Universitat de Lleida
Escola Politècnica Superior
Enginyeria en Informàtica

Sistemes informàtics (Treball de final de carrera)

Aplicació web de generació i autoavaluació d'exàmens tipus test basat en el criptosistema knapsack

Autor: Xavier Fdez-Baldor Font

Directors: Joan Manel Gimeno Illa
Magda Valls Marsal

Setembre 2011

Als meus pares,
sóc qui sóc gràcies a ells.

A ma germà,
m'ha cuidat i recolzat
des de que vaig néixer.

Als meus tutors,
m'han ajudat enormement
quan els he necessitat.

I sobretot a la Lara,
què sempre ha estat al meu costat;
gràcies pel suport i ànims que m'has donat en tot moment.

Índex

Índex de figures

Índex de llistats

Índex de taules

Capítol 1

Introducció

En tot projecte o treball, sempre existeix una raó que justifiqui la realització d'aquest; una motivació que ens mostri la necessitat de crear un nou producte que veritablement sigui útil per a les noves generacions, i ens faci adonar dels motius que ens mouen a l'hora de realitzar-ho.

Durant la realització dels meus estudis d'Enginyeria en Informàtica a la Universitat de Lleida, vaig cursar les assignatures de *Protocols criptogràfics* i *Seguretat computacional*. Dins del contingut curricular d'aquestes vaig gaudir de les matemàtiques i la possibilitat que s'obria d'una nova visió de com aplicar-les a la informàtica, podent arribar a barrejar així les meves dues passions: les matemàtiques i la informàtica.

Dins de l'assignatura de *Protocols criptogràfics* vaig haver de desenvolupar un petit treball d'investigació. Entre el ventall de treballs oferts, vaig escollir el de *Exàmens de tipus test amb el criptosistema knapsack*, qüestió que posteriorment em va fer possible abordar el disseny i la implementació d'aquest projecte.

De fet, vaig escollir aquest treball per un motiu molt clar i entenedor: la meva professió actual és la de professor. Desenvolupo la tasca de professor de l'ESO en una escola de Lleida. En el moment de plantejar-me aquest treball vaig llegir l'article *Eina per al disseny i correcció d'exàmens de tipus test* [10], escrit per tres professors de la UPC. Article que, indubtablement, em va captivar.

Vaig escometre el treball de l'assignatura amb gran interès. El fet de poder crear exàmens tipus test i que la correcció d'aquests fos automàtica era una cosa que, com a professor, m'atreia i m'estimulava molt.

Quan vaig haver d'escollir un projecte final de carrera no ho vaig dubtar ni un segon, volia portar a terme una aplicació basat en el que havia après d'aquest treball. Vaig

proposar-ho a la Magda Valls, com a tutora de la part de criptografia, i al JM Gimeno, com a tutor de la part d'informàtica, atorgant-me el seu vist-i-plau.

Després de parlar molt amb ells, vam consensuar, els tres, els requeriments d'aquest projecte (Capítol 2).

No hi ha cap dubte que la memòria d'un projecte ha d'estar perfectament estructurada; analitzant primer els continguts i vertebrant els diversos apartats i seccions que aquesta haurà de tenir. Un cop copsat tot això, desenvolupar-los degudament.

A continuació s'elaborarà una breu explicació de tots els apartats que es poden trobar en aquesta memòria, amb la finalitat d'obtenir una idea de tots els continguts que formaran part. Tot això, abans d'endinsar-nos completament en la realització d'aquesta.

En primer lloc, al capítol 2, es presentaran els requeriments de l'aplicació; requeriments basats en l'aplicació del criptosistema *knapsack* per generar exàmens de tipus test on la finalitat és que siguin autoavaluables. Requeriments que, òbviament, són els que ens vam plantejar els tutors i jo mateix.

A continuació, capítol 3, es farà una introducció a la criptografia on es descriuran els fonaments matemàtics necessaris utilitzats en criptografia; així com l'evolució d'aquesta última. Tot seguit, dins el capítol 4, s'explicarà el criptosistema *knapsack* i com s'aplica per a la creació dels exàmens i de l'aplicació.

Posteriorment, al capítol 5, s'exposaran les tecnologies emprades i la justificació de la seva elecció, per donar pas a la descripció del disseny de l'aplicació, capítol 6.

Finalment, dins el capítol 7, s'exposarà a mode de conclusió final, la valoració del treball realitzat. Aquesta avaluació final ens ha de permetre valorar si hem arribat a assolir les nostres fites i objectius inicials, així com fer un balanç final i integrat de l'aplicació creada. Tot plegat, ens ha de permetre valorar, críticament, els punt forts i els punts febles de tot el projecte. S'inclourà, així mateix, un recull de possibles ampliacions per fer més completa l'aplicació, amb noves opcions i indicant les possibles mancances que ens hi podríem trobar.

No cal fer molt d'esment per observar que a la bibliografia s'inclouran tots els llibres, pàgines webs i documentació que s'ha consultat per la realització del projecte, fonts primàries i secundàries relacionades amb la temàtica de tots els recursos utilitzats.

Capítol 2

Requeriments

L'anàlisi de requeriments és el primer pas en el desenvolupament d'una aplicació. És l'etapa on s'intenta esbrinar quines són les necessitats del client que ha de satisfer l'aplicació.

A partir de la definició dels requeriments generals i dels usuaris del sistema es podran redactar els casos d'ús que servirà com a base indispensable per a la implementació. Una forma que tenim d'estructurar requeriments són els casos d'ús.

2.1 Requeriments generals del sistema

Es desitja crear una aplicació web que giri entorn a la creació d'exàmens tipus test autoavaluables, és a dir, que la correcció d'aquests sigui ràpida i el mateix alumne sigui capaç de realitzar-la. S'haurà de crear una versió diferent de cada examen, una per a cada alumne, on surtin les mateixes preguntes en ordre diferent i les possibles respostes també en ordre aleatori. Per aconseguir que sigui autoavaluable, al costat de cada opció no hi haurà lletres (a,b,c...), en aquest cas es ficarà un número. El resultat de l'examen serà la suma dels números que acompanyen a l'opció que l'estudiant escull com a correcta. Gràcies a aquest número es podrà saber, de forma ràpida i satisfactòria, el resultat de l'examen.

A cada versió de l'examen li correspondrà un número de versió. Gràcies a aquest número de versió i al número resultant de l'examen, l'aplicació haurà de ser capaç de: donar el resultat de l'examen, indicar la nota, i mostrar les respostes correctes i incorrectes seleccionades per l'alumne. Tant el professor com el mateix alumne podran accedir a l'aplicació per saber el resultat de l'examen.

L'aplicació haurà de ser capaç d'emmagatzemar preguntes de tipus test amb les corresponents possibles respostes, tot sabent, quina és la resposta correcta, és a dir, l'aplicació tindrà un banc de preguntes dins la base de dades. També és podrà

especificar si una pregunta ha d'aparèixer sempre en l'última posició, ja que, com s'ha especificat anteriorment, l'ordre de les respostes és aleatori. Els professors podran anar afegint les preguntes que vulguin sobre les assignatures que donen per poder recuperar-les en el moment desitjat a l'hora de crear un examen.

Quan el professor desitgi crear l'examen, té la possibilitat de realitzar dues opcions: escollir les preguntes que vulgui que surtin o fer que l'aplicació les agafi aleatòriament. També hi haurà la possibilitat de realitzar ambdues a la vegada, és a dir, escollir algunes preguntes a mà i la resta aleatòriament. En qualsevol cas, l'aplicació crearà una versió per a cada alumne matriculat a l'assignatura o tantes versions com indiqui el professor.

El professor haurà de tenir un historial de totes les notes. Haurà de poder consultar totes les notes de tots els exàmens realitzats a les seves assignatures. L'alumne també tindrà un historial sobre totes les notes, tant les assignatures que està cursant com les assignatures cursades en anys anteriors.

Finalment, haurà d'existir la figura de l'administrador, ja que s'ha de poder administrar els usuaris registrats, professors i alumnes, així com les assignatures existents.

Dins el capítol 3 es pot trobar una explicació molt més detallada del funcionament dels exàmens tipus test.

2.2 Usuaris de l'aplicació

Segons els requeriment hi ha dos tipus d'usuaris: el professor i l'alumne. Per qüestions tècniques d'infraestructura existeix un tercer usuari; l'administrador.

2.2.1 Usuari professor

Aquests usuaris són l'eix principal de l'aplicació.

Posseiran la capacitat de realitzar les següents accions:

- Inserir/consultar preguntes de les seves assignatures a la base de dades.
- Crear/consultar exàmens de les seves assignatures.
- Consultar/modificar notes així com obtenir la nota d'un examen, amb el sistema d'autoavaluació.
- Modificar la seva contrasenya.

2.2.2 Usuari alumne

Són els usuaris que tenen menys permisos.

Posseiran la capacitat de realitzar les següents accions:

- Consultar l'historial de notes.
- Obtenir la nota d'un nou examen realitzat, amb el sistema d'autoavaluació.
- Modificar la seva contrasenya.

2.2.3 Usuari administrador

Seran els usuaris que tindran un control sobre els aspectes administratius de l'aplicació.

Podran:

- Matricular alumnes a les assignatures.
- Consultar/afegir/esborrar/modificar professors i les assignatures que donen els professors.
- Consultar/afegir/esborrar/modificar alumnes i les assignatures que cursen els alumnes.
- Consultar/afegir/esborrar/modificar assignatures.

2.3 Casos d'ús

Un cas d'ús especifica una seqüència d'accions, incloent variants, que el sistema pot executar i que produeix un resultant observable de valor per a un particular actor.

El seu component més important és reunir en tots els casos d'ús els requeriments del sistema de software que s'està construint.

Un cas d'ús descriu *què* fa el sistema, no *com* ho fa.

A continuació es descriuran els casos d'ús:

2.3.1 Inserir una pregunta

Cas d'ús: Inserir una pregunta.

Objectius: Inserir una pregunta d'una assignatura a la base de dades.

Actors: Professor.

Tipus: Primari.

Precondicions: El professor s'ha autenticat correctament.

Postcondicions: La nova pregunta s'ha inserit amb èxit.

Flux principal d'esdeveniments:

1. El sistema mostra les assignatures que dóna el professor.
2. El professor indica l'assignatura a la que pertany la pregunta.
3. El sistema mostra el formulari corresponent.
4. El professor completa el formulari i l'envia.
5. El sistema emmagatzema la nova pregunta, mostra per pantalla un missatge d'èxit i mostra de nou el formulari per inserir una nova pregunta.

Flux alternatiu d'esdeveniments:

5.1. Apareix un error ja que el formulari no s'ha omplert correctament, és a dir, algun dels camps demanats s'han deixat en blanc o no s'ha escollit quina és la resposta correcta. S'indica per pantalla l'error produït.

5.1.1. Es torna al formulari.

2.3.2 Consultar una pregunta

Cas d'ús: Consultar una pregunta.

Objectius: Consultar una pregunta d'una assignatura.

Actors: Professor.

Tipus: Primari.

Precondicions: El professor s'ha autenticat correctament.

Postcondicions: La consulta s'ha realitzat amb èxit.

Flux principal d'esdeveniments:

1. El sistema mostra les assignatures que dona el professor.
2. El professor indica l'assignatura a consultar.
3. El sistema mostra totes les preguntes de l'assignatura.
4. El professor selecciona una pregunta.
5. El sistema desplega les possibles respostes d'aquesta pregunta.

Flux alternatiu d'esdeveniments:

4.1. El professor selecciona *Editar* una pregunta.
  4.1.1. El sistema mostra la pregunta i les respostes.
  4.1.2. El professor realitza les modificacions pertinents.
  4.1.3. El sistema emmagatzema les modificacions, mostra per pantalla un missatge d'èxit i mostra de nou la taula amb les preguntes.
4.2. El professor selecciona *Eliminar* una pregunta.
  4.2.1. El sistema mostra un missatge de confirmació.
    4.2.1.1. El professor confirma.
      4.2.1.1.1. El sistema elimina la pregunta, mostra per pantalla un missatge d'èxit i mostra de nou les preguntes.
    4.2.1.2. El professor cancel·la.
      4.2.1.2.1. El missatge es tanca i es continua visualitzant les preguntes.

2.3.3 Crear un examen

Cas d'ús: Crear un examen.

Objectius: Crear un examen d'una assignatura.

Actors: Professor.

Tipus: Primari.

Precondicions: El professor s'ha autenticat correctament.

Postcondicions: El nou examen ha estat creat amb èxit.

Flux principal d'esdeveniments:

1. El sistema mostra les assignatures que dona el professor.
2. El professor indica l'assignatura de l'examen a crear.
3. El sistema mostra el formulari corresponent.
4. El professor completa el formulari i l'envia.
5. El sistema crea, emmagatzema i mostra el nou examen.

Flux alternatiu d'esdeveniments:

5.1. Apareix un error ja que el formulari no s'ha omplert correctament, és a dir, alguns dels camps demanats s'han deixat en blanc o la data seleccionada és anterior a l'actual. S'indica per pantalla l'error produït.
5.1.1. Es torna al formulari.

2.3.4 Consultar un examen

Cas d'ús: Consultar un examen.

Objectius: Consultar un examen d'una assignatura.

Actors: Professor.

Tipus: Primari.

Precondicions: El professor s'ha autenticat correctament.

Postcondicions: La consulta s'ha realitzat amb èxit.

Flux principal d'esdeveniments:

1. El sistema mostra les assignatures que dona el professor.
2. El professor indica l'assignatura a consultar.
3. El sistema mostra tots els exàmens de l'assignatura.
4. El professor selecciona un examen.
5. El sistema mostra totes les versions creades d'aquest examen.

Flux alternatiu d'esdeveniments:

4.1. El professor selecciona *Eliminar* un examen.
4.1.1. El sistema mostra un missatge de confirmació.
4.1.1.1. El professor confirma.
4.1.1.1.1. El sistema elimina l'examen, mostra per pantalla un missatge d'èxit i mostra de nou els exàmens.
4.1.1.2. El professor cancel·la.
4.1.1.2.1. El missatge es tanca i es continua visualitzant els exàmens.

2.3.5 Consultar una nota

Cas d'ús: Consultar una nota.

Objectius: Consultar una nota d'un examen.

Actors: Professor.

Tipus: Primari.

Precondicions: El professor s'ha autenticat correctament.

Postcondicions: La consulta s'ha realitzat amb èxit.

Flux principal d'esdeveniments:

1. El sistema mostra les assignatures que dona el professor.
2. El professor indica l'assignatura a consultar.
3. El sistema mostra tots els exàmens de l'assignatura.
4. El professor indica un examen.
5. El sistema mostra totes les notes d'aquest examen.
6. El professor selecciona la nota d'un alumne.

7. El sistema desplega la resta de dades de la nota.

Flux alternatiu d'esdeveniments:

6.1. El professor selecciona *Eliminar* una nota.
6.1.1. El sistema mostra un missatge de confirmació.
6.1.1.1. El professor confirma.
6.1.1.1.1. El sistema elimina la nota, mostra per pantalla un missatge d'èxit i mostra de nou les notes.
6.1.1.2. El professor cancel·la.
6.1.1.2.1. El missatge es tanca i es continua visualitzant les notes.
6.2. El professor selecciona *Veure Examen*.
6.2.1. El sistema obre un arxiu pdf amb l'examen realitzat per l'alumne on es poden observar les respostes de l'alumne i les respostes correctes.

2.3.6 Canviar contrasenya

Cas d'ús: Canviar contrasenya.

Objectius: Canviar la contrasenya d'un usuari.

Actors: Professor i Alumne.

Tipus: Primari.

Precondicions: El professor o l'alumne s'ha autenticat correctament.

Postcondicions: El canvi de contrasenya s'ha realitzat amb èxit.

Flux principal d'esdeveniments:

1. El sistema mostra el formulari corresponent.
2. El professor o l'alumne completa el formulari i l'envia.
3. El sistema emmagatzema els nous canvis.

Flux alternatiu d'esdeveniments:

3.1. Apareix un error ja que el formulari no s'ha omplert correctament, és a dir, la contrasenya actual inserida no és correcta o les contrasenyes noves inserides no coincideixen. S'indica per pantalla l'error produït.
3.1.1. Es torna al formulari.

2.3.7 Obtenir la nota d'un examen

Cas d'ús: Obtenir la nota d'un examen.

Objectius: Saber la nota d'un examen just després de realitzar-lo.

Actors: Alumne.

Tipus: Primari.

Precondicions: L'alumne s'ha autenticat correctament.

Postcondicions: L'obtenció de la nota s'ha realitzat amb èxit.

Flux principal d'esdeveniments:

1. El sistema mostra totes les assignatures que està o ha estat matriculat l'alumne.
2. L'alumne indica l'assignatura corresponent.
3. El sistema desplega tots els exàmens realitzats de l'assignatura.

4. L'alumne selecciona *Afegir nota.*
5. El sistema mostra inputs en blanc.
6. L'alumne insereix el número de versió de l'examen que ha realitzat i el número resultant.
7. El sistema mostra a l'alumne la nota que ha tret de l'examen.

Flux alternatiu d'esdeveniments:

7.1. Apareix un error ja que un dels valors inserits no es correcte, és a dir, la versió inserida no existeix o no es corresponen la versió i el número resultant. S'indica per pantalla l'error produït.
7.1.1. Es torna al formulari.

2.3.8 Donar d'alta a un alumne

Cas d'ús: Donar d'alta a un alumne.

Objectius: Donar d'alta a un alumne a l'aplicació.

Actors: Administrador.

Tipus: Primari.

Precondicions: L'administrador s'ha autenticat correctament.

Postcondicions: L'alta de l'alumne s'ha realitzat amb èxit.

Flux principal d'esdeveniments:

1. El sistema mostra el formulari pertinent.
2. L'administrador escull l'arxiu .css on es troba el llistat dels alumnes nous.
3. El sistema dona d'alta als alumnes, emmagatzema les dades a la base de dades i els hi envia un e-mail de confirmació.

Flux alternatiu d'esdeveniments:

3.1. Apareix un error ja que el formulari no s'ha omplert correctament, és a dir, l'arxiu seleccionat no es correcte. S'indica per pantalla l'error produït.
3.1.1. Es torna al formulari.

2.3.9 Consultar un professor

Cas d'ús: Consultar un professor.

Objectius: Consultar les dades d'un professor .

Actors: Administrador.

Tipus: Primari.

Precondicions: L'administrador s'ha autenticat correctament.

Postcondicions: La consulta s'ha realitzat amb èxit.

Flux principal d'esdeveniments:

1. El sistema mostra algunes dades dels professors.
2. L'administrador indica un professor.
3. El sistema desplega la resta de dades del professor.

Flux alternatiu d'esdeveniments:

2.1. L'administrador selecciona *Editar* un professor.

2.1.1. El sistema mostra inputs amb les dades del professor.
2.1.1.1. L'administrador realitza les modificacions pertinents.
2.1.1.1.1. El sistema emmagatzema les modificacions, mostra per pantalla un missatge d'èxit i mostra de nou els professors.
2.2. L'administrador selecciona *Eliminar* un professor.
2.2.1. El sistema mostra un missatge de confirmació.
2.2.1.1. L'administrador confirma.
2.2.1.1.1. El sistema elimina el professor, mostra per pantalla un missatge d'èxit i mostra de nou els professors.
2.2.1.2. L'administrador cancel·la.
2.2.1.2.1. El missatge es tanca i es continua visualitzant els professors.

2.3.10 Consultar un alumne

Cas d'ús: Consultar un alumne.

Objectius: Consultar les dades d'un alumne .

Actors: Administrador.

Tipus: Primari.

Precondicions: L'administrador s'ha autenticat correctament.

Postcondicions: La consulta s'ha realitzat amb èxit.

Flux principal d'esdeveniments:

1. El sistema mostra algunes dades dels alumnes.
2. L'administrador indica un alumne.
3. El sistema desplega la resta de dades de l'alumne.

Flux alternatiu d'esdeveniments:

2.1. L'administrador selecciona *Editar* un alumne.
2.1.1. El sistema mostra inputs amb les dades de l'alumne.
2.1.1.1. L'administrador realitza les modificacions pertinents.
2.1.1.1.1. El sistema emmagatzema les modificacions, mostra per pantalla un missatge d'èxit i mostra de nou els alumnes.
2.2. L'administrador selecciona *Eliminar* un alumne.
2.2.1. El sistema mostra un missatge de confirmació.
2.2.1.1. L'administrador confirma.
2.2.1.1.1. El sistema elimina l'alumne, mostra per pantalla un missatge d'èxit i mostra de nou els alumnes.
2.2.1.2. L'administrador cancel·la.
2.2.1.2.1. El missatge es tanca i es continua visualitzant els alumnes.

2.3.11 Consultar una assignatura

Cas d'ús: Consultar una assignatura.

Objectius: Consultar les assignatures existents.

Actors: Administrador.

Tipus: Primari.

Precondicions: L'administrador s'ha autenticat correctament.

Postcondicions: La consulta s'ha realitzat amb èxit.

Flux principal d'esdeveniments:

1. El sistema mostra les assignatures.

Flux alternatiu d'esdeveniments:

1.1. L'administrador selecciona *Editar* una assignatura.
1.1.1. El sistema mostra inputs amb les dades de l'assignatura.
1.1.1.1. L'administrador realitza les modificacions pertinents.
1.1.1.1.1. El sistema emmagatzema les modificacions, mostra per pantalla un missatge d'èxit i mostra de nou les assignatures.
1.2. L'administrador selecciona *Eliminar* una assignatura.
1.2.1. El sistema mostra un missatge de confirmació.
1.2.1.1. L'administrador confirma.
1.2.1.1.1. El sistema elimina l'assignatura, mostra per pantalla un missatge d'èxit i mostra de nou les assignatures.
1.2.1.2. L'administrador cancel·la.
1.2.1.2.1. El missatge es tanca i es continua visualitzant les assignatures.

Capítol 3

Introducció a la criptografia

L'objectiu d'aquest capítol és el d'introduir els principals conceptes matemàtics que han estat necessaris per desenvolupar el projecte. En cap cas es pretén explicar de forma rigorosa i complexa cadascun dels conceptes però sí fer factible la seva utilització com a petita guia o referència per a totes aquelles persones que es vulguin endinsar en aquest món, o bé, per afavorir que una persona no iniciada en aquest camp pugui assolir el grau suficient per a comprendre el projecte desenvolupat.
En cadascun dels apartats del capítol es procura donar una explicació general, basada en els conceptes que s'utilitzen en el projecte, acompanyat d'un exemple senzill d'utilització.
En aquest capítol es tracta l'aritmètica modular, i en concret, aquells teoremes que són necessaris per poder entendre els criptosistemes de clau pública i el criptosistema knapsack, eix principal d'aquest projecte. També es veu la criptografia, des del seus orígens fins a la criptografia actual, tot explicant breument els criptosistemes més destacats.

3.1 Aritmètica modular

Molts dels criptosistemes emprats en l'actualitat utilitzen eines d'aritmètica modular per xifrar i desxifrar. En aquesta secció es mostrarà una petita introducció a alguns conceptes bàsics.

3.1.1 Divisió entera

El concepte de divisió entera es remunta fins a l'antiguitat i el següent teorema es demostra fàcilment.

Teorema.- Donats $a, b \in \mathbb{Z}, \mathrm{b} \neq 0$, existeixen dos únics enters q i r (anomenats, respectivament, quocient i resta de la divisió entera de a per b), que compleixen $a = b \cdot q + r, 0 \leq r < |b|$.

Exemple

Es trien 2 enters qualsevol a i b, per exemple $a = 13$ i $b = 3$. La divisió entera de 13 i 3 consisteix a fer $13 = 3 \cdot 4 + 1$. Apareixen dos enters "nous", el 4, anomenat quocient, i l'1, anomenat resta, o residu, de la divisió. S'observa que la resta, 1, és menor que $b = 3$.

3.1.2 Algorisme d'Euclides

Una conseqüència immediata del teorema de la divisió entera és que pot esser emprat a l'algorisme d'Euclides, que ens permet calcular de forma eficient el mcd de dos nombres. Aquest teorema es basa en el fet que $mcd(a, b) = mcd(b, r)$, sent r el residu de la divisió entre a i b.

Teorema.- Donats $a, b \in \mathbb{Z}, \mathrm{b} \neq 0$, es pot aplicar reiteradament el teorema de la divisió entera, fins que $r = 0$, de la següent manera:

$$\left.\begin{array}{rr} a = b \cdot q_1 + r_1, & r_1 < |b| \\ b = r_1 \cdot q_2 + r_2, & r_2 < r_1 \\ \vdots & \\ r_{n-2} = r_{n-1} \cdot q_n + r_n, & r_n < r_{n-1} \\ r_{n-1} = r_n \cdot q_{n+1} + r_{n+1}, & r_{n+1} = 0 \end{array}\right\} m.c.d(a, b) = r_n$$

Exemple

Es trien 2 enters qualssevol a i b, per exemple $a = 88$ i $b = 26$, i s'aplica l'algoritme d'Euclides, fent successives divisions enteres:

$$88 = 26 \cdot 3 + 10$$
$$26 = 10 \cdot 2 + 6$$
$$10 = 6 \cdot 1 + 4$$
$$6 = 4 \cdot 1 + 2$$
$$4 = 2 \cdot 2 + 0$$

Es pot afirmar que el $m.c.d.(88{,}26) = 2$

3.1.3 Identitat de Bézout

La identitat de Bézout relaciona dos nombres amb el seu màxim comú divisor de la forma següent:

Teorema.- Si $a, b \in \mathbb{Z}$ i d és el màxim comú divisor de a i b, existeixen enters r i s de manera que $d = a \cdot r + b \cdot s$.

L'algorisme que donats a i b permet retornar d, r i s s'anomena Algorisme d'Euclides Extès. Consisteix en resseguir en el sentit invers les equacions obtingudes a partir de les múltiples divisions enteres efectuades, és a dir, es fa Euclides i després es desfà.

Exemple

Continuant amb l'exemple anterior, $a = 88$ i $b = 26$, s'ha fet Euclides, ara es desfà, començant per la darrera igualtat en que el reste no és zero:

$$10 = 6 \cdot 1 + 4 \rightarrow 4 = 10 - 6 \cdot 1$$

$$2 = 6 - 4 \cdot 1 = 6 - \overbrace{(10 - 6 \cdot 1)} \cdot 1 = 6 - 10 \cdot 1 + 6 \cdot 1 = 6 \cdot 2 - 10 \cdot 1 =$$
$$= (26 - 10 \cdot 2) \cdot 2 - 10 \cdot 1 = 26 \cdot 2 - 10 \cdot 4 - 10 \cdot 1 = 26 \cdot 2 - 10 \cdot 5 =$$
$$= 26 \cdot 2 - (88 - 26 \cdot 3) \cdot 5 = 26 \cdot 2 - 88 \cdot 5 + 26 \cdot 15 = 26 \cdot 17 - 88 \cdot 5$$

$$2 = 88 \cdot (-5) + 26 \cdot 17, on\ r = -5\ i\ s = 17$$

3.1.4 Nombres primers

Qualsevol nombre enter p és divisible per ± 1 i $\pm p$. Si aquests són els seus únics divisors, direm que p és primer.

Exemple

El 13 és primer ja que només es pot dividir entre ell i entre 1.

Donat un enter qualsevol, es pot descompondre de manera única com a producte de nombres primers.

Exemple

$15 = 5 \cdot 3 \cdot 1$

Direm que dos enters a i b són primers entre ells si no tenen cap divisor comú (tret, és clar, de l'1), és a dir, $m.c.d.(a, b) = 1$.

Exemple

Donats $a = 9$ i $b = 4$, $mcd(9,4) = 1$, llavors a i b són primers entre ells.

3.1.5 Congruències. Els conjunts $\mathbb{Z}_m$

Fixem un enter $m > 0$. Direm que dos enters a i b són congruents mòdul m si, quan dividim a i b entre m, obtenim la mateixa resta en un i altre cas. Per indicar que a i b són congruents mòdul m escriurem $a \equiv b(mod\ m)$

Es trien 2 enters qualssevol a i b, per exemple $a = 7$ i $b = 15$, i $m = 4$.

$$\left.\begin{matrix} 7 = 1 \cdot 4 + 3 \\ 15 = 3 \cdot 4 + 3 \end{matrix}\right\} r = 3, llavors\ 7 \equiv 15(mod\ 4)$$

Des d'aquest punt de vista, definim el conjunt de residus d'ordre m simplement com el conjunt dels enters entre 0 i $m - 1$: $\mathbb{Z}_m = \{0,1,..,m-1\}$ amb un parell d'operacions, suma i producte (modulars), definides a $\mathbb{Z}_m$: si $a, b \in \mathbb{Z}_m$, definim la seva suma com $a + b\ (mod\ m)$ i el seu producte com $a \cdot b(mod\ m)$. D'aquesta manera, la suma i el producte de dos elements de $\mathbb{Z}_m$ és també un element de $\mathbb{Z}_m$. Per exemple, si es prenen

$2,5 \in \mathbb{Z}_7$, es té $2+5=7\equiv 0(mod\ 7)$ i $2\cdot 5=10\equiv 3(mod\ 7)$. Per tant, pensant a $\mathbb{Z}_7$, es pot dir que "$2+5=0$" i que "$2\cdot 5=3$".

3.1.6 Inversos modulars

Un element $a\in\mathbb{Z}_m$ diem que és invertible si existeix un altre element $b\in\mathbb{Z}_m$, tal que $a\cdot b\equiv 1(mod\ m)$. Si aquest existeix el denotarem $a^{-1}(mod\ m)$.

Per exemple, s'ha de mirar si 3 té invers a $\mathbb{Z}_{10}$. Es pot multiplicar successivament 3 per 0,1,..,9 i comprovar si algun d'aquests productes dóna un resultat congruent amb 1 (mod 10):

$$3\cdot 0\equiv 0;\quad 3\cdot 1\equiv 3;\quad 3\cdot 2\equiv 6;\quad 3\cdot 3\equiv 9;\quad 3\cdot 4\equiv 12\equiv 2;$$
$$3\cdot 5\equiv 15\equiv 5;\quad 3\cdot 6\equiv 18\equiv 8;\quad 3\cdot 7\equiv 21\equiv 1\ (mod\ 10)$$

S'ha trobat que 3 té un invers a $\mathbb{Z}_{10}$: $3^{-1}(mod\ 10)=7$. Si s'intenta trobar l'invers de 6 a $\mathbb{Z}_{10}$, després de fer tots els càlculs es veurà que no existeix cap invers de 6.

És fàcil saber si un element és invertible? El teorema següent ens dona la resposta.

Teorema.- Un element a té un invers a $\mathbb{Z}_m$ si i només si $m.c.d.(a,m)=1$.
En particular, s'observa que, quan m és un nombre primer, qualsevol enter $a\in\mathbb{Z}_m$ tindrà un invers a $\mathbb{Z}_m$.

Calcular l'invers modular comprovant tots els productes és inviable quan es tracta de números grans. Se sap que un enter a té invers a $\mathbb{Z}_m$ quan $m.c.d.(a,m)=1$. Aleshores, per la identitat de Bézout se sap que $1=r\cdot a+s\cdot m$. Si considerem aquesta igualtat a $\mathbb{Z}_m$ se sap que $s\cdot m\equiv 0(mod\ m)$, llavors $1\equiv r\cdot a\ (mod\ m)$. Per tant, l'invers que es busca és la constant r de la identitat de Bézout.

Es busca l'invers modular de 7438 en $\mathbb{Z}_{79533}$ fent algoritme d'Euclides Extés:

$$79533=7438\cdot 10+5153$$
$$7438=5153\cdot 1+2285$$
$$5153=2285\cdot 2+583$$
$$2285=583\cdot 3+536$$
$$583=536\cdot 1+47$$
$$536=47\cdot 11+19$$
$$47=19\cdot 2+9$$
$$19=9\cdot 2+1$$

Com que $r=1$, se sap que $mcd(79533,7438)=1$, llavors ∃ invers, es continua,

$$1=19-9\cdot 2=19-(47-19\cdot 2)\cdot 2=19-47\cdot 2+19\cdot 4=19\cdot 5-47\cdot 2=$$
$$=(536-47\cdot 11)\cdot 5-47\cdot 2=536\cdot 5-47\cdot 55-47\cdot 2=536\cdot 5-47\cdot 57=$$
$$=536\cdot 5-(583-536\cdot 1)\cdot 57=536\cdot 5-583\cdot 57+536\cdot 57=536\cdot 62-583\cdot 57=$$
$$=(2285-583\cdot 3)\cdot 62-583\cdot 57=2285\cdot 62-583\cdot 186-583\cdot 57=2285\cdot 62-583\cdot 243=$$
$$=2285\cdot 62-(5153-2285\cdot 2)\cdot 243=2285\cdot 62-5153\cdot 243+2285\cdot 486=$$
$$=2285\cdot 548-5153\cdot 243=(7438-5153\cdot 1)\cdot 548-5153\cdot 243=$$
$$=7438\cdot 548-5153\cdot 548-5153\cdot 243=7438\cdot 548-5153\cdot 791=$$
$$=7438\cdot 548-(79533-7438\cdot 10)\cdot 791=7438\cdot 548-79533\cdot 791+7438\cdot 7910=$$
$$=7438\cdot 8458-79533\cdot 791$$

Per lo tant, $r=8458$ que és l'invers de 7438 en $\mathbb{Z}_{79533}$.

3.1.7 La funció φ d'Euler

Teorema.- Si n és un enter positiu, definim la funció phi d'Euler de n, $\varphi(n)$, com el nombre d'enters entre 1 i n-1 que són coprimers amb n, i ho definim com:

$$\varphi(n) = |\{m \in \mathbb{N} | m < n \wedge mcd(n,m) = 1\}|$$

La funció φ d'Euler satisfà les propietats següents:

1. Si p és primer, $\varphi(p) = p - 1$.
2. Si $m.c.d.(m,n) = 1$, llavors $\varphi(m \cdot n) = \varphi(m) \cdot \varphi(n)$.
3. Si $m.c.d.(a,m) = 1$, llavors $a^{\varphi(m)} \equiv 1\ (mod\ m)$.

3.1.8 Teorema d'Euler

És una generalització del petit teorema de Fermat. Aquest teorema permet simplificar el càlcul de les potències de mòdul n.

Teorema.- Sigui n un nombre natural i a un enter primer amb n, llavors:

$$a^{\varphi(n)} \equiv 1\ (mod\ n)$$

Exemple
Es vol trobar el valor de 7^{222} mòdul 10. Es pot veure que 7 i 10 són primers entre ells. Segons la funció d'Euler (3.1.7), $\varphi(10) = \varphi(5 \cdot 2) = \varphi(5) \cdot \varphi(2) = 4 \cdot 1 = 4$. Per tant el teorema d'Euler indica que:

$$7^4 \equiv 1\ (mod\ 10)$$

S'en dedueix que

$$7^{222} \equiv 7^{4 \cdot 55+2} \equiv (7^4)^{55} \cdot 7^2 \equiv 1^{55} \cdot 7^2 \equiv 1 \cdot 49 \equiv 49 \equiv 9\ (mod\ 10)$$

3.1.9 Algorisme d'exponenciació ràpida

Proporciona una manera de calcular de forma ràpida grans potencies i es basa en la conversió binaria de l'exponent.

Donada l'operació a^b, representar el número b en binari, $b = b_k, \dots, b_0$, sent k el número de bits que representen el valor b en binari. Començar per $x_{k+1} = 1$ i calcular $x_{i-1} = x_i^2$, sent $i = k + 1, \dots, 0$. Si $b_i = 1$, fer $x_i = x_i \cdot a$

Exemple
Es suposa que volem calcular 3^{17}. Es converteix l'exponent 17 a binari:

$$17_{10} = 10001_2 = b_4 b_3 b_2 b_1 b_0.$$

A continuació, es parteix de $x = 1$. A cada pas, s'ha de fer x^2 i si el bit és igual a 1, es multiplica per la base.

$$\begin{array}{ll} b_4 = 1 & x = 1^2 \cdot 3 = 3 \\ b_3 = 0 & x = 3^2 = 9 \\ b_2 = 0 & x = 9^2 = 81 \\ b_1 = 0 & x = 81^2 = 6561 \\ b_0 = 1 & x = 6561^2 \cdot 3 = 129140163 \end{array}$$

Ja s'ha calculat el resultat i ens s'ha estalviant moltes operacions, ja que en lloc de fer 17 multiplicacions només s'ha realitzat 7.

3.1.10 Exponenciació modular

En molts criptosistemes de clau pública es necessita efectuar exponenciacions modulars, tant en el xifrat com en el desxifrat. Habitualment, tant les bases com els exponents d'aquestes expressions són nombres molt grans i efectuar l'exponenciació a base de fer multiplicacions successives seria inviable.

Per a calcular aquestes potencies es pot utilitzar del teorema d'Euler (apartat 3.1.7) i l'algorisme d'exponenciació ràpida (apartat 3.1.8).

Si es vol calcular a^s mòdul n els passos a seguir són:

1. En cas que s sigui més gran que $\varphi(n)$, reduir l'exponent utilitzant el Teorema d'Euler.
2. Utilitzar l'algorisme d'exponenciació ràpida amb el nou exponent.

Suposem que es vol calcular 3^{2016} a $\mathbb{Z}_{235}$:

1. Se sap que $\varphi(235) = \varphi(5 \cdot 47) = 4 \cdot 46 = 184$.
   Com que $m.c.d.(3{,}235) = 1$, pel Teorema d'Euler se sap que $3^{184} \equiv 1\ (mod\ 235)$
   Per tant,
   $$3^{2016} \equiv 3^{184 \cdot 11 + 82} \equiv (3^{184})^{11} \cdot 3^{82} \equiv 3^{82}\ (mod\ 235)$$

2. $82_{10} = 1010010_2 = b_6 b_5 b_4 b_3 b_2 b_1 b_0$
   Per tant,
   $$\begin{array}{ll} b_6 = 1 & x = 1^2 \cdot 3 = 3\ (mod\ 235) = 3 \\ b_5 = 0 & x = 3^2 = 9\ (mod\ 235) = 9 \\ b_4 = 1 & x = 9^2 \cdot 3 = 243\ (mod\ 235) = 8 \\ b_3 = 0 & x = 8^2 = 64\ (mod\ 235) = 64 \\ b_2 = 0 & x = 64^2 = 4096\ (mod\ 235) = 101 \\ b_1 = 1 & x = 101^2 \cdot 3 = 30603\ (mod\ 235) = 53 \\ b_0 = 0 & x = 53^2 = 2809\ (mod\ 235) = 224 \end{array}$$

Es pot concloure que 3^{2016} a $\mathbb{Z}_{235}$ = 224

3.1.11 NP-complet

Els problemes polinomials (P) són aquells que es poden resoldre en temps polinomial.

NP és l'acrònim en anglès de *nondeterministic polynomial time* (temps polinomial no determinista), i són el conjunt de problemes els quals no s'ha trobat cap algoritme que el resolgui en temps polinomial, però sí que es pot comprovar en temps polinomial si una determinada instància és solució.

Si existís una solució polinomial per a un problema NP-complet, tots els problemes de NP tindrien també una solució en temps polinomial. Si es demostres que un problema NP-complet no es pugues resoldre en temps polinomial, la resta dels problemes NP-complets tampoc es podrien resoldre en temps polinomial, i a l'inrevés. Però justament saber si les classes P i NP són o no iguals constitueix el problema no resolt més important de les ciències de la computació en l'actualitat.

La Figura 3.1 mostra la relació entre els problemes P, NP i NP-complet.

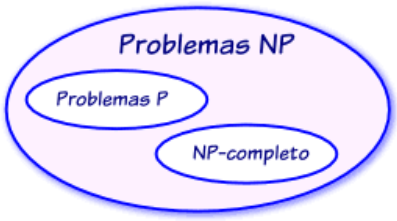


Figura 3.1: Diagrama problemes NP

3.2 Criptografia

En aquesta secció es redactaran els conceptes bàsics sobre criptografia i s'analitzaran alguns dels criptosistemes més destacats. Primerament, s'especificarà una nomenclatura estàndard que s'utilitzarà al llarg d'aquest capítol i posteriors.

3.2.1 Nomenclatura

S'enten com a "missatge" (m) un conjunt de dades de qualsevol tipus (no necessàriament un text escrit, típicament serà un número). Es parla de *text inicial* o *text en clar* (P) quan es refereix al missatge que es desitja transmetre, i de *text xifrat* o *criptograma* (C) quan es fa referència al missatge transformat que finalment es transmet. *Xifrar* (E) és el procés de convertir el text inicial en un text xifrat. El procés invers s'anomena *desxifrar* (D).

El conjunt de caràcters amb els quals s'escriu el text s'anomena *alfabet* (A) (e.g., es pot utilitzar les lletres de la A a la Z, numerals, l'espai en blanc o qualsevol altre símbol que vulguem permetre).

Es refereix a emissor a la persona que envia el missatge, i a receptor com la persona que és la destinatària d'aquest missatge. Canal és el mitjà de transmissió del missatge o les claus secretes, pot ser en mà, carta, Internet,... i es parla de canal segur quan es pot afirmar, amb tota seguretat, que el missatge entre l'emissor i el receptor no ha estat interferit. Aquest canal segur s'utilitza normalment per compartir la clau secreta, normalment a mà o amb correus personals.

El fet de xifrar i desxifrar és una aplicació bijectiva, és a dir, a un text xifrat només li correspon una única possible interpretació, el que és el mateix, un text desxifrat. Si dos textos inicials diferents x i y es convertissin en un mateix text z i rebéssim el text xifrat z, no seríem capaços de decidir amb seguretat de quin text inicial prové z.

$$P \xrightarrow{f} C \xrightarrow{f^{-1}} P$$

S'anomena *criptoanàlisi* el fet de vulnerar la seguretat d'un criptosistema. Un atac al sistema pot ser de diferents tipus, segons el nivell d'informació de què disposi l'atacant: pot interceptar un fragment de missatge xifrat, o bé pot disposar també d'un petit text en clar acompanyat del seu corresponent text xifrat, ... Aquest dos exemples requereixen un atac físic al canal de transmissió. La criptografia no s'ocupa de la seguretat física d'aquest canal. Es parteix de la hipòtesi que sempre hi haurà algú capaç d'interceptar text xifrat, i cal assegurar-se que a aquest individu li serà molt difícil deduir el procediment de xifratge disposant només d'aquesta informació.

3.2.2 Conceptes bàsics

Tal com indica la seva etimologia, criptografia consisteix simplement en escriure en clau (*crypto*: secret i *grafos*: escriptura). Aquest va ser en principi l'únic objectiu de la denominada criptografia clàssica, obtenir la confidencialitat dels missatges. Per aconseguir això, aquests missatges es transmetien xifrats amb una clau secreta, i el receptor dels mateixos podia desxifrar-los utilitzant la mateixa clau secreta amb el qual l'emissor els va xifrar. Per aquesta raó, ha estat anomenada *criptografia simètrica o de clau secreta*, ja que els dos comunicants haurien de posseir la mateixa clau (secreta) compartida per xifrar i desxifrar. En aquestes condicions, un eventual atacant desconeixedor de la clau secreta hauria de fer ús de tècniques de criptoanàlisi per intentar descobrir-la i així poder desxifrar el missatge.

Les claus secretes utilitzades a la criptografia clàssica havien de ser transmeses a través de canals segurs, generalment, correus personals. Per aquesta raó, els comunicants podien canviar les claus secretes compartides sempre i quan poguessin disposar d'emissaris fiables per transportar-les, i sembla que això no era massa freqüent com mostra la nombrosa documentació sobre labors d'espionatge recollida a la literatura històrica. A més, si el correu personal no tenia accés al receptor, aquest es quedava incomunicat. La criptografia clàssica era una ciència, o més ben dit, un art secret i gairebé exclusiu dels àmbits oficials dels exercits i cossos diplomàtics. La seva utilització en altres àmbits era també secreta, ja que es practicava al comerç d'alt nivell normalment associat amb les classes governants, així com al món de la màgia i l'alquímia, amb l'objecte de transmetre coneixements que permetessin exercir poder per sobre dels no iniciats.

Aquest tipus de criptografia va començar a emprar-se a l'antiguitat i va ser igualment utilitzada als temps anteriors a la Segona Guerra Mundial, en els quals el criptoanàlisi encara era lent i laboriós. En la majoria de casos, quan es descobria la clau i es desxifrava el missatge, aquest havia perdut gran part del seu valor. Durant la guerra es continuà utilitzant la criptografia simètrica i els anglesos van desenvolupar en secret el primer ordinador electrònic, el *Colossus*[1], per desxifrar els missatges de l'enemic. El progressiu augment de la capacitat dels ordinadors, així com el considerable increment de la seva velocitat van disminuir el temps necessari per dur a terme la violació d'aquests sistemes de xifrat simètric, això va exigir el desenvolupament de noves

[1] L'ordinador Colossus fou construït pels anglesos per desxifrar els missatges xifrats per les màquines alemanyes ENIGMA. La importància política i estratègica d'aquest fet va fer que es mantingués en secret fins l'any 1976. Aquesta es la raó de que la computadora ENIAC, desenvolupada pels EUA a la Universitat de Pennsylvania al 1946, fos considerat durant molts anys el primer ordinador electrònic del mon.

tècniques amb nivells de seguretat tals que el cost del seu criptoanàlisi fos superior al valor intrínsec del document eventualment recuperat. Per aquest motiu fou necessari dissenyar eines criptogràfiques modernes capaces de resistir atacs criptoanalitics. En l'actualitat disposem de criptosistemes simètrics robustos, com l'AES, i també s'han desenvolupat tècniques de criptografia de clau pública, com RSA, ElGamal o el mateix knapsack. En la criptografia de clau pública o asimètrica cada persona posseeix dues claus, una secreta i una altra pública. La clau secreta és totalment personal i s'evita el problema de compartir-la mitjançant un canal segur.

3.2.3 Criptografia simètrica clàssica

En aquesta secció s'exposaran els criptosistemes clàssics més destacats al llarg de la història.

Els criptosistemes clàssics es poden classificar segons el procés de xifratge que empren entre:

- Criptosistemes de transposició.
- Criptosistemes de substitució.

3.2.3.1 Criptosistemes de transposició

Consisteix en alterar l'ordre dels caràcters dins el missatge a xifrar, seguint una certa permutació que és la clau.

Hi ha diferents mètodes:
Un mètode possible és escriure el missatge en dues línies intercalant una caràcter a cada línia, posteriorment, s'agafen les línies i s'escriuen una darrera l'altra[1].

Exemple

Text en clar: A L'ALBA ATACAREM LA TORRE DOS
Intercalat de símbols:

A	A	B	A	A	A	E	L	T	R	E	O
L	L	A	T	C	R	M	A	O	R	D	S

Text xifrat: AABAAAELTREOLLATCRMAORDS

Un altre mètode consisteix en dividir el missatge amb blocs de ***n*** lletres (xifrat amb bloc) i mitjançant una clau fer permutacions dins cada bloc[2].

Exemple

Text en clar: A L'ALBA ATACAREM LA TORRE DOS
Clau $k = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 6 & 3 & 1 & 5 & 2 & 4 \end{pmatrix}$
Es divideix el text en clar en blocs de sis caràcters, marcat per la clau:

$$m_1 = "ALALBA"$$
$$m_2 = "ATACAR"$$
$$m_3 = "EMLATO"$$
$$m_4 = "RREDOS"$$

S'aplica a cada bloc la permutació marcada per la clau:

$$k(m_1) = "ABLDLA"$$
$$k(m_2) = "AATRCA"$$
$$k(m_3) = "LTMOAE"$$
$$k(m_4) = "EORSDR"$$

Text xifrat: ABLDLAAATRCALTMOAEEORSDR
Quant el text xifrat arribi al receptor, només ha d'executar el mateix procediment que l'emissor però utilitzant la permutació inversa:

$$k^{-1} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 5 & 2 & 6 & 4 & 1 \end{pmatrix}$$

Scytale
Està considerat com el primer aparell criptogràfic de la història. El van utilitzar els Grecs al S.V a.C. per encobrir missatges escrits en tires pe papir.

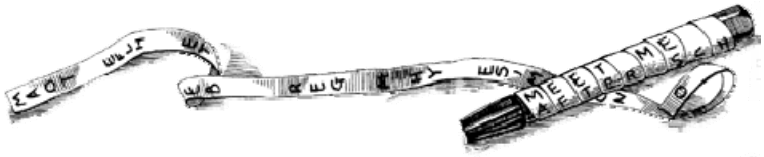


Figura 3.2: Scytale (Imatge extreta d'Internet)

La tècnica consistia en enrotllar una tira de papir al voltant d'un cilindre anomenat Scytale (Veure Figura 3.2). Una vegada enrotllat, s'escrivia el missatge. Al treure el papir del cilindre només es podien visualitzar lletres sense cap sentit. Per desxifrar el missatge l'emissor havia d'enrotllar el papir a un altre Scytale de mateix diàmetre.

3.2.3.2 Criptosistemes de substitució

Consisteix en reemplaçar un o més caràcters d'un missatge a xifrar per un o més caràcters diferents.

Existeixen diferents tipus de criptosistemes:

Criptosistemes de substitució simple.
Consisteix en reemplaçar cadascun dels caràcters del missatge per un altre caràcter de l'alfabet. Cada caràcter serà substituït sempre pel mateix caràcter. El mètode més conegut és el *Xifrat de Cèsar*.

Xifrat de Cèsar.
Definició. Se suposa que tenim un alfabet A de n lletres i unitats de missatge d'una lletra que s'anomena x. Es pot pensar, doncs, l'alfabet com el conjunt d'enters $\mathbb{Z}_n$. Es considera la transformació per xifrar:

$$E_k: \mathbb{Z}_n \rightarrow \mathbb{Z}_n$$
$$x \rightarrow f(x) = x + k \ (mod\ n), on\ 0 < k < n - 1$$

Aquesta transformació desplaça cada lletra *x* del text original *k* lletres endavant[2].
Per desxifrar:

$$D_k: \mathbb{Z}_n \rightarrow \mathbb{Z}_n$$
$$x \rightarrow f(x) = x - k \ (mod\ n)$$

Exemple. Es considera *k*=3 i l'alfabet conegut nostre. La Taula 3.1 mostra la taula de substitucions i la Figura 3.3 mostra exemples visuals de com s'efectuen els canvis.

En clar	a b c d e ... v w x y z
Xifrat	d e f g h ... y z a b c

Taula 3.1: Taula substitució per =3

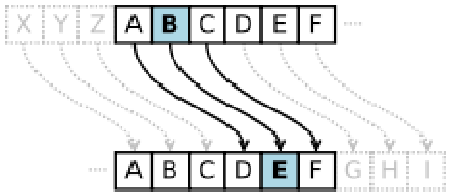
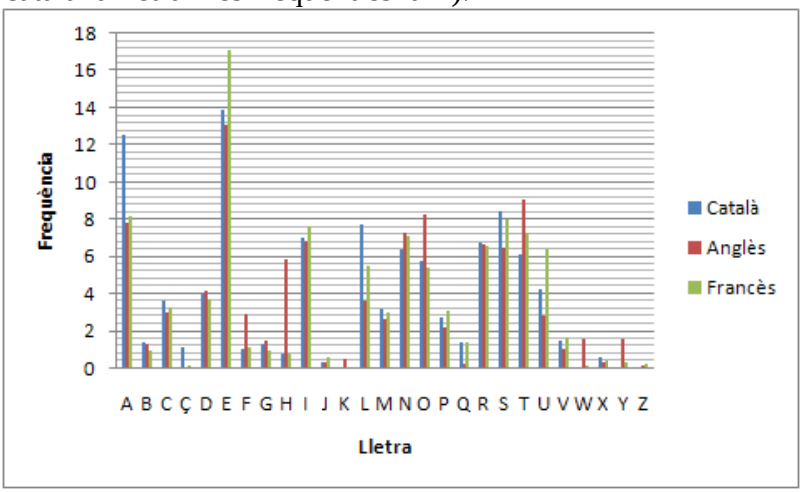


Figura 3.3: Representació substitucions de lletres per =3

Text en clar: A L'ALBA ATACAREM LA TORRE DOS
Text xifrat: DODOEDDWDFDUHPODWRUUHGRV

Aquest criptosistema té una vulnerabilitat molt alta a l'atac per freqüència. L'atac per freqüència consisteix a comparar les freqüències relatives d'aparició d'una lletra en el text inicial i en el seu corresponent text xifrat (les lletres més freqüents en un determinat idioma estan tabulades, i depenen de l'alfabet utilitzat; per exemple, veient la Taula 3.2, en un text en català la lletra més freqüent és la E).



Taula 3.2: Taula de freqüències

Si el text és prou llarg, gràcies a la taula de freqüències es pot intuir per quina lletra s'ha substituït la lletra E, i després de fer algunes proves es pot obtenir la clau.

Criptosistemes de substitució poligràfica.

Consisteix en reemplaçar un grup de caràcters d'un missatge per un altre grup de caràcters. Cada grup de caràcters serà substituït pel mateix grup de caràcters.

Xifrat de Playfair (1854)

Consisteix en un algoritme que xifra parell de lletres (dígrafs). L'algoritme utilitza una taula de 5x5. La taula s'omple amb una frase o paraula secreta descartant les lletres repetides. La resta de la taula s'omple amb les lletres de l'alfabet amb ordre i sense repetir les ja utilitzades a la frase o paraula secreta. Com que l'alfabet es composa de 26 símbols i la taula té 25 espais, hi ha dues possibilitats, ometre la W i utilitzar la V al seu lloc, o utilitzar a la mateixa casella la I i la J.

Per xifrar és realitzen les següents operacions:

- Si les dues lletres apareixen a la mateixa fila de la taula, cadascuna és reemplaçada per la lletra adjacent de la seva dreta.
- Si les dues lletres apareixen a la mateixa columna de la taula, cadascuna és reemplaçada per la lletra adjacent que es troba a sota.
- Si les dues lletres no es troben a la mateixa fila ni a la mateixa columna, cadascuna és reemplaçada per la lletra que es troba a la mateixa fila formant un triangle entre les dues lletres.
- Si les dues lletres són la mateixa, es reemplacen la segona aparició per una X i xifrem seguint les regles anteriors.

Exemple
Text en clar: A L'ALBA ATACAREM LA TORRE DOS
Clau k =PLAYFAIR
La Taula 3.3 és la taula generada mitjançant la clau que s'utilitza pel nostre exemple per xifrar i desxifrar. En aquest cas s'ha omès la W.

P	L	A	Y	F
I	R	B	C	D
E	G	H	J	K
M	N	O	Q	S
T	U	V	X	Z

Taula 3.3: Exemple de taula Playfair per xifrar/desxifrar

Es fan parells: AL AL BA AT AC AR EM LA TO RR ED OS

Es xifra AL:

P	**L** →	**A** →	**Y**	F	Mateixa fila
I	R	B	C	D	Regla: Agafar les lletres adjacents de la seva dreta
E	G	H	J	K	AL → YA
M	N	O	Q	S	
T	U	V	X	Z	

Es xifra BA:

P	L	**A**	Y	F	Mateixa columna
I	R	**B** ▼	C	D	Regla: Agafar les lletres adjacents de sota
E	G	**H** ▼	J	K	BA → HB
M	N	O	Q	S	
T	U	V	X	Z	

Es xifra AT:

P ←	~~L~~	**A**	Y	F	Rectangle
I	R	B	C	D	Regla: Agafar de la mateixa fila, el cantó oposat.
E	G	H	J	K	AT → PV
M	N	O	Q	S	
T →	~~U~~	**V**	X	Z	

Es xifra RR: Com és la mateixa lletra, es modifica la segona per una X, és a dir, s'ha de xifrar RX

P	L	A	Y	F	Rectangle
I	R	B	C	D	Regla: Agafar de la mateixa fila, el cantó oposat.
E	G	H	J	K	RX → CU
M	N	O	Q	S	
T	U	V	X	Z	

Text xifrat: YA YA HB PV YB LB MT AY VM CU KI QM

Aquest criptosistema millora la seguretat dels criptosistemes de substitució simple en l'àtac per freqüències ja que l'anàlisi no és de 26 caràcters sinó que al tractar-se de dígrafs es té $26^2 = 676$ posibilitats.

Criptosistemes de substitució polialfabètica.
S'utilitzen múltiples alfabets per practicar la substitució d'un mateix missatge. Els alfabets no necessiten esser d'orígens diferents, per exemple, un alfabet romànic i un altre ciríl·lic. El simple fet d'alterar l'ordre en la seqüència de les lletres ja és considera un nou alfabet. La característica principal d'aquest criptosistema és que la mateixa lletra es substituïda per lletres diferents, evitant l'atac per freqüències.

Xifrat de Vigenère
Definició.[8] Se suposa que es té un alfabet A de n lletres i es divideix el missatge en blocs de longitud *t*.

i. Existeix una clau $k = (p_1, p_2, \dots, p_t)$ on p_i consisteix en la permutació a realitzar
ii. Per xifrar el missatge m= $(m_1, m_2, \dots, m_t)$ amb la clau $k = (p_1, p_2, \dots, p_t)$: $E_k(m) = (p_1(m_1), p_2(m_2), \dots, p_t(m_t))$;
iii. Per desxifrar la clau serà: $k^{-1} = (p_1^{-1}, p_2^{-1}, \dots, p_t^{-1})$

Exemple. Es considera t=3, l'alfabet conegut nostre i k=(3,7,10).
Text en clar: A L'ALBA ATACAREM LA TORRE DOS
Es fica en blocs de 3: ALA ALB AAT ACA REM LAT ORR EDO SXX
A cada bloc se li aplica la permutació que ens marca la clau, la primera lletra es permuta 3 posicions a la dreta, la segona 7 posicions i la tercera 10.
Text xifrat: DSK DSL DHD DJK ULW OHD RYB HKY VEH
Com es pot observar, la lletra A s'ha xifrat amb diferents lletres evitant, com s'ha comentat anteriorment, l'atac per freqüència.

La màquina Enigma (1919)
Mecanisme electromecànic, és a dir, feia servir una combinació de parts mecàniques i parts elèctriques. Bàsicament consistia en una sèrie de rotors mòbils que giraven a cada tecla que es polsava. D'aquesta forma, en lloc de la lletra escollida apareixia una altra escollida per la màquina segons diferents regles en un codi polialfabètic complex. El fet de que els rotors giressin possibilitava que la mateixa lletra és xifrés amb lletres diferents.
Aquesta màquina és molt coneguda per la utilització que van fer els nazis a la Segona Guerra Mundial, els quals pensaven que era inviolable, sense saber que acceleraria la seva derrota al ser capaços els aliats de desxifrar els seus missatges.

3.2.4 Criptografia simètrica moderna

Amb l'inici de l'ús comercial dels ordinadors, els criptosistemes clàssics ja no són eficients, ja que si se sap el procediment del criptosistema es pot atacar fent servir anàlisi de freqüències o qualsevol altre mètode, encara que es desconeguin les claus.
Utilitzant el concepte d'entropia, Shannon va definir el **secret perfecte** que tenia aquell criptosistema en el que la interceptació del missatge xifrat no proporciona cap informació sobre l'original[8].
Per a que es considerin criptosistemes segurs, si es xifra una seqüència de tot 1's, una de tot 0's i una aleatòria, amb els resultats no es pot saber quin es el resultat de la seqüència aleatòria i quines són de les altres dues seqüències, anomenat Principi de confusió (Veure Figura 3.4)

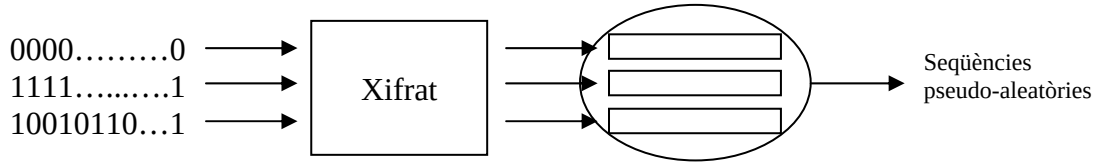


Figura 3.4: Principi de confusió

Si es xifren dues seqüències que només es diferencien per una sola xifra, el resultat ha de ser computacionalment indistingible, és a dir, no es coneix cap algorisme que en temps polinomial sigui capaç de distingir-los. Les seqüències resultants són totalment diferents, no es poden assemblar, anomenat Principi de difusió (Veure Figura 3.5)

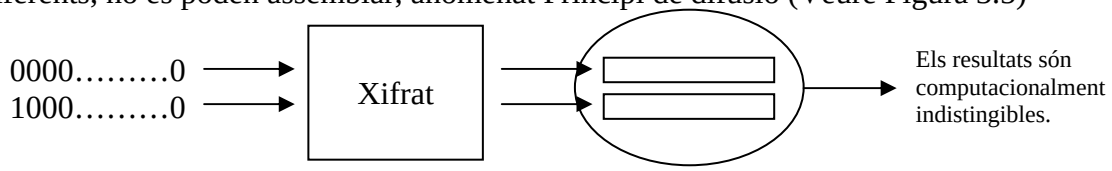


Figura 3.5: Principi de difusió

3.2.4.1 DES (Data Encryption Standard)

És el criptosistema de clau compartida més conegut i el més utilitzat mundialment[2,8]. Va sorgir l'any 1976 i va ser adoptat com estàndard per a la criptografia simètrica de clau secreta en Estats Units i posteriorment a la resta del món. Va ser trencat l'estiu del 1998.
Basat en els treballs dels investigadors d'IBM, que van aplicar els resultats de Shannon, presenta la particularitat d'utilitzar els mateixos circuits per xifrar que per desxifrar, això permet una realització tècnica molt competent i econòmica[5].

El DES es un exemple de mètode de xifratge en bloc. Això vol dir que el text inicial es divideix en blocs de 64 bits i cada bloc es xifra separadament emprant una clau de 54 bits. La Figura 3.6 mostra la seva simbologia.

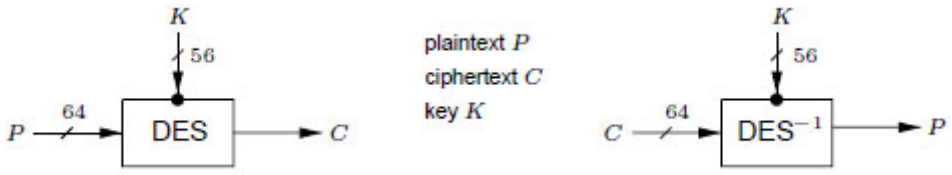


Figura 3.6: Xifrat – Desxifrat DES

Algoritme DES[8]:
Entrada: missatge de 64 bits $m_1 \dots m_{64}$, clau de 56 bits $k_1 \dots k_{56}$
Sortida: text xifrat de 64 bits $c_1 \dots c_{64}$

1. (llistat de claus) Càlcul de 16 claus de 48 bits partint de la clau K.
2. $(L_0, R_0) \leftarrow IP(m_1 m_{2\dots} m_{64})$. (Usar la Taula 3.4 per permutar els bits)
3. (16 vegades) per i des de 1 a 16, calcular L_i i R_i usant les equacions:
   $L_i = R_{i-1}$
   $R_i = L_{i-1} \oplus f(R_{i-1,} K_i)$, on $f(R_{i-1,} K_i) = P(S(E(R_{i-1}) \oplus K_i))$
   on E es una transformació d'expansió de R_{i-1} usant la Taula 3.5
   P es una permutació fixa usant la Taula 3.5
   i S és una transformació no lineal.
4. $b_1 b_2 \dots b_{64} \leftarrow (R_{16}, L_{16})$. (Intercanvi de blocs final L_{16}, R_{16})
5. $C \leftarrow IP^{-1}(b_1 b_2 \dots b_{64})$. (Permutació usant la Taula 3.4)

IP							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

IP^{-1}							
40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

Taula 3.4: Permutació inicial i inversa (IP i IP^{-1})

E					
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

P			
16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

Taula 3.5: Expansió E i Permutació P

Funcionament.
El xifrat és realitza en tres etapes diferents. (Figura 3.7)

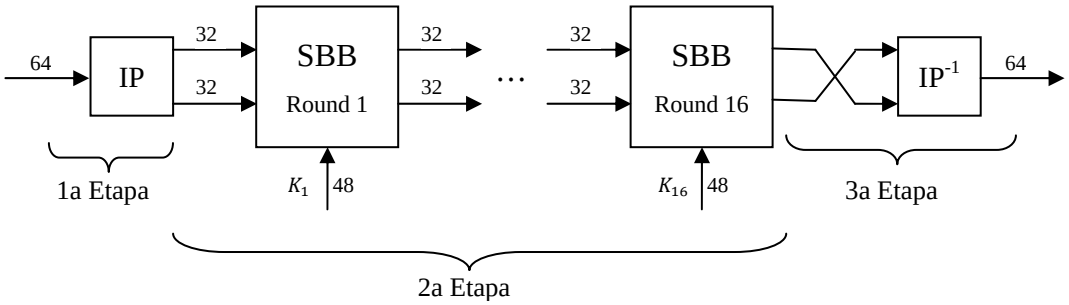


Figura 3.7: Funcionament DES per etapes

1a Etapa (Punt 2 de l'algoritme):
Se li aplica una permutació inicial IP al bloc m.

2a Etapa (Punt 3 de l'algoritme):
El bloc de 64 bits es divideix en dues parts iguals (L_1, R_1).
Aquests dos subblocs, conjuntament amb una subclau K_1de 48 bits són l'entrada del mòdul anomenat Standard Building Block, en el qual s'efectuen un seguit de transformacions (veure Figura 3.8).
Aquest procediment iteratiu es realitza 16 vegades.

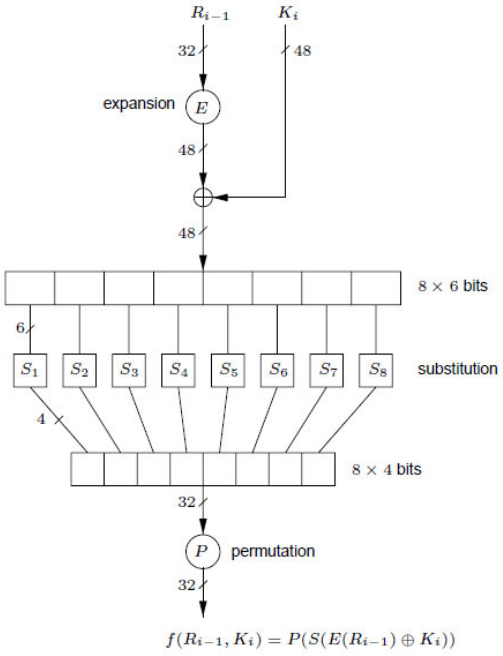


Figura 3.8: SBB del DES

3a Etapa (Punts 4 i 5 de l'algoritme):
El bloc resultant (L_{16}, R_{16}) es transposa i se li aplica una permutació inversa.

La clau del DES són les S-boxes que realitzen una transformació no lineal. A cada capsa li entren 6 bits i en surten 4 bits.

Generació de subclaus.
Les subclaus $K_1, \ldots, K_{16}$ es generen seguint la Figura 3.9 i la Taula 3.6.

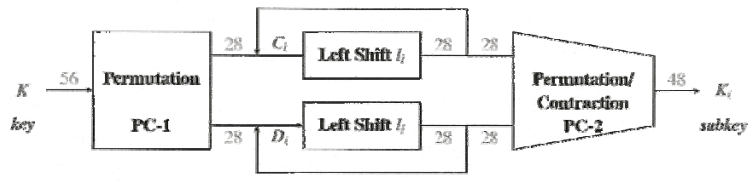


Figura 3.9: Generació de les subclaus del DES

PC-1						
57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
adalt per C_i; abaix per D_i						
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

PC-2					
14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

Taula 3.6: Permutació/Contracció PC-1 i PC-2

A la clau original de 56 bits se li afegeix 8 bits de paritat, es per aquesta raó que a la taula PC-1 (Taula 3.6) es pot observar nombres més grans de 56. Aquesta pràctica no repercuteix en el resultat final, ja que aquestos bits de paritat que s'han afegit no s'agafen mai, no surten reflectits a la taula PC-2.

Xifrat i Desxifrat.

El DES té la propietat que els procediments de xifratge i desxifratge són exactament els mateixos. El receptor parteix de la mateixa clau K que l'emissor ja que s'està parlant d'un algoritme de clau compartida. El receptor obté les subclaus $K_1, \dots, K_{16}$ seguint el mateix mecanisme que l'emissor. Posteriorment, emprarà el mateix procediment que en el xifrat, però utilitzant les claus en ordre invers.

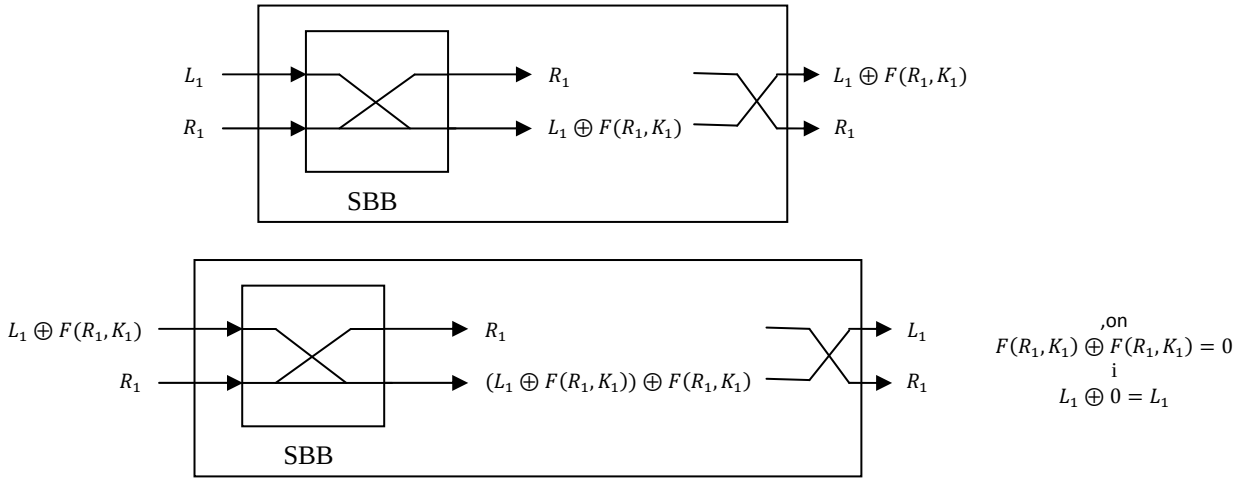


Figura 3.10: Xifrat i Desxifrat amb una SBB box

Invertir l'ordre de les dues meitats i fent un XOR d'una d'elles amb una funció que depén de l'altra meitat i de la clau (operacions efectuades dins les SBB) combinat amb la transposicó final permet aquesta facilitat al desxifrar.

Es pot fer la prova imaginant que només hi hagués una SBB box tal i com mostra la Figura 3.10

Seguretat.

Malgrat la seva feblesa, el DES va resistir durant molts anys tots els atacs de criptòlegs, els quals van inventar procediments cada vegada més sofisticats com el criptoanàlisi diferencial o el criptoanàlisi lineal, però no aconseguien ficar en gaires problemes la seguretat del DES. No va ser fins la construcció de màquines especialitzades realitzant una exploració exhaustiva (atac per força bruta) del conjunt de claus possibles (2^{56}, és a

dir, més de 72 milions de possibilitats) per la EFF (Electronic Frontier Foundation) quan va ser demostrada la seva vulnerabilitat a l'any 1998.
Amb aquesta resolució es va demostrar que la mida de la clau era massa curta, cosa que havia generat una gran controvèrsia.

3.2.4.2 Millores del DES

Triple DES.

Consisteix en un xifrat DES en tres etapes i amb dos claus en un ordre particular. Donat un missatge de text, la primera clau DES es utilitzada per xifrar el missatge. La segona clau es utilitzada per desxifrar mitjançant DES el missatge xifrat. Donat que la segona clau no es la correcta, aquest desxifratge xifra encara més les dades. Aquest missatge xifrat dues vegades es torna a xifrar una tercera vegada amb la primera clau. El text resultant ha set xifrat mitjançant el mètode Triple DES. (Vegis Figura 3.11)

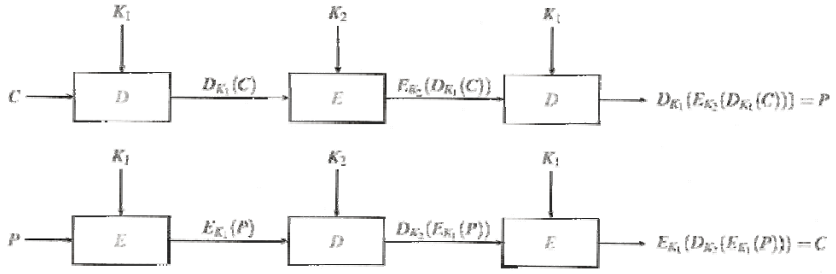


Figura 3.11: Triple DES

Si s'empra el Triple DES amb $k_1 = K_2 = K$ el que s'obté és exactament un xifrat amb DES. Això facilita la compatibilitat entre gent que treballa amb DES i gent que treballa amb Triple DES, sobretot té sentit si està implementat amb Hardware aprofitant els antics circuits tot augmentant la seguritat.
El Triple DES està desapareixent lentament, sent reemplaçat per l'algoritme AES. Encara això, la majoria de targetes de crèdit i altres mitjans de pagament electrònic tenen com estàndard l'algoritme Triple DES (anteriorment usaven el DES). El codi PIN (Personal Identification Number) de les targetes de crèdit no figura directament sobre la targeta, sinó que es calculat mitjançant dades fixes que es troben a la targeta. Aquestes es xifren amb l'ajuda del triple DES utilitzant una clau que només coneix el banc emissor de la targeta. D'aquesta forma es determina el codi PIN que ha de correspondre a l'insertat pel portador de la targeta.[5]

IDEA (International Data Encryption Algorithm)

Va ser un algoritme proposat per substituir al DES. També xifra per blocs. Va ser descrit per primera vegada al 1991 per Xuejia Lai i James L. Massey. Va ser utilitzat en les primeres versions de PGP[2].
Xifra blocs de 64 bits usant una clau de 128 bits. Consisteix en vuit transformacions idèntiques i una transformació de sortida. El concepte de disseny principal de IDEA és barrejar operacions de tres diferents grups algebraics de 2^n elements, que són l'XOR, la suma i la multiplicació modular.[8]
El procés de xifrat i desxifrat és similar.

[2] Pretty Good Privacy.- Software lliure de seguretat que permet el xifrat de dades, arxius i missatges, dissenyat inicialment pel correu electrònic.

AES o Rijndael

A partir de 1998, el NIST (National Institute of Standards and Technology) va llençar un concurs internacional per trobar el successor del DES. El nou estàndard AES (Adavanced Encryption Standard) va ser publicat al 2001. L'algoritme escollit es va anomenar Rijndael i va ser proposat per dos investigadors belgues[5].
Rijndael treballa amb blocs de 128 bits i es poden escollir contrasenyes de 128, 192 i 256 bits, fet que fa resistir els atacs per recerca exhaustiva. L'algoritme està basat en fonaments matemàtics sòlids i es conegut per resistir al criptoanàlisi diferencial i lineal[5].
Els motius per escollir Rijndael com a substitut del DES van ser la bona combinació de seguretat, velocitat, eficiència, senzillesa i flexibilitat. Pot arribar a ser fins a 6 vegades més ràpid que el Triple DES.

3.2.5 Criptografia asimètrica

La criptografia simètrica o de clau compartida té el gran problema de com compartir la clau mitjançant un canal segur. Aquest problema va ser resolt per W. Diffie i M.E. Hellman, de la Universitat de Stanford en California, en 1976. Van crear un protocol criptogràfic que permetia distribuir claus secretes mitjançant canals oberts sense protecció. Aquest invent fou el primer concepte innovador i revolucionari des dels temps de la criptografia clàssica.
El concepte és que cada persona té dues claus: una secreta, que ha de conservar, i una altra pública, que ha de difondre entre la resta dels usuaris de la xarxa. Es publica un llistat amb tots els usuaris i les seves respectives claus públiques. Quan l'usuari A vol enviar un missatge m a un altre membre B de la xarxa, només ha de xifrar el missatge amb la clau pública de B (e_B) i enviar-li el missatge. Ara, només B podrà desxifrar aquest missatge perquè és l'únic que té la clau privada (d_B), encara que una altra persona intercepti aquest missatge, no el podria desxifrar sense la clau privada de B.
Per assegurar aquest compliment, es va introduir el concepte de **funció unidireccional**, és a dir, funcions el càlcul directe de les quals és fàcil però l'invers té tal complexitat que és impossible de realitzar amb els coneixements matemàtics actuals i amb les prestacions dels ordinadors actuals. Per exemple, multiplicar dos nombres grans és fàcil i ràpid, però no existeixen algorismes eficients per factoritzar un nombre gran donat.
La transformació inversa pot ser factible si es coneix una dada addicional, la qual s'anomena trampa. En aquest cas s'anomena funció unidireccional amb trampa i serà totalment necessari per poder desxifrar el missatge.

3.2.5.1 Autenticació

De vegades, la part més important a l'enviar un missatge no és la seva confidencialitat, sinó l'autenticitat, és a dir, que el receptor estigui segur que la persona que firma el missatge és la que realment l'ha tramés.

Se suposa que l'usuari A envia un missatge xifrat amb la seva clau privada d_A, en lloc d'utilitzar la clau pública del destinatari. El missatge no serà secret, ja que tothom podrà desxifrar-lo amb la clau pública de A, e_A. Pel contrari, si a l'utilitzar aquest clau pública e_A el destinatari obté el missatge desxifrat, es pot afirmar que el remitent del missatge és A, ja que cap altra persona pot xifrar aquest missatge amb la clau privada de A excepte aquest. Aquest sistema s'utilitza per la signatura digital.

Si a més d'autentificació també és vol seguretat, es poden combinar els dos mètodes. Suposem que A vol enviar un missatge secret a B i aquest vol estar segur que ha set A qui l'ha enviat. A xifrarà el missatge amb la seva clau privada, per donar autentificació, i posteriorment tornarà a xifrar el resultat amb la clau pública de B, per donar seguretat. Quan B rep aquest missatge fa el procediment invers. Primer, desxifrar amb la seva clau privada i, posteriorment, continua desxifrant amb la clau pública de A. B ja té el missatge secret i a més pot assegurar que l'ha enviat A.

La criptografia de clau pública no soluciona només el problema de compartir les claus, sinó que també pot assegurar l'autenticació de l'emissor del missatge, la signatura digital.

La clau pública no substitueix el sistema clàssic perquè és menys eficient, és a dir, consumeix més recursos de temps i memòria[2]. A la pràctica s'utilitza AES per xifrar i RSA per intercanviar la clau compartida de AES.

3.2.5.2 RSA

Aquesta fou la primera realització del model teòric proposat per Diffie i Hellman. El van desenvolupar Rivest, Shamir i Adleman (les inicials dels quals donen nom al criptosistema) al 1978 i és el mètode de clau pública més estudiat i emprat.

Algoritme Generació de claus[8]:

1. Es generen dos nombres primers (i diferents), p i q, extremadament grans i de llargada similar.
2. Es calcula $n = p \cdot q$, i $\varphi(n) = (p-1)(q-1)$, usant les propietats 1 i 2 de la funció phi d'Euler (apartat 3.1.7)
3. S'escolleix un enter $e, 1 < e < \varphi(n)$, tal que $m.c.d.(e, \varphi(n)) = 1$.
4. Usant l'algoritme d'Euclides Extés (apartat 3.1.3) i inversos modulars (apartat 3.1.6) es calcula $d, 1 < d < \varphi(n)$, tal que $e \cdot d \equiv 1 (mod\ \varphi(n))$.
5. Les claus públiques són (n, e) i les claus privades són $(\varphi(n), d)$.

Algoritme Xifrat i Desxifrat RSA[8]:

Resum: B xifra un missatge m per A i després A desxifra.

1. *Xifrar*: B ha de seguir els següents passos:
    a. Obtenir les claus públiques de A (n, e).
    b. Representar el missatge com un enter m en intervals $[0, n-1]$.
    c. Calcular $c = m^e\ mod\ n$, usant exponenciació modular (apartat 3.1.9)
    d. Enviar el text xifrat a A.
2. *Desxifrar*: Per recuperar el missatge m des de c, A ha de seguir els següents passos:
    a. Usar la seva clau privada d i calcular $m = c^d\ mod\ n$

Per desxifrar s'utilitza el Teorema d'Euler (apartat 3.1.7):

$$c^d = (m^e)^d mod\ n$$

Aplicant Euler se sap que $e \cdot d \equiv 1(\varphi(n))$, llavors $c^d = m\ mod\ n$

Exemple.

L'usuari A tria els nombres primers $p = 281$ i $q = 167$. Això implica que $n = p \cdot q = 46927$ i $\varphi(n) = (p-1)(q-1) = 280 \cdot 166 = 46480$. Tot seguit, tria l'enter $e = 39423$, que està entre 1 i 46480 i compleix que $m.c.d.(e, \varphi(n)) = 1$. Finalment, A calcula $d = e^{-1}\big(mod\ \varphi(n)\big) = 39423^{-1}(mod\ 46480)$ mitjançant l'algorisme d'Euclides Extés, i obté $d = 26767$. La clau que l'usuari A publica és $\{46927,39423\}$.

Un usuari B vol enviar el missatge ENCARA a l'usuari A. Primer dividirà el missatge en dues parts ENC i ARA, amb equivalents numèrics:

$$\text{ENC} = 4\ 13\ 2 = 4 \cdot 26^2 + 13 \cdot 26 + 2 = 3044$$
$$\text{ARA} = 0\ 17\ 0 = 0 \cdot 26^2 + 17 \cdot 26 + 0 = 442$$

S'obté $m_1 = 3044$ i $m_2 = 442$. Aleshores es xifra:

$$c = m^e\ mod\ n$$
$$c_1 = 3044^{39423}\ (mod\ 46927) = 39896$$
$$c_2 = 442^{39423}\ (mod\ 46927) = 31017$$

Finalment s'expressa en base 26 aquestes unitats xifrades i es recodifica a l'alfabet original: 39896= 2 7 0 12 = C H A M i 31017 = 1 19 22 25 = B T W Z. L'usuari A rebrà, doncs, el següents missatge: CHAMBTWZ

L'usuari A divideix el missatge rebut en dues parts, busca els equivalents numèrics corresponents i utilitza la seva clau privada $d = 26767$ per desxifrar:

$$m = c^d\ mod\ n$$
$$m_1 = 39896^{26767}\ (mod\ 46927) = 3044 = \text{ENC}$$
$$m_2 = 31017^{26767}\ (mod\ 46927) = 442 = \text{ARA}$$

Com s'ha observat al llarg del capítol, la criptografia a avançat molt al llarg de la història, sobretot, gràcies a les matemàtiques.

3.2.5.3 Altres criptosistemes de clau pública

S'ha vist amb més detall el criptosistema RSA al ser el criptosistema de clau pública més estès i utilitzat. Altres criptosistemes de clau pública molt utilitzat són ElGamal i el criptosistema knapsack, que es veurà amb més detall en el capítol 4 ja que és el criptosistema emprat en aquest projecte.

Capítol 4

Examen tipus test amb criptosistema knapsack

En el capítol anterior s'ha fet una introducció als fonaments matemàtics i a la criptografia per poder seguir curosament el capítol present, on s'explica com funciona el criptosistema knapsack. Aquest criptosistema és la base de l'aplicació web realitzada, és a dir, totes les aplicacions matemàtiques realitzades en aquest projecte es basen en aquest criptosistema. Posteriorment, s'explica detalladament com es generen els exàmens tipus test i com s'autoavaluen.

4.1 El criptosistema knapsack

El criptosistema knapsack és un criptosistema de clau pública basat en el problema de la suma de subconjunts (*subset sum*) o problema de la motxilla, el qual és NP-Complet (3.1.10), descrit a continuació:

Donat $S \in \mathbb{N}$ i una seqüència $A = (a_1, \dots, a_k), a_i \in \mathbb{N}$, determinar si existeix una seqüència $X = (x_1, \dots, x_k), x_i \in \{0,1\}$ tals que $S = \sum_{i=1}^{k} x_i a_i$

Exemple
Donats $S = 114$ i $A = (3,28,15,8,90,45,78)$ sí hi ha solució, per exemple $X = (0,1,0,1,0,0,1)$, és a dir, $28 + 8 + 78 = 114$. En canvi, si $S = 16$ no hi ha solució, no hi ha cap combinació de sumands que dins de A que doni 16 com a resultat.

Per tal de resoldre aquest problema per força bruta, es pot observar clarament que la quantitat de combinacions a provar es de 2^k. Si k és un nombre suficientment gran garanteix la seguretat enfront d'atacs per força bruta.

No obstant això, en el cas concret de que la successió A sigui supercreixent, és a dir, $a_i > \sum_{j=1}^{i-1} a_j \, \forall i$, llavors és fàcil trobar una solució.

Exemple
Donats $S = 292$ i $A = (5,7,13,28,56,113,231)$ hi ha una única solució, que serà $X = (1,0,0,0,1,0,1)$. El resultat s'obté restant S amb el primer element de la successió menor o igual. Es fica un 1 al resultat a la mateixa posició que ocupa aquest element dins A. S'agafa el resultat de la resta i es repeteix el procés fins que el número sigui 0. Es fica un 0 a les posicions del resultat que resulten buits. Per $S = 292$ el primer element més proper és el 231, es fica un 1 a la posició 7 de la llista resultant i es fa la resta, $292 - 231 = 61$. Es repeteix el procés, l'element més proper a 61 és el 56, es fica un 1 a la posició 5 de la llista resultant i es fa la resta, $61 - 56 = 5$. L'element més proper a 5 és 5, es fica un 1 a la primera posició de la llista i com que $5 - 5 = 0$ ja s'ha acabat. Es fica un 0 a la resta de posicions de la llista resultant.

Merkle i Hellman van aprofitar aquesta facilitat per trobar una solució per crear el **criptosistema knapsack**, l'any 1978. Mitjançat multiplicacions modulars i permutacions, van aconseguir convertir-lo en un problema de difícil solució, per tant, ja tenien el criptosistema, per una part, fàcil de resoldre amb les claus privades i, per una altra, difícil de resoldre sense aquestes.

Partint d'una successió $A = (\mathrm{a}_1, \dots, \mathrm{a}_\mathrm{k})$ supercreixent, es consideren $n > \sum_{i=1}^{k} a_i$ i $t \in \mathbb{Z}_n$, es calcula una nova successió $P = (c_1, \dots, c_k)$, on $c_i = a_i t (mod\ n) \forall i = 1, \dots, k$. Es pot veure la descripció del criptosistema a continuació[10]:

- Clau pública: $P = (\mathrm{c}_1, \dots, \mathrm{c}_\mathrm{k})$.
- Clau privada: n, t.
- Missatges: $M = (x_1, \dots, x_k)$, on $x_i \in \{0,1\}$.
- Xifrat: $C = \sum_{i=1}^{k} x_i c_i$.
- Desxifrat: $C \cdot t^{-1} = (\sum_{i=1}^{k} x_i c_i) \cdot t^{-1} (mod\ n) = \sum_{i=1}^{k} x_i \cdot c_i \cdot t^{-1} = \sum_{i=1}^{k} x_i \cdot a_i$ ja que $c_i \cdot t^{-1} = a_i$.

Exemple
$A = (1,3,5,11,21,44,87,175)$

$n = 2311 \; i \; t = 57$

Es mira que n i t siguin primers entre ells (3.1.4) calculant $m.c.d(2311,57) = 1$. Això es fa per saber que t té invers a $\mathbb{Z}_n$ (3.1.6). Es calcula l'invers utilitzant l'algorisme d'Euclides Extès (3.1.3) i inversos modulars (3.1.6):

$$2311 = 57 \cdot 40 + 31$$

$$57 = 31 \cdot 1 + 26$$

$$31 = 26 \cdot 1 + 5$$

$$26 = 5 \cdot 5 + 1$$

Com la resta és igual a 1, es pot afirmar que t i n són primers entre ells, llavors ∃ invers, i es continua per trobar-lo:

$$1 = 26 - 5 \cdot 5 = 26 - 5(31 - 26 \cdot 1) = 26 - 31 \cdot 5 + 26 \cdot 5 = 26 \cdot 6 - 31 \cdot 5 =$$
$$= 6(57 - 31 \cdot 1) - 31 \cdot 5 = 57 \cdot 6 - 31 \cdot 6 - 31 \cdot 5 = 57 \cdot 6 - 31 \cdot 11 =$$
$$= 57 \cdot 6 - 11(2311 - 57 \cdot 40) = 57 \cdot 6 - 2311 \cdot 11 + 57 \cdot 440 =$$
$$= 57 \cdot 446 - 2311 \cdot 11$$

$t^{-1} = 446$

Es pot comprovar el resultat ja que se sap que $t \cdot t^{-1} \equiv 1(mod\ n)$ (2.1.6):

$$57 \cdot 446 \equiv 25422 \equiv 2312 + 2311 \cdot 10 \equiv 2312(mod\ 2311) \equiv 1(mod\ 2311)$$

Posteriorment, s'aplica $c_i = a_i t(mod\ n) \forall i = 1, \dots, k$ per obtenir la clau pública:

$$c_1 = 1 \cdot 57(mod\ 2311) \equiv 57(mod\ 2311)$$
$$c_2 = 3 \cdot 57(mod\ 2311) \equiv 171(mod\ 2311)$$
$$c_3 = 5 \cdot 57(mod\ 2311) \equiv 285(mod\ 2311)$$
$$c_4 = 11 \cdot 57(mod\ 2311) \equiv 627(mod\ 2311)$$
$$c_5 = 21 \cdot 57(mod\ 2311) \equiv 1197(mod\ 2311)$$
$$c_6 = 44 \cdot 57(mod\ 2311) \equiv 2508(mod\ 2311) \equiv 197(mod\ 2311)$$
$$c_7 = 87 \cdot 57(mod\ 2311) \equiv 4959(mod\ 2311) \equiv 337(mod\ 2311)$$
$$c_8 = 175 \cdot 57(mod\ 2311) \equiv 9975(mod\ 2311) \equiv 731(mod\ 2311)$$

La clau serà: $P = (57,171,285,627,1197,197,337,731)$

El missatge a xifrar que s'ha escollit és M=KNAPSACK. S'ha de xifrar lletra a lletra, i, com s'ha vist a l'algorisme, cada lletra ha de ser mostrada en binari. Per obtenir el número binari de cada lletra, primer s'obté el seu equivalent decimal gracies al codi ASCII, i posteriorment, es passa a binari. Subsegüentment, es xifra segons l'algorisme:

$K \equiv 75 \equiv 01001011 \rightarrow 171 + 1197 + 337 + 731 = 2436$

$N \equiv 78 \equiv 01001110 \rightarrow 171 + 1197 + 197 + 337 = 1902$

$A \equiv 65 \equiv 01000001 \rightarrow 171 + 731 = 902$

$P \equiv 80 \equiv 01010000 \rightarrow 171 + 627 = 798$

$S \equiv 83 \equiv 01010011 \rightarrow 171 + 627 + 337 + 731 = 1866$

$C \equiv 67 \equiv 01000011 \rightarrow 171 + 337 + 731 = 1239$

Xifrant cadascuna de les lletres, s'envien els 8 missatges xifrats següents:

$$(2436,1902,902,798,1866,902,1239,2436)$$

El receptor dels missatges té com a claus privades t i n. Mitjançant aquestes dues claus, obté t^{-1} com s'ha vist amb anterioritat. A continuació, aplica $a_i = t^{-1} c_i (mod\ n)$ per elaborar la successió supercreixent original A. Es fa la prova

$$a_1 = 446 \cdot 57(mod\ 2311) \equiv 1(mod\ 2311)$$
$$a_2 = 446 \cdot 171(mod\ 2311) \equiv 3(mod\ 2311)$$

... ...

$$a_8 = 446 \cdot 2436(mod\ 2311) \equiv 175(mod\ 2311)$$

Subsegüentment, cada missatge cal multiplicar-lo per $t^{-1}(mod\ n)$. Es fa el primer:

$$2436 \cdot 446(mod\ 2311) \equiv 1086456(mod\ 2311) \equiv 286(mod\ 2311)$$

Per tant, s'ha de saber si el problema knapsack té solució quan $S = 286$ i la successió supercreixent és $A = (1,3,5,11,21,44,87,175)$

Si es té 286, el primer element de la successió més proper és el 175, bit 7, llavors es fica el bit setè a 1 i es resta $286 - 175 = 111$. L'element més proper, ara, és el 87, bit 6, es fica el sisè bit a 1 i es resta $111 - 87 = 24$. El més proper és el 21, bit 4, es fica el quart a 1 i es resta $24 - 21 = 3$. Finalment, el més proper és el 3, bit 1, es fica a 1 i es resta $1 - 1 = 0$. S'ha arribat a 0, es fica la resta de bits a 0 i ja s'ha acabat amb el resultat:

$$286 \equiv 0,1,0,0,1,0,1,1$$

Es converteix aquesta seqüència binària a decimal obtenim el número 75 que equival a la lletra K, per tant, el primer missatge passat és K. Es fa el mateix per la resta de missatges:

$$1902 \cdot 446(mod\ 2311) \equiv 155 \equiv 0,1,0,0,1,1,1,0 \equiv 78 \equiv N$$

$$(155 - 87 = 68 - 44 = 24 - 21 = 3 - 3 = 0)$$

$$902 \cdot 446(mod\ 2311) \equiv 178 \equiv 0,1,0,0,0,0,0,1 \equiv 65 \equiv A$$

$$(178 - 175 = 3 - 3 = 0)$$

$$798 \cdot 446(mod\ 2311) \equiv 14 \equiv 0,1,0,1,0,0,0,0 \equiv 80 \equiv P$$

$$(14 - 11 = 3 - 3 = 0)$$

$$1866 \cdot 446(mod\ 2311) \equiv 276 \equiv 0,1,0,1,0,0,1,1 \equiv 83 \equiv S$$

$$(276 - 175 = 101 - 87 = 14 - 11 = 3 - 3 = 0)$$

$$1239 \cdot 446(mod\ 2311) \equiv 265 \equiv 0,1,0,0,0,0,1,1 \equiv 67 \equiv C$$

$$(265 - 175 = 90 - 87 = 3 - 3 = 0)$$

Si s'ajunten tots els missatges rebuts amb ordre s'obté el text original, KNAPSACK.

4.2 Examen tipus test

En aquesta secció es veurà com aplicar el criptosistema knapsack per a la creació d'exàmens tipus test. L'objectiu es crear una versió diferent de l'examen per cada alumne. Cada versió té les mateixes preguntes i les mateixes possibles respostes però desordenades, tant les preguntes com les respostes. D'aquesta forma cada alumne respondrà les mateixes preguntes però el fet que estiguin desordenades millora el possible frau per còpia per part dels estudiants, ja que cadascun rebrà una versió d'examen amb aparença distinta.

El criptosistema knapsack permet codificar les respostes del test de manera senzilla, és a dir, a cada possible resposta se l'associa un número en lloc de les típiques lletres (a, b, c, ...). Això suposa poc esforç addicional a l'alumne que haurà d'anar sumant els números enters associats a les respostes que creu correctes. El resultat de l'examen serà un sol número. Mitjançant aquest, l'estudiant podrà obtenir la seva nota immediatament després de finalitzar l'examen, consultant la pàgina web de l'aplicació i introduint el seu número resultant i la versió que ha realitzat.

Aquesta aplicació permet, a més, alliberar al professor de la correcció de l'examen, ja que el resultat que obtingui l'alumne es guarda automàticament a la base de dades de l'aplicació després de la consulta d'aquest.

4.2.1 Preparació

Per a la creació d'un examen es necessita:

- Una successió supercreixent $a_1, \dots, a_k$, on $k = p \cdot r$ sabent que p és el número de preguntes que té l'examen i r és el número de possibles respostes que té cada pregunta.
- Un mòdul $n > 2a_k$.
- Tants multiplicadors t com versions es volen crear (una versió per alumne).

Aplicant el criptosistema knapsack (4.1), es transforma la successió supercreixent en la clau pública aplicant $c_i = a_i t (mod\ n) \forall i = 1, \dots, k$. S'ha de fer el mateix procés per a cada versió de l'examen, la qual cosa, s'obtenen tantes claus públiques com versions es volen crear.

Finalment, per a cada versió, s'agafa la clau pública corresponent i s'assigna a cada possible resposta un element d'aquesta clau pública.

4.2.2 Funcionament

Cada alumne rep una versió de l'examen. Escollir una resposta consistirà en agafar el número associat a la resposta que l'alumne considera correcta. El resultat final de l'examen serà la suma de totes les respostes que l'alumne cregui correctes. Al finalitzar l'examen, l'alumne marxarà amb un rebut composat per dos números:

- El número de versió que ha realitzat.
- El número resultant de l'examen.

Mentre que el professor tindrà un altre rebut on, a més dels dos números anteriors, s'afegirà el DNI de l'alumne.

4.2.3 Correcció

Hi ha tres possibles formes de corregir l'examen:

- **Directe**.- L'alumne entrega el rebut i el professor introdueix a l'aplicació el DNI de l'alumne, la versió que ha realitzat i el número resultant, mitjançant un ordinador portàtil a la mateixa aula on es realitza l'examen. El professor notifica la nota a l'alumne i tota aquesta informació es queda enregistrada a la base de dades.
- **Diferit**.- El professor recull tots els rebuts i, al seu despatx, calcula les notes repetint el procés del punt anterior.
- **Autoservei**.- L'alumne accedeix a l'aplicació i facilitant el seu DNI, la versió de l'examen realitzat i el número resultant obté la nota de l'examen. Aquesta

informació s'emmagatzema a la base de dades i el professor pot consultar les notes de l'examen sense la necessitat de ser ell l'encarregat d'anar inserint les dades de cada examen per obtenir el resultat.

La correcció, per part de l'aplicació, es senzilla. Es desxifra el número resultant de l'examen amb el criptosistema kanpsack (4.1) obtenint un vector amb 1's i 0's. Aquest vector ens indica les respostes que ha escollit l'alumne (1's) i comparant amb les respostes correctes es determina la nota.

4.2.4 Seguretat

Es important introduir un sistema de detecció d'eventuals errors per part de l'estudiant en el càlcul de la suma. Una bona possibilitat és que tots els números associats a les possibles respostes siguin múltiples de 7, i s'aconsegueix multiplicant cada element de la clau pública per 7. Quant l'alumne hagi sumat totes les seves respostes es demanarà que divideixi aquest número entre 7 on el resultat haurà de ser enter, si no és així, l'estudiant ha comés un error a la suma.

Per augmentar la seguretat a eventuals fraus, es pot demanar a l'estudiant que no faci cap marca a la fulla de l'examen ja que amb la suma resultant ja es té tota la informació necessària per la seva correcció.

A més a més, cada versió d'examen té els números de les respostes diferents, ja que cada versió té associada una clau pública distinta. Això ens assegura que un alumne no es copiï el número resultant d'un altre company, ja que aquest número, amb la seva versió, no serà correcte.

I, com s'ha nombrat a la introducció de la secció 4.1, al tenir les respostes i les possibles respostes desordenades, permutades, s'evita el possible frau per copia.

4.2.5 Versió optimitzada

Fins ara s'ha treballat basant-se amb el criptosistema knapsack sense adaptar-lo a les necessitats dels exàmens tipus test. La successió supercreixent més lenta és $a_i = 2^{i-1}$, $i = 1 \dots k$, sabent que $k = p \cdot r$, on p és el número de preguntes que té l'examen i r és el número de possibles respostes que té cada pregunta. S'ha de tenir present que les calculadores utilitzades pels alumnes no podran operar amb números majors de 10 dígits. S'ha de calcular el màxim de preguntes i respostes per a que els números associats a les respostes no superi els 10 dígits. A més, els números han de ser múltiples de 7 (4.2.4). Per a que tinguin com a màxim 10 dígits: $7(2a_k - 1) < 10^{10}$ i, si es pren la successió supercreixent més lenta, equival a que $k \leq 30$, per exemple, 10 preguntes amb 3 opcions per pregunta.

Però, si a cada pregunta només li correspon una possible resposta, existeix una solució més eficient que permet utilitzar números més petits. S'ha de pensar que a cada pregunta li correspon un sac amb tants números com possibles respostes. El vector supercreixent és la unió de tots els sacs. De cada sac només es pot escollir una opció, llavors ja no fa falta que $a_i = 2^{i-1}$. Dins un mateix sac $a_i = i + 1$ i el primer element d'un sac serà la suma del primer i últim del sac anterior i aquest servirà per la successió d'elements d'aquest sac. D'aquesta manera el vector supercreixent serà més petit i es podrà obtenir més preguntes per vector.

Exemple.- Suposem que es vol crear un examen de 5 preguntes amb 3 possibles respostes per pregunta. Segons la versió optimitzada, s'ha de crear 5 sacs amb 3 elements a cada sac. Es comença pel primer, aplicant $a_i = i + 1$, com hem vist:

$$[1,2,3]$$

Ja es té el primer sac. El primer element del següent sac serà la suma del primer i últim de l'actual, és a dir, $1 + 3 = 4$, la resta d'elements del grup es calcularan sumant aquest primer número:

$$[4,8,12]$$

El primer element del tercer sac serà $4 + 12 = 16$, i la resta d'elements s'obtindran sumant 16 més cada vegada:

$$[16,32,48]$$

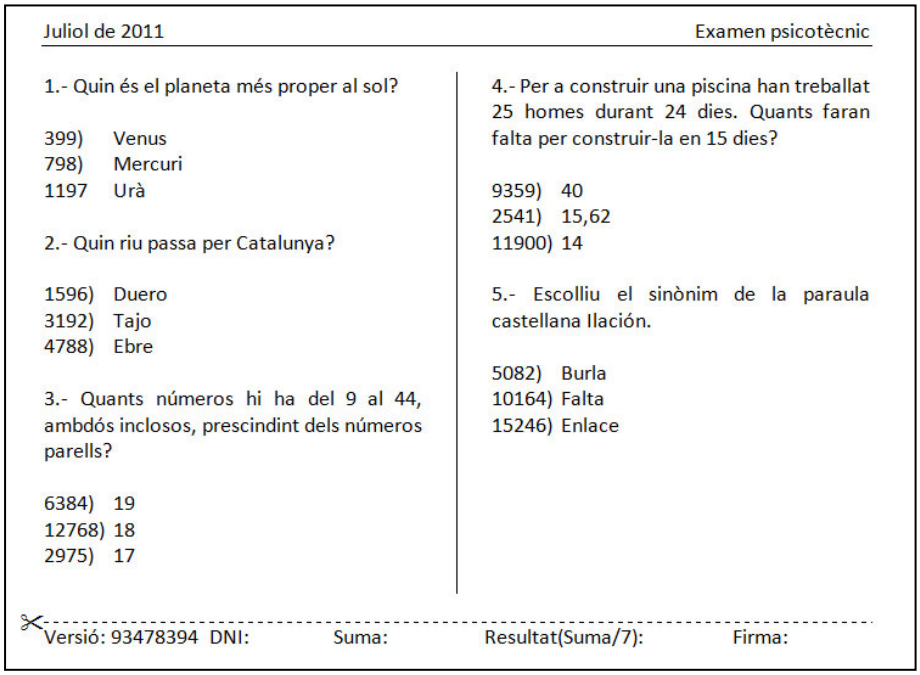
Si es repeteix el procés, el resultat final serà:

$$([1,2,3], [4,8,12], [16,32,48], [64,128,192], [256,512,768])$$

Amb aquest vector supercreixent s'assegura que al resoldre el knapsack, només hi haurà un 1 dins de cada sac, com que cada sac representa les possibles respostes de cada pregunta, la resposta escollida per l'alumne serà la que té l'1.

4.2.6 Exemple d'examen tipus test

Es suposa que tenim un examen amb 5 preguntes i cada pregunta té 3 possibles respostes. La Figura 4.1 mostra com quedarà una versió d'aquest examen.



Juliol de 2011 — Examen psicotècnic

1.- Quin és el planeta més proper al sol?

399) Venus
798) Mercuri
1197 Urà

2.- Quin riu passa per Catalunya?

1596) Duero
3192) Tajo
4788) Ebre

3.- Quants números hi ha del 9 al 44, ambdós inclosos, prescindint dels números parells?

6384) 19
12768) 18
2975) 17

4.- Per a construir una piscina han treballat 25 homes durant 24 dies. Quants faran falta per construir-la en 15 dies?

9359) 40
2541) 15,62
11900) 14

5.- Escolliu el sinònim de la paraula castellana Ilación.

5082) Burla
10164) Falta
15246) Enlace

Versió: 93478394 DNI: Suma: Resultat(Suma/7): Firma:

Figura 4.1: Exemple d'examen tipus test

Es veurà com s'ha generat l'examen. Primer, es crea el vector supercreixent. Se sap que l'examen consta de 5 preguntes i cada pregunta té 3 possibles respostes, llavors, el vector estarà composat de 15 elements, podent aprofitar el vector òptim de l'exemple de la secció anterior:

$$([1,2,3],[4,8,12],[16,32,48],[64,128,192],[256,512,768])$$

A continuació, i seguint els passos de l'algorisme knapsack (3.1), s'agafa un mòdul i un multiplicador:

- $n = 2311$.
- $t = 57$.

S'ha escollit els mateixos valors que l'exemple de la secció 4.1, que compleixen les condicions necessàries, per no haver de repetir els càlculs. Es calcula l'invers de t sent $t^{-1} = 446$. Es calcula la clau pública aplicant $c_i = a_i t (mod\ n) \forall i = 1, \dots, k$:

$$P = (57,114,171,228,456,684,912,1824,425,1337,363,1700,726,1452,2178)$$

Ara, es multiplica cada element del vector per 7 (3.2.4):

$$P' = (399,798,1197,1596,3192,4788,6384,12768,2975,$$
$$,9359,2541,11900,5082,10164,15246)$$

Ja es té la clau pública. Seguidament, s'assigna un número a cada possible resposta. Com es pot veure a la Figura 4.1, les 3 respostes de la primera pregunta tenen assignats els 3 primers números, les 3 respostes de la segona pregunta, els 3 següents i així successivament.

Una vegada es té l'examen creat, es continua l'exemple per veure com el farà un alumne. Se suposa que l'alumne selecciona les respostes $a), a), c), b), c)$. Si fa la suma dels números associats a les respostes s'obté:

$$399 + 1596 + 2975 + 2541 + 15246 = 22757$$

Per comprovar que no s'ha equivocat al fer la suma ha de dividir aquest número per 7:

$$22757/7 = 3251$$

Aquest és el número resultant de l'examen que s'adjuntarà al rebut de l'alumne, juntament amb la versió de l'examen que ha realitzat.

Per corregir l'examen desxifrem el criptosistema (4.1):

$$3251 \cdot 446\ (mod\ 2311) \equiv 949$$

Es compara aquest resultat amb el vector supercreixent per obtenir un vector amb 1's i 0's on es poden veure les respostes de l'alumne:

$$949 = 1,0,0,1,0,0,0,0,1,0,1,0,0,0,1$$
$$(949 - 768 = 181 - 128 = 53 - 48 = 5 - 4 = 1 - 1 = 0)$$

Es separa el vector en sacs:

$$[1,0,0],[1,0,0],[0,0,1],[0,1,0],[0,0,1]$$

Es pot veure que de la primera pregunta a respòs la $a)$, de la segona també la $a)$, de la tercera la $c)$, de la quarta la $b)$ i de la última pregunta la $c)$. Ara que ja se saben les respostes de l'alumne ja es pot ficar-li una nota.

Capítol 5

Tecnologies implicades

L'objectiu d'aquest capítol és el d'introduir les principals tecnologies amb les que ha estat desenvolupat el projecte. En cap cas es pretén explicar de forma rigorosa i completa cada una de les tecnologies però sí fer factible la seva utilització com a petita guia o referència per a totes aquelles persones que es vulguin endinsar en aquest món, o bé, per afavorir que una persona no iniciada en alguna d'aquestes matèries pugui assolir el grau suficient per a comprendre el projecte desenvolupat.

En cada un dels apartats del capítol es procura donar una explicació general de cada una de les tecnologies involucrades, comparant-les amb altres amb capacitats similars.

Cap de les tecnologies implicades han estat marcades pels requeriments no funcionals ja que no hi havia cap client darrera de l'aplicació que marqués alguna pauta i s'ha pogut escollir lliurement.

5.1 PHP

PHP és un llenguatge de programació (originàriament PHP Tools, Personal Home Page Tools), que serveix principalment per proporcionar característiques dinàmiques a una pàgina web. El PHP sol combinar-se amb bases de dades MySQL, i amb servidor Apache sobre Linux, oferint resultats molt interessants per totes aquelles pàgines web que pretenguin figurar com actives i dinàmiques.

S'enten com a aplicació web aquella web que no es limita simplement a mostrar la informació a l'usuari en forma de text i fotografies, sinó que apart d'això aquest hi pot interactuar, emplenant formularis, o introduint qualsevol tipus de dades per a qualsevol propòsit.

El llenguatge PHP té la característica de poder emprar-se conjuntament amb llenguatge HTML, utilitzat per crear pàgines web. PHP, al contrari d' HTML, té la característica que s'interpreta i s'executa directament en el servidor on està penjada la web, mitjançant el qual el visitant d'aquesta només rep el resultat buscat pel codi en el que està escrit.

El PHP és Open Source. És gratuït, i el seu desenvolupament depèn de programadors que l'utilitzen i el van perfeccionant i desenvolupant segons les necessitats que van apareixent dia a dia en el disseny web.

S'utilitzarà PHP per desenvolupar gran part de l'aplicació, que permetrà, entre d'altres coses, realitzar càlculs, paginar resultats, validar dades, emmagatzemar informació al servidor o realitzar consultes de informació específica.

Perquè s'ha escollit PHP?

El llenguatge HTML per sí sol no podia ser l'únic llenguatge a emprar, ja que l'aplicació ha de ser dinàmica, necessita d'accés a una base de dades, i l'HTML no és capaç de dur a terme aquest propòsit.

Les possibilitats estudiades en un principi foren PHP, JSP, ASP. ASP es va descartar ràpidament per diferents motius. És una tecnologia propietària, es necessari escriure molt codi per realitzar funcions senzilles, el hosting web és costós i és la única de les tres nombrades amb la qual no he treballat.

L'elecció havia de ser, doncs, entre PHP i JSP. S'ha decantat per PHP bàsicament perquè era el llenguatge amb el que he treballat més últimament. JSP ofereix un gran potencial, encara que és un llenguatge multi-propòsit, per tant, moltes de les seves llibreries no ens serviran en una aplicació web. PHP està més pensat per a web i totes les seves llibreries estan pensades per a web. Tanmateix, s'ha complementat aquesta mancança de PHP treballant amb jQuery (secció 5.2), cosa que ha provocat que es pugues escollir PHP encara que JSP tingues més potencial a priori.

Avantatges i inconvenients de l'ús de PHP

A continuació s'enumeren les principals avantatges i inconvenients d'aquesta tecnologia.

Avantatges:

- És un llenguatge multiplataforma.
- El codi font escrit en PHP és invisible al navegador web i al client ja que és el servidor el que s'encarrega d'executar el codi i enviar el resultat HTML al navegador. Això fa que la programació en PHP sigui segura i confiable.
- Té maneig d'excepcions.
- Biblioteca nativa de funcions amplia i inclosa.
- Permet tècniques de programació orientada a objectes.
- Àmplia documentació a la seva pàgina oficial, entre la qual destaca que totes les funcions dels sistema estan explicades i exemplificades en un únic arxiu d'ajuda.
- Capacitat de connexió amb la majoria de motors de base de dades que s'utilitzen actualment, destacada la seva connectivitat amb MySQL i PostgreSQL.
- És lliure.
- Capacitat d'expandir el seu potencial usant mòduls.

Inconvenients:

- Promou la creació de codi desestructurat i amb un manteniment complex.
- No posseeix un adequat maneig d'unicode.
- És molt difícil d'optimitzar.
- Com és un llenguatge que s'interpreta en execució, per cert usos pot resultar un inconvenient que el codi no pugi ser ocultat. L'ofuscació és una tècnica que pot dificultar la lectura del codi però no la impedeix i, en ocasions, representa un cost en temps d'execució.

5.2 jQuery

JavaScript és un llenguatge de programació interpretat. Es defineix com orientat a objectes, basat en prototips, imperatiu i dinàmic. S'utilitza principalment per a crear pàgines webs dinàmiques.

jQuery és una llibreria o framework de JavaScript, que permet simplificar la forma d'interactuar amb els elements d'una web mitjançant el DOM, manejar events, desenvolupar animacions, agregar interacció amb la tècnica AJAX[3] a pàgines web. El que el fa tan especial és la seva senzillesa i la poca extensió del codi que es necessita escriure.

La característica principal de la biblioteca és que permet canviar el contingut d'una pàgina web sense la necessitat de recargar-la, mitjançant la manipulació de l'arbre DOM i peticions AJAX.

jQuery és software lliure i codi obert, permeten el seu ús en projectes lliures i privats.

Un dels gran avantatges que té treballar amb JavaScript utilitzant jQuery, en lloc de treballar directament amb JavaScript, és que jQuery ens permet programar sense preocupar-nos del navegador amb el que ens visitarà l'usuari, ja que funcionen d'exacta forma en totes les plataformes més habituals, mentre que directament en JavaScript generalment ha de preocupar-se per fer scripts compatibles amb diferents navegadors i afegir codi per detectar el navegador de l'usuari.

Perquè s'ha escollit jQuery

És important comentar que jQuery no és l'únic framework que existeix al mercat. Existeixen varies solucions similars que també funcionen molt bé, que bàsicament serveixen per fer el mateix. Les altres opcions estudiades van ser Mootools i Prototype.

Després de molt comparar, els tres framework eren bastants semblants i amb qualsevol dels tres s'hagués pogut treballar i aconseguir el que es volia fer. jQuery té coses de sèrie, com pot ser l'agafar elements DOM amb la sintaxi pròpia de CSS. Això en Mootools es un Addon i pràcticament sempre s'ha de seleccionar i descarregar-lo. També jQuery fa les coses una mica més senzilles que Mootools. jQuery té una millor gestió d'esdeveniments que Prototype i una lleugera millora en la rapidesa de peticions.

Finalment, s'ha decantat per jQuery per que és un producte amb una acceptació per part dels programadors molt bona i un grau de penetració al mercat molt ampli. És un

[3] Acrònim de JavaScript Asíncron y XML. Tècnica de desenvolupament web per a crear aplicacions interactives. Aquestes s'executen al client, és a dir, al navegador dels usuaris mentre es manté una comunicació asíncrona amb el servidor en un segon pla.

producte serio, estable, ben documentat i amb un gran equip de desenvolupadors a càrrec de la millora i actualització.

Una altra cosa molt important és la dilatada comunitat de creadors de plugins o components, la qual cosa fa fàcil trobar solucions ja creades en jQuery.

Avantatges i inconvenients de l'ús de jQuery

A continuació s'enumeren les principals avantatges i inconvenients d'aquesta tecnologia.

Avantatges:

- Reducció del codi.
- Codi més concís i de fàcil lectura.
- Facilitat per operacions AJAX.
- Compatibilitat crossbrowser (multi-navegador).
- Reutilització de codi.
- Facilitat en el maneig d'arrays.

Inconvenients:

- Alguns desenvolupadors tenen problemes amb "this" en les funcions de callback.

5.3 MySQL

MySQL és un gestor de bases de dades relacionals fàcil d'utilitzar i increïblement ràpid. També és un dels motors de bases de dades més utilitzats a Internet. La principal raó d'això és perquè és gratuït per aplicacions no comercials.

Les característiques principals de MySQL són:

- **És un gestor de base de dades**. Partint de la idea de que una base de dades és un conjunt de dades organitzades, es pot dir que un gestor de bases de dades és una aplicació capaç de manejar tota la informació que aquestes tenen de manera eficient i còmoda.
- **És Open Source**. El codi font de MySQL es pot descarregar gratuïtament d'Internet, i és accessible per a tothom que el necessiti. Utilitza la llicència GPL per aplicacions no comercials, orientada principalment a protegir la lliure distribució, modificació i ús de software.
- **És una base de dades ràpida, segura i fàcil d'utilitzar**. Gràcies a la col·laboració dels usuaris, i al tractar-se d'una de les bases de dades més utilitzades a Internet, aquesta ha anat millorant optimitzant la seva velocitat i facilitat d'ús.
- **Existeix una gran quantitat de software que l'utilitza**. Degut a la seva gran popularitat, hi ha una gran varietat de software que s'ha adaptat per tal de poder-lo utilitzar.
- **La connexió entre PHP i MySQL és excel·lent**. Aquest és un punt que cal remarcar degut a l'elecció de PHP com a llenguatge dinàmic de l'aplicació.

5.4 CSS

CSS és un sistema i llenguatge creat per a la implementació de classes i atributs de disseny en les pàgines HTML, un estàndard establert pel W3C que pretén establir un mecanisme de disseny independent del contingut.

Mitjançant CSS és possible crear i definir classes o estils de taules, llistes, capes, text,... sense haver de definir els seus paràmetres de disseny cada vegada que són emprats en una pàgina web. Això ajuda a fer l'aplicació una mica més portàtil, poden ser integrada en diferents tipus de dispositius sense la necessitat d'haver de modificar el codi bàsic d'aquesta.

Es poden redefinir unes classes predeterminades d'una pàgina web, com són els encapçalaments (<H1>,..), els links (<A>) i altres elements. O es poden definir unes classes en concret, utilitzant nombres identificatius, que permetin ser cridats quan sigui necessari.

Mitjançant aquest sistema s'aconsegueix, a part d'una reducció en la mida del codi de les pàgines web, una gran facilitat en la modificació del disseny. Això es degut a que únicament modificant un atribut d'una classe determinada s'esta modificant implícitament el disseny de tots els elements que existeixen en la web pertanyents a aquesta classe.

5.5 FPDF

FPDF és una classe desenvolupada en PHP per a poder realitzar documents PDF dinàmicament a partir de scripts PHP. Aquesta classe treballa totalment autònoma, per la qual cosa no requereix utilitzar la llibreria PDFlib ni qualsevol altre producte similar.

Es tracte d'una classe amb moltes possibilitats, ja que es pot modificar la unitat de mesura, el format de les pàgines, els marges, les capçaleres i els peus de pàgina, els salts de línia, les imatges, colors, enllaços, ...

Aquest framework en PHP està desenvolupat amb orientació a objectes, essent l'objecte FPDF l'encarregat d'anar emmagatzemant l'estructura, i mostrant-la amb la funció Output, tenint diferents sortides tant per la pantalla com per la impressora o simplement oferint la possibilitat de descarregar l'arxiu. FPDF ofereix l'avantatge de permetre crear un PDF des de PHP amb una relativa senzillesa fent d'intermediari entre les funcions elementals de sortida de dades que pinten el PDF i l'usuari.

La primera F de FPDF significa *Free* (gratis i lliure), una llibreria gratuïta, tant comercial com professional, per tant resulta molt interessant per qualsevol ús.

Capítol 6

Disseny de l’eina

En aquest capítol s’explicarà el procés de creació de l’aplicació. No pretén ser un manual d’usuari de dita aplicació, si no que, es vol explicar com s’ha utilitzar la tecnologia descrita al capítol anterior.

Primer es parlarà del disseny de la base de dades mostrant el model conceptual i l’esquema relacional, tot explicant com ha anat evolucionant el seu disseny a mesura que s’ha avançat en l’aplicació. A continuació es mostrarà l’ús de jQuery, exposant algun exemple i la seva aplicació dins de l’eina. Posteriorment, es descriuran algunes funcions matemàtiques creades amb PHP i utilitzades al llarg de l’aplicació. Finalment, es detallarà el procediment utilitzat per crear els pdf’s necessaris amb l’eina FPDF mostrant algun exemple.

El disseny de la plataforma gira entorn als requeriments que es van especificar i que s’han procurat complir.

6.1 Disseny de la Base de Dades

Com s’ha exposat en els requeriments, s’ha emprat MySQL (4.3) com a gestor de la base de dades.

La Figura 6.1 mostra l’esquema relacional de la base de dades creada. La base de dades s’ha creat amb el suport del software DBDesigner 4.

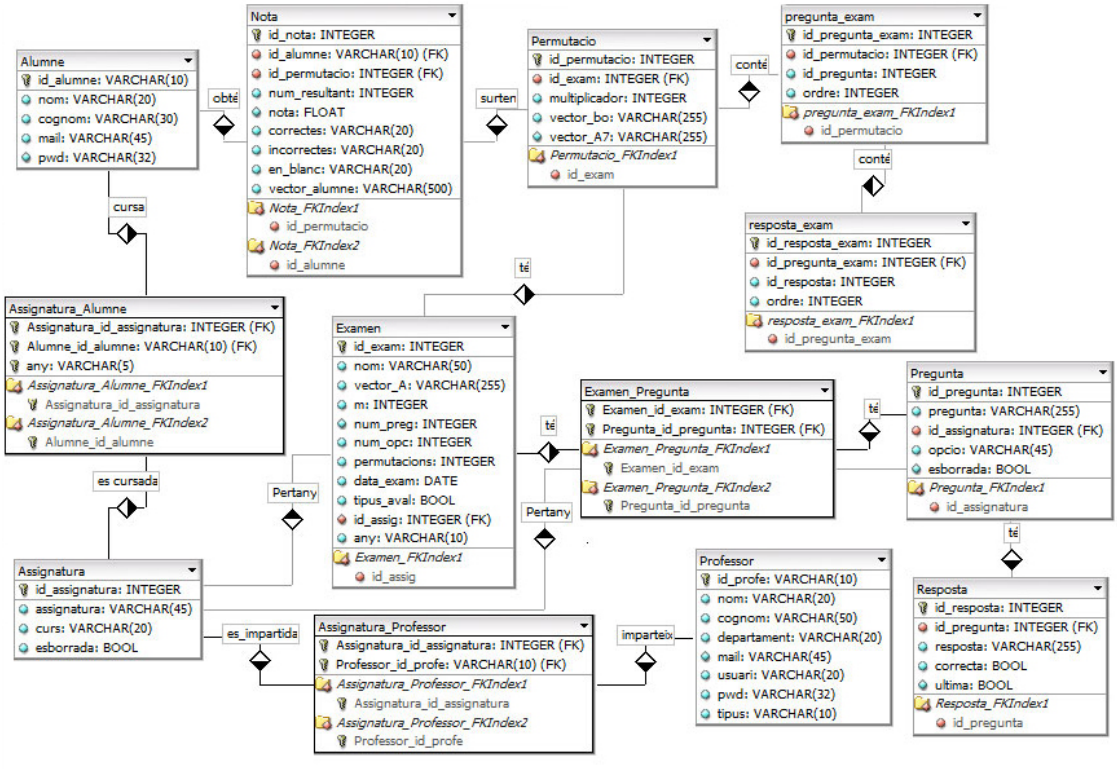


Figura 6.1: Esquema relacional de la base de dades definitiva

La base de dades ha canviat molt al llarg del desenvolupament de l'aplicació. A la Figura 6.2 es pot observar el model conceptual inicial i a la Figura 6.3 l'esquema relacional de la primera versió creada. Com es pot observar hi ha hagut molts canvis significatius degut a les necessitats que anaven sorgint.

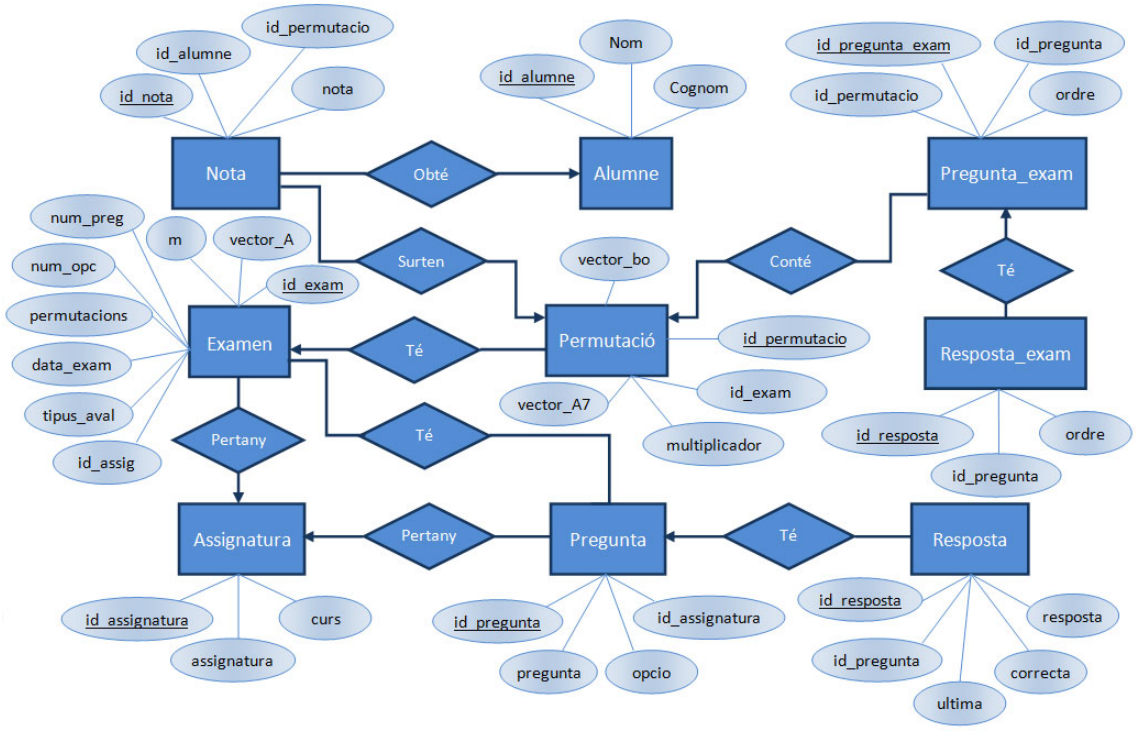


Figura 6.2: Model conceptual de la base de dades inicial

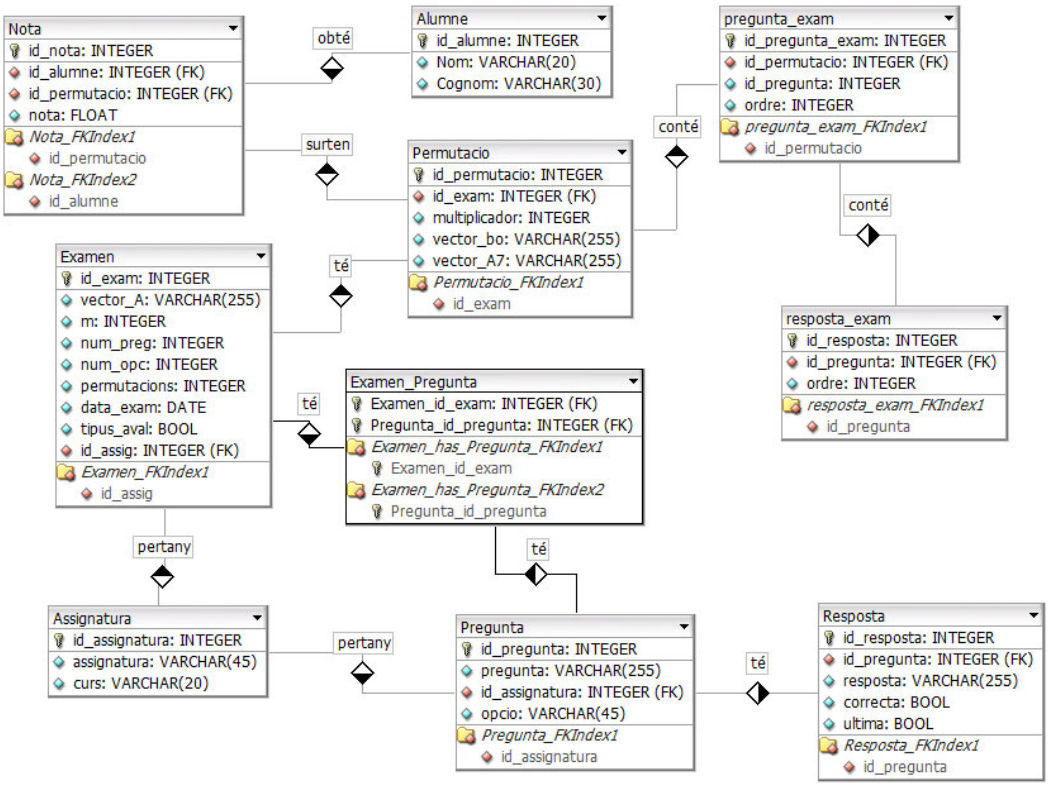


Figura 6.3: Esquema relacional de la base de dades inicial

En un primer moment no es va veure la necessitat de que haguessin usuaris registrats, ràpidament és va modificar i es va afegir la taula de professors i es va modificar la taula d'alumnes. Aquest canvi va provocar que l'atribut del camp id_alumne passés de ser *integer* i *auto_increment* a *varchar* i *no auto_increment* ja que s'emmagatzema el DNI.

Altres canvis que es van anar produint va ser la inserció de nous camps, per exemple, es va afegir el camp *esborrada* a la taula *Pregunta*. Amb aquest canvi es pretenia que a l'esborrar una pregunta no s'eliminés per complet de la base de dades si ja estava escollida en algun examen, sinó, al recuperar un examen anterior no trobava aquesta pregunta i es produïa un error.

A la taula *Nota* també s'han afegit diversos camps per tal de poder donar més informació sobre la nota de l'alumne, quines preguntes a respòs correctament, quines ha deixat en blanc,...

Un altre canvi a destacar és l'afegir el camp *any* a la taula *Examen* per saber en quin curs s'ha fet l'examen i que es pugui veure a l'historial de l'alumne totes les notes de tots els cursos acadèmics.

6.2 Utilització de jQuery

En aquesta secció es mostraran tots els casos diferents en els quals s'ha emprat jQuery. Només es donarà un exemple per cada cas, encara que s'ha pogut reutilitzar bastant codi, un dels avantatges d'aquesta tecnologia.

6.2.1 Eliminar

Cada cop que es vol eliminar dades, ja sigui un examen, una pregunta, una nota, una assignatura, un professor, ... s'utilitza una funció jQuery. Per exemple, quan es vol eliminar un examen s'utilitza el codi jQuery que mosta el Llistat 6.1.

```
82  $("a.elimina_exam").click(function(){
83      a1 = "/pfc/prof_consul_exam.php?id=";
84      a2 = $("#id_exam").html();
85      b1 = "&assig=";
86      b2 = $("#assig").html();
87
88      exam = $(this).parents("tr").find("td").eq(1).html();
89      id = $(this).parents("tr").find("td").eq(0).html();
90      respuesta = confirm("Desitja eliminar l'examen: " + exam + "?");
91      if (respuesta){
92          $(this).parents("tr").fadeOut("normal", function(){
93              $.post("varis.php", {del_exam: 1,id_ex:id}, function(data){
94                  if(data=='ok') {
95                      alert("Examen " + exam + " eliminat")
96                      adress = a1+a2+b1+b2;
97                      window.location.replace(adress)
98                  }
99                  else {alert("Hi ha hagut un error i no s'ha pogut eliminar l'examen. Torni a
    intertar-ho si us plau.")}
100             });
101         })
102     }
103 });
```

Llistat 6.1: Codi jQuery per eliminar exàmens

Mitjançant les línies 88 i 89 es recupera el nom de l'examen i l'identificador que es necessita per eliminar-lo. A la linia 90 es demana confirmació. Si la confirmació és positiva es crida l'arxiu *varis.php*. Dins aquest arxiu es troba el codi PHP necessari per eliminar de la base de dades l'examen escollit. El Llistat 6.2 mostra aquest codi en el qual es pot observar els passos seguits.

```
340 // Per eliminar un examen des del modul professor, es cridat per myjquery
341 if ($del_exam){
342
343     // Busco i elimino totes les preguntes i les respostes de cada permutació
344     $result_permu_del = mysql_query("SELECT * FROM permutacio WHERE id_exam='$id_ex'");
345     while ($row_permu_del = mysql_fetch_array($result_permu_del))
346         {
347         $result_pregexam_del = mysql_query("SELECT * FROM pregunta_exam WHERE
    id_permutacio='$row_permu_del[id_permutacio]'");
348         while ($row_pregexam_del = mysql_fetch_array($result_pregexam_del))
349             {
350             $sql_del_resposexam = "DELETE FROM resposta_exam WHERE
    id_pregunta_exam=$row_pregexam_del[id_pregunta_exam]";
351             $result_del_resposexam = mysql_query($sql_del_resposexam);
352             }
353
354         $sql_del_pregexam = "DELETE FROM pregunta_exam WHERE
    id_permutacio=$row_permu_del[id_permutacio]";
355         $result_del_pregexam = mysql_query($sql_del_pregexam);
356         }
357
358     // Elimino totes les premutacions de l'examen
359     $sql_del_permu = "DELETE FROM permutacio WHERE id_exam=$id_ex";
360     $result_del_permu = mysql_query($sql_del_permu);
361     // Elimino l'examen
362     $sql_del_exam = "DELETE FROM examen WHERE id_exam=$id_ex";
363     $result_del_exam = mysql_query($sql_del_exam);
364     // Elimino la relació de l'examen amb les preguntes escollides per aquest examen
365     $sql_del_exampreg = "DELETE FROM examen_pregunta WHERE examen_id_exam=$id_ex";
366     $result_del_exampreg = mysql_query($sql_del_exampreg);
367
368     echo 'ok';
369 }
```

Llistat 6.2: Codi PHP per eliminar exàmens

Mitjançant aquest codi s'elimina l'examen, i totes les entrades relacionades amb l'examen, com són les permutacions creades, la relació de totes les preguntes i respostes de cada permutació, així com la relació entre l'examen i les preguntes que surten en aquest examen.

Una vegada tot ha anat correctament, es retorna un *ok*. El codi jQuery rep l'*ok*, llavors informa que l'examen s'ha eliminat amb èxit i es torna a recarregar la pàgina per a que s'actualitzi la pantalla amb les noves modificacions.

6.2.2 Editar

Quan es vol modificar una dada sempre existeix l'opció d'editar-la. Amb aquesta opció podrem modificar les dades dels professors i dels alumnes, el nom de les assignatures i les notes inserides. El Llistat 6.3 mostra el codi jQuery necessari per modificar les dades d'un professor.

```
551 // Modificar els profes amb les noves dades
552 $("a.edit_profe").click(function(){
553     id = $(this).parents("tr").find("td").eq(0).html();
554     id2 = $("#cognomprofe_" + id + "").get(0).value;
555     id3 = $("#nomprofe_" + id + "").get(0).value;
556     id4 = $("#departamentprofe_" + id + "").get(0).value;
557     id5 = $("#mailprofe_" + id + "").get(0).value;
558 
559     a1 = "/pfc/admin_profe.php";
560 
561     respuesta = confirm("Desitja modificar les dades del professor amb DNI " + id + "?");
562     if (respuesta){
563         $(this).parents("tr").fadeOut("normal", function(){
564             $.post("varis.php", {modificar_profe: 1, dni: id, cognom: id2, nom: id3, depart: id4
    , mail: id5}, function(data){
565                 if(data=='ok') {
566                     alert("Dades modificades")
567                     adress = a1;
568                     window.location.replace(adress)
569                 }
570                 else {alert("Hi ha hagut un error i no s'ha pogut modificar les dades. Torni a
    intertar-ho si us plau.")}
571             });
572         })
573     }
574 });
```

Llistat 6.3: Codi jQuery per modificar les dades dels professors

El codi de les línies 553 a 557 recupera les dades necessàries. A la línia 561 es demana confirmació per modificar les dades del professor seleccionat. Com es veia amb anterioritat, a la línia 564 es crida a l'arxiu *varis.php* i se li passen les dades recuperades en les primeres línies. El Llistat 6.4 reflecteix aquest codi PHP.

```
199 // Per modificar dades dels professors des del modul d'administrador, es cridat per myjquery
200 if ($modificar_profe){
201 
202     $cognom=utf8_decode($cognom);
203     $nom=utf8_decode($nom);
204     $depart=utf8_decode($depart);
205     $sql_edit_profe = "UPDATE professor SET cognom='$cognom', nom='$nom', departament='$depart',
     mail='$mail' WHERE id_profe='$dni'";
206     $result_edit_profe = mysql_query($sql_edit_profe);
207     // Comprovar si s'ha modificat les dades
208     if( $result_edit_profe )    echo 'ok';
209 }
```

Llistat 6.4: Codi PHP per modificar les dades dels professors

Aquest codi rep les noves dades del professor i la línia 205 les actualitza. Anteriorment, les línies 202 a 204 serveixen per emmagatzemar correctament els caràcters especials, com poden ser els accents. A la línia 208 es comprova que s'hagin actualitzat les dades correctament, llavors retorna un *ok*.

Al rebre l'*ok*, el codi jQuery mostra per pantalla un missatge d'èxit i actualitza per poder visualitzar correctament els canvis.

6.2.3 Inserir

Quan es volen inserir dades noves també es fa mitjançant jQuery. Quan s'insereix una assignatura nova, un nou professor o un nou alumne es fa mitjançant codi jQuery. Quan es vol assignar una assignatura a un professor o a un alumne també s'utilitza. Finalment, es fa ús de jQuery per saber la nota d'un examen i emmagatzemar tota la informació del resultat a la base de dades.

El Llistat 6.5 mostra el codi jQuery necessari per inserir un nou professor. Com es pot comprovar, el codi és molt similar als anteriors analitzats. Primer es recuperen les dades a inserir i es demana confirmació. Si la confirmació es positiva es crida a *varis.php* i li passem les dades. Si tot ha anat correctament, surt una finestra emergent amb la confirmació i es recarrega la pàgina amb el nou professor.

```
// Inserir professor des del modul d'administrador
$("a.ficar_profe").click(function(){
    id1 = $("#nou_dni_profe").get(0).value;
    id2 = $("#nou_cognom_profe").get(0).value;
    id3 = $("#nou_nom_profe").get(0).value;
    id4 = $("#nou_depart_profe").get(0).value;
    id5 = $("#nou_mail_profe").get(0).value;

    a1 = "/pfc/admin_profe.php";

    respuesta = confirm("Desitja afegir el professor amb DNI " + id1 + "?");
    if (respuesta){
        $(this).parents("tr").fadeOut("normal", function(){
            $.post("varis.php", {insertar_profe: 1, dni: id1, cognom: id2, nom: id3, depart: id4
, mail: id5}, function(data){
                if(data=='ok') {
                    alert("El professor amb DNI " + id1 + " s'ha donat d'alta correctament")
                    adress = a1;
                    window.location.replace(adress)
                }
                else {alert("Hi ha hagut un error i no s'ha pogut donar d'alta al/a professor/a.
 Torni a intertar-ho si us plau.")}
            });
        })
    }
});
```

Llistat 6.5: Codi jQuery per inserir un nou professor

Al Llistat 6.6 es pot apreciar el codi PHP que complementa aquesta funció jQuery. Quan es dona d'alta un nou professor l'aplicació li envia automàticament un e-mail amb les seves noves dades. Per defecte, s'utilitza com a nom d'usuari i contrasenya el seu DNI. Posteriorment, el professor hauria de canviar la contrasenya per qüestions lògiques de seguretat. A la línia 125 s'observa com s'utilitza l'algoritme criptogràfic MD5 per emmagatzemar, posteriorment, la contrasenya a la base de dades encriptada, guanyant amb seguretat. A la línia 133 es pot veure el codi necessari per l'enviament de l'e-mail.

```
// Per inserir professors des del modul d'administrador, es cridat per myjquery
if($insertar_profe) {

        $cognom=utf8_decode($cognom);
        $nom=utf8_decode($nom);
        $depart=utf8_decode($depart);
        // Utilitzo el dni com a nom d'usuari i pwd (encriptat amb MD5)
        $pwd=md5($dni);
        // Envio un mail al professor per a que sapigui el nom d'usuari i el pwd
        $subject="Contrasenya servei correcció examens";
        $message="El seu nom d'usuari i la seva contrasenya per accedir al servei de correcció
automàtica d'examens és: ".$dni;
        $cabeceras = 'From: xavibaldor@hotmail.com' . "\r\n" .
            'Reply-To: xavibaldor@hotmail.com' . "\r\n" .
            'X-Mailer: PHP/' . phpversion();
        //Enviem un mail a l'alumne amb el seu pwd
        mail($mail,$subject,$message,$cabeceras);

        $sql = mysql_query("INSERT INTO professor VALUES
('$dni','$nom','$cognom','$depart','$mail','$dni','$pwd','profe')");
        if( $sql )  echo 'ok';
}
```

Llistat 6.6: Codi PHP per inserir un nou professor

Quant es vol saber el resultat d'un examen també s'utilitza jQuery. Tal com està explicat al cas d'ús (2.3.5), abans de començar s'ha d'haver realitzat l'examen per part de l'alumne, llavors es tenen tres dades necessàries, el DNI de l'alumne, la versió de l'examen i el número resultant de l'examen. Al Llistat 6.7 s'observa el codi jQuery necessari per dur a terme aquesta tasca. Aquest codi és el mateix tant pel mòdul professors com pel mòdul alumne, ja que tant els professors com els alumnes tenen la possibilitat de desxifrar la nota de l'examen. Evidentment, la nota quedarà enregistrada a la base de dades.

```
292 // Inserir notes des del modul del professor
293 $("a.ficar_nota").click(function(){
294     id = $(this).parents("tr").find("td").eq(0).html();
295     id2 = $("#dni_insert").get(0).value;
296     id3 = $("#versio_insert").get(0).value;
297     id4 = $("#resultant_insert").get(0).value;
298 
299     a1 = "/pfc/prof_consul_notes.php?exam=";
300     a2 = $("#ex_del_nota").html();
301     b1 = "&assig=";
302     b2 = $("#assig_del_nota").html();
303     c1 = "&nom=";
304     c2 = $("#nom_del_nota").html();
305 
306     respuesta = confirm("Desitja inserir la nota de l'alumne " + id2 + "?");
307     if (respuesta){
308         $(this).parents("tr").fadeOut("normal", function(){
309             $.post("varis.php", {insertar_nota: 1, dni: id2,versio: id3,resultat: id4}, function(data){
310                 if(data=='ok') {
311                     alert("Nota inserida correctament")
312                     adress = a1+a2+b1+b2+c1+c2;
313                     window.location.replace(adress)
314                 }
315                 else {
316                     if (data=='ko1') {alert("Ja s'ha inserit una nota per aquesta versió.")}
317                     if (data=='ko2') {alert("El DNI inserit no és correcte.")}
318                     if (data=='ko3') {alert("La versió inserida no és correcta.")}
319                 }
320             });
321         })
322     }
323 });
```

Llistat 6.7: Codi jQuery per inserir la nota de l'examen

El codi jQuery emprat és molt similar als casos vistos anteriorment. Com sempre primer es recuperen les variables, es demana confirmació i es crida a *varis.php* passant totes les variables. Si tot es correcte surt un missatge de confirmació i es recarrega la pàgina web per poder visualitzar la nota i les seves dades. Si hi ha hagut algun error s'especifica quin. Pot haver-hi tres errors, que la nota ja estigui ficada, és a dir, que la versió inserida ja té associada una nota, que el DNI ficat no sigui correcte, que no hi hagi cap alumne amb aquest DNI, o que la versió no existeixi, que hagi hagut un error al teclejar la versió. En un principi hauria de ser bastant difícil que es produeixin els errors, ja que, tan a l'inserir el DNI com la versió, surt un desplegable amb els DNI's de tots els alumnes matriculat i les versions que encara no tenen nota assignada, encara que s'ha estimat adient controlar els possibles errors per si s'escriu el DNI o la versió a mà sense fer ús d'aquests desplegables.

Al Llistat 6.8 es pot veure el codi PHP per dur a terme aquest propòsit. De la línia 216 a 221 es on es tracten els possibles errors explicats anteriorment. A partir de la línia 222 comença el procés per saber la nota de l'examen. Com ja s'ha vist al capítol 3, es necessita l'invers del multiplicador (línia 227) utilitzant la funció pròpia *Bezout* (6.3). Per saber la nota es crida a la funció, també pròpia, *Desxifrar* (6.3) (línia 229). Ja s'obté el resultat de l'examen. Només queda emmagatzemar-lo en la base de dades.

```php
// Per inserir notes des del modul del professor i de l'alumne, es cridat per myjquery
if($insertar_nota) {
    $cerca_versio= mysql_query("SELECT id_permutacio FROM nota WHERE id_permutacio='$versio'");
    $cerca_dni= mysql_query("SELECT id_alumne FROM alumne WHERE id_alumne='$dni'");
    $result_permu = mysql_query("SELECT * FROM permutacio WHERE id_permutacio='$versio'");
    if((mysql_num_rows($cerca_versio)!=0) || (mysql_num_rows($cerca_dni)==0) || (mysql_num_rows($result_permu)==0)) {
        if (mysql_num_rows($cerca_versio)!=0) {
          echo "ko1";
        } elseif (mysql_num_rows($cerca_dni)==0) {
          echo "ko2";
        } else {echo "ko3";}
    } else {
        $row_permu = mysql_fetch_array($result_permu);
        $result_exam = mysql_query("SELECT * FROM examen WHERE id_exam='$row_permu[id_exam]'");
        $row_exam = mysql_fetch_array($result_exam);
        // Calculem l'invers del multiplicador
        $t1=Bezout($row_permu[multiplicador],$row_exam[m]);
        // Cridem a Desxifrar per saber les respostes de l'alumne
        $vector_alumne=Desxifrar($resultat,$t1,$row_exam[m],$row_exam[vector_A],$row_exam[num_preg],$row_exam[num_opc]);
        if ($vector_alumne!=-1) {
            // Cridem a Corregir per saber la nota
            $id_nota=Corregir($vector_alumne,$row_permu[vector_bo],$row_exam[num_preg],$row_exam[num_opc],$row_exam[tipus_aval]);
            // Inserim els resultats a la taula nota
            $sql = mysql_query("UPDATE nota SET id_alumne='$dni', id_permutacio='$versio', num_resultant='$resultat', vector_alumne='$vector_alumne' WHERE id_nota=$id_nota");
            echo 'ok';
        } else {echo 'ko';}
    }
}
```

Llistat 6.8: Codi PHP per inserir la nota de l'examen

6.2.4 Format

jQuery també és emprat per modificar el format de la web. Amb poques línies de codi es pot aconseguir un format dinàmic o estalviar-se'n moltes.

S'ha utilitzat per l'estil de les taules. Les taules es carreguen de la base de dades i mitjançant jQuery es fa que el fons de les línies senars surtin amb color i el de les parells en blanc. La Figura 6.4 mostra un exemple.

Assignatura	Curs	Editar	Eliminar
Algebra	1r ESO		
Física	4t ESO		
Informàtica	4t ESO		
Llengua catalana	4t ESO		
Matemàtiques	3r ESO		
Matemàtiques	4t ESO		

Figura 6.4: Format de les taules

Per aconseguir aquest efecte, al Llistat 6.9 es pot veure el codi jQuery emprat. El funcionament és senzill, quan detecta una línia senar crida a la classe *odd*, i quan la línia es parell crida a la classe *even*. El Llistat 6.10 mostra el format CSS d'aquestes dues classes.

```javascript
// Taules admin
$('#taula_admin tr:not([class^="editassig"]):odd').addClass('odd');
$('#taula_admin tr:not([class^="editassig"]):even').addClass('even');
```

Llistat 6.9: Codi jQuery per donar format a les taules

```css
.odd {
    background-color: #E5E0EC;
}

.even {
    background-color: #FFFFFF;
}
```

Llistat 6.10: Codi CSS per donar format a les taules

El disseny de la pàgina web s'ha creat mitjançant *div's* i descomponent les parts de la web en arxius php, per exemple, la capçalera de la web s'ha creat dins l'arxiu *header.php*, el menú de l'esquerra dins *menu.php*,... Dins del *div* corresponent es crida l'arxiu necessari i aquest es carrega mitjançant jQuery. El Llistat 6.11 presenta el codi PHP emprat i el Llistat 6.12 el codi jQuery corresponent.

```
29      <div id="header"></div>
30
31      <div id="esq"></div>
32
```

Llistat 6.11: Codi PHP per carregar arxius

```
59  // Carreguem les pag header,menu i footer
60  $(document).ready(function() {
61    $('#header').load('header.php');
62    $('#header2').load('header2.php');
63    $('#esq').load('menu.php');
64    $('#esq_alumne').load('menu_alumne.php');
65    $('#esq_admin').load('menu_admin.php');
66    $('#footer').load('footer.php');
67  });
```

Llistat 6.12: Codi jQuery per carregar arxius

Va sorgir la necessitat de que l'amplada d'una taula variés al ficar el mouse a sobre d'aquesta. Per aconseguir aquest efecte també s'ha emprat jQuery. A la Figura 6.5 s'observa aquest canvi d'amplada aconseguit gràcies al codi jQuery que apareix al Llistat 6.13. Al Llistat 6.14 es visualitza el codi PHP necessari per al bon funcionament, simplement s'ha utilitzat l'atribut *id* del tag *table* i un parell de tags *span*.

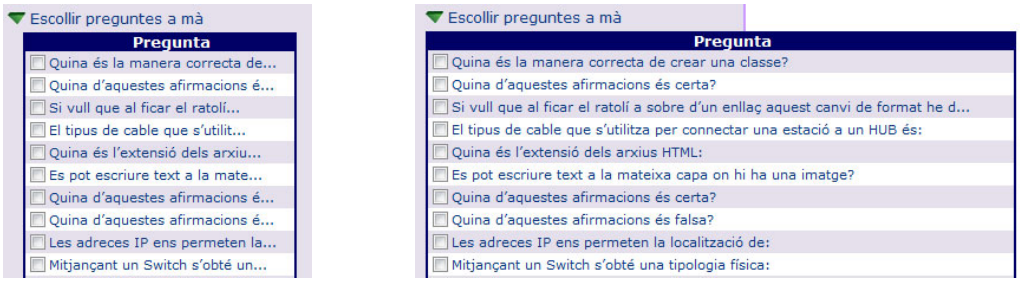


Figura 6.5: Canvi d'amplada d'una taula

```
275 <table width="90%" border="0" cellspacing="0"  cellpadding="0" class="cospetit" id="taula_crear_exam" align="center">
```

Llistat 6.13: Codi PHP per modificar l'amplada d'una taula

```
596 // A l'escollir preguntes a mà, per a que es faci la taula més gran al ficar el mouse a sobre per
    poder llegir més part de la pregunta.
597   $('#taula_crear_exam').hover(
598     function() {
599         $("#taula_crear_exam").attr('width',500);
600         $('span.esc_preg_curt').hide();
601         $('span.esc_preg_llarg').show();
602         $('label.espai_blanc').show();
603     },
604     function() {
605         $("#taula_crear_exam").attr('width','90%');
606         $('span.esc_preg_llarg').hide();
607         $('span.esc_preg_curt').show();
608         $('label.espai_blanc').hide();
609     }
610   );
```

Llistat 6.14: Codi jQuery per modificar l'amplada d'una taula

Una altra necessitat que va sorgir va ser la de poder modificar el format de les taules desplegant informació addicional i, evidentment, es va emprar jQuery per aquest propòsit. La Figura 6.6 presenta un exemple del que s'ha aconseguit, en aquest cas en concret, es té un llistat dels professors i mitjançant la icona es desplegua informació addicional com són les assignatures que dona el professor.

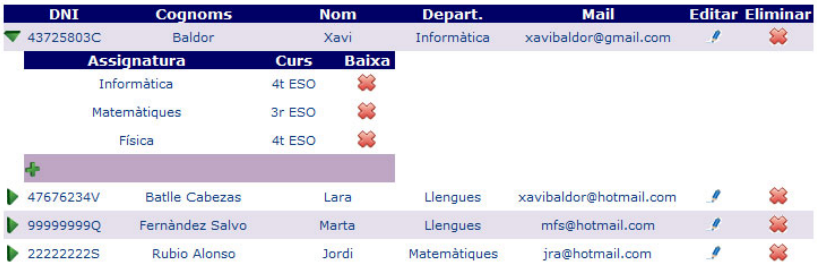


Figura 6.6: Taula desplegada amb informació addicional

Al Llistat 6.15 apareix el codi jQuery necessari. La informació addicional només es mostrarà quan es fa clic a sobre de la icona . Aquesta icona es situa dins una etiqueta *span* amb el nom *masprofe*. La informació a mostrar es troba dins una fila de la taula, *tr*, amb el nom *profe_veure_assig* seguit de l'identificador del professor. Quan es fa clic a sobre de la icona s'entra a la funció que es veu a la línia 737. Aquesta funció fa que es visualitzi la informació addicional, línia 740, del professor escollit gràcies al seu identificador capturat a la línia 738. Al mateix temps la icona canvia, . Si es fa clic sobre aquesta nova icona, s'entrarà a la segona funció, línia 745, que fa que la informació ja no es visualitzi, línia 748, i que la icona torni a la imatge inicial.

```
733  // Per visualitzar/ocultar les assignatures dels professors a Consultar Professors des del modul
     Administrador
734  $(document).ready(function() {
735    $('tr[class^="profe_veure_assig"]').hide();
736    $('td[class^="profe_noveure_assig"]').hide();
737    $('span.masprofe').click(function() {
738      id_alu = $(this).parents("tr").find("td").eq(0).html();
739      if (id_alu){
740          $('tr.profe_veure_assig'+id_alu).fadeIn('slow');
741          $('td.profe_veure_assig'+id_alu).hide();
742          $('td.profe_noveure_assig'+id_alu).show();
743      }
744    });
745    $('span.menysprofe').click(function() {
746      id_alu = $(this).parents("tr").find("td").eq(0).html();
747      if (id_alu){
748          $('tr.profe_veure_assig'+id_alu).fadeOut('slow');
749          $('td.profe_noveure_assig'+id_alu).hide();
750          $('td.profe_veure_assig'+id_alu).show();
751      }
752    });
753  });
```

Llistat 6.15: Codi jQuery per desplegar informació addicional

A les seccions 6.2.2 i 6.2.3 s'ha vist com inserir o editar informació, però per poder realitzar-ho el format varia, ha d'aparèixer uns camps per poder escriure la nova informació o modificar-ne la existent. La Figura 6.7 ofereix un exemple de com es modifica el format d'una taula que presenta dades, en aquest cas dels professors, i al fer clic a sobre de la icona d'editar, apareixen camps *inputs* per poder escriure les modificacions.

Figura 6.7: Aparició de camps per poder modificar les dades

El funcionament és semblant a l'anterior. El Llistat 6.16 mostra el codi jQuery emprat. Dins el codi PHP, s'ha creat dues línies, *tr*, per cada professor. En una línia es mostra la informació del professor i a l'altra apareixen els camps *inputs* amb les dades del professor preparades per ser modificades. En un primer moment només es visualitza la línia amb les dades. Al codi jQuery es tenen dues funcions, la primera funció es cridada al fer clic a sobre de la icona d'editar, línia 942. En aquesta funció es recupera l'identificador del professor, línia 943, i fa que aparegui la línia amb els *inputs* (línia 946) i que s'oculti la línia amb la informació (línia 945). Quan ja s'ha modificat les dades o s'ha cancel·lat el procés tot torna com al principi gràcies a la segona funció, línia 949.

```
939 // Carreguem camps inputs per poder editar els professors i modificar-los
940 $(document).ready(function() {
941   $('tr[class^="edit"]').hide();
942   $('span.editarprofe').click(function() {
943     id_profe = $(this).parents("tr").find("td").eq(0).html();
944     if (id_profe){
945         $('tr.noeditprofe'+id_profe).hide();
946         $('tr.editprofe'+id_profe).show();
947     }
948   });
949   $('span.noeditarprofe').click(function() {
950     id_profe = $(this).parents("tr").find("td").eq(0).html();
951     if (id_profe){
952         $('tr.editprofe'+id_profe).hide();
953         $('tr.noeditprofe'+id_profe).show();
954     }
955   });
956 });
957
```

Llistat 6.16: Codi jQuery per fer aparèixer camps per poder modificar les dades

6.2.5 Altres aplicacions

També s'ha usat jQuery per alguns cassos més concrets. Quan es crea un examen escollim el número de versions en concret, però també es té l'opció de poder crear alguna versió més en el moment que es vulgui. Per poder crear aquestes noves versions s'ha utilitzat jQuery. El funcionament es similar als explicats en els apartats anteriors, apareix un camp on es pot escollir el número d'exàmens a crear tal i com mostra la Figura 6.8, i el codi jQuery que mostra el Llistat 6.17 crida a *varis.php* on es fan els càlculs necessaris, molt semblants als de crear un examen nou i que es veurà en el següent apartat (6.3).

```
382 // Crear més versions dels exàmens des del modul del professor
383 $("a.crear_versions").click(function(){
384     id = $(this).parents("tr").find("td").eq(0).html();
385     id2 = $("#quant_versions").get(0).value;
386     id3 = $(this).parents("tr").find("td").eq(1).html();
387
388     a1 = "/pfc/prof_consul_exam.php?exam=";
389     a2 = id;
390     b1 = "&assig=";
391     b2 = id3;
392     c1 = "&nom=";
393     c2 = $("#nom").html();
394
395     respuesta = confirm("Desitja crear " + id2 + " versions més per l'examen de " + c2 + "?");
396     if (respuesta){
397         $(this).parents("tr").fadeOut("normal", function(){
398             $.post("varis.php", {crear_versions: 1, id_exam: id, quant: id2}, function(data){
399                 if(data=='ok') {
400                     alert("Les versions s'han creat correctament.")
401                     adress = a1+a2+b1+b2+c1+c2;
402                     window.location.replace(adress)
403                 }
404                 else {alert("Hi ha hagut un error i no s'ha pogut crear les noves versions. Torni a
 intertar-ho si us plau.")}
405             });
406         })
407     }
408 });
```

Llistat 6.17: Codi jQuery per afegir noves versions

Figura 6.8: Afegir noves versions

Quan es crea un examen es té la possibilitat d'escollir les preguntes aleatòriament o poder escollir algunes o totes les preguntes entre les que hi ha disponibles a la base de dades. Davant de cada pregunta hi ha un box per poder escollir-la, tal i com s'ha vist a la Figura 6.5. A mesura que s'escull una pregunta s'emmagatzema l'elecció i, a més, es comptabilitza la pregunta per a no escollir més preguntes de les permeses.

```
612 // Per enregistrar les preguntes que es van escollint a mà.
613   $('.esco_preg').click(function() {
614     id_pregunta = $(this).parents("tr").find("td").eq(0).html();
615     array_pregunta = $("#escollides").get(0).value;
616     max_pregunta = $("#preg").get(0).value; // Miro les preg que té l'examen
617     if (array_pregunta=="") {
618         array_pregunta=id_pregunta;
619         $("#escollides").attr('value', array_pregunta);
620     }
621     else {
622         $.post("varis.php", {proces_escollir_preg: 1, id_preg:id_pregunta, max_preg:max_pregunta,
    array_preg:array_pregunta}, function(data){
623             if(data=='ok' || data=='max') {
624                 array_pregunta=array_pregunta+","+id_pregunta;
625                 $("#escollides").attr('value', array_pregunta);
626                 if(data=='max') {$(".esco_preg").attr('disabled',true);}
627             }
628             else{
629                 $("#escollides").attr('value', data);
630             }
631         });
632     }
633   });
```

Llistat 6.18: Codi jQuery per emmagatzemar les preguntes escollides a mà

```
428 // Per escollir preguntes a mà al crear un examen nou des del modul professor, es cridat per
    myjquery
429 if($proces_escollir_preg){
430     $array_preguntes = explode(",",$array_preg);
431     $llargada=count($array_preguntes);
432     if (in_array($id_preg, $array_preguntes)) { // Si s'ha tornat a escollir la mateixa pregunta
    vol dir que es deselecciona
433         for($i=0;$i<$llargada;$i++){
434             if($array_preguntes[$i]==$id_preg){
435                 unset($array_preguntes[$i]);
436             }
437         }
438         $array_preguntes = array_values($array_preguntes);// elimino l'espai que ha quedat despres
    d'eliminar-se
439         $array_preguntes = implode(",",$array_preguntes);
440         echo $array_preguntes;
441     }
442     else if ($llargada==$max_preg-1) {  echo 'max'; }
443     else { echo 'ok';}
444 }
```

Llistat 6.19: Codi PHP per emmagatzemar les preguntes escollides a mà

Els Llistats 6.18 i 6.19 reflecteixen el codi jQuery i PHP necessari per dur a terme aquesta finalitat. L'objectiu es anar emmagatzemant les preguntes escollides en un array (línia 624, Llistat 6.18), i si es seleccionen el nombre total de preguntes que té l'examen, ja no deixa escollir més preguntes (línia 626, Llistat 6.18). Si es deselecciona una pregunta, aquesta s'elimina de l'array (línies 432 a 438, Llistat 6.19).

6.3 Funcions matemàtiques en PHP

Per poder aconseguir l'objectiu de l'aplicació, i com s'ha anat explicant al llarg de la memòria, s'ha utilitzat bastants operacions matemàtiques necessàries en criptografia. Recordar que l'aplicació es basa en el criptosistema knapsack (4.1), i com tot criptosistema, les matemàtiques són fonamentals. S'ha creat un arxiu, *math.php*, dins el qual s'han creat un seguit de funcions que es van cridant a mesura que es necessiten. Algunes de les funcions s'han pogut aprofitar en més d'una pàgina de l'aplicació. En aquesta secció es mostrarà algunes d'aquestes funcions creades.

6.3.1 Identitat de Bézout (3.1.3)

El Llistat 6.20 mostra el codi de la funció *Bezout*. Aquesta funció es crida al crear l'examen i serveix per trobar l'invers d'un número t mòdul m (4.1), paràmetres passats a la funció.

```
//Identitat de Bézout.
//Fem extés d'Euclides per calcular l'invers del multiplicador t.
function Bezout($t,$m) {
    $a=$t;
    $f=$m;
    $quocient=array();
    $reste=array();
    $rs=array();
    $resul=array();
    $temp=0;
    $i=0;

    //Baixem
    while ($temp==0) {
        $q=floor($f/$a);
        $r=fmod($f,$a);
        if ($r!=0) {
            $q1=$q*(-1);
            $quocient[]=$q1;
            $cont=$i;
            }
        $reste[]=$r;
        $r=$reste[$i];
        if ($reste[$i]==1 || $reste[$i]==0) $temp=1;
        $f=$a;
        $a=$r;
        $i++;
    }

    //Pujem
    $quocient=array_reverse($quocient);
    $rs[0]=1;
    $rs[1]=$quocient[0];
    $i=1;
    while ($i<=$cont) {
        $temp = $quocient[$i]*$rs[1]+$rs[0];
        $rs[0]=$temp;
        $rs=array_reverse($rs);
        $i++;
    }
    $invers=$rs[1];
    if ($invers<0) $invers=$invers+$m;
    return $invers;
}
// Fi funció Bezout
```

Llistat 6.20: Codi PHP de la funció Bezout

6.3.2 Desxifrar l'algorisme knapsack

Una altra funció creada es la funció *Desxifrar*, cridada al corregir els exàmens, que serveix, com el seu nom indica, per desxifrar el número resultant de l'examen basat en l'algorisme knapsack. Gracies a aquesta funció es poden saber les respostes que ha escollit l'alumne. Al Llistat 6.21 s'observa el codi emprat.

```php
// Desxifrar.
// Primer desxifrem el resultat de l'examen fent un producte modular amb t'.
// Posteriorment comparem el resultat amb el vector A per saber quines respostes ha escollit l'alumne.
function Desxifrar ($resposta,$multi,$mod,$A,$preguntes,$opc) {
    $elements=$preguntes*$opc;
    $resul=array();
    $num=Modul($resposta,$multi,$mod);
    $array_A=array();
    $array_A=explode(",",$A);
    $array_A=array_reverse($array_A);
    $i=0;
    while ($i<$elements) {
        if ($num>=$array_A[$i]) $resul[]=1;
        else $resul[]=0;
        if ($num>=$array_A[$i]) $num=$num-$array_A[$i];
        $i++;
    }
    $result=array_reverse($resul);

    // Comprovem que el vector resultant sigui correcte, que el numero inserit per l'alumne sigui correcte
    // De moment nomes si hi ha 3 opcions per pregunta
    $k=0;
    $error=0;
    while ($k<$elements) {
            if (($result[$k] == 1 && $result[($k+1)] == 1) || ($result[($k+1)] == 1 && $result[($k+2)] == 1) || ($result[$k] == 1 && $result[($k+2)] == 1)) {$error=1;}
        $k=$k+3;
    }

    if ($error==1) return -1; else return $result=implode(",",$result);
}
// Fi funció Desxifrar
```

Llistat 6.21: Codi PHP de la funció Desxifrar

6.3.3 Corregir exàmens

Una altra funció destacada, i també cridada al corregir l'examen, és la funció *Corregir*. Mitjançant aquesta funció s'obté la nota de l'alumne, així com les respostes correctes, les respostes incorrectes i les respostes que ha deixat en blanc. El Llistat 6.22 ofereix el codi d'aquesta funció.

```php
// Correcció de l'examen.
// Si la resposta es correcta li sumem un punt, despres calculem la nota real depenent de les preguntes de l'examen.
function Corregir ($vector_resul,$vector_bo,$preguntes,$opc,$tipus) {
    $elements=$preguntes*$opc;
    $resp_correc=0;
    $resp_incorrec=0;
    $i=0;
    $correctes=array();
    $incorrectes=array();
    $en_blanc=array();
    $vectorbo=array();
    $vectorbo=explode(",",$vector_bo);
    $resul=array();
    $resul=explode(",",$vector_resul);
    while ($i<$elements) {
        if ($vectorbo[$i]==1 && $resul[$i]==1) {
            $resp_correc++;
            $pregunta=floor($i/3)+1;
            $correctes[]=$pregunta;
        }
        if ($vectorbo[$i]==0 && $resul[$i]==1) {
            $resp_incorrec++;
            $pregunta=floor($i/3)+1;
            $incorrectes[]=$pregunta;
            }
        if ($i%3 == 0) {
            if ($resul[$i]==0 && $resul[$i+1]==0 && $resul[$i+2]==0) {
                $pregunta=floor($i/3)+1;
                $en_blanc[]=$pregunta;
                }
            }
        $i++;
    }
    if ($tipus == 1) {
        $nota=(($resp_correc-$resp_incorrec*(1/($preguntes-1)))*10)/$preguntes;
    } else {
        $nota=($resp_correc*10)/$preguntes;
    }
```

```
240     $correct=implode(',',$correctes);
241     $incorrect=implode(',',$incorrectes);
242     $blanc=implode(',',$en_blanc);
243     $nota = round($nota*100)/100;
244     // Si la nota es negativa, ja que les respostes dolentes resten, li fiquem un 0
245     if ($nota<0) {$nota=0;}
246
247     // Fiquem les preguntes correctes, les incorrectes i la nota a la taula nota
248     $q1 = "INSERT INTO nota SET nota='$nota', correctes='$correct', incorrectes='$incorrect',
en_blanc='$blanc'";
249     mysql_query ($q1);
250     // Recuperem el id_nota per tornar-lo i podem afegir els camps que falten
251     $id_nota = mysql_insert_id();
252
253     return $id_nota;
254 }
255 // Fi funció Corregir
```

Llistat 6.22: Codi PHP de la funció Corregir

6.3.4 Altres funcions

S'han creat moltes altres funcions utilitzades per l'aplicació, com són:

- *Multiplicador*.- Passat un número qualsevol que fa de mòdul *m* aquesta funció troba un altre número que sigui primer entre ells i compleixi les condicions per ser multiplicador.
- *Primer*.- Calcula el número primer més proper a un número donat.
- *VectorA*.- Calcula la llista supercreixent més eficient depenen del número de preguntes i respostes.
- *VectorA1*.- Calcula la clau pública del criptosistema knapsack.
- *Modul*.- Utilitzat en varies ocasions per calcular multiplicacions modulars.
- *NumVersio*.- Genera números de versions, aleatòriament, per ser assignats quan creem un examen.

I altres petites funcions que s'han utilitzat puntualment al llarg del disseny de l'aplicació.

6.4 Utilització de FPDF

S'ha utilitzat FPDF per crear arxius en PDF llestos per ser impresos. Concretament s'ha emprat per a que els professors puguin imprimir els exàmens creats, així com per imprimir totes les notes d'un examen. Els alumnes poden imprimir un resguard de l'examen amb la nota obtinguda i tant els professors com els alumnes poden veure el resultat d'un examen amb les preguntes que s'han escollit i les preguntes correctes.

Al Llistat 6.23 mostra el codi bàsic necessari per crear un arxiu PDF.

```
186 $pdf=new PDF();
187 $pdf->AliasNbPages(np);
188 $page = np;
189 $title='Escola Politècnica';
190 $opc = explode(",",$row_permu[vector_A7]);
191 $pdf->SetTitle($title);
192 $pdf->SetAuthor('Xavi Baldor');
193 $pdf->AddPage();
194 $pdf->Inici();
195 $i=1;
196 while ($row_preg_exam = mysql_fetch_array($result_preg_exam)){ // num preguntes
197     $resp=$pdf->PrintPreguntes($row_preg_exam,$i);
198     $i++;
199 }
200 //Open the print dialog
201 $pdf->AutoPrint(true);
202
203 $modo="I";
204 $nom_arxiu="Examen_".$row_exam[nom]."_".$row_exam[data_exam]."_v.".$row_permu[id_permutacio].".pdf";
205 $pdf->Output($nom_arxiu,$modo);
```

Llistat 6.23: Codi bàsic per crear PDF

Primer es crea l'objecte FPDF amb el constructor (línia 186). S'estableix el títol del document així com l'autor (línies 191 i 192). Es crea una pàgina nova (línia 193) i es carregua la funció *Inici* creada per carregar la capçalera de cada pàgina. També s'ha creat la funció *AutoPrint* per a que l'arxiu s'imprimeixi directament. A més, s'ha donat un nom a l'arxiu per si s'emmagatzema al disc dur de l'ordinador (línies 203-205)

Les versions creades d'un mateix examen es poden imprimir totes alhora o imprimir cada versió individualment. A tots els exàmens hi surt la capçalera amb el nom de l'examen, la data, el logotip de la universitat i alguna informació més tal i com s'observa a la Figura 6.9.

Figura 6.9: Capçalera dels exàmens

El Llistat 6.24 mostra el codi necessari per aconseguir-ho. Cal destacar que s'utilitzen variables globals que s'han estret de la base de dades. La resta es anar utilitzant les classes existents de FPDF. Una de les més importants es *Cell* que imprimeix una cel·la. *Ln* serveix per crear un salt de línia i *SetFont* per escollir la font.

```
86  function Inici()
87  {
88      global $data;
89      global $assig;
90      global $nom_exam;
91
92      //Arial bold 10
93      $this->SetFont('Arial','B',10);
94      //Título
95      $this->Cell(150,'','Escola Politècnica Superior');
96      $this->Cell(40,'','Universitat de Lleida',0,1);
97      //Salt de linia
98      $this->Ln(5);
99      // Logo
100     $this->Image('imatges/logo_udl.jpg',170,13,20);
101     $this->Cell(60,'','Enginyeria en Informàtica',0,1);
102     $this->Ln(5);
103     $this->Cell(0,'',$nom_exam.' - '.$assig,'','',C);
104     $this->Ln(5);
105     $this->SetFont('Arial','',8);
106     $this->Cell(0,'',$data,'','',C);
107     $this->Ln(10);
108     //Gruix de linia i fiquem linia
109     $this->SetLineWidth(0.3);
110     $this->Line(10,30,200,30);
111 }
```

Llistat 6.24: Codi capçalera dels exàmens

A més de la capçalera, els exàmens tenen un peu de pàgina situat a les seves pàgines parell. Si l'examen consta d'un número senar de pàgines, sortirà el peu de pàgina a la darrera de les pàgines encara que sigui senar. La Figura 6.10 presenta el peu de pàgina i el Llistat 6.24 el codi necessari per obtenir-ho.

La resta de codi emprat per a la creació dels exàmens són accessos a la base de dades per recuperar tota la informació necessària i bolcar les preguntes amb les seves possibles respostes. Al Llistat 6.23 s'ha pogut veure que es creava un bucle que cridava a la funció *PrintPreguntes* (línies 196-199). Aquesta funció es cridarà tantes vegades com preguntes tingui l'examen. Dins d'aquesta funció es crida a una altra, *PrintOpcions*, per escriure les possibles respostes de cada pregunta.

```php
// Funcio per al peu de pàgina que es crida automàticament
function Footer()
{
    global $id;
    global $page;
    global $np;

    //Posició a 1,5 cm del final
    $this->SetY(-15);
    //Gruix de linia i fiquem linia
    $this->SetLineWidth(0.3);
    $this->Line(10,280,200,280);
    //Arial itàlica 8
    $this->SetFont('Arial','I',8);
    //Color del text gris
    $this->SetTextColor(128);
    // Només fico versio,... a les pàgines parell o a l'última pàgina en cas de ser senar
    $a = $page;
    $a1 = (string)$a;
    $b = (string)$this->PageNo();
    $r = 0;
    if ($a1==$b) $r=1;
    if (($this->PageNo()%2 == 0) || ($r==0)) {
        $this->Cell(28,10,'Versió: '.$id);
        $this->Cell(30,10,'DNI:');
        $this->Cell(37,10,'Suma:');
        $this->Cell(45,10,'Resultat (Suma/7):');
        $this->Cell(40,10,'Firma:');
    }
    //Número de pàgina a totes les pàgines
    $this->Cell(0,10,'Pàgina '.$this->PageNo().'/'.$page,'','',C);
}
```

Llistat 6.25: Codi peu de pàgina dels exàmens

Figura 6.10: Peu de pàgina dels exàmens

L'alumne té l'opció de veure el seu examen corregit. Una vegada ja sap la nota podrà visualitzar, en un arxiu pdf, l'examen que ha fet. Dins d'aquest arxiu es marca en verd si la resposta que ha escollit es correcta. Si la resposta es incorrecta, aquesta es marca en color vermell i en color blau la resposta correcta, per a que l'alumne pugui saber quina era l'opció correcta. Finalment, si l'alumne a deixat la resposta en blanc, es pot veure, també en color blau, la resposta correcta. Per aconseguir això s'ha emprat el mateix codi que s'ha explicat anteriorment, només modificant el color de les respostes depenent de les respostes de l'alumne.

També s'ha utilitzat FPDF per crear un resguard per l'alumne amb la nota de l'examen, la versió que ha realitzat, les preguntes encertades, i més informació que es pot visualitzar a la Figura 6.11.

Figura 6.11: Resguard de la nota per l'alumne

Una de les petites diferencies amb el que s’ha vist anteriorment és la mida del resguard, bastant més petit que un DIN A4, i s’ha aconseguit modificant els valors per defecte del constructor `$pdf=new PDF('L','mm',array(100,150));` A més, s’ha adaptat un script, ja existent, per crear el codi de barres amb la versió de l’examen.

Capítol 7

Conclusions i treball futur

Després d'un llarg procés d'estudi, disseny, aprenentatge i programació, s'ha aconseguit complir els requisits inicials, obtenint un bon resultat i un producte satisfactori.

S'ha pogut implementar una aplicació totalment llesta per ser utilitzada. Professors i alumnes es podran beneficiar de tots els avantatges que representa l'aplicació. Aquesta eina ha de permetre al cos docent, entre altres moltes coses, crear exàmens amb molta facilitat, tenir un historial d'aquest, i tan els professors com els alumnes, podran obtenir el resultat de l'examen immediatament després de finalitzar-lo, alliberant al professor de la feixuga tasca de correcció i reduint a l'alumne l'espera en la qualificació.

He de fer particular esment al fet que des d'un punt de vista més dirigit a la implementació, he après a treballar amb jQuery amb els qual no estava familiaritzat, així com he assolit coneixements en el desenvolupament d'un projecte d'aquestes magnituds. Un altre aspecte molt ressenyable, que també s'ha de fer esment, són els esforços aconseguits en el aspecte més estilístic i gramatical tenint cura en la millora de la redacció de la present memòria.

A mesura que s'ha avançat en l'aplicació s'han evidenciat algunes modificacions respecte els requisits inicials. L'aplicació s'ha anat modificant per poder complir amb els nous requisits, encara que alguns no s'han pogut assolir pel fet de tenir l'aplicació bastant avançada o perquè s'allunyaven molt dels inicials.

Un dels aspectes que trobem més interessants i engrescadors de cara a futures versions és fer un càlcul estadístic de la dificultat de les preguntes en funció dels alumnes que encerten aquesta. Seria bo pel professor que pugues visualitzar si una pregunta és estadísticament fàcil, de dificultat mitjana o difícil. Així el professor, a l'hora de fer l'examen, podria barrejar les tres dificultats de preguntes per realitzar un examen equitatiu.

Una altra tasca d'ampliació de l'aplicació, podria ser la possibilitat de que a l'inserir el professor una pregunta a la base de dades, tingues l'opció d'utilitzar equacions matemàtiques. Això es podria escometre inserint dins del formulari una barra d'eines que possibilites l'escriptura matemàtica.

Realment ha estat una experiència personal molt gratificant poder combinar les meves dues passions: les matemàtiques i la informàtica. Tot plegat he d'agrair al projecte la possibilitat d'endinsar-me molt més en aquests móns tan fascinants. A més a més, m'ha permès veure una nova aplicació de les matemàtiques que no coneixia en profunditat com és la criptografia i la gran importància que ha tingut al llarg de la història, així com, la que tindrà en un futur.

Bibliografia

Llibres

[1] M. Valls. *Seguretat Computacional*. Document no publicat formalment, Escola Politècnica Superior, Universitat de Lleida.

[2] J. Gutierrez and J. Tena. *Protocolos Criptográficos y Seguridad en redes*. Servicio de Publicaciones de la Universidad de Cantabria, 2003.

[3] D. Juher. *Introducció a la criptografia*. Servei de publicacions de la Universitat de Girona, 2000.

[4] J. Pastor and M.A. Sarasa. *Criptografía digital: Fundamentos y aplicaciones.* Prensas Universitarias de Zaragoza, 1998.

[5] T. Ebrahimi, F. Leprévost and B. Warusfel. *Enjeux de la sécurité multimédia.* Lavoisier, 2006.

[6] S. Vaudenay. *A Classical Introduction to Modern Cryptography: Aplications for Communications Security.* Springer Science + Business Media Inc., 2006

[7] N. Koblitz. *A course in Number Theory and Cryptography.* Springer-Verlag, 2001.

[8] A. Menezes, P. van Oorschot and S. Vanstone. *Handbook of Applied Cryptography.* CRC Press, 1996.

Internet i adreces Web

[9] M.A. Gallardo. *Seguridad del comercio electrónico.* Disponible en Internet (http://cita.es/textos/protesis.htm), 1997. [17/07/2009]

[10] M. Dalmau, S. Martín and J. Saludes. *Eina per al disseny i correcció d'exàmens de tipus test.* Universitat politècnica de Catalunya. Disponible en Internet (http://upcommons.upc.edu/video/bitstream/2099.2/271/6/271_Article.pdf). [13/07/2009]

[11] *Sistemas de clave pública.* Disponible en Internet (http://www.textoscientificos.com/criptografia/publica). [8/7/2010]

Annex A

Manual d'usuari

A continuació es presenta el manual d'usuari de l'aplicació per tal d'ajudar als usuaris inexperts a començar ràpidament i amb la menor dificultat possible, així com facilitar als usuaris amb experiència a que aconsegueixin ser productius el més aviat possible.

A.1 Instal·lació

L'aplicació necessita una base de dades relacional configurada i un servidor web HTTP que suporti PHP. En aquest cas s'ha utilitzat MySQL 5.0.27 per a la base de dades i Apache 2.2.4 per al servidor web. També s'ha utilitzat PHP 5.2.1.

A.1.1 Requeriment mínims

Com es tracta d'una eina multiplataforma, els requeriments de maquinari dependran del sistema operatiu sota el qual s'estigui executant l'aplicació. Els requeriments de programari són els següents:

Entorn Windows:

- Windows 95 o superior.
- Navegador Web i visualitzador de documents PDF.

Entorn Linux:

- Linux (kernel 2.6 o superior).
- Navegador Web i visualitzador de documents PDF.

És recomanable, també, configurar la pantalla de la màquina on es vagi a executar l'aplicació a una resolució de 1024x768 píxels o similar.

A.1.2 Instal·lació de l'aplicació

Primer de tot s'ha de crear una base de dades anomenada *test*. Una vegada creada, s'ha d'importar l'arxiu *db.sql* que es troba dins el CD adjunt a la memòria. Es crearan totes les taules necessàries pel funcionament de l'aplicació així com un usuari administrador per defecte, amb nom d'usuari i contrasenya *admin*.

El següent pas es copiar els arxius que es troben dins la carpeta *Codi* a la carpeta corresponent del servidor web. Una vegada tenim tots els arxius i carpetes copiades dins el servidor web, obrirem l'arxiu *db.php* que es troba dins la carpeta *Connections*. Dins aquest arxiu s'ha de ficar la contrasenya d'accés a la vostra base de dades.

Ja es pot accedir a l'aplicació i començar a treballar amb l'usuari administrador creat per defecte.

A.2 Funcionament

Dins d'aquest subapartat s'especifica totes les utilitats de l'aplicació així com una explicació de com dur-ho a terme.

A.2.1 L'accés d'usuaris

Per poder accedir a qualsevol funcionalitat de l'aplicació s'ha d'estar identificat prèviament. D'aquesta forma l'usuari rep un tipus d'informació depenent del tipus d'usuari que és.

Existeixen tres tipus d'usuaris: el de professor, el d'alumne i el d'administrador. Una vegada l'usuari escull el tipus corresponent s'ha de validar correctament omplint el formulari que es mostra a la Figura A.1.

Figura A.1: Formulari d'identificació d'usuari

A.2.2 Usuari professor

En aquesta subsecció s'especificaran totes les possibilitats que té un professor dins de l'aplicació.

Figura A.2: Menú mòdul professor

Quan el professor es valida correctament podrà visualitzar un llistat de les assignatures que cursa en l'actualitat així com altres assignatures que ha cursat en cursos anteriors. La Figura A.2 ofereix el menú que apareix dins del mòdul professor.

A.2.2.1 Inserir una pregunta

Quan un professor vol inserir una pregunta, el primer que ha de fer és seleccionar l'assignatura corresponent. A continuació, apareixerà un formulari on podrà inserir l'enunciat de la pregunta, així com les possibles respostes d'aquesta. A més a més, haurà de marcar quina pregunta és la correcta, i si hi ha alguna possible resposta que sempre haurà de sortir en l'últim lloc, com per exemple, *cap de les anteriors*.

L'aplicació retornarà un error si es deixa algun camp en blanc, ja sigui la pregunta o alguna de les opcions, o si no es marca quina és la resposta correcta. Si això succeeix, es mostrarà per pantalla l'error i tornarà a sortir el formulari.

Si tot ha anat correctament, la pregunta es guardarà a la base de dades i es tornarà a visualitzar el formulari per si es vol continuar inserint preguntes.

A.2.2.2 Consultar, eliminar o modificar una pregunta

Quan un professor vol consulta una pregunta ja existent a la base de dades, haurà d'escollir l'assignatura corresponent, es pot escollir entre les assignatures que dona en el curs vigent o assignatures que ha donat amb anterioritat.

A la Figura A.3 s'observa la taula amb les preguntes que es visualitzarà després d'escollir l'assignatura desitjada.

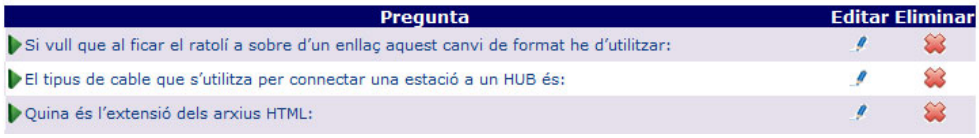


Figura A.3: Taula amb les preguntes a consultar

A qualsevol de les preguntes es pot visualitzar les possibles respostes assignades. Per poder visionar-les s'ha de fer un clic a sobre de la icona i apareixen. A la Figura A.4 apareix un exemple del resultat. Per fer que desapareguin, s'ha de fer un clic sobre la icona .

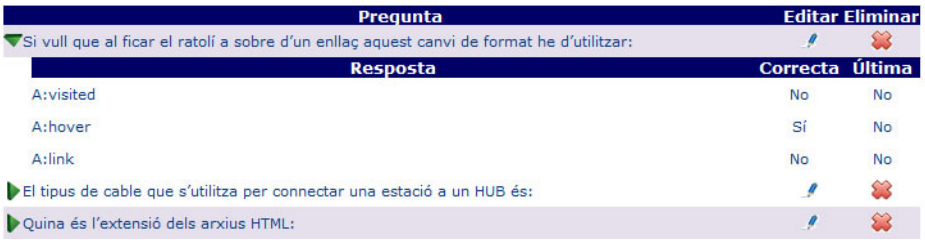


Figura A.4: Taula amb les respostes d'una pregunta

Qualsevol pregunta o resposta es pot modificar mitjançant l'opció *Editar*. Si fem clic a sobre de la icona corresponent apareix un formulari amb la pregunta i les possibles respostes en el qual es pot escriure.

També es pot eliminar una pregunta. Demanarà sempre un confirmació abans d'eliminar-la. Si s'accepta, la pregunta s'elimina, ja no es podrà utilitzar en els propers exàmens a crear, però continuarà visualitzant-se als exàmens ja creats amb anterioritat.

A.2.2.3 Crear un examen

Quan un professor vol crear un examen, el primer que ha de fer és seleccionar l'assignatura corresponent. Una vegada seleccionada, apareixerà un formulari a omplir com es pot visualitzar a la Figura A.5

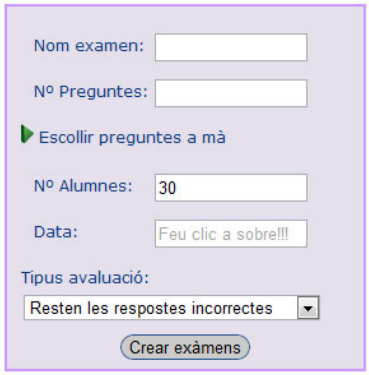


Figura A.5: Formulari per crear un examen

El professor haurà de ficar-li un nom a l'examen i indicar el número de preguntes. Mitjançant l'opció *Escollir preguntes a mà*, podrà escollir les preguntes que sortiran a l'examen, l'únic que s'ha de fer és un clic a la icona  i apareixeran les preguntes de l'assignatura com s'observa a la Figura A.6. Es poden escollir totes les preguntes a mà o només escollir algunes, la resta de les preguntes es completaran aleatòriament. Si el professor no escull cap pregunta a mà, totes les preguntes s'agafen aleatòriament.

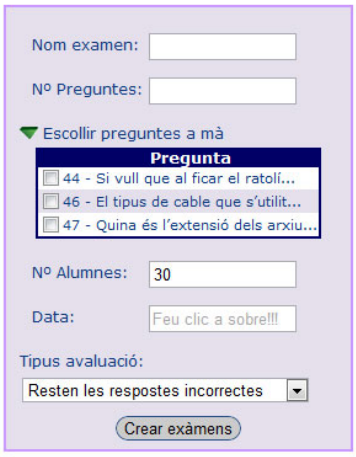


Figura A.6: Escollir preguntes a mà al crear un examen

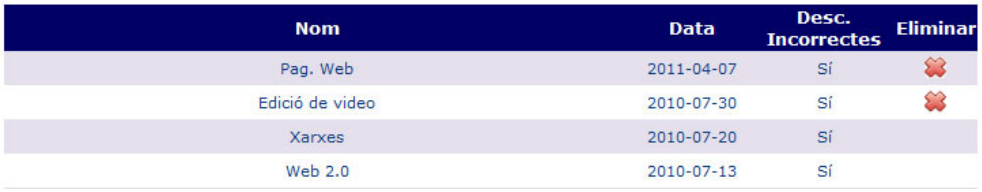
Per defecte surten el número d'alumnes matriculats a l'assignatura. Cal recordar que es crea una versió de l'examen per a cada alumne (Capítol 4). Si el professor vol modificar el número de versions ho pot fer a mà. També s'ha d'inserir la data de l'examen, que haurà de ser igual o posterior al dia de la creació d'aquest. Si es fa un clic a sobre del camp data apareix un calendari per inserir amb més comoditat el dia. Finalment, s'ha d'indicar si les preguntes incorrectes resten o no a l'hora d'avaluar l'examen.

En el cas de que hi hagi algun error a l'omplir el formulari apareixerà per pantalla i es tornarà a visualitzar el formulari per corregir-lo.

Una vegada l'examen s'ha creat amb èxit, el professor podrà imprimir totes les versions en un sol arxiu pdf o podrà fer-ho cada versió individualment. Quan s'obre un arxiu pdf automàticament s'obre el menú d'impressió.

A.2.2.4 Consultar o eliminar un examen

Quan es vol consultar un examen, primer s'ha d'escollir l'assignatura. Una vegada escollida, apareix una taula amb tots els exàmens creats de l'assignatura com presenta la Figura A.7. Aquesta taula informa al professor de la data en que s'ha efectuat o s'efectuarà l'examen, així com si les preguntes incorrectes descompten la nota. A més, es podrà eliminar l'examen, sempre i quan l'examen no s'hagi fet i corregit.



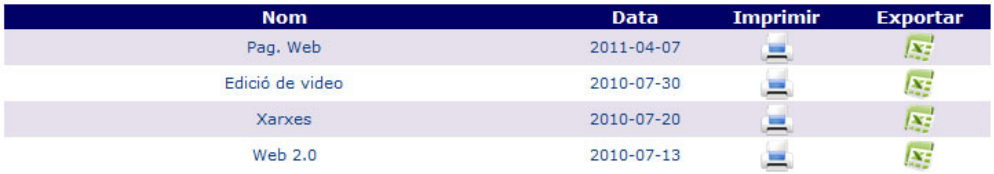
Nom	Data	Desc. Incorrectes	Eliminar
Pag. Web	2011-04-07	Sí	
Edició de video	2010-07-30	Sí	
Xarxes	2010-07-20	Sí	
Web 2.0	2010-07-13	Sí	

Figura A.7: Taula amb els exàmens a consultar

Si es selecciona qualsevol dels exàmens, apareix la possibilitat de visualitzar i imprimir l'examen, totes les versions en un mateix arxiu pdf o cada versió en un arxiu individual.

A.2.2.5 Administrar o obtenir notes

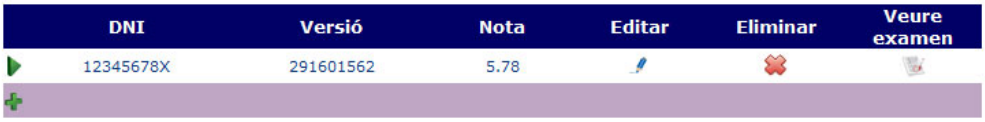
Quan un professor vol consultar les notes d'un examen, ha d'escollir l'assignatura corresponent i apareixerà la taula que es mostra a la Figura A.8. Dins d'aquesta taula té la possibilitat d'imprimir totes les notes d'un examen en un arxiu pdf o la possibilitat d'exportar-les mitjançant un arxiu csv.



Nom	Data	Imprimir	Exportar
Pag. Web	2011-04-07		
Edició de video	2010-07-30		
Xarxes	2010-07-20		
Web 2.0	2010-07-13		

Figura A.8: Taula amb els exàmens per consultar les notes

Si es selecciona un examen apareix una nova taula com ofereix la Figura A.9. En aquesta taula es mostraran totes les notes obtingudes pels alumnes, mostrant el DNI i la versió que ha realitzat.



DNI	Versió	Nota	Editar	Eliminar	Veure examen
12345678X	291601562	5.78			

Figura A.9: Taula amb les notes d'un examen

A més, es podrà eliminar la nota o editar-la per si es vol canviar alguna dada. També es pot visualitzar l'examen que ha realitzat l'alumne, mitjançant un arxiu pdf, on es trobaran les respostes de l'alumne i les respostes correctes. Si es desplega més informació sobre la nota, sortirà el número resultant de l'examen, les preguntes correctes, les incorrectes i les preguntes que l'alumne a deixat en blanc.

Tant el professor com l'alumne poden obtenir les notes dels nous exàmens realitzats, evidentment, només s'haurà de fer una vegada. El professor ho pot fer fent un clic al signe més. Haurà d'inserir el DNI de l'alumne, la versió d'examen que aquest a realitzat i el número resultant. Per evitar errors en la inserció del DNI o de la versió, a mesura que el professor insereix els números, l'aplicació mostra tots els DNI's dels alumnes que coincideixin amb el número parcialment inserit i totes les versions coincidents. Si

es produeix algun error d'inserció, es mostra per pantalla. Si tot es correcte, es mostra la nota. Un dels possibles errors és que el número resultant no coincideixi amb la versió inserida.

A.2.2.6 Canviar la contrasenya

Un professor té l'opció de canviar la seva contrasenya. Quan l'administrador de l'aplicació dona d'alta a un professor nou, l'aplicació li envia automàticament un e-mail de confirmació d'alta amb el nom d'usuari i la contrasenya. El nom d'usuari sempre serà el DNI. La contrasenya per defecte també serà el DNI. Els professors hauran de modificar la seva contrasenya la primera vegada que accedeixin a l'aplicació, per qüestions clares de seguretat. Posteriorment, la podran modificar sempre que vulguin mitjançant el formulari corresponent que es mostra a la Figura A.10. Com es habitual, es demana la contrasenya actual i la nova contrasenya, dues vegades per a que no hi hagi un error d'escriptura. Si la contrasenya actual inserida no fos correcta, sortirà un missatge d'error per pantalla.

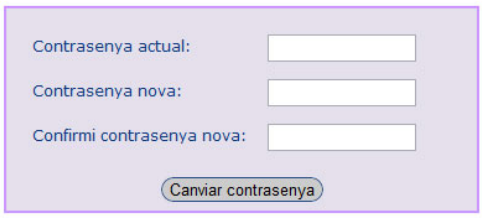


Figura A.10: Formulari per modificar la contrasenya

A.2.3 Usuari alumne

En aquesta subsecció s'especificaran totes les possibilitats que té un alumne dins de l'aplicació. Quan l'alumne es valida correctament podrà visualitzar un llistat amb l'historial de les seves assignatures, tant les que cursa en l'any acadèmic actual, com les que ha cursat en anys anterior. A més, apareixerà un botó per poder modificar la seva contrasenya.

A.2.3.1 Consultar o obtenir notes

Quan una alumne entra a l'aplicació visualitzarà una taula amb l'historial de les seves assignatures com apareix a la Figura A.11. En aquesta taula s'especifica l'any acadèmic en que ha cursat l'assignatura. Una mateixa assignatura pot sortir dues o més vegades, com és el cas de la informàtica de 4t d'ESO, com s'aprecia a la figura. Això significa que l'alumne repeteix aquesta assignatura.

Assignatura	Curs	Curs acadèmic
Informàtica	4t ESO	2010-2011
Matemàtiques	3r ESO	2010-2011
Informàtica	4t ESO	2009-2010
EVP	3r ESO	2009-2010

Figura A.11: Taula amb l'historial de les assignatures d'un alumne

Si l'alumne selecciona una assignatura obtindrà un llistat amb tots els exàmens d'aquesta. En aquest llistat apareixeran les notes dels exàmens realitzats o la possibilitat d'obtenir la nota d'un examen que acaba de realitzar. La Figura A.12 presenta un exemple. Com es pot observar, l'alumne pot consultar les notes dels exàmens ja realitzats, pot visualitzar l'examen que va fer en un arxiu pdf on estaran marcades les

preguntes que va escollir i les preguntes correctes. A més, podrà imprimir un resguard de l'examen. Si selecciona un examen que ja te nota, podrà veure el número resultant de l'examen, així com les preguntes correctes, les incorrectes i les preguntes que va deixar en blanc ,en cas d'haver-n'hi.

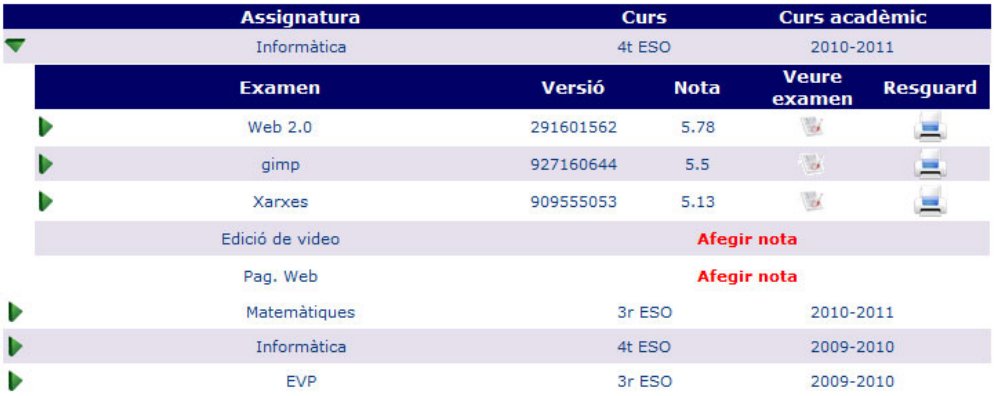


Figura A.12: Taula amb les exàmens d'una assignatura

Si el que vol fer l'alumne és obtenir la nota d'un examen que acaba de realitzar, ha de fer un clic a sobre d'*Afegir nota*. L'aplicació li demanarà la versió d'examen que ha realitzat i el número resultant obtingut. Si hi hagués algun error a l'inserir les dades apareixerà un missatge i la possibilitat de corregir-ho, si tot es correcte, automàticament, apareixerà per pantalla la nota i tota la informació addicional de l'examen.

A.2.3.2 Canviar la contrasenya

Quan l'administrador de l'aplicació dona d'alta a un alumne nou, l'aplicació li envia automàticament un e-mail de confirmació d'alta amb el nom d'usuari i la contrasenya. El nom d'usuari sempre serà el DNI. La contrasenya serà una cadena 10 de caràcters generats aleatòriament. L'alumne te la possibilitat de modificar aquesta contrasenya per una altra més fàcil de memoritzar i la podrà modificar sempre que vulguin mitjançant el formulari corresponent que es mostra a la Figura A.10. Com es habitual, es demana la contrasenya actual i la nova contrasenya, dues vegades per a que no hi hagi un error d'escriptura. Si la contrasenya actual inserida no fos correcta, sortirà un missatge d'error per pantalla.

A.2.4 Usuari administrador

En aquesta subsecció s'especificarà totes les possibilitats que té l'administrador dins de l'aplicació. Quan l'administrador es valida correctament entrarà dins el mòdul d'administrador i apareixerà el menú que s'observa a la Figura A.13.

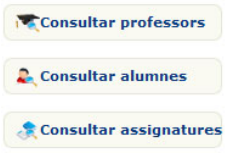


Figura A.13: Menú mòdul administrador

A.2.3.1 Donar d'alta als alumnes a l'aplicació i a una assignatura

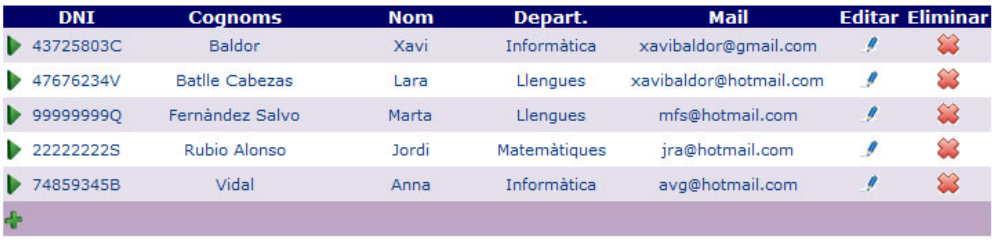
Quan l'administrador accedeix al mòdul, apareix un formulari com es reflecteix a la Figura A.14. Mitjançant aquest formulari es donaran d'alta els alumnes a les assignatures i a l'aplicació, en el cas que no ho estiguin.

Figura A.14: Formulari d'alta d'alumnes a l'aplicació i a les assignatures

Tots els alumnes matriculats en una assignatura es troben dins un arxiu csv. L'administrador escollirà l'arxiu csv de l'assignatura mitjançant el formulari, i el curs acadèmic actual. L'aplicació donarà d'alta a tots els alumnes a l'assignatura i, si és el primer any de l'alumne, també el donarà d'alta a l'aplicació, enviant-li un e-mail de confirmació amb el nom d'usuari, que serà el DNI, i la contrasenya, generada aleatòriament. Si l'arxiu seleccionat no compleix amb el format adient, es mostrarà un error per pantalla.

A.2.3.2 Consultar, modificar, eliminar o donar d'alta un professor

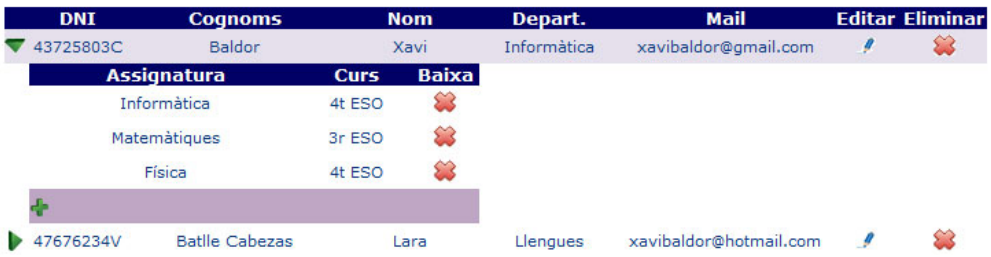
L'administrador podrà gestionar tot el relacionat amb els professors. Quant s'accedeix a l'opció de *Consultar professors*, apareix una taula amb tots els professors existents com presenta la Figura A.15. Dins d'aquesta taula es podrà consultar les dades del professor i el departament del qual forma part. A més, es poden modificar les seves dades i eliminar-lo.



DNI	Cognoms	Nom	Depart.	Mail	Editar	Eliminar
43725803C	Baldor	Xavi	Informàtica	xavibaldor@gmail.com		
47676234V	Batlle Cabezas	Lara	Llengues	xavibaldor@hotmail.com		
99999999Q	Fernàndez Salvo	Marta	Llengues	mfs@hotmail.com		
22222222S	Rubio Alonso	Jordi	Matemàtiques	jra@hotmail.com		
74859345B	Vidal	Anna	Informàtica	avg@hotmail.com		

Figura A.15: Taula amb els professors existents

Si es selecciona un professor es podrà veure les assignatures que cursa en l'actualitat, eliminar-ne'n alguna o associar-se'n de noves, com mostra la Figura A.16.



DNI	Cognoms	Nom	Depart.	Mail	Editar	Eliminar
43725803C	Baldor	Xavi	Informàtica	xavibaldor@gmail.com		

Assignatura	Curs	Baixa
Informàtica	4t ESO	
Matemàtiques	3r ESO	
Física	4t ESO	

47676234V	Batlle Cabezas	Lara	Llengues	xavibaldor@hotmail.com		

Figura A.16: Taula amb els professors i les assignatures associades

En aquesta subsecció de l'aplicació l'administrador pot donar d'alta a un nou professor mitjançant el signe més que es pot observar a la Figura A.15. Quan s'insereixen les dades del nou professor, l'aplicació el dona d'alta agafant el DNI com a nom d'usuari i contrasenya, i li envia un e-mail amb la confirmació d'alta. Més endavant, el professor haurà de modificar la contrasenya donada per defecte.

A.2.3.2 Consultar, modificar, eliminar o donar d'alta un alumne

El procediment és molt semblant al de l'apartat anterior. L'administrador pot consultar tots els alumnes existents a l'aplicació mitjançant la taula que apareix a la Figura A.17.

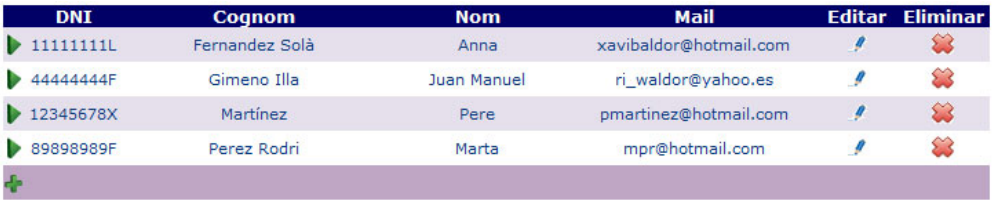


Figura A.17: Taula amb els alumnes existents

Com succeeix a l'administrar els professors, l'administrador pot consultar les dades dels alumnes, modificar-les i eliminar-los quan sigui convenient. Si es selecciona un alumne, apareix l'historial d'assignatures de l'alumne. Es pot esborrar una assignatura o afegir-ne una altra. Com s'ha vist en l'apartat A.2.3.1, normalment els alumnes es donen d'alta a les assignatures al inici del curs acadèmic mitjançant el corresponent arxiu csv. Aquesta opció de donar d'alta un alumne a una assignatura serveix per quan arribin alumnes amb el curs començat. Pel mateix motiu, també es pot donar d'alta a un alumne fent clic al signe més que es visualitza a la Figura A.17.

A.2.3.2 Consultar, modificar, eliminar o afegir una assignatura

L'administrador pot visualitzar les assignatures existents mitjançant una taula com presenta la Figura A.18. Es pot modificar alguna assignatura existent o eliminar-la. Mitjançant el signe més es pot donar d'alta una nova assignatura.

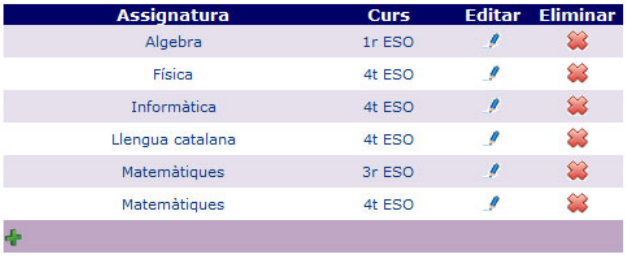


Figura A.18: Taula amb les assignatures existents