emplena el formulari.
4. L'aplicació mostra un missatge d'èxit i guarda la nova categoria de quirks.

**Extensions:**

3a. L'administrador cancel·la el procés: l'aplicació descarta la informació i torna a la pantalla d'administració de categories.

4b. Ja existeix una categoria amb el mateix nom: l'aplicació retorna un missatge d'error i continua al formulari.

## CU17\_Esborrar una categoria de quirks

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** s'esborra la categoria.

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració de categories.
2. L'administrador selecciona la categoria de quirks a esborrar.
3. L'aplicació demana confirmació de la operació.
4. Si l'administrador accepta, l'aplicació esborra la categoria.
5. Se guarda al log una descripció del canvi incloent la data.

**Extensions:**

4a. L'administrador no accepta, l'aplicació no fa res.

4b. La categoria encara té quirks associats: l'aplicació torna un missatge d'error i no fa res més.

## CU18\_Modificar una categoria de quirks

**Actor principal:** administrador

**Precondició:** l'administrador està loguejat

**Postcondició:** se modifica la categoria.

**Escenari principal d'èxit:**

1. L'administrador accedeix a la pantalla d'administració de categories.
2. L'administrador selecciona la categoria de quirks a modificar.
3. L'aplicació mostra el formulari d'edició amb les dades de la categoria seleccionada.
4. L'administrador modifica les dades que desitja i les guarda.
5. L'aplicació demana confirmació de la operació.
6. Si l'administrador accepta, l'aplicació actualitza la categoria.
7. Se guarda al log una descripció del canvi incloent la data.

**Extensions:**

4a. L'administrador cancel·la el procés: l'aplicació descarta la informació i torna a la pantalla d'administració.

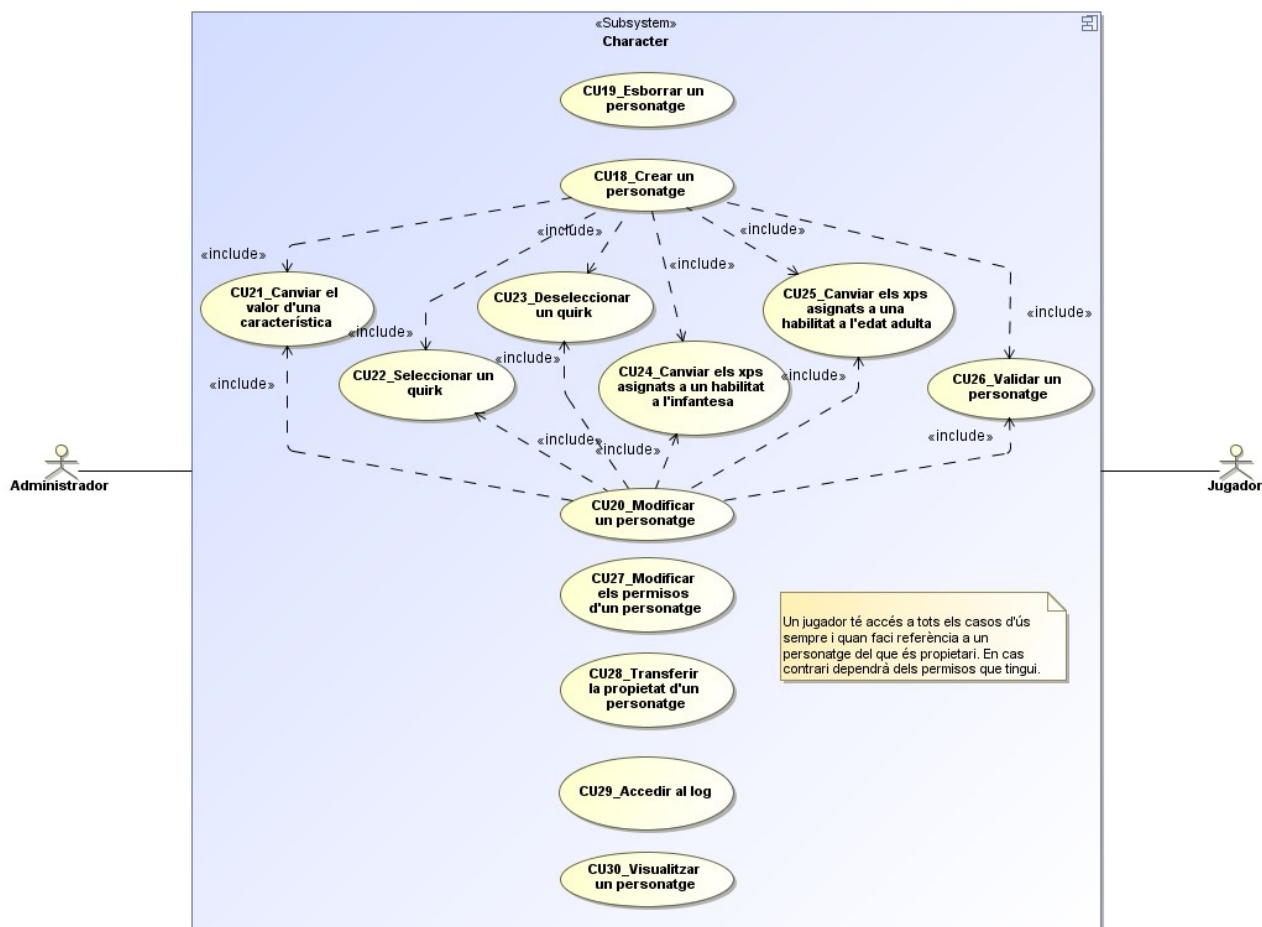
6a. L'usuari no accepta, l'aplicació no fa res.

Administració de personatges

Aquest subsistema és comú tant a jugadors com l'administrador i engloba els casos d'us referents a la creació i administració de personatges.

Els casos 18 i 20 (creació i modificació de personatges) inclouen sis casos addicionals. Aquests casos representen operacions prou complexes com per a demanar de ser definits amb detall i separatament.

# Administració de personatges



Il·lustració 3: Administració de personatges

I els seus casos d'ús són:

## CU18\_Crear un personatge

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat

**Postcondició:** se crea el personatge

**Escenari principal d'èxit:**

1. L'usuari accedeix a la pantalla d'administració de personatges.
2. L'usuari accedeix al l formulari de creació de personatges.
3. L'usuari introdueix les dades, tria les virtuts/defectes i posa els punts a les habilitats. En acabar, ho guarda.
4. L'aplicació valida el personatge i, en cas que no s'ajusti a les regles del sistema de joc, el guarda marcant-lo amb "Pending revision"

**Extensions:**

- 3a. L'usuari cancel·la el procés: l'aplicació descarta la informació i torna a la pantalla d'administració d'habilitats.
- 4b. Ja existeix un personatge amb el mateix nom: l'aplicació retorna un missatge d'error i continua al formulari.

## CU19\_Esborrar un personatge

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat

**Postcondició:** s'esborra el personatge

**Escenari principal d'èxit:**

1. L'usuari accedeix a la pantalla d'administració de personatges.
2. L'usuari selecciona esborrar el personatge.
3. L'aplicació demana confirmació de la operació.
4. Si l'usuari accepta, l'aplicació esborra el personatge.

**Extensions:**

- 2a. L'usuari no té cap permís sobre el personatge: aquest no apareix a la pantalla d'administració de personatges.
- 2b. L'usuari no és propietari del personatge: no és possible ordenar que s'esborri el personatge.
- 4a. L'usuari no accepta, l'aplicació no fa res.

## CU20\_Modificar un personatge

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat

**Postcondició:** se modifica el personatge amb les noves dades.

**Escenari principal d'èxit:**

1. L'usuari accedeix a la pantalla d'administració de personatges.
2. L'usuari selecciona editar el personatge.
3. L'aplicació mostra la pantalla d'edició de personatges, l'usuari canvia les dades que desitja i les guarda.
4. L'aplicació demana confirmació de la operació.
5. L'usuari accepta, l'aplicació valida el personatge i, depenent del resultat, el marca com "Pending revision"
6. L'aplicació guarda les noves dades.

**Extensions:**

- 2a. L'usuari no té cap permís sobre el personatge: aquest no apareix a la pantalla d'administració de personatges.
- 2b. L'usuari no té permisos d'escriptura sobre el personatge: no és possible de modificar el personatge.
- 4a. L'usuari cancel·la el procés: l'aplicació descarta la informació i torna a la pantalla d'administració de personatges.
- 5a. L'usuari no accepta, l'aplicació no fa res.

## CU21\_Canviar el valor d'una característica

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat i està creant/editant un personatge

**Postcondició:** la característica agafa el nou valor.

**Escenari principal d'èxit:**

1. L'usuari accedeix a la pantalla de dades generals del formulari de creació de personatges.
2. L'usuari insereix el nou valor al camp de la característica.
3. El sistema calcula el cost per a totes les característiques (consultar apèndix A per veure la taula) i el suma.
4. L'aplicació resta aquest total de 7 i el mostra al camp de punts de característiques.

## CU22\_Seleccionar un quirk

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat i està creant/editant un personatge

**Postcondició:** el quirk s'incorpora a la llista de quirks seleccionats per l'usuari.

**Escenari principal d'èxit:**

1. L'usuari accedeix a la pantalla de quirks al formulari de creació de personatges.
2. L'usuari selecciona quin quirk vol.
3. L'aplicació calcula el valor total de la llista actualitzada de quirks de tipus virtut seleccionats fins ara i li resta el valor total dels quirks de tipus defecte seleccionats i mostra el resultat.
4. L'aplicació actualitza les llistes de quirks i habilitats accessibles en funció de la nova llista de quirks seleccionats.

**Extensions:**

- 1a. El quirk no és accessible ja que no s'acompleixen els requisits: l'usuari no pot seleccionar el quirk.

## CU23\_Deseleccionar un Quirk

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat i està creant/editant un personatge

**Postcondició:** el quirk se treu de la llista de quirks seleccionats per l'usuari.

**Escenari principal d'èxit:**

5. L'usuari accedeix a la pantalla de quirks al formulari de creació de personatges.
6. L'usuari selecciona quin quirk vol treure de la llista de quirks seleccionats.
7. L'aplicació calcula el valor total de la llista actualitzada de quirks de tipus virtut seleccionats fins ara i li resta el valor total dels quirks de tipus defecte seleccionats i mostra el resultat.
8. L'aplicació actualitza les llistes de quirks i habilitats accessibles en funció de la nova llista de quirks seleccionats.

## **CU24\_Canviar el punts d'experiència assignats a una habilitat a la infantesa**

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat i està creant/editant un personatge

**Postcondició:** el punts d'experiència assignats a l'habilitat a la infantesa canvien al nou valor.

**Escenari principal d'èxit:**

1. L'usuari accedeix a la pantalla d'habilitats al formulari de creació de personatges.
2. L'usuari introdueix el nou valor de punts d'experiència assignats al camp de punts d'experiència infantil de l'habilitat que vol modificar.
3. L'aplicació calcula el valor total de punts d'experiència infantils assignats entre totes les habilitats, el resta de 45 i mostra el valor resultant a la interfície.

**Extensions:**

- 2a. Aquesta habilitat no pot rebre punts d'experiència a la infantesa: el camp no és accessible.
- 2b. L'habilitat no és accessible degut als requisits: el camp no és editable.
- 3a. El total ja és igual a 45: no és possible introduir valors al camp.
- 3b. El total era inferior a 45 i ara és superior: al camp s'hi assigna la diferència entre 45 i el total anterior(p.e. si el total anterior era 38 i s'assignen 20, només n'apareixeran 7).

## **CU25\_Canviar el punts d'experiència assignats a una habilitat a l'edat adulta**

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat i està creant/editant un personatge

**Postcondició:** el punts d'experiència assignats a l'habilitat a l'edat adulta canvien al nou valor.

**Escenari principal d'èxit:**

1. L'usuari accedeix a la pantalla d'habilitats al formulari de creació de personatges.

2. L'usuari introdueix el nou valor de punts d'experiència assignats al camp de punts d'experiència adulta de l'habilitat que vol modificar.
3. L'aplicació calcula el valor total de punts d'experiència infantils assignats entre totes les habilitats, el resta del límit de punts disponibles per edat(veure apèndix A) i mostra el valor resultant a la interfície.

**Extensions:**

- 2a. Aquesta habilitat no pot rebre punts d'experiència a la infantesa: el camp no és accessible.  
2b. L'habilitat no és accessible degut als requisits: el camp no és editable.  
3a. El total ja és igual al límit de punts disponibles per edat: no és possible introduir valors al camp.  
3b. El total era inferior al límit i ara és superior: al camp s'hi assigna la diferència entre el límit i el total anterior(p.e. si el total anterior era 130, amb un límit de 165 i s'assignen 40, només n'apareixeran 35).

## CU26\_Validar un personatge

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat

**Postcondició:** l'aplicació informa al usuari si el personatge és correcte.

**Escenari principal d'èxit:**

1. L'usuari accedeix a la pantalla d'administració de personatges.
2. L'usuari accedeix a la pantalla d'edició del personatge.
3. L'usuari demana validació.
4. L'aplicació analitza el personatge i retorna una llista amb totes les violacions de les regles que ha trobat.

**Extensions:**

- 2a. L'usuari no té cap permís sobre el personatge: aquest no apareix a la pantalla d'administració de personatges.  
2b. L'usuari no té permisos d'escriptura sobre el personatge: no és possible d'accedir a la pantalla d'edició del personatge.  
3a. L'usuari cancel·la el procés:l'aplicació descarta la informació i torna a la pantalla d'administració de personatges.  
3b. L'usuari guarda el personatge: l'aplicació valida el personatge de forma automàtica i el guarda amb el resultat de la validació.

## CU27\_Modificar els permisos d'un personatge

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat

**Postcondició:** se modifiquen els permisos del personatge per a la resta de jugadors de l'aplicació.

### **Escenari principal d'èxit:**

1. L'usuari accedeix a la pantalla d'administració de personatges.
2. L'usuari accedeix a la pantalla d'edició de permisos del personatge.
3. La pantalla llista només tots els usuaris de perfil jugador excepte del jugador propietari, amb l'estat actual dels permisos.
4. L'usuari canvia els permisos que desitja i els guarda.
5. L'aplicació demana confirmació de la operació.
6. L'usuari accepta, l'aplicació modifica els permisos d'accés del personatge.

### **Extensions:**

- 2a. L'usuari no té cap permís sobre el personatge: aquest no apareix a la pantalla d'administració de personatges.
- 2b. L'usuari no és propietari del personatge: no és possible d'editar els permisos del personatge.
- 3a. L'usuari cancel·la el procés: l'aplicació descarta la informació i torna a la pantalla d'administració de personatges..
- 5a. L'usuari no accepta, l'aplicació no fa res.

**Nota:** per defecte, l'administrador és copropietari inamovible de tots els personatges creats.

## **CU28\_Transferir la propietat d'un personatge**

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat

**Postcondició:** se transfereix la propietat del personatge a un altre jugador i l'anterior propietari perd tots els permisos.

### **Escenari principal d'èxit:**

1. L'usuari accedeix a la pantalla d'administració de personatges.
2. L'usuari accedeix a la pantalla d'edició de permisos del personatge.
3. La pantalla llista només tots els usuaris de perfil jugador excepte del jugador propietari, amb l'estat actual dels permisos.
4. L'usuari selecciona l'usuari que serà el nou propietari del personatge.
5. L'aplicació demana confirmació de la operació.
6. L'usuari accepta, l'aplicació transfereix la propietat del personatge i retira a l'usuari els permisos de lectura i escriptura.

### **Extensions:**

- 2a. L'usuari no té cap permís sobre el personatge: aquest no apareix a la pantalla d'administració de personatges.
- 2b. L'usuari no és propietari del personatge: no és possible canviar-ne els permisos.
- 4a. L'usuari cancel·la el procés: l'aplicació torna a la pantalla d'administració de personatges..
- 6a. L'usuari no accepta, l'aplicació no fa res.

## CU29\_Accedir al log

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat

**Postcondició:** s'accedeix a la pantalla de log

**Escenari principal d'èxit:**

1. L'usuari, des de qualsevol part de l'aplicació, selecciona la pantalla principal.
2. L'aplicació redirecciona l'usuari a la pantalla principal on hi ha el log.

**Extensions:**

1a. L'usuari, quan entra a l'aplicació, és enviat directament a la pantalla principal.

## CU30\_Visualitzar un personatge

**Actor principal:** usuari

**Precondició:** l'usuari està loguejat

**Postcondició:** s'accedeix a la fitxa del personatge en mode visualització.

**Escenari principal d'èxit:**

1. L'usuari accedeix a la pantalla d'administració de personatges.
2. L'usuari selecciona visualitzar el personatge.
3. L'aplicació mostra la fitxa del personatge en mode de lectura.

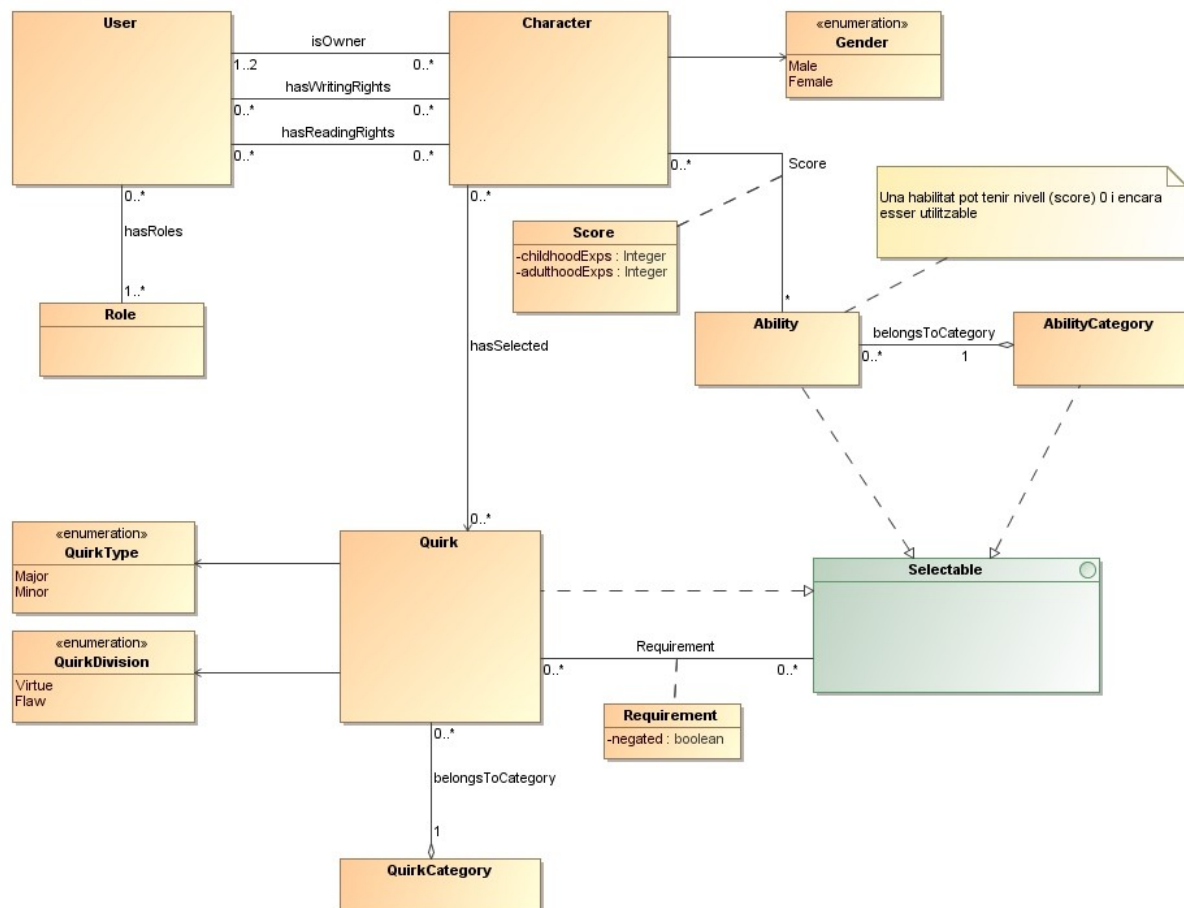
**Extensions:**

2a. L'usuari no té cap permís sobre el personatge: aquest no apareix a la pantalla d'administració de personatges.

2b. L'usuari no té permís de lectura sobre el personatge: no és possible de visualitzar el personatge.

# Disseny

## Diagrama de classes



Il·lustració 4: Diagrama de classes

## Característiques del diagrama de classes

### Usuaris amb múltiples rols

La versió actual del projecte no demana usuaris amb més d'un rol, però se va decidir que si en cas d'ampliació fos necessari, seria interessant disposar des d'el principi d'una estructura que pogués suportar aquesta funcionalitat.

### Multipropietat dels personatges

Com pot observar-se, la relació "IsOwner" té multiplicitat 1..2 a la part de l'usuari. El motiu és que

tots els personatges sempre tenen dos propietaris: l'administrador i el jugador que l'ha creat, excepte quan el creador és també l'administrador, aleshores només hi ha un propietari.

## **Punts d'experiència a les habilitats**

Una habilitat pot rebre punts d'experiència de dues fonts: infantesa i edat adulta, i pot passar que certes habilitats només en puguin rebre a l'edat adulta. Per això és necessari guardar aquests valors per separat a la classe "Score".

## **Diferències entre Quirks i virtuts/defectes**

Funcionalment, virtuts i defectes són idèntics: ambdós tenen les mateixes categories (General, sobrenatural, etc.) i els mateixos tipus (Major/Minor). La única diferència és semàntica i per això, és senzill simplificar el diagrama i englobar-los dins la classe "Quirks" i fent servir una enumeració per diferenciar-los.

## **La interfície "Selectable"**

Si se revisa el codi podrà observar-se que no existeix cap interfície anomenada "Selectable": aquesta interfície és una forma d'expressar que les classes "Quirk", "AbilityCategory" i "Ability" comparteixen el fet de poder tenir elements de la classe "Quirk" com requisits.

## **Base de dades**

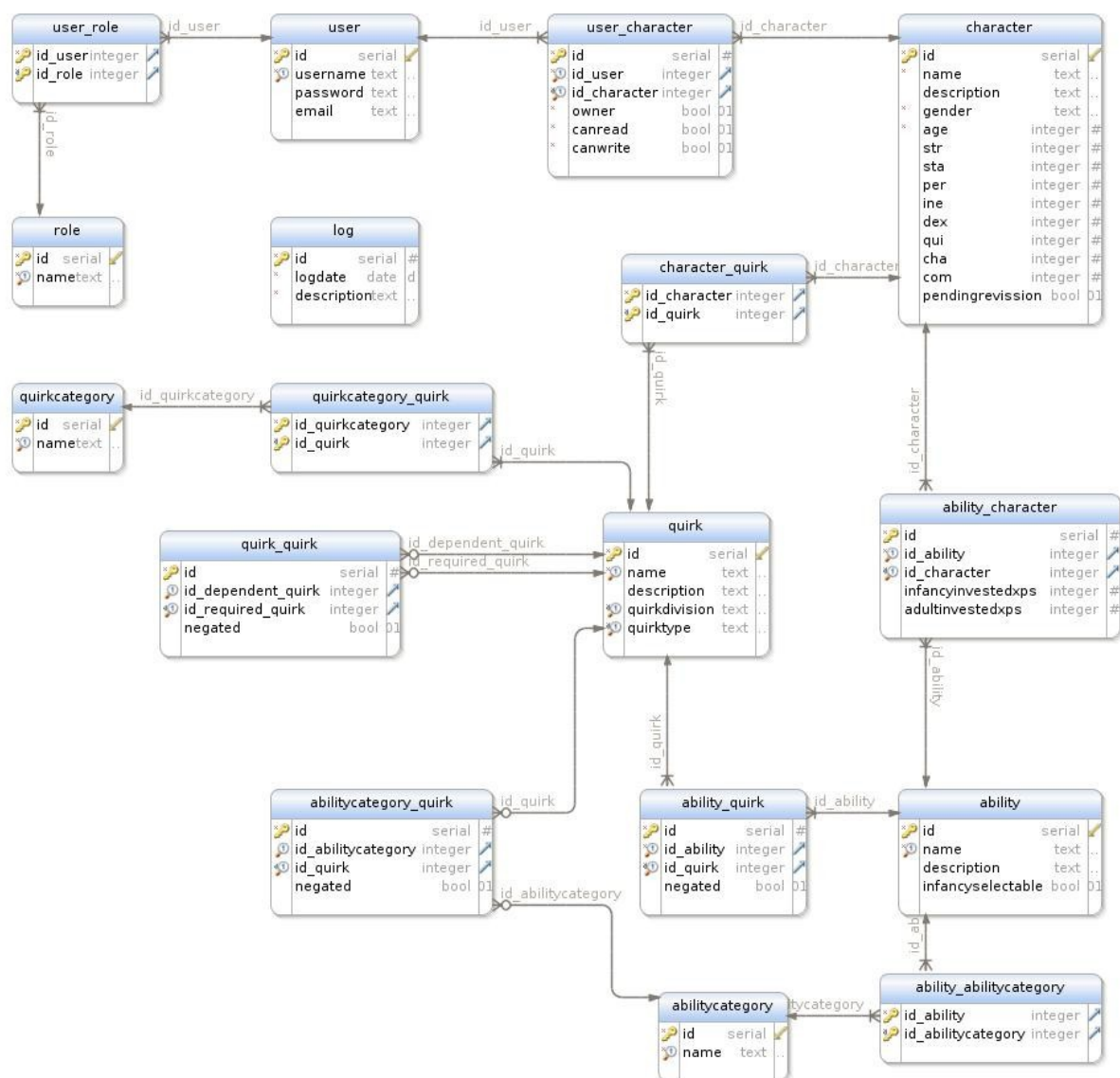
En essència, la base de dades emprada és un reflex del diagrama de classes mostrat al punt anterior.

## **Descripció de les taules**

El diagrama ER se divideix en dos grups: taules principals, amb el seu corresponent al diagrama de classes i taules de relació o "join-tables", que expressen les relacions entre les taules principals.

Com pot observar-se al diagrama les taules principals són aquelles que tenen el mateix nom que els seus equivalents al diagrama de classes. Per altre banda, les "join-tables" tenen com a nom la concatenació dels noms de les dues taules que connecta separats per un guió baix.

La funció d'aquestes "join-tables" és representar relacions 1-N o N-M, encara que algunes tenen informació addicional de la relació i estan mapejades a les seves pròpies classes al projecte.



Il·lustració 5: Esquema ER

## Tecnologies

Per a la construcció del projecte s'ha escollit J2EE així com un conjunt de frameworks i llibreries especialitzades aplicades a cada capa del projecte.

Com se pot veure al diagrama, s'ha optat per no fer servir cap framework de presentació (ExtJS, GWT, etc.) sinó que s'ha preferit d'emprar HTML5+CSS3 pur. La única excepció en aquest sentit és incloure alguns widgets de la llibreria d'efectes JQuery UI.

La funcionalitat de les pàgines depèn de la llibreria JQuery. El motiu d'emprar aquesta llibreria és facilitar l'ús de Javascript i, sobretot, independència del navegador ja que fa de capa intermèdia entre el codi de la pàgina i la implementació local de Javascript.

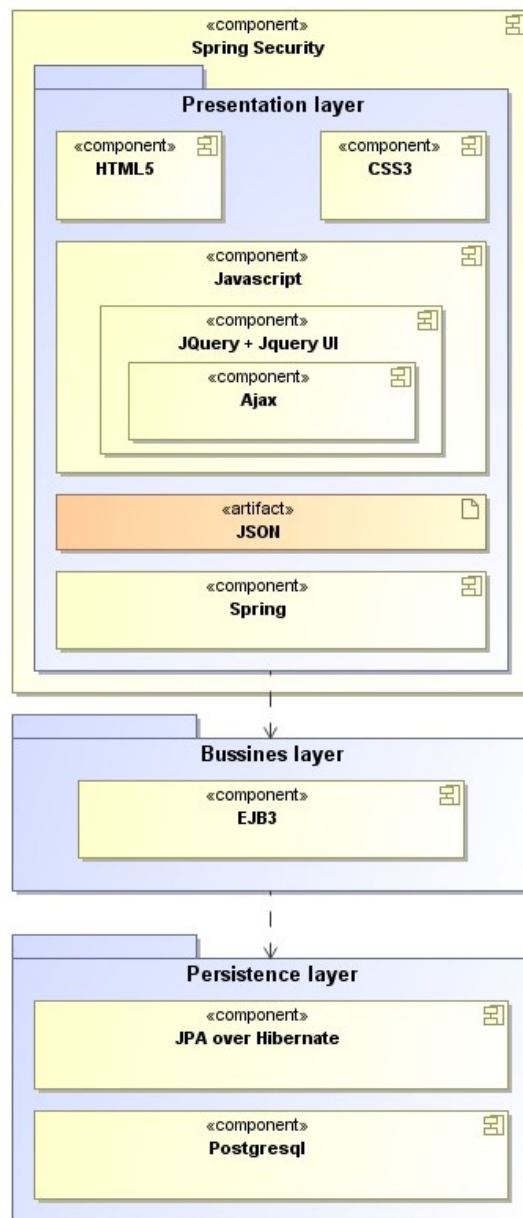
La base de la capa de presentació està construïda sobre Spring MVC. Se va escollir aquest framework perquè és relativament lleuger i senzill d'entendre, a més de integrar-se fàcilment amb altres frameworks i estandars.

En aquest projecte porta tota la lògica de la capa de presentació i fa d'intermediari entre la part de client i la capa de negoci seguint el patró MVC.

La comunicació entre Spring i el client se fa mitjançant cridades POST i GET sobre Ajax als controladors de Spring, emprant JSON per a la transferència d'informació.

La capa de negoci està construïda integrament sobre el estandar EJB3. Aquest és un estandar reconegut i emprat dins la indústria.

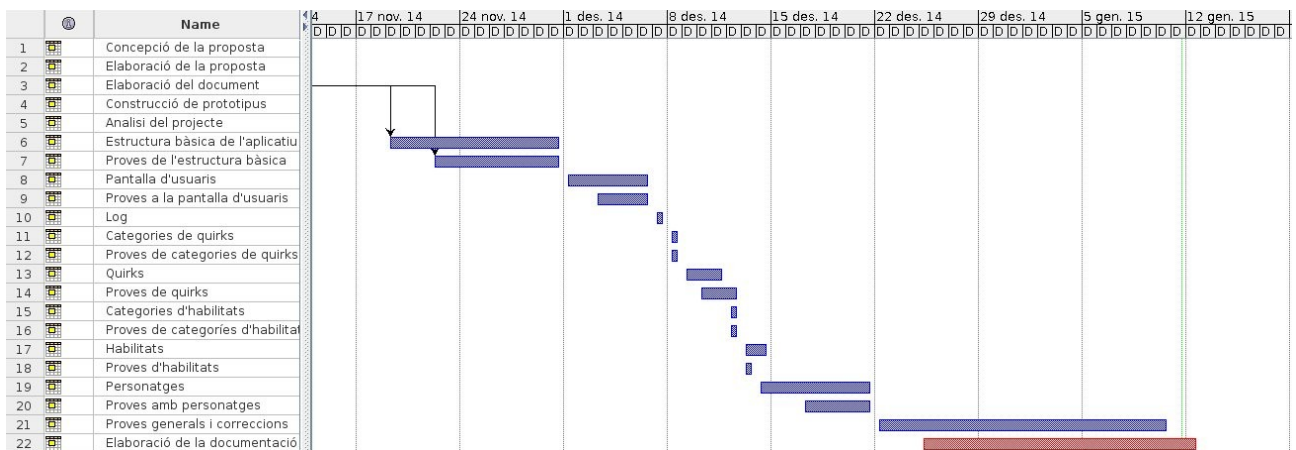
Per a la capa de persistència s'ha triat Hibernate, sobre el que se fa servir l'estandar JPA i que ataca a una base de dades PostgreSql.



*Il·lustració 6: Esquema de tecnologies*

# Implementació

## Calendari final del projecte



El principal factor destacable en la planificació és el buit que arrenca el dia 7/11 i s'estén durant 12 dies. Aquest buit, degut a causes laborals va trastocar completament el calendari i va forçar un retràs molt greu a tot el projecte.

# Anàlisi del procés d'implementació i de les tecnologies emprades

A continuació s'exposa una anàlisi de procés d'implementació així com de les tecnologies emprades, el seu comportament durant el procés, i la impressió que m'han causat.

## Capa de presentació

### HTML5+CSS3

*Eines emprades:* Firefox<sup>2</sup> 32.0 a 34.0, més els plugins Firebug<sup>3</sup> i Dust-Me Selectors<sup>4</sup>

En començar aquesta pràctica vaig optar per no fer servir cap framework de presentació amb la intenció de "jugar" i aprendre sobre HTML5 i CSS3 "purs". Vist en perspectiva, no ha estat una mala decisió ja que, apart d'aprendre, he pogut comprovar que, una vegada superat el procés inicial de construir les pantalles i, sobretot, fer les fulles d'estil que és la part més complexa, el resultat és que satisfactori. La despesa afegida de temps no és excessiva, sobretot si s'estructura la interfície per a reutilitzar codi com he intentat en aquesta pràctica, p.e., totes les pàgines tenen aquesta estructura (obviant els casos partiquals de fulles d'estil, js, etc):

```
<html>
  <head>
    <%@ include file="../../fragments/links.jspf" %>
    <link href="<c:url value='/css/userManagement.css'/>" rel="stylesheet" type="text/css"
media="screen" />
    <link href="<c:url value='/css/editUser.css'/>" rel="stylesheet" type="text/css"
media="screen" />
    <script type="text/javascript" src="<c:url value='/webcontent/js/editUser.js'/>"></script>
  </head>
  <body>
    <%@ include file="../../fragments/header.jspf" %>
    <%@ include file="../../fragments/navBar.jspf" %>
    <%@ include file="../../fragments/errorModalDialog.jspf" %>
    <div class="modalDialog" id="dialog">
```

Mitjançant la sentència `<%@ include file="../../fragments/fragment_name.jspf" %>` és possible d'incloure un tros de codi html dintre d'un altre abans d'enviar-ho al navegador. Així és possible incloure elements com la capçalera, la barra de navegació, etc. sense haver de reescriure-los cada vegada i sabent que basta modificar un sol arxiu per a tota l'aplicació.

La part negativa que veig, és que un framework d'interfícies complet com p.e. ExtJS, posa una capa d'abstracció entre el navegador i el codi, cosa que simplifica tot el procés si se vol que aquesta pagina se pugui emprar correctament a diversos navegadors i dispositius (tables, smartphones, etc), mentre que amb HTML5+CSS3, la tasca és més artesanal i potencialment prohibitiva.

Al final, la qüestió està entre llibertat de fer qualsevol cosa i el cost.

<sup>2</sup> <https://www.mozilla.org/ca/firefox/new/>

<sup>3</sup> <http://getfirebug.com/>

<sup>4</sup> <https://addons.mozilla.org/ca/firefox/addon/dust-me-selectors/>

L'altre inconvenient és que el codi resultant, especialment CSS, pot resultar caòtic i redundant si no s'organitza bé. Aquí he d'admetre que s'agriria una refactorització, potser agrupant els CSS en funció del mòdul(p.e. llista de quirks) més que no per pàgines.

## JavaScript, JQuery i JQuery UI

*Eines emprades:* Firefox 32.0 a 34.0 amb el plugin Firebug

Encara que és perfectament possible fer tota la pràctica fent servir únicament javascript pur, sense recórrer a cap altra llibreria vaig optar per JQuery<sup>5</sup> degut a no poques males experiències en el passat amb diferents navegadors i com aquets entenen el DOM i implementen javascript.

Actualment, aquest problema ha minvat molt (una plataforma com nodeJS seria implensable en temps del IE6) però encara prefereixo emprar JQuery per les següents raons:

- Incorpora una selecció de selectors que permeten treballar comodament amb col·leccions d'elements.
- Encara que les actuals versions dels navegadors han millorat molt en aquest aspecte, encara presenten incompatibilitats entre ells i com implementen el javascript. Molts d'aquestes diferències són detectades i tractades per JQuery.
- Incorpora suport per Ajax i JSON. Basicament és un wrapper sobre javascript però amb algunes millores afegides.

Pel que fa a JQuery UI<sup>6</sup>, aquesta és una llibreria d'efectes gràfics i widgets. He triat emprar-la per no haver d'implementar manualment els següents elements:

- Spinners a la pantalla de creació de personatges.
- Pestanyes.
- Popups modals i normals.

Trobo un detall interessant que sigui possible baixar-se una versió personalitzada de JQueryUI, amb només els efectes seleccionats per si la mida de l'arxiu és un factor determinant.

En quant al procés d'implementació, cap de les dues llibreríes va representar cap dificultat excepte quan vaig voler fer servir els spinners pels punts d'experiència de les habilitats. Això m'hauria estalviat la verificació de les dades ja que, per configuració, haurien estat nombres sencers.

Desgraciadament, no va ser possible integrar correctament els diferents spinners de cada habilitat entre si així que vaig haver de desistir i emprar inputs de tipus text.

---

5 <http://jquery.com/>

6 <http://jqueryui.com/>

## Spring 4.1.2

*Eines emprades:* Eclipse Luna(4.4.0)

Des d'un primer moment vaig descartar els servlets purs per a la interfície entre la part de client i el servidor. En la meua opinió afegeixen massa complexitat i codi a lo que hauria de ser senzill i clar.

També vaig descartar JSF perquè, en la meua opinió, encara que faciliten molt el procés de creació de la interfícia web creen un codi molt lleig i difícil d'entendre.

Hi ha altres frameworks sobre Java (Struts<sup>7</sup>, p.e.) però vaig triar Spring<sup>8</sup> perquè és lleuger i el codi resultant és molt net i fàcilment comprensible, sobretot si se treballa amb etiquetes.

En aquest projecte Spring fa de gestor general, s'integra amb Hibernate i Spring Security però principalment interactua amb la part de client via Ajax/JSON.

La seva configuració pot fer-se amb classes o xmls(jo he preferit el segon mètode), i consta de dos arxius diferents en aquesta aplicació:

- springmvc-servlet.xml: conté tota la configuració de spring MVC, p.e. on trobar els controladors, com construir les vistes, etc.
- applicationContext.xml: aquí hi ha la part relativa a la integració amb Hibernate i Spring Security. P.e se configura un entity manager per a fer servir JPA.

Per altre banda, la part java és molt senzilla. P.e. aquest és un tros editat de la classe

```
@Controller
public class UserController {

    @Inject
    private UserManagementEJBLocal userEJB;

    @RequestMapping(value="/admin/users", method=RequestMethod.GET)
    public String userManagement(Map<String, Object> model) {

        List<UserJPA> userList = userEJB.getAllPlayers();

        model.put("userList", userList);

        return "admin/views/userManagementView";
    }

    @RequestMapping(value="admin/deleteUser", method=RequestMethod.POST)
    @ResponseBody
    public String deleteUser(@RequestParam int id) {

        userEJB.deleteUserById(id);

        return "correct";
    }
}
```

El tag `@Controller` identifica la classe com a un controlador que serà carregat i mapejat en

---

<sup>7</sup> <http://struts.apache.org/>

<sup>8</sup> <https://spring.io/>

desplegar el WAR.

El tag `@Inject` és un tag estandar JSR-330 per injectar el ejb directament a la classe sense haver d'instanciar-lo. Spring implementa aquest estandar encara que té un tag similar (`@Autowired`)

El tag `@RequestMapping` informa a Spring que aquest mètode s'ha d'executar quan se rebí una petició amb un mètode concret a una direcció determinada. P.e. si se rep una petició GET dirigida a `/admin/users`, aleshores s'executarà el mètode `userManagement`

Pot observar-se que `userManagement` retorna un string i té com a paràmetre d'entrada un `Map<String, Object>` anomenat `model`. Bàsicament, aquest mètode recupera unes dades via EJB, les afegeix a `model` i retorna el que pareix un URL. Mirat des de el punt de vista de Spring, el que fa aquest mètode és afegir unes dades a la resposta per a que estiguin disponibles a la part de client, i retorna el que és efectivament part d'una URL. Aquest string serà formatat segons la configuració que pot trobar-se a `springmvc-servlet.xml` per a trobar la URL on cal redirigir l'usuari.

Per contra, el segon mètode `deleteUser` és molt diferent, aquí hi ha dos tags nous:

`@RequestParam`, `id` és un parametre d'entrada que s'ha de recuperar de la petició que ha fet el client.

`@ResponseBody`, la resposta no és part de la URL d'una vista sinó que són dades.

En general puc afirmar que Spring ha estat de gran ajut en tot el procés d'implementació. Té una configuració minimalista i s'adapta molt bé a aquest sistema de cridades asíncrones desde client.

L'únic inconvenient l'he tingut amb JSON, o més concretament, amb les llibreríes Jackson<sup>9</sup> que són les llibreríes per defecte per serialitzar i deserialitzar JSON. El seu comportament ha estat impecable fins que he hagut de serialitzar objectes amb relacions circulars (p.e. una categoria d'habilitats té habilitats, que pertanyen a una categoria que al seu torn té habilitats, i així *ad infinitum*). En aquest cas les llibreríes Jackson fallen miserablement i cap de les solució que he provat ha funcionat<sup>10</sup>. Al final he emprat el tag `@JsonIgnore` que, posat sobre un camp o el seu getter, diu a les llibreríes Jackson que aquell camp no s'ha de serialitzar. Això ha funcionat al rompre el cercle però condiona la informació que s'envia i, en una aplicació on s'accedeix a la mateixa informació des de múltiples fonts cercant diferents parts d'ella, pot ser un handicap greu.

---

9 <https://github.com/FasterXML/jackson>

10 <http://keenformatics.blogspot.com.es/2013/08/how-to-solve-json-infinite-recursion.html>

## Spring Security 3.2.5

Eines emprades: Eclipse Luna(4.4.0)

Per la part de seguretat de l'aplicació vaig triar d'emprar Spring Security<sup>11</sup> per dos motius:

- S'integra molt bé amb Spring
- No ho havia emprat mai.

La instal·lació és també molt senzilla, encara que pot ser una mica més enrevesada ja que consta de dues parts:

L'arxiu spring-security.xml, amb la configuració propiament dita.

La classe CustomUserDetailsService.java, que s'utilitza per accedir a les dades dels usuaris quan s'ha de fer una verificació de permisos.

L'estructura del xml és simple:

```
<http create-session="never" use-expressions="true">
  <intercept-url pattern="/login" access="permitAll" />
  <intercept-url pattern="/logout" access="permitAll" />
  <intercept-url pattern="/webcontent/**" access="permitAll" />
  <intercept-url pattern="/admin/**" access="hasRole('ROLE_ADMIN')" />
  <intercept-url pattern="/**" access="hasAnyRole('ROLE_ADMIN', 'ROLE_PLAYER')" />
  <form-login login-page="/login"
    login-processing-url="/login_security_check"
    username-parameter="username"
    password-parameter="password"
    default-target-url="/mainView"
    authentication-failure-url="/login?authfailed"
    always-use-default-target="true" />
  <session-management>
    <concurrency-control max-sessions="1" expired-url="/login" />
  </session-management>
  <logout delete-cookies="JSESSIONID" invalidate-session="true" logout-url="/logout" logout-success-
url="/login?logout" />
</http>

<authentication-manager alias="authenticationManager">
  <authentication-provider user-service-ref="customUserDetailsService" />
</authentication-manager>
```

El bloc d'instruccions `<intercept-url pattern=... />` intercepten les URLS de les peticions en funció de la seva forma i comproven que l'usuari que les realitza tingui les autoritzacions adequades: p.e. una crida a `/admin/deleteUser` serà processada per `<intercept-url pattern="/admin/**" access="hasRole('ROLE_ADMIN')" />` i a menys que l'usuari sigui administrador, retornarà error.

El bloc

```
<form-login login-page="/login"
  login-processing-url="/login_security_check"
  username-parameter="username"
  password-parameter="password"
```

---

<sup>11</sup> <http://projects.spring.io/spring-security/>

```
default-target-url="/mainView"  
authentication-failure-url="/login?authfailed"  
always-use-default-target="true"/>
```

defineix com és el procés de login, especificant:

- URL de la pàgina de login.
- URL on s'ha de fer la petició per a que s'executi la verificació.
- Nom que a la petició tindran els parametres nom d'usuari i paraula clau.
- URL on redirigir l'usuari en cas d'èxit.
- URL on redirigir l'usuari en cas de fracàs.

La darrera línia és per assegurar que sempre s'empraran aquestes URLs tal i com estan escrites.

A continuació se defineix el nombre màxim de sessions simultànies per un usuari,

```
<session-management>  
  <concurrency-control max-sessions="1" expired-url="/login" />  
</session-management>
```

i com és el procés de logout, incloent l'esborrat de cookies, de la sessió, la URL on fer logout i on redirigir l'usuari en cas d'èxit.

```
<logout delete-cookies="JSESSIONID" invalidate-session="true" logout-url="/logout" logout-success-  
url="/login?logout" />
```

Personalment he quedat molt satisfet amb Spring Security: una vegada resolt l'escull de la configuració ha resultat una solució molt efectiva.

## Capa de negoci

### EJB3

*Eines emprades:* Eclipse Luna(4.4.0)

Arribats a aquesta secció cal dir que vaig triar els EJB3 perquè els he fet servir sempre per a la capa de negoci. Spring va ser concebut originalment com alternativa molt més lleugera a EJB2, i presenta la seva alternativa mitjançant `@Component`<sup>12</sup> i altres tags associats com `@Service` i `@Repository`. Però donat que no conec aquesta part de Spring vaig preferir de no arriscar-me en una capa tan sensible com és la de negoci.

Dels EJB3 i de la capa de negoci en general només puc dir que fa la seva feina correctament i que, en aquesta aplicació, atès que no hi ha accés remot s'han fet servir només interfícies locals.

## Capa de persistència

### JPA 2 sobre Hibernate 4.0.1

*Eines emprades:* Eclipse Luna(4.4.0)<sup>13</sup>

Vaig decidir d'emprar el estandar JPA en lloc de, p.e., les instruccions específiques de Hibernate sobretot per compatibilitat. Vist en perspectiva, som ambivalent sobre el tema, tant sobre JPA com Hibernate ja he tingut bastans problemes a la capa de persistència:

- En condicions normals, la instrucció `remove(object)` esborra un objecte de la base de dades però si hi ha algún altre objecte amb una relació de persistència cap aquest objecte el procés fallarà sense cap missatge d'error<sup>14</sup>.
- Arrencant amb Hibernate 4, aquest va reemplaçar `slf4j` amb `JBoss Logging` però la documentació no diu res al respecte ni com integrar-ho amb `log4j`<sup>15,16</sup>. En conseqüència no he pogut disposar d'un log detallat del funcionament d'hibernate.

Cal dir que, al final i tenint en compte els problemes anteriors, tant Hibernate com JPA han funcionat correctament però, personalment, considero un error greu no poder aconseguir més informació durant el funcionament d'Hibernate.

---

12 <http://stackoverflow.com/questions/6827752/whats-the-difference-between-component-repository-service-annotations-in>

13 <http://www.eclipse.org/home/index.php>

14 <http://stackoverflow.com/questions/16898085/jpa-hibernate-remove-entity-sometimes-not-working>

15 [https://docs.jboss.org/hibernate/orm/4.0/manual/en-US/html\\_single/#configuration-logging](https://docs.jboss.org/hibernate/orm/4.0/manual/en-US/html_single/#configuration-logging)

16 [https://docs.jboss.org/hibernate/orm/4.0/devguide/en-US/html\\_single/](https://docs.jboss.org/hibernate/orm/4.0/devguide/en-US/html_single/)

## PostgreSql 9.3

*Eines emprades:* pgAdmin III i DbSchema

A l'hora de triar una base de dades vaig considerar dues opcions principals: MySql i PostgreSql<sup>17</sup>. De les pràctiques de la UOC m'havia quedat amb les ganes de emprar la segona en un projecte complet i per això la vaig seleccionar. Puc afirmar que no m'ha decebut.

Sobre pgAdmin<sup>18</sup>, l'eina que he fet servir per administrar la base de dades, ha resultat ser una eina molt potent i senzilla de fer servir a pesar de ser relativament simple. Desgraciadament no té cap mòdul per generar un esquema ER de la base de dades així que he hagut de recórrer a una altra eina.

En quant a DbSchema<sup>19</sup>, l'he fet servir per importar el disseny de la base de dades (veure Esquema ER). No la he emprat en profunditat així que no puc opinar més sobre ella però la primera impressió és bona.

## Altres eines

### Jboss 7.1.1<sup>20</sup> as Final

Coneixia aquesta versió de Jboss gràcies a les pràctiques de la UOC i me va pareixer una tria lògica. En general el seu comportament ha estat excel·lent tret d'un bug molt molest que vaig patir al principi.:

En general, quan se desenvolupa sobre HTML5+CSS3 és freqüent haver de redesplegar constantment la capa de presentació a mesura que se fan canvis. Teòricament Jboss suporta el desplegament en calent d'arxius html, js i css fent servir un WAR descomprimit<sup>21</sup>, però degut a un bug a la versió 7.1.1 això va deixar de funcionar. Al final en vaig trobar un parxe<sup>22</sup> però fins aleshores la única forma de redesplegar la capa de presentació era redesplegar tot el WAR amb la conseqüent perdua de temps.

### Eclipse Luna (4.4.0)

La meua opinió sobre el IDE és ambivalent, l'he fet servir Eclipse durant anys i, encara que m'agrada molt, la meua impressió és que amb cada versió no fa sinó augmentar de tamany i consum de memòria. És molt bon software però una versió minimalista d'Eclipse seria molt d'agraïr.

## Kubuntu 14.04

Tota l'aplicació ha estat desenvolupada sobre Kubuntu 14.04. Donat que pràcticament totes les eines importants (PostgreSQL, Eclipse, etc) són multiplataforma l'únic problema ha vingut a l'hora de

---

17 <http://www.postgresql.org/>

18 <http://www.pgadmin.org/>

19 <http://www.db-schema.com/>

20 <http://jbossas.jboss.org/>

21 <http://stackoverflow.com/questions/487363/how-can-i-get-jboss-to-explode-a-deployed-war-file>

22 <https://developer.jboss.org/thread/196890?tstart=0>

cercar eines per tasques com p.e. generar l'esquema ER de la base de dades que he hagut de mirar un parell d'opcions.

## **MagicDraw**

No hi ha molt a dir en aquest punt: no el coneixia fins entrar a la UOC i immediament me va pareixer una eina excel·lent. Aquesta opinió no ha canviat.

## **Control de versions**

Pel control de versions he fet servir Git. Aquesta eina està optimitzada a treball en grup amb diferents branques així que en un projecte amb un sol programador és, potser, excessiva. Ara bé, compta amb un excel·lent suport per internet amb servidors gratuïts on poder muntar un repositori remot i això la fa la tria ideal.

Al meu cas he triat Bitbucket<sup>23</sup> com servidor remot i la implementació estandar de git 1.9.1 en linux com a client local i el resultat ha estat molt satisfactori.

---

<sup>23</sup> <https://bitbucket.org/>

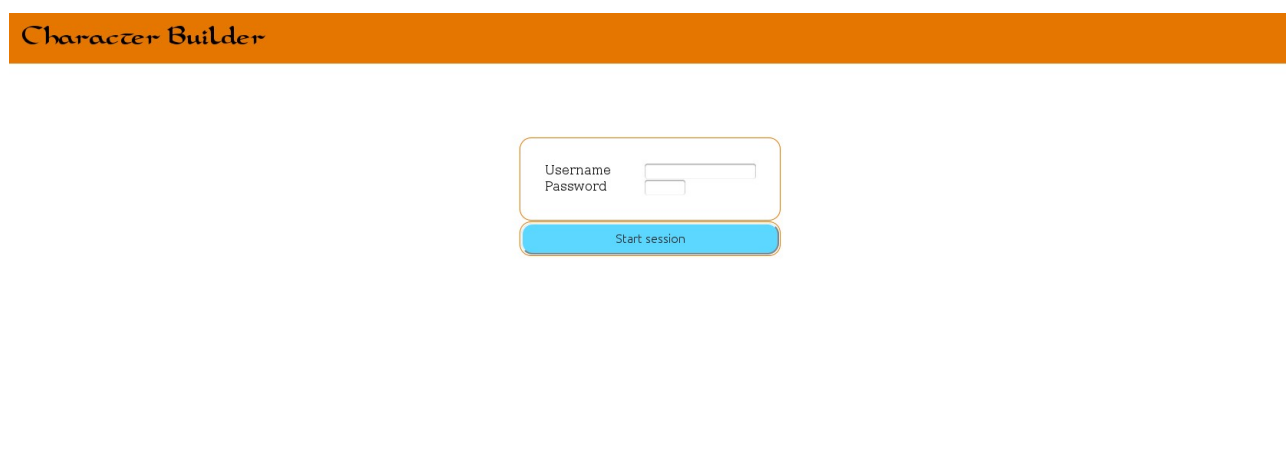
# Manual d'usuari

## Concepte: propietat d'un personatge

Un usuari és propietari d'un personatge quan té tots els permisos sobre ell: lectura, edició, esborrat i edició de permisos. Per defecte, el creador d'un personatge n'és el propietari encara que és possible de transferir-ne la propietat.

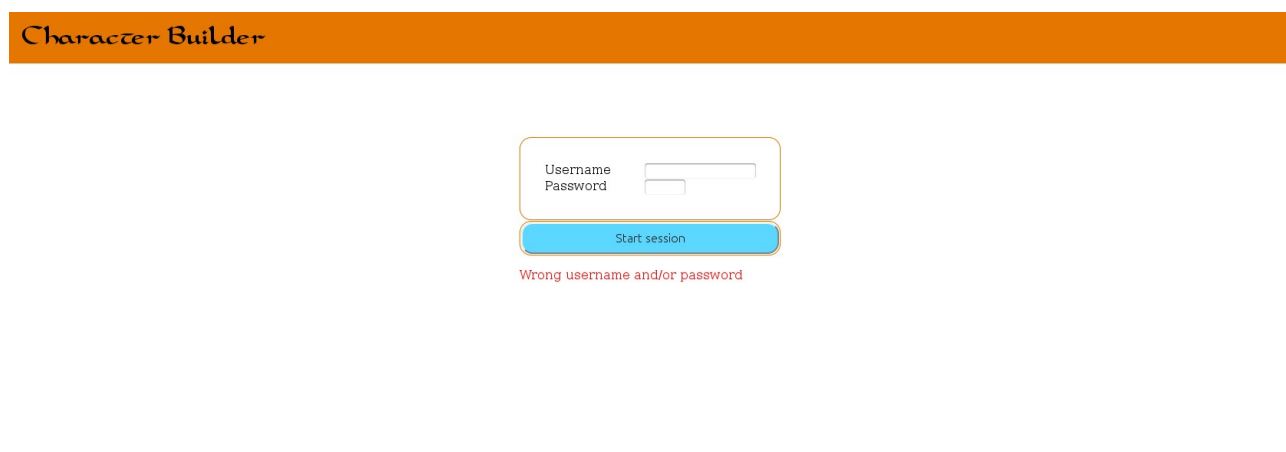
L'administrador és copropietari de tots els personatges presents a l'aplicació.

## Autenticar-se a l'aplicació



### *Il·lustració 7: Pantalla de login*

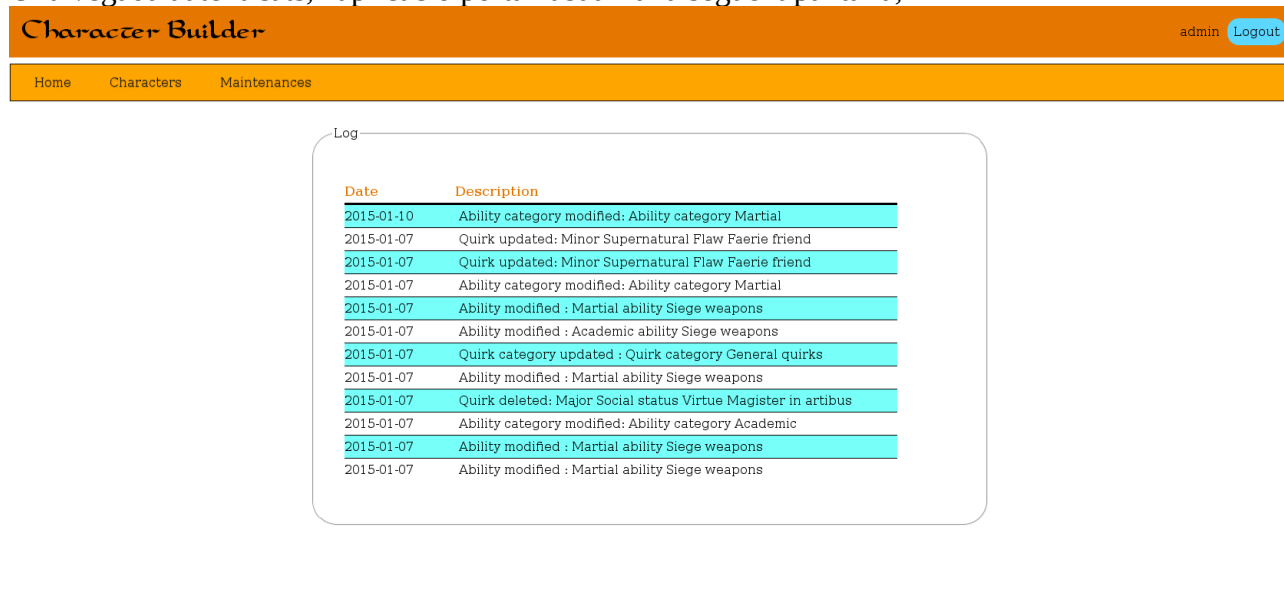
Per autenticar-se a l'aplicació basta amb accedir a la pantalla de login (veure il·lustració 7) i emplenar el formulari amb el nom d'usuari i la paraula clau. A continuació, cal premer el botó "Start Session" per a que l'aplicació verifiqui les dades entrades. En cas que se corresponguin a algún usuari, l'aplicació redirigirà l'usuari a la pantalla principal, en cas contrari donarà un missatge d'error(veure il·lustració 8).



### *Il·lustració 8: Error d'autenticació*

## Estructura bàsica de la interfícia

Una vegada autenticats, l'aplicació porta l'usuari a la següent pantalla,



*Il·lustració 9: Pantalla principal*

que pot dividir-se en les 3 seccions següents:

### Capçalera

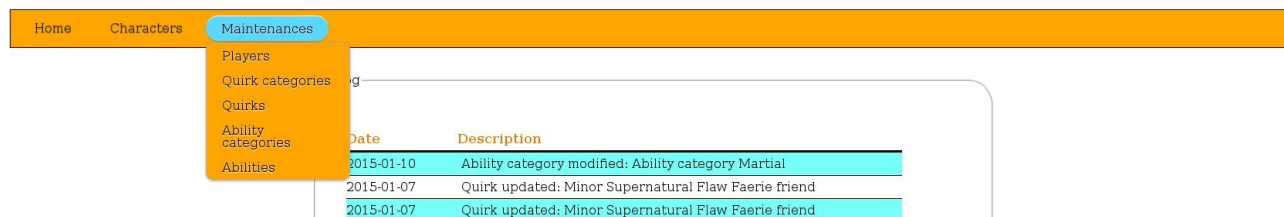


*Il·lustració 10: Capçalera*

La capçalera és bàsicament informativa, tret de dos elements:

- Nom de l'usuari: si se prem damunt, s'accedeix a la pantalla d'edició del propi compte(veure Edició d'usuaris).
- Botó de logout: si se prem, tanca la sessió i torna l'usuari a la pantalla de login.

### Barra de navegació



*Il·lustració 11: Barra de navegació*

Permet accedir a les diferents pantalles de l'aplicació, encara que no totes estan disponibles

depenent del rol de l'usuari. Les opcions actualment existents són:

- Home: porta a la pantalla principal(veure il·lustració 9)
- Characters: porta a la pantalla de personatges.
- Maintenances: és un submenu amb les diferents pantallaes dels manteniments. És exclussiva de l'administrador.

## Cos

Log

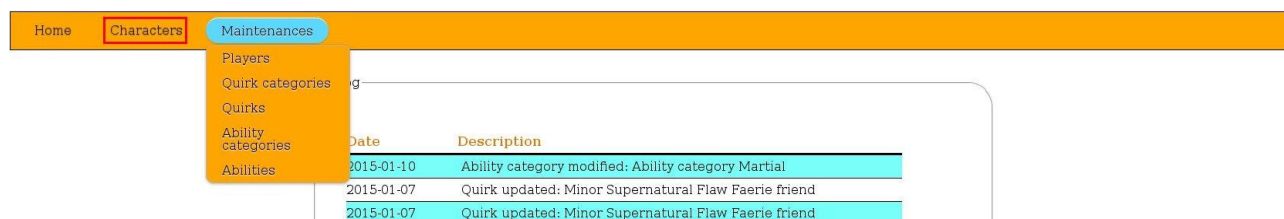
Date	Description
2015-01-10	Ability category modified: Ability category Martial
2015-01-07	Quirk updated: Minor Supernatural Flaw Faerie friend
2015-01-07	Quirk updated: Minor Supernatural Flaw Faerie friend
2015-01-07	Ability category modified: Ability category Martial
2015-01-07	Ability modified : Martial ability Siege weapons
2015-01-07	Ability modified : Academic ability Siege weapons
2015-01-07	Quirk category updated : Quirk category General quirks
2015-01-07	Ability modified : Martial ability Siege weapons
2015-01-07	Quirk deleted: Major Social status Virtue Magister in artibus
2015-01-07	Ability category modified: Ability category Academic
2015-01-07	Ability modified : Martial ability Siege weapons
2015-01-07	Ability modified : Martial ability Siege weapons

### Il·lustració 12: Cos principal de la pantalla

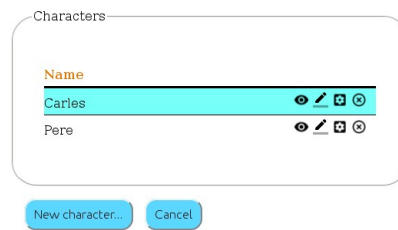
És l'area de treball de l'aplicació i on se carreguen les diferents pantalles quan l'usuari les necessita. Així, la capçalera i la barra de navegació romanen constants mentre el contingut del cos canvia depenent de la opció triada.

## Administració de personatges

S'accedeix a aquesta pantalla seleccionant aquesta opció,



que porta a la pantalla d'administració de personatges:



*Il·lustració 13: Pantalla d'administració de personatges*

L'elements més destacat és la llista de personatges als que l'usuari té accés. Se considera que un usuari té accés a un personatge quan se dona una de les condicions següents:

- L'usuari és propietari del personatge.
- El propietari del personatge ha donat a l'usuari permís de lectura o escriptura sobre el personatge.

Cada línia representa un personatge i consta dels següents elements:



*Il·lustració 14: Línia de personatge*

- Nom del personatge
- Una sèrie d'icones que representen les opcions disponibles.

Depenent dels permisos que tingui l'usuari apareixeran més o menys icones. El seu significat és el següent:

- Veure la fitxa del personatge en mode lectura. Demana permís o esser propietari. Veure Visualització de personatges
- Editar la fitxa del personatge. Demana permís o esser propietari. Veure Edició de personatges
- Editar els permisos d'accés d'aquest personatge pels altres jugadors. Demana esser propietari. Veure Edició de permisos d'accés
- Esborrar el personatge. Demana esser propietari. Veure Esborrat de personatges

A més, hi dos botons: un per cancel·lar i tornar a la pantalla principal i un altre que porta a la creació de personatges.

## Creació de personatges

Per a una major comprensió d'aquest apartat se recomana llegir prèviament l'apèndix A.

La pantalla de creació de personatges consta de 3 pestanyes diferents,

### Dades generals

The screenshot shows a web interface for creating a new character. It has three tabs: 'Character data' (selected), 'Quirks', and 'Abilities'. The 'New Character' form contains the following elements:

- Name:** A text input field with a red asterisk indicating it is required.
- Gender:** A dropdown menu with the text 'Choose gender' and a red asterisk.
- Age:** A spinner box currently showing '16'.
- Description:** A large text area.
- Attributes:** Eight spinner boxes for Intelligence, Perception, Charisma, Communication, Quickness, Dexterity, Strength, and Stamina, all currently set to 0.
- Characteristic points:** A spinner box currently showing 7.
- Validation:** A red asterisk and the text '\* : Required fields' are located below the description field.
- Buttons:** 'Save', 'Cancel', and 'Validate' buttons are at the bottom of the form.

*Il·lustració 15: Dades generals*

Aquí hi ha les dades generals del personatge. La primera part és un formulari amb els següents camps per emplenar:

- Nom del personatge (obligatori).
- Un desplegable amb el sexe del personatge (obligatori)
- Edat (té un mínim de 16 anys): determina el nombre de punts d'experiència disponibles a l'edat adulta.
- Descripció del personatge (opcional).

La segona part està formada per les característiques, amb el següent format:

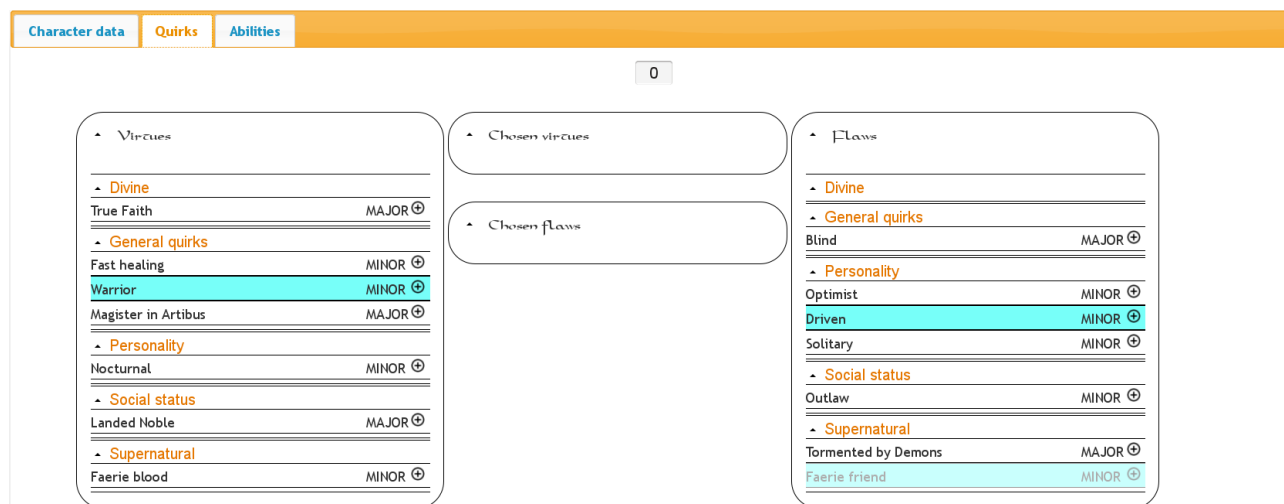
Intelligence

Com s'explica a l'apèndix A i al video, una característica pot prendre valors entre 3 i -3. L'usuari disposa de 7 punts de característiques que pot gastar per comprar característiques positives segons la taula 4 de l'apèndix A. També pot posar valors negatius a característiques per aconseguir més punts per gastar. El recompte final de punts se mostra al camp següent:


Characteristic points

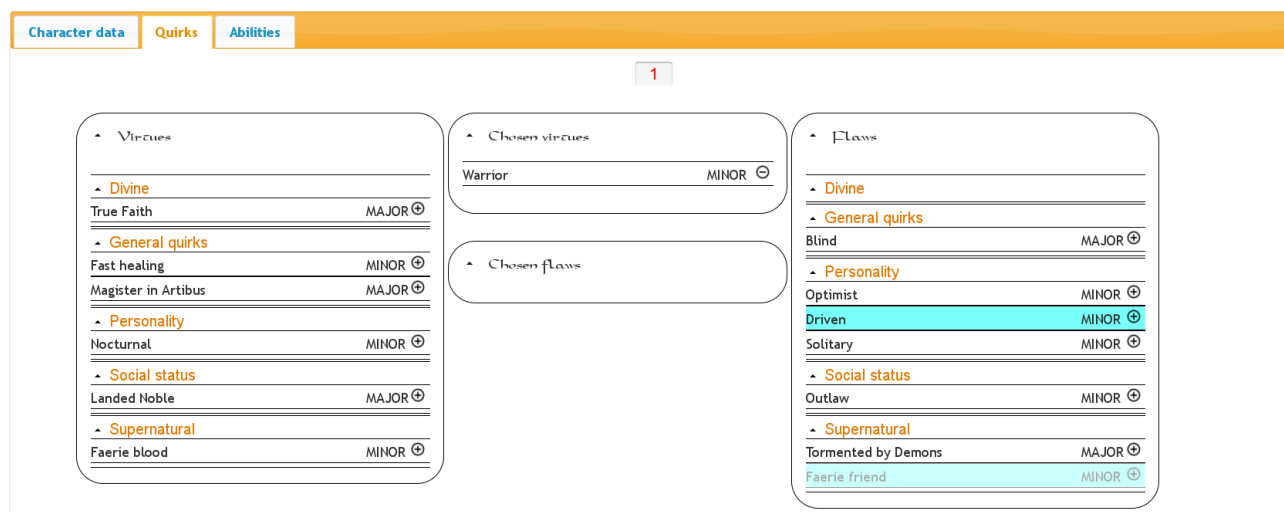
Es possible quedar amb punts negatius però això invalida el personatge fins que se corregeixi.


## Quirks



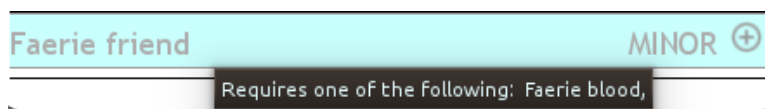
Il·lustració 16: Pantalla de quirks

En aquesta pantalla l'usuari pot seleccionar les virtuts i defectes que vulgui pel seu personatge. Per seleccionar-ne un basta amb fer click sobre la icona . Això afegeix el quirk a la seva llista corresponent d'elements seleccionats.



Per deseleccionar un quirk basta premer la icona .

Pot passar que un quirk no sigui seleccionable, això se deu a que aquest quirk té requisits, és a dir cal seleccionar un altre quirk o deseleccionar-ne un altre abans. Per saber quins requisits té un quirk, basta amb passejar el cursor sobre ell fins que apareix un tooltip amb la informació.



Pot passar que hi hagi més d'un requisit però basta satisfer-ne un per a que el quirk sigui accessible.

Al igual que amb les característiques hi ha certes regles que s'han d'acomplir amb els quirks per a que el personatge sigui considerat vàlid (veure apèndix A). En el cas dels quirks aquestes són:

- Les virtuts i defectes han d'estar equilibrats en punts : cada quirk equival a un nombre de punts depenent del seu tipus (1 punt els Minor i 3 punts els Major), i se cerca que al final el sumatori de punts de virtuts i defectes estigui en equilibri. Per ajudar a aquesta tasca, hi ha un indicador a la pantalla de quirks amb la diferència actual



- No pot haver més de 10 punts en virtuts o defectes.
- Els requeriments de tots els quirks seleccionats han d'estar satisfets.

Finalment, cada quirk només pot esser seleccionat una vegada.

## Habilitats

Character data

Quirks

Abilities

Free childhood xp points: 45

Free adulthood xp points: 165

Abilities

Name	Childhood	Adulthood	XP's	Score
<b>Academic</b>				
Medicine		0	0	0
Artes liberales		0	0	0
<b>General</b>				
Athletics	0	0	0	0
Swin	0	0	0	0
Guile	0	0	0	0
Brawl	0	0	0	0
<b>Martial</b>				
Single weapon		0	0	0
Bows		0	0	0
Siege weapons		0	0	0
Great weapon		0	0	0
<b>Supernatural</b>				
Second Sight	0	0	0	0

Il·lustració 17: Pantalla d'habilitats

Segons les regles de l'apèndix A, un personatge passa per dues etapes:

- Infantesa(0 a 5 anys), guanya 45 xps a repartir entre un nombre concret d'habilitats.
- Edat adulta (6+), guanya 15 punts per any de vida, amb una edat mínima de 16.

Això se representa a la pantalla amb els següents camps:

Free childhood xp points: 45

Free adulthood xp points: 165

No és possible afegir més punts a la infantesa però si a l'edat adulta augmentant l'edat del personatge a un ritme de 15 punts/any.

Cada línia de la taula se correspon amb una habilitat i tenen el següent format:

Name	Childhood	Adulthood	XP's	Score
▸ Academic				
Medicine		0	0	0
Artes liberales		0	0	0
▸ General				
Athletics	0	0	0	0

D'esquerra a dreta hi ha:

- El nom de l'habilitat.
- Camp de punts d'experiència de la infantesa("Childhood").
- Camp de punts d'experiència de l'edat adulta("Adulthood").
- Total de punts d'experiència.
- Score

D'aquests camps, només són editables els camps etiquetats com "Childhood" i "Adulthood" i representen els punts assignats a aquella habilitat a la infantesa i a l'edat adulta respectivament.

De totes formes, poden observar-se dues coses:

- No totes les habilitats tenen el camp "Childhood".
- Algunes de les habilitats no se poden editar.

El motiu de perquè algunes habilitats no tenen camp "Childhood" és que, per configuració, se considera que no se poden aprendre a la infantesa i per tant només poden rebre punts a l'edat adulta. En quant al segon punt, la causa és senzilla: les habilitats poden tenir requisits que s'han de satisfer per a que l'habilitat sigui accessible. Per saber quins requisits té una habilitat basta passar el cursor a sobre per a que aparegui el tooltip amb la informació.

Posant punts a una habilitat

Per afegir punts basta amb editar un dels dos camps i escriure directament la quantitat de xps que se vol posar (veure taula 5). L'aplicació calcula automàticament el total de les dues columnes així com el score final i els punts encara no gastats.

Name	Childhood	Adulthood	XP's	Score
▸ Academic				
Medicine		0	0	0
Artes liberales		0	0	0
▸ General				
Athletics	15	30	45	3

*Il·lustració 18: Assignant punts d'experiència*

Nota: si se posen punts a una habilitat amb requisits i després se canvien els quirks de forma que ja

no se'n satisfan els requisits, aquests punts se perden automàticament.

L'usuari pot assignar tots els punts que vulgui fins arribar al seu màxim que és:

- 45 xps, a la columna "Childhood"
- $(\text{Edat} - 5) * 15$ , a la columna "Adulthood".

És possible augmentar l'edat per guanyar més punts però no baixar-la si el nombre de punts d'experiència resultant queda per sota del total assignat fins ara.

## Validació de personatges

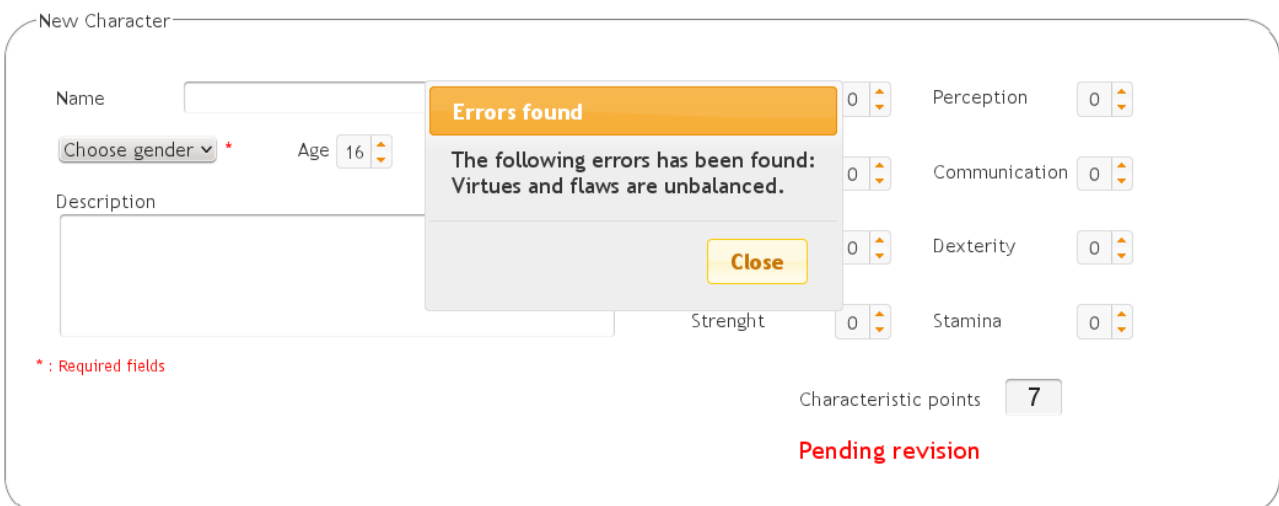
Com s'ha explicat abans, una fitxa ha d'acomplir certes condicions per a que sigui considerada vàlida. Per a comprovar-ho és possible cridar a un validador en qualsevol moment per a que la verifiqui.

El validador se pot cridar de dues maneres:

Fent servir el boto "Validate" de la pestanya de dades generals del personatge.



El validador repassarà la fitxa i si troba res incorrecte, mostrarà un missatge d'error i marcarà la fitxa com "Pending revision".



*Il·lustració 19: Error de validació*

En cas que no trobi res, no mostrarà cap missatge i marcarà la fitxa com a validada.

New Character

Name  \*

Choose gender  \* Age

Description

\* : Required fields

Intelligence  Perception

Charisma  Communication

Quickness  Dexterity

Strenght  Stamina

Characteristic points

**Validated**

*Il·lustració 20: Validació correcta*

Alternativament, se pot guardar directament personatge, l'aplicació cridarà el validador i guardarà el personatge amb el resultat de la validació.

## Visualització de personatges

Aquesta opció demana que l'usuari tingui assignat permís de lectura o que sigui el propietari del personatge. Bàsicament obre la pantalla de creació de personatges en mode lectura, de forma que no és possible editar res.

New Character

Name  \*

Male \* Age

Description

\* : Required fields

Intelligence  Perception

Charisma  Communication

Quickness  Dexterity

Strenght  Stamina

Characteristic points

**Validated**

*Il·lustració 21: Visualització de personatges*

## Edició de personatges

Aquesta opció demana que l'usuari tingui assignat permís d'escriptura o que sigui el propietari del personatge. Obre la pantalla de creació de personatges carregant-hi primer el personatge seleccionat i permet de fer-hi canvis.

## Esborrat de personatges

Aquesta opció demana que l'usuari sigui el propietari del personatge. Esborra el personatge de l'aplicació.

## Edició de permisos d'accés

Aquesta pantalla permet al propietari de donar i treure permisos sobre el seu personatge a usuaris de perfil jugador.



*Il·lustració 22: Pantalla d'edició de permisos de personatges*



La pantalla llista tots els usuaris de perfil jugador del sistema tret de l'usuari propietari. No llista l'administrador ja que aquest és copropietari permanent de tots els personatges.

Cada línia representa un usuari amb els següent camps:

Name	ReadWrite		
player1	<input type="checkbox"/>	<input type="checkbox"/>	
player2	<input type="checkbox"/>	<input type="checkbox"/>	
player3	<input type="checkbox"/>	<input type="checkbox"/>	

- Nom d'usuari
- Un checkbox etiquetat com "Read"
- Un checkbox etiquetat com "Write"
- Un link per a canviar el propietari del personatge.

L'edició de personatge és senzilla: basta amb marcar els checkboxes depenent de si volem donar només permís de lectura o també d'escriptura. P.e.:

Name	ReadWrite		
player1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
player2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
player3	<input type="checkbox"/>	<input type="checkbox"/>	

Amb aquesta configuració l'usuari player1 guanya el permís de lectura mentre l'usuari player2 guanya també permís de lectura i escriptura. L'usuari player3 continua sense tenir-ne accés.

Per a canviar el personatge basta amb premer sobre la icona, l'aplicació demanarà confirmació i si d'accepta, farà el següent:

1. Treure tots els permisos a l'antic propietari de forma que el personatge deixarà d'apareixer a la llista de personatges.
2. Donarà tots els permisos al nou propietari, és a dir: lectura, escriptura, edició de permsos i esborrat.

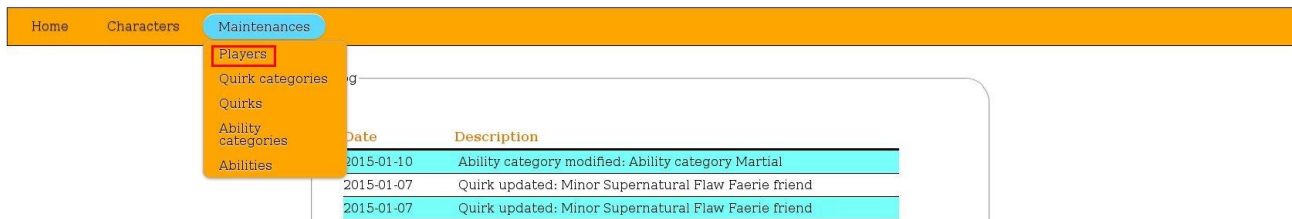
La única excepció és si el personatge només té l'administrador com a propietari, en aquest cas només s'executa el segon punt, convertint el jugador triar en copropietari.

## Manteniments

Aquest bloc comprén el manteniment de la informació necessària pel funcionament de l'aplicatiu i és exclusiva de l'administrador.

## Manteniment d'usuaris

Des d'aquesta opció se porta el manteniment d'usuaris de perfil jugador del sistema.






I aquesta és la pantalla de jugadors:



*Il·lustració 23: Pantalla d'usuaris*

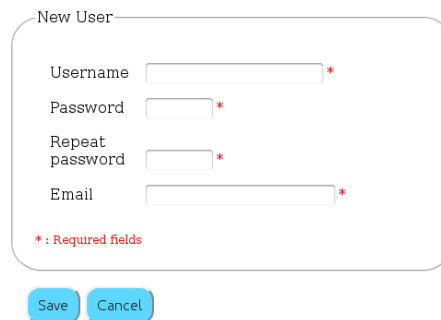
Cada línia de la llista és un usuari amb els següents camps:

Name	Email	
player1	playeremail@email.com	 

- Nom d'usuari
- Email
- Edició del compte de l'usuari 
- Esborrat de l'usuari 

A més hi ha un botó per crear nous usuaris.

## Creació d'usuaris



New User

Username  \*

Password  \*

Repeat password  \*

Email  \*

\* : Required fields

Save Cancel

*Il·lustració 24: Creació d'usuaris*

Per crear un usuari basta emplenar el formulari i guardar. Si el nom coincideix amb el d'un usuari ja existent l'aplicació tornarà un missatge d'error i no continuarà.

## Edició d'usuaris

La pantalla és la mateixa que a la creació tret que hi ha carregades les dades de l'usuari a modificar. Si se tria un nom d'usuari idèntic al d'un altre usuari ja existent l'aplicació donarà error.

## Esborrat d'usuaris

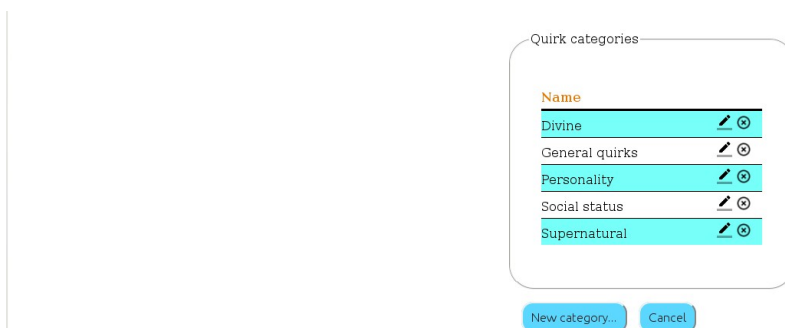
Si se prem sobre la opció d'esborrar l'aplicació demanarà confirmació i si s'accepta, esborrarà l'usuari del sistema. Tots els seus personatges passaran a pertanyer únicament a l'administrador.

## Categoríes de quirks

Els quirks s'agrupen en categoríes encara que, funcionalment, aquestes categoríes no aporten res tret d'organitzar millor la informació.





Aquesta és la pantalla de categoríes de quirks:



*Il·lustració 25: Pantalla de categories de quirks*


Cada línia de la llista és un categoria amb les següents opcions:



- Nom de la categoria
- Edició de la categoria 
- Esborrat de la categoria 

A més, hi ha un botó que porta a la creació de categoríes.

## Creació de categorías de quirks



The screenshot shows a web form titled "New category". It contains a single text input field labeled "Name" with a red asterisk to its right, indicating it is a required field. Below the input field, there is a red error message that reads " \*: Required fields". At the bottom of the form, there are two buttons: "Save" and "Cancel".

*Il·lustració 26: Pantalla de creació de categories de quirks*

Per crear una categoria basta amb posar-li nom i guardar. Si el nom coincideix amb el d'una categoria ja existent, l'aplicació donarà error.

## Edició de categorías de quirks

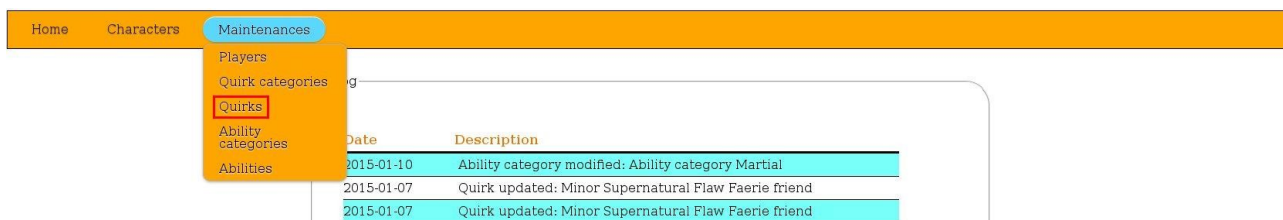
La pantalla és la mateixa que a la creació tret que hi ha carregades les dades de la categoria a modificar. Si se tria un nom de categoria idèntic al d'una altre categoria ja existent l'aplicació donarà error.

## Esborrat de categorías de quirks

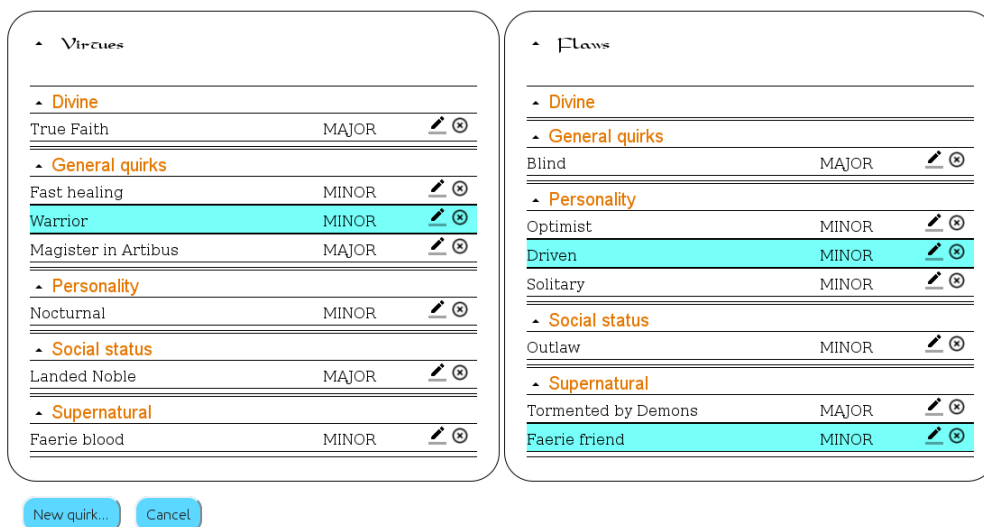
Si se prem sobre la opció d'esborrar, l'aplicació demanarà confirmació i si s'accepta, esborrarà la categoris del sistema si aquesta està "buïda", és a dir, no hi ha cap quirk que hi pertanyi. En cas contrari, donarà error.

## Quirks

Els quirks s'administren des d'aquesta pàgina. Això inclou també els seus requisits.







I aquesta és la pàgina d'administració de quirks:





Il·lustració 27: Pantalla de quirks

Els quirks se mostren en dues llistes separades depenent de si són virtuts o defectes i organitzats per categories.

Cada línia és un quirk amb els següents camps:

Warrior	MINOR	 
Magister in Artibus	MAJOR	 

- Nom del quirk
- Tipus (Major / Minor)
- Edició del quirk 
- Esborrat del quirk 

A més hi ha un botó per crear-ne de nous.

## Creació de quirks

**New Quirk**

Name  \*

Choose kind  \* Choose type  \* Choose category  \*

Description

\* : Required fields

Save Cancel

**Required Virtues**

Divine			
Selected	<input type="checkbox"/>	Negated	<input type="checkbox"/>
True Faith			MAJOR
General quirks			
Selected	<input type="checkbox"/>	Negated	<input type="checkbox"/>
Fast healing			MINOR
Selected	<input checked="" type="checkbox"/>	Negated	<input type="checkbox"/>
Warrior			MINOR
Selected	<input type="checkbox"/>	Negated	<input type="checkbox"/>
Magister in Artibus			MAJOR
Personality			
Selected	<input type="checkbox"/>	Negated	<input type="checkbox"/>
Nocturnal			MINOR
Social status			
Selected	<input type="checkbox"/>	Negated	<input type="checkbox"/>
Landed Noble			MAJOR
Supernatural			
Selected	<input type="checkbox"/>	Negated	<input type="checkbox"/>
Faerie blood			MINOR

**Required Flaws**

Divine			
General quirks			
Selected	<input type="checkbox"/>	Negated	<input type="checkbox"/>
Blind			MAJOR

Il·lustració 28: Pantalla de creació de quirks

La pantalla consta de dues seccions, una de dades del quirk i l'altre de requisits.

## Dades generals del quirk

**New Quirk**

Name  \*

Choose kind  \* Choose type  \* Choose category  \*

Description

\* : Required fields

Save Cancel

Il·lustració 29: Dades generals del quirk

Aquesta secció és obligatòria i consta dels següents camps:

- Nom del quirk
- Un desplegable on seleccionar si és virtut o defecte.
- Un desplegable on seleccionar el seu tipus(Major o Minor).
- Un desplegable on seleccionar-ne la categoria.
- Una descripció(opcional) del quirk.

## Requisits del quirk

Aquesta secció és opcional ja que se poden crear quirks sense requisits. Consta d'un llistat amb tots els quirks del sistema.

Required Virtues

Divine

Selected

Negated

True Faith

MAJOR

General quirks

Selected

Negated

Fast healing

MINOR

Selected

Negated

Warrior

MINOR

Selected

Negated

Magister in Artibus

MAJOR

Personality

Selected

Negated

Nocturnal

MINOR

Social status

Selected

Negated

Landed Noble

MAJOR

Supernatural

Selected

Negated

Faerie blood

MINOR

Cada línia és un quirk amb següents camps:

Selected

Negated

Warrior

MINOR

- Un checkbox marcat com "Selected"
- Un checkbox marcat com "Negated" i que normalment està desactivat.
- Nom del quirk
- Tipus del quirk

Per assignar requisits basta amb clicar els checkboxes "Selected" dels quirks que se vulguin com requisits. S'ha de tenir en compte que si hi ha més d'un requisit, bastarà amb que se'n stisfaci un per a que el quirk sigui seleccionable a la fitxa.

La funció del checkbox "Negated" és invertir la relació entre el quirk i el seu requisit. Si suposam un quirk A que pren un altre B com requisit:

<b>Selected</b>	<b>Negated</b>	<b>Relació amb el quirk</b>
Deseleccionat	Desactivat	B no és requisit de A
Seleccionat	Deseleccionat	Per poder seleccionar A és necessari haver seleccionat B
Seleccionat	Seleccionat	Per poder seleccionar A cal NO haver seleccionat B

*Taula 1: Requisits d'un quirk*

En acabar de seleccionar d'emplenar el formulari i de triar els requisits, basta guardar les dades. Si s'ha triat com a nom el d'un altre quirk ja existent i que a més coincideixin el tipus (Major/Minor) i la classe (Virtut / Defecte), l'aplicació donarà error.

### **Edició de quirks**

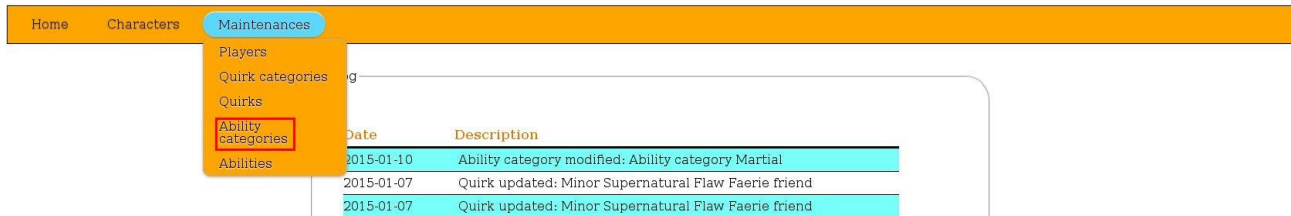
La pantalla és la mateixa que a la creació tret que hi ha carregades les dades del quirk a modificar. Si s'ha triat com a nom el d'un altre quirk ja existent i que a més coincideixin el tipus (Major/Minor) i la classe (Virtut / Defecte), l'aplicació donarà error.

### **Esborrat de quirks**

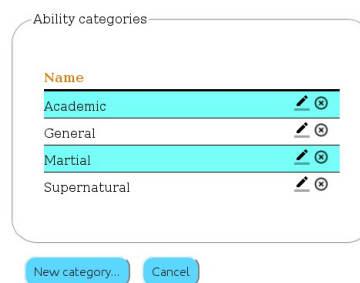
Si se prem sobre la opció d'esborrar, l'aplicació demanarà confirmació amb un missatge exposant si és requisit d'algún altre element. Si s'accepta, esborrarà el quirk del sistema. Si aquest quirk era requisit d'algún altre element, deixarà de ser-ho.

## Categoríes d'habilitats

A diferència de les categories de quirks, les d'habilitats tenen importància funcional ja que poden tenir requisits que se propaguen a les habilitats que en formen part.





I aquesta és la pantalla:



*Il·lustració 30: Pantalla de categories d'habilitats*

Cada línia és una categoria amb els següents camps:



- Nom de la categoria
- Edició de la categoria 
- Esborrat de la categoria 

A més, hi ha un botó que porta a la creació de categories.

## Creació de categories d'habilitats

The screenshot shows a 'New category' form with two main sections. The first section, 'New category', contains a 'Name' field with an asterisk indicating it is required, and 'Save' and 'Cancel' buttons. The second section, 'Required Virtues', contains a list of virtues with checkboxes for 'Selected', 'Negated', and 'Required'. The 'Required' column is highlighted in blue. The virtues listed are: True Faith (MAJOR), Fast healing (MINOR), Warrior (MINOR), Magister in Artibus (MAJOR), Nocturnal (MINOR), Landed Noble (MAJOR), and Faerie blood (MINOR). Below this, there is a section for 'Required Flaws' with a similar list of flaws.

New category			
Name *			
*: Required fields			
Save Cancel			
Required Virtues			
Divine			
Selected	Negated	True Faith	MAJOR
General quirks			
Selected	Negated	Fast healing	MINOR
Selected	Negated	Warrior	MINOR
Selected	Negated	Magister in Artibus	MAJOR
Personality			
Selected	Negated	Nocturnal	MINOR
Social status			
Selected	Negated	Landed Noble	MAJOR
Supernatural			
Selected	Negated	Faerie blood	MINOR
Required Flaws			
Divine			
General quirks			
Selected	Negated	Blind	MAJOR

Il·lustració 31: Creació de categories

La pantalla de creació consta de dues seccions: una de dades generals i una de requisits.

## Dades generals d'una categoria d'habilitats

The screenshot shows the 'New category' form with the 'Name' field highlighted. The field has an asterisk indicating it is required. Below the field, there is a red text label: '\*: Required fields'.

New category	
Name *	
*: Required fields	

Il·lustració 32: Dades generals d'una categoria d'habilitats

L'únic camp present és el nom de la categoria i, com a altres casos, si coincideix amb el d'una altra categoria l'aplicació donarà error en guardar la categoria.

## Requisits d'una categoria d'habilitats

Aquesta secció és opcional ja que se poden crear categories sense requisits. Consta d'un llistat amb tots els quirks del sistema.

Required Virtues

Divine

Selected

Negated

True Faith

MAJOR

General quirks

Selected

Negated

Fast healing

MINOR

Selected

Negated

Warrior

MINOR

Selected

Negated

Magister in Artibus

MAJOR

Personality

Selected

Negated

Nocturnal

MINOR

Social status

Selected

Negated

Landed Noble

MAJOR

Supernatural

Selected

Negated

Faerie blood

MINOR

Cada línia és un quirk amb següents camps:

Selected

Negated

Warrior

MINOR

- Un checkbox marcat com "Selected"
- Un checkbox marcat com "Negated" i que normalment està desactivat.
- Nom del quirk
- Tipus del quirk

Per assignar requisits basta amb clicar els checkboxes "Selected" dels quirks que se vulguin com requisits. S'ha de tenir en compte que si hi ha més d'un requisit, bastarà amb que se'n satisfaci un per a que els requisits de la categoria se considerin satisfets.

La funció del checkbox "Negated" és invertir la relació entre la categoria i el seu requisit. Si suposam una categoria A pren un quirk B com requisit:

Selected	Negated	Relació amb el quirk
Deseleccionat	Desactivat	B no és requisit de A
Seleccionat	Deseleccionat	Per poder posar punts a habilitats d'A és necessari haver seleccionat B
Seleccionat	Seleccionat	Per poder posar punts a habilitats d'A cal NO haver seleccionat B

Taula 2: Requisits d'una categoria d'habilitats

Els requisits no afecten directament la categoria però si les seves habilitats. Quan se posen requisits a una categoria d'habilitats, passa el següent:

- Totes les habilitats de la categoria guanyen els mateixos requisits.
- Si se crea una habilitat d'aquesta categoria guanya els seus requisits.
- Si una habilitat canvia de categoria, perd els requisits de la categoria anterior i guanya els de la nova.

### **Edició de categories d'habilitats**

La pantalla és la mateixa que a la creació tret que hi ha carregades les dades de la categoria a modificar.

En acabar d'editar-la, l'aplicació repassa tots els personatges que tenen alguna habilitat d'aquesta categoria per comprovar si, legalment, encara la poden tenir. Si no és així, esborra l'habilitat de la fitxa i posa el personatge com "Pending Revision".

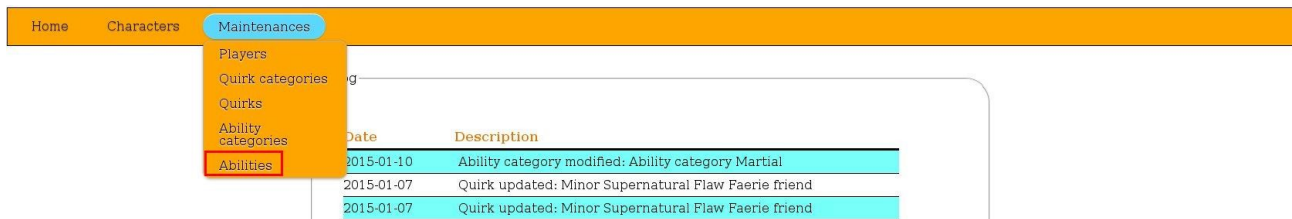
Si se tria un nom de categoria idèntic al d'una altre categoria ja existent l'aplicació donarà error.

### **Esborrat de categories d'habilitats**

Si se prem sobre la opció d'esborrar, l'aplicació demanarà confirmació i si s'accepta, esborrarà la categoria del sistema si aquesta està "buïda", és a dir, no hi ha cap habilitat que hi pertanyi. En cas contrari, donarà error.

## Habilitats

Les habilitats s'administren des d'aquesta pàgina. Això inclou també els seus requisits.





Aquesta és la pantalla d'habilitats:



*Il·lustració 33: Pantalla de categories d'habilitats*

Les habilitats s'organitzen per categories i cada línia de la llista és una habilitat amb els següents camps:

Medicine  

- Nom de l'habilitat
- Edició de l'habilitat 
- Esborrat de l'habilitat 

A més hi ha un botó per crear habilitats.

## Creació d'habilitats

The screenshot shows a 'New ability' form with two main sections. The left section is for general data, and the right section is for required virtues and flows.

**New ability**

Name  \*

Infancy selectable ☐ \*

Description

\* : Required fields

**Required Virtues**

**Divine**

Selected <input type="checkbox"/> Negated <input type="checkbox"/>	True Faith	MAJOR
--	------------	-------

**General quirks**

Selected <input type="checkbox"/> Negated <input type="checkbox"/>	Fast healing	MINOR
Selected <input checked="" type="checkbox"/> Negated <input type="checkbox"/>	Warrior	MINOR
Selected <input type="checkbox"/> Negated <input type="checkbox"/>	Magister in Artibus	MAJOR

**Personality**

Selected <input type="checkbox"/> Negated <input type="checkbox"/>	Nocturnal	MINOR
--	-----------	-------

**Social status**

Selected <input type="checkbox"/> Negated <input type="checkbox"/>	Landed Noble	MAJOR
--	--------------	-------

**Supernatural**

Selected <input type="checkbox"/> Negated <input type="checkbox"/>	Faerie blood	MINOR
--	--------------	-------

**Required Flaws**

**Divine**

**General quirks**

Selected <input type="checkbox"/> Negated <input type="checkbox"/>	Blind	MAJOR
--	-------	-------

Il·lustració 34: Creació d'habilitats

La pantalla de creació consta de dues seccions: una de dades generals i una de requisits.

## Dades generals d'una categoria d'habilitats

The screenshot shows the 'New ability' form with the general data section highlighted.

**New ability**

Name  \*

Infancy selectable ☐ \*

Description

\* : Required fields

Il·lustració 35: Dades generals d'una habilitat

Aquesta secció consta dels següents camps:

- Nom de l'habilitat.
- Un checkbox anomenat "Infancy selectable".
- Un desplegable on triar la categoria d'habilitat.

- Una descripció(opcional) de l'habilitat.

La funció del checkbox és permetre que, a la secció d'habilitats de la pantalla de creació de personatges, sigui possible de posar punts d'experiència a la columna "Childhood".

Si se tria un nom d'habilitat idèntic al d'una altre habilitat ja existent l'aplicació donarà error.

## Requisits d'una habilitat

Aquesta secció és opcional ja que se poden crear habilitats sense requisits. Consta d'un llistat amb tots els quirks del sistema.

Required Virtues

Divine

Selected

Negated

True Faith

MAJOR

General quirks

Selected

Negated

Fast healing

MINOR

Selected

Negated

Warrior

MINOR

Selected

Negated

Magister in Artibus

MAJOR

Personality

Selected

Negated

Nocturnal

MINOR

Social status

Selected

Negated

Landed Noble

MAJOR

Supernatural

Selected

Negated

Faerie blood

MINOR

Cada línia és un quirk amb següents camps:

Selected

Negated

Warrior

MINOR

- Un checkbox marcat com "Selected"
- Un checkbox marcat com "Negated" i que normalment està desactivat.
- Nom del quirk
- Tipus del quirk

Per assignar requisits basta amb clicar els checkboxes "Selected" dels quirks que se vulguin com requisits. S'ha de tenir en compte que si hi ha més d'un requisit, bastarà amb que se'n satisfaci un per a que els requisits de l'habilitat se considerin satisfets.

La funció del checkbox "Negated" és invertir la relació entre ll'habilitat i el seu requisit. Si suposam una habilitat A que pren un quirk B com requisit:

Selected	Negated	Relació amb el quirk
Deseleccionat	Desactivat	B no és requisit de A
Seleccionat	Deseleccionat	Per poder posar punts a A és necessari haver seleccionat B
Seleccionat	Seleccionat	Per poder posar punts a A cal NO haver seleccionat B

*Taula 3: Requisits d'una habilitat*

Com s'ha explicat a la secció de categories d'habilitat (veure ), una habilitat hereta els requisits de la categoria a la que pertany però en pot tenir, a més, d'altres de diferents als de la categoria a la que pertany.

## Edició d'habilitats

La pantalla és la mateixa que a la creació tret que hi ha carregades les dades de l'habilitat a modificar, incloent-ne els requisits seus i de la categoria a la que pertany. Es possible modificar completament aquets requisits fins el punt que siguin completament diferents o canviar de categoria, cosa que implica perdre els requisits de la categoria sortint i guanyar els de l'entrant.

En acabar d'editar-la, l'aplicació repassa tots els personatges que tenen aquesta habilitat per comprovar si, legalment, encara la poden tenir. Si no és així, l'esborra de la fitxa i posa el personatge com "Pending Revision".

Si se tria un nom de categoria idèntic al d'una altre categoria ja existent l'aplicació donarà error.

## Esborrat d'habiliats

Si se prem sobre la opció d'esborrar, l'aplicació demanarà confirmació i si s'accepta, esborrarà l'habilitat del sistema.

## Pantalla de log

Quan se dona alguna de les operacions següents, l'aplicació inserta un log publicament accessible a tots els usuaris per informar-los del que ha passat.

- Esborrat i edició d'una categoria de quirks.
- Esborrat i edició d'un quirk.
- Esborrat i edició d'una categoria d'habilitats.
- Esborrat i edició d'una habilitat.

Aquest log se troba a la pantalla principal i és accessible tant al fer login com directament desde la barra de navegació.



El contingut de la pantalla de log té una longevitat de 10 dies i està ordenat data de creació.

Log

Date	Description
2015-01-10	Ability category modified: Ability category Martial
2015-01-07	Quirk updated: Minor Supernatural Flaw Faerie friend
2015-01-07	Quirk updated: Minor Supernatural Flaw Faerie friend
2015-01-07	Ability category modified: Ability category Martial
2015-01-07	Ability modified : Martial ability Siege weapons
2015-01-07	Ability modified : Academic ability Siege weapons
2015-01-07	Quirk category updated : Quirk category General quirks
2015-01-07	Ability modified : Martial ability Siege weapons
2015-01-07	Quirk deleted: Major Social status Virtue Magister in artibus
2015-01-07	Ability category modified: Ability category Academic
2015-01-07	Ability modified : Martial ability Siege weapons
2015-01-07	Ability modified : Martial ability Siege weapons

*Il·lustració 36: Pantalla de log*

## Milllores

Des de el principi, aquesta aplicació se va concebre com la base sobre la que construir un sistema integrat de gestió de personatges i campanyes. Només la primera part implica els següents canvis:

- Afegir equipament i les seves regles associades.
- Incloure altres tipus de personatges.
- Afegir el sistema de màgia.
- Permetre impressió de personatges, ja sigui a arxiu o paper.
- Inclure progrés del personatge en temps de joc.

Altres millores van referides a la mateixa aplicació:

- Determinar els avantatges de passar a Spring+Hibernate purs a les capes de persistència i negoci.
- Determinar els avantatges de refactoritzar la interfície per a que utilitzi un framework de presentació.

# Bibliografia

Nota prèvia: tota la informació necessària pel projecte ha estat recollida d'internet, incloent forums, pàgines web, tutorials, etc. Aquesta llista no és completa.

## Informació bàsica

<http://sevenset.org/2014/12/23/how-to-create-the-perfect-jpa-entity/>

<http://viralpatel.net/blogs/hibernate-self-join-annotation-mapping-many-to-many-example/>

[https://docs.jboss.org/hibernate/orm/4.0/devguide/en-US/html\\_single/](https://docs.jboss.org/hibernate/orm/4.0/devguide/en-US/html_single/)

<http://www.baeldung.com/spring-security-logout>

<http://krams915.blogspot.com.es/2012/01/spring-security-31-implement.html>

<http://www.baeldung.com/hibernate-4-spring>

<http://howtodoinjava.com/2013/04/16/custom-userdetailsservice-example-for-spring-3-security/>

<http://www.java2s.com/Code/Java/JPA/CatalogJPA.htm>

[http://en.wikibooks.org/wiki/Java\\_Persistence](http://en.wikibooks.org/wiki/Java_Persistence)

[javahash.com/spring-security-hello-world-example/](http://javahash.com/spring-security-hello-world-example/)

<http://stackoverflow.com/questions/8416021/spring-securitydatabase-authentication-provider>

<http://stackoverflow.com/questions/13102792/hibernate-update-with-entitymanager>

<http://web.archive.org/web/20110721191046/http://particletree.com/features/rediscovering-the-button-element/>

<http://stackoverflow.com/questions/20704656/javaconfiguration-for-spring-4-0-security-3-2-j-spring-security-check>

<http://www.baeldung.com/2011/12/13/the-persistence-layer-with-spring-3-1-and-jpa/>

## Errors i dubtes

<http://stackoverflow.com/questions/13818210/how-to-remove-relationship-between-many-to-many-table-with-jpa-and-hibernate>

<http://stackoverflow.com/questions/6049687/jquery-ui-dialog-box-need-to-return-value-when-user-presses-button-but-not-wor>

<http://stackoverflow.com/questions/8800855/create-session-stateless-usage>

<http://tomee.apache.org/examples-trunk/>

<http://webdesignerwall.com/tutorials/css3-dropdown-menu>

<http://stackoverflow.com/questions/2421339/how-to-inject-jpa-entitymanager-using-spring>

<http://stackoverflow.com/questions/7414794/difference-between-contextannotation-config-vs-contextcomponent-scan>

<http://stackoverflow.com/questions/16732607/role-name-in-association-relationship>

<http://www.uml-diagrams.org/association.html>

## Apèndix A: Resum del sistema de creació de personatges d'Ars Magica

Nota prèvia: Ars Magica inclou un sistema de màgia molt complet però, per desgràcia, també complex fins el punt que implementar-lo requeriria un projecte propi. Per això, el sistema implementat només permetrà la creació de personatges sense capacitats màgiques.

Dins Ars Màgica, un personatge està definit per tres conjunts d'atributs:

### Característiques físiques

Concretament, són *Intelligence*, *Perception*, *Presence*, *Communication*, *Strength*, *Stamina*, *Dexterity* i *Quickness* i representen com és físicament el personatge. Les característiques prenen valors numèrics entre -3 i +3, entenent 0 com el valor medi per a un humà.

Un personatge comença amb totes les seves característiques a 0 i el jugador rep 7 punts que pot gastar a la taula següent per a "comprar" característiques. També és possible de comprar valors negatius per a tenir més punts positius.

Valor de la característica	Cost
+3	-6
+2	-3
+1	-1
0	0
-1	+1
-2	+3
-3	+6

Taula 4: Valor de les característiques

### Habilitats

Les habilitats representen el coneixement que un personatge té a un determinat camp. Les habilitats s'agrupen en cinc categories : *General*, *Academic*, *Arcane*, *Supernatural* i *Martial*, algunes de les quals només són permeses durant la creació del personatge si s'han agafat determinades virtuts o defectes.

Totes les habilitats comencen amb un valor de 0 i fan servir la mateixa taula de progressió:

Cost	Puntuació
5	1
15	2
30	3
50	4
75	5
105	6
140	7
180	8
225	9
275	10

*Taula 5: Cost en xps dels nivells d'habilitat*

Nota: no hi ha un límit efectiu a la taula, simplement només ho he apuntat fins el nivell 10.

Les habilitats se compren amb punts d'experiència que, a la creació del personatge, són proporcionals a la seva edat, així, com més vell és el personatge, més experiència ha acumulat.

El període comprés entre el naixement i els primers 5 anys de vida se considera la infantesa del personatge, on rep 45 punts que només pot fer servir per comprar nivells a les habilitats *Area Lore*, *Athletics*, *Brawl*, *Charm*, *Folk Ken*, *Guile* i *Living Language*.

Per cada any posterior, el jugador rep 15 punts addicionals que pot emprar on desitgi.

Es perfectament possible afegir punts a una mateixa habilitat, primer a la infantesa i, posteriorment, a l'etapa adulta, en aquest cas se sumen els punts i se mira la taula per a treure el nivell final.

Se poden posar punts a una habilitat de forma que no basti per arribar a un nivell.

## **Virtuts i defectes**

Opcional ment, un personatge pot tenir virtuts i defectes que representen circumstàncies especials i capacitats extraordinàries. Les virtuts i els defectes també se classifiquen en categories: *Generals*, *Supernaturals*, *Personality* i *Social Status* i poden ser *Majors* o *Minors* depenent de lo poderoses que són.

No és obligatori d'agafar virtuts i defectes però si se'n volen agafar, és necessari d'equilibrar-ho amb el mateix nivell de l'altre tipus seguint aquestes equivalències:

Una virtut o defecte..	Equival a
Major	3 punts
Menor	1 punt

*Taula 6: Equivalència en punts dels quirks*

p.e. si un jugador agafa 4 punts de virtuts, haurà de triar 4 nivells de defectes per a compensar-ho.

Addicionalment, se pot agafar un màxim de 10 punts en virtuts o defectes.

Depenent de les virtuts/defectes triats l'usuari tindrà accés a determinats virtuts/defectes, categories d'habilitats i habilitats.