

12/03/2014

PracticaClic

Autor: Josep Antoni Rey Vila

Consultor: Jordi Ceballos Villach

PracticaClic

Autor: Josep Antoni Rey Vila

Agraïments

ÍNDIX

1. Introducció al projecte PracticaClic	6
1.1. Operativa del projecte	6
2. Objectius	6
3. Funcionalitats principals	7
4. Calendari del projecte	7
4.1. Lliuraments establerts	7
4.2. Calendari	8
4.3. Detall de l'entrega de la PAC1	9
4.4. Detall de l'entrega de la PAC2	10
4.5. Detall de l'entrega de la PAC3	12
4.6. Detall de l'entrega del Lliurament Final	13
5. Recursos i infraestructura	14
5.1. Recursos hardware	14
5.2. Recursos software ordinador desenvolupament	14
5.3. Recursos software ubicats al servidor	15
6. Tecnologies a aplicar	15
6.1. Tecnologies del servidor web	16
6.2. Tecnologies mòbils	16
7. Riscos del projecte	17
8 Anàlisi funcional	18
8.1. Requeriments funcionals	18
8.2. Requeriments no funcionals	19
8.3. Funcionalitats del sistema	19
8.4. Usuaris del sistema	20
8.5. Casos d'ús	20
9. Disseny tècnic	27
9.1. Arquitectura plataforma web (part servidora)	27
9.2. Arquitectura plataforma mòbil (part client)	28
9.3. Arquitectura física	28
9.4. Arquitectura de base de dades	28
9.5. Diagrama de classes	36
10. Prototip	39

10.1. Característiques de les interfícies	39
10.2. Pantalla d'inici.	40
10.3. Pantalla assignar feina alumnes.	41
10.4. Pantalla Assignar-se alumnes.	41
10.5. Pantalla exercicis (interfície web).	42
10.6. Pantalla exercicis (aplicació mòbil)	45
11. Implementació	46
11.1. Premisses de la implementació	46
11.2. Interfície	46
11.3. Implementació de base de dades servidor	47
11.4. Implementació de la plataforma web	49
11.5. Implementació de la plataforma mòbil	62
11.6. Eines de validació i debug.	64
12. Funcionament de l'aplicació	66
12.1. Funcionament aplicació web.	67
12.2. Funcionament aplicació mòbil.	79
13. Conclusions	83
13.1. Assoliment d'objectius	83
13.2. Variacions del producte final respecte el disseny inicial previst	83
13.3. Valoració personal	83
13.4. Futures millores	83
14. Fonts d'informació	83
14.1. Bibliografia	83
14.2. Formació Online	83
14.3. Consultes Android	83
14.4. Consultes PHP	83
14.5. Consultes generals	83

1. Introducció al projecte PracticaClic

El “mobile learning” o l’aprenentatge basat en l’ús de dispositius mòbils, és una de les tecnologies emergents que tindran un impacte important en educació de cares als propers anys, ja que la facilitat d’accés a la informació en qualsevol moment i en qualsevol lloc permet una experiència d’aprenentatge flexible i personalitzada en la què el context és molt important.

Aprofitant un dels grans avantatges d’aquest tipus d’aprenentatge, el projecte va dirigit en fomentar l’aprenentatge fora de les aules, que tingui lloc en qualsevol part i en qualsevol moment, on l’estudiant pugui obtenir més coneixement d’una matèria obligatòria determinada.

PracticaClic és una plataforma educativa dirigida a nens i nenes de Primària, i en concret, el projecte està centrat en la matèria de les Matemàtiques.

1.1. Operativa del projecte

L’operativa del projecte és molt senzilla, en la qual, hi haurà dos tipus de usuaris :

- Mestres/Pares/Tutors.
- Alumnat.

Mestres/Pares/Tutors: Aquests tipus d’usuaris podran enregistrar-se al sistema PracticaClic. El sistema els hi permetrà donar d’alta els seus alumnes/fills. Així doncs, el mestre/pare/tutor podrà escollir i assignar els exercicis que posteriorment els alumnes hagin de realitzar.

Un cop els alumnes finalitzin els exercicis, el mestre/pare/tutor podrà veure els resultats d’aquests exercicis.

Alumne: L’alumne podrà veure els exercicis proposats pel mestre/pare/tutor i lògicament, executar-los.

2. Objectius

La implantació d’aquest projecte pretén fomentar que els estudiants de primària aprenguin amb les noves tecnologies de manera divertida i alhora incentivar-los a l’aprenentatge.

Els alumnes tenen accés al sistema per mitjà d'Internet a través d'un dispositiu mòbil o ordinador.

El projecte també consta d'una aplicació realitzada pels dispositius mòbils, tot i que aquesta aplicació és portable a diferents plataformes (IOS, Windows Phone, Android...) però en aquest cas, només es provarà la plataforma Android.

Aquest sistema educatiu també va dirigit als mestres de primària, per aconseguir optimitzar la formació dels seus alumnes i veure en quins exercicis hi ha hagut més errors i així poder-los corregir a classe, ja sigui de la forma tradicional o amb una pissarra digital.

Per tant, la idea és que qualsevol formador/a, professor/a, tutor/a, pugui escollir els exercicis que més li convinguin als alumnes, i posteriorment pugui avaluar-ne els resultats.

El projecte està dissenyat per ser utilitzat en diferents assignatures, tot i que el projecte només consta de diferents exercicis de matemàtiques.

3. Funcionalitats principals

- **Donar d'alta usuaris tipus alumne:** L'usuari "mestre" pot donar d'alta usuaris del tipus "alumnes" els quals li queden associats.
- **Identificar usuaris:** Permet l'accés a usuaris registrats del sistema a les funcionalitats de l'aplicació.
- **Assignar exercicis:** L'usuari "mestre", pot assignar exercicis als alumnes que té assignats.
- **Realització d'exercicis:** L'usuari "alumne", ha de poder realitzar els exercicis assignats pel "mestre".
- **Estadístiques:** Un usuari "mestre" pot veure el resultats obtinguts pels alumnes.

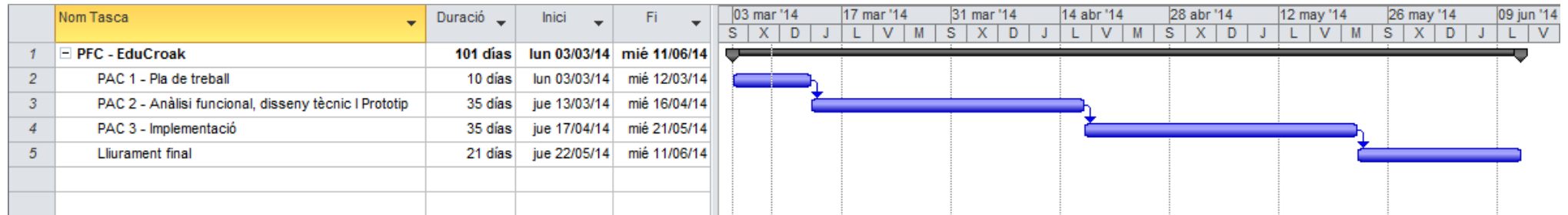
4. Calendari del projecte

4.1. Lliuraments establerts

Data de lliurament	Fita avaluació continuada	Lliurable
12 de març	PAC 1	Pla de treball
16 d'abril	PAC 2	Anàlisi funcional, disseny tècnic i Prototip.
21 de maig	PAC 3	Implementació
11 de juny	Lliurament final	Memòria i vídeo amb la presentació.

4.2. Calendari

L'elaboració del projecte s'ha de realitzar en 101 dies. He considerat els caps de setmana com a dies laborables, ja que són els dies dels que disposo de més hores de dedicació per a la realització del projecte.



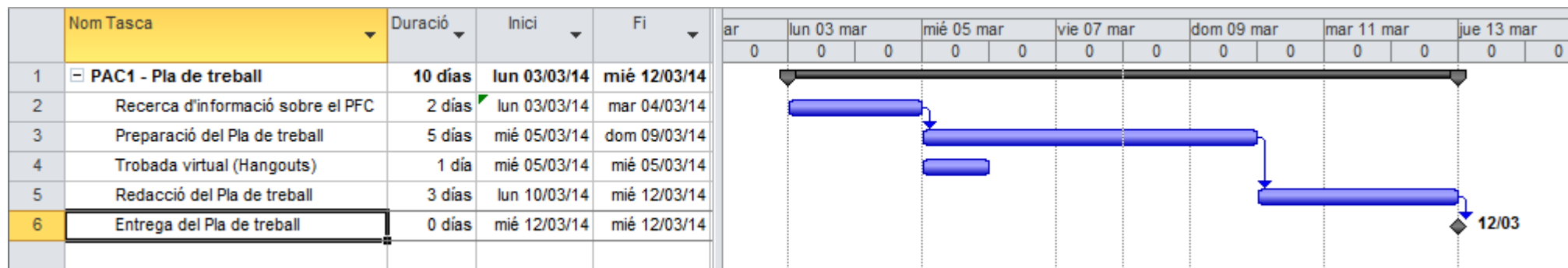
Hi ha 4 fases ben definides:

- PAC1: Construcció del pla de treball que es seguirà durant el desenvolupament del projecte.
- PAC2: Entrega de l'anàlisi funcional, disseny tècnic i prototip.
- PAC3: Realització de la implementació de projecte.
- Lliurament final: Entrega del projecte final de carrera.

4.3. Detall de l'entrega de la PAC1

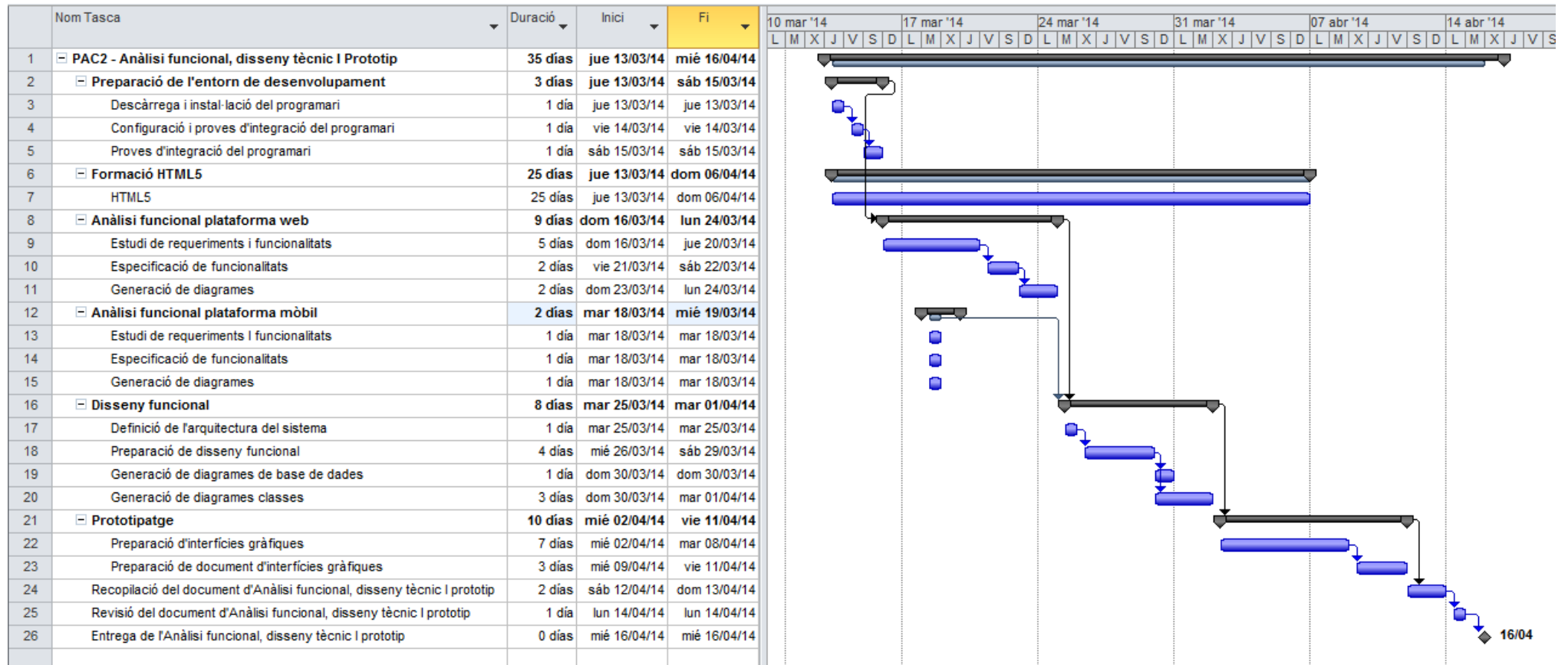
El pla de treball s'inicia el dia 3 de març ja que és el dia de l'aprovació de la proposta del projecte.

El pla de treball s'ha realitzat amb 10 dies, distribuïts tal com es veu en la següent imatge:



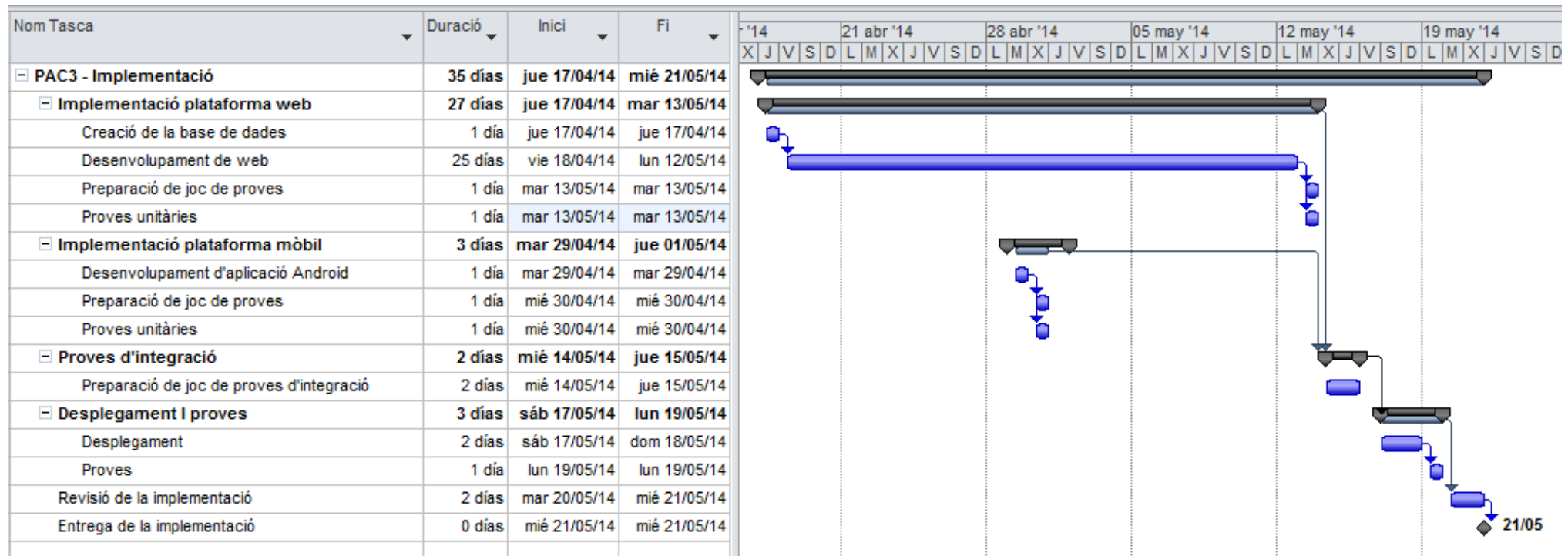
Tasca	Definició
Recerca d'informació sobre el PFC	Cerca d'informació per definir de forma general el projecte i definir els objectius.
Preparació del Pla de treball	Definició dels calendaris, Tecnologies i possibles riscos.
Trobada virtual (Hangouts)	Trobada virtual amb el tutor i companys per exposar els diferents projectes i solucionar possibles dubtes.
Redacció del Pla de treball	Es redacta de forma formal, el pla de treball.
Entrega del pla de treball	Es fa l'entrega del pla de treball.

4.4. Detall de l'entrega de la PAC2



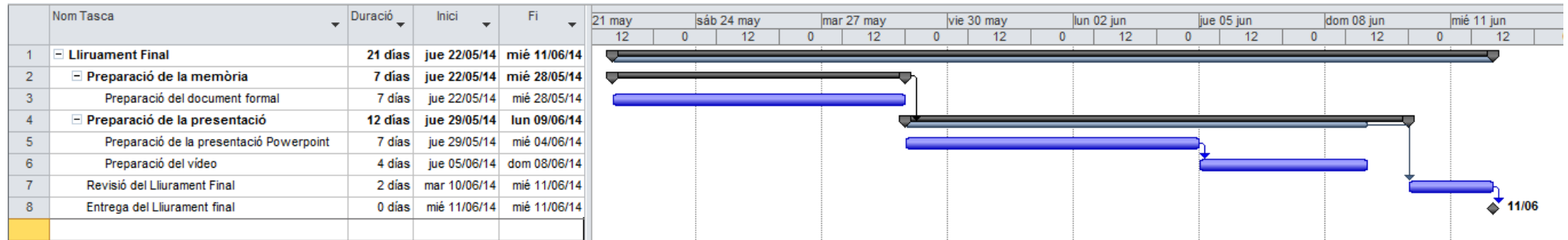
Tasca	Definició
Preparació de l'entorn de desenvolupament	Es destinen 3 dies a realitzar la descàrrega, instal·lació i realització de proves del programari destinat al desenvolupament del projecte.
Formació HTML5	Es destinen 25 dies a la formació de HTML5 i JavaScript.
Anàlisi funcional plataforma web	Es destinen 9 dies, per l'estudi de requeriments, especificació de funcionalitats i generació de diagrames de la plataforma web.
Anàlisi funcional plataforma mòbil	Es destinen 3 dies, per l'estudi de requeriments, especificació de funcionalitats i generació de diagrames de la plataforma mòbil.
Disseny funcional	8 dies destinats a la realització del disseny funcional.
Prototipatge.	Es destinen 10 dies per a realitzar el prototip amb la documentació de les interfícies gràfiques.

4.5. Detall de l'entrega de la PAC3



Tasca	Definició
Implantació plataforma web	Realització de tota la implementació del costat del servidor en 27 dies.
Implementació plataforma mòbil	3 dies per realitzar la implementació de l'aplicació per Android.
Proves d'integració	2 dies per realitzar les proves d'integració.
Desplegament i proves	Realització de diverses proves.

4.6. Detall de l'entrega del Lliurament Final



Tasca	Definició
Preparació de la memòria	Es finalitza la realització de la memòria.
Preparació de la presentació PowerPoint	Es prepara una presentació amb PowerPoint del projecte realitzat.
Preparació del vídeo	Es realitza la preparació del vídeo a partir de la presentació de PowerPoint.
Revisió del lliurament final	Es revisa el lliurament final per si hi ha algun detall a millorar.
Entrega del lliurament final	Es fa l'entrega del projecte.

5. Recursos i infraestructura

5.1. Recursos hardware

Component	Característiques tècniques	Funció
ordinador desenvolupament	Processador: Pentium (R) Dual-Core CPU E5700 3GHz Memòria RAM: 4Gb Sistema Operatiu: Windows 7 Home Premium Sistema operatiu de 32bits	Es realitzaran totes les tasques de disseny i implementació del projecte.
Servidor web	Processador: Pentium (R) Dual-Core CPU E5700 3GHz Memòria RAM: 4Gb Sistema Operatiu: Windows 7 Home Premium Sistema operatiu de 32bits	Hosting del projecte, també s'utilitzarà per a fer els tests del programari web.
Mòbil Nexus 5	CPU: Qualcomm Snapdragon 800 a 2,26GHz GPU: Adreno 330 a 450 MHz Sistema Operatiu: Android 4.4 KitKat	S'utilitzarà per a realitzar els test amb un dispositiu mòbil.

5.2. Recursos software ordinador desenvolupament

En la següent taula podem veure tot el programari que conté l'ordinador de desenvolupament.

Programari desenvolupament	Funció
Eclipse IDE	Entorn de desenvolupament per implementar el projecte amb HTML, PHP i JavaScript.
Magic Draw UML	Eina per al disseny de diagrames de casos d'ús, diagrames de classes, diagrames de seqüències, etc
Microsoft Project	Eina per definir els calendaris del projecte.
Microsoft Word	Eina per redactar la memòria del projecte.
Microsoft Visio	Eina per dissenyar esquemes.
Xampp	paquet de programari lliure XAMPP.
SDK Android	SDK d'android per compilar l'aplicació mòbil amb cordovà.
PhoneGap 3.0	Eina que ens permet <i>compilar</i> en diferents plataformes mòbils. Els projectes han d'estar desenvolupats amb HTML5 i JavaScript. Amb PhoneGap podem fer la compilació en Local o bé al núvol.
Cordova	Es el mateix que PhoneGap, però nomes ens deixa

	compilar en local.
MySQL Workbench	Eina oficial de MySQL per realitzar el disseny visual d'esquemes de base de dades.

5.3. Recursos software ubicats al servidor

En la següent taula podem veure tot el programari que conté el servidor.

Programari servidor	Funció
Apache Web Server	Servidor web que permet l'accés al sistema.
MySQL	Sistema gestor de base de dades del sistema.
phpMyAdmin	Mòdul que facilita la gestió de les bases de dades MySQL
php	Intèrpret que habilita l'execució de codi escrit en PHP al servidor web

6. Tecnologies a aplicar

El projecte té dues parts molt ben definides: la part del **servidor** i la part del **client**.

El servidor està desenvolupat amb les següent tecnologies: HTML5, JavaScript, CSS, MySQL i PHP. El motiu pel qual m'he decantant per HTML5, ha sigut perquè amb aquesta tecnologia fa que sigui accessible des dels diferents dispositius mòbils que hi ha al mercat, fins arribar a un ordinador.

Aquest projecte també consta d'una aplicació mòbil on l'alumne té accés per a realitzar els exercicis assignats.

Quan volem portar una aplicació a un dispositiu mòbil, podem emprar l'eina PhoneGap o bé Cordova, si s'ha desenvolupat amb HTML5 i amb JavaScript.

PhoneGap és una eina que ens permet "*compilar*" aplicacions desenvolupades amb HTML5 i JavaScript per les diferents plataformes mòbils. A més a més, aquesta eina ens permet accedir a funcions del dispositiu que en un principi només són accessibles per una aplicació nativa; com per exemple, podria ser el GPS.

Aquesta aplicació mòbil permet l'accés al sistema d'una forma més fàcil i així evitar la utilització del navegador del dispositiu mòbil. Aquesta aplicació en un futur es podria millorar incorporant serveis que ens dona el propi sistema operatiu; com per exemple, rebre una notificació dels nous exercicis a realitzar, etc. Si tenim en compte que utilitzem l'eina de PhoneGap serà prou àgil dur l'aplicació en les diferents plataformes mòbils.

L'ús de PhoneGap, també ens permet que tota la implementació necessària estigui guardada en el dispositiu mòbil, per tant, no s'han de descarregar tots els arxius com ho fa el navegador.

Cal dir que, que tots els casos de proves, els he realitzat únicament en dispositius mòbils Android.

6.1. Tecnologies del servidor web

El servidor web del que consta el sistema és l'*Apache Web Server* amb el mòdul de *PHP*. Aquest mòdul és el que s'utilitza per donar l'accés a la base de dades *MySQL* i obtenir tant els exercicis que s'assignaran als alumnes com els resultats de l'avaluació dels exercicis realitzats pels alumnes.

6.2. Tecnologies mòbils

Com s'ha comentat anteriorment, el projecte s'ha desenvolupat amb tecnologia *HTML5*, això fa que l'accés a l'aplicació sigui possible amb els diferents dispositius mòbils actuals, ja sigui un sistema Android o un iOS; és a dir, qualsevol dispositiu que sigui compatible amb *HTML5*.

7. Riscos del projecte

Risc	Descripció	Probabilitat d'aparició	Impacte	Accions mitigadores
Manca de coneixements en desenvolupament HTML5.	Al no disposar de coneixements en la tecnologia HTML5, pot dur un endarreriment del projecte.	Alta	Alta	Cerca de documentació per Internet. Comprar algun llibre o manual de HTML5.
Planificació incorrecta del projecte o imprevist.	El projecte està molt ajustat al temps de que es disposa, un endarreriment en alguna de les tasques, pot dur el projecte al fracàs	Mitja	Alta	Seguir al màxim la planificació, i en cas ser possible avançar les tasques. Si alguna tasca s'endarrereix al no disposar d'un horari laboral fixa, s'hauran d'augmentar les hores dedicades.
Malaltia o baixa.	És imprevisible saber si durant el projecte poden caure malalts, patir una lesió per practicar esport o tenir accident.	Baixa	Alta	Evitar el màxim a les possibles exposicions a riscos innecessaris.
Avaria d'equipament tècnic.	En el cas d'una avaria en el ordinador de desenvolupament, pot endarrerir el projecte i evitar complir amb els terminis establerts.	Baixa	Mitja	Es disposa d'un portàtil amb la instal·lació del software necessari pel desenvolupament.
Pèrdua de dades.	En el cas d'una pèrdua parcial o total de les dades, per un error de software o hardware, pot portar al fracàs el projecte.	Baixa	Alt	Instal·lació d'un sistema de còpies de seguretat mitjançant Backuppc en un servidor.

8 Anàlisi funcional

En aquest apartat es tracten els elements obtinguts durant la fase d'anàlisi del sistema. Es detallen els requisits funcionals i els no funcionals, juntament amb l'especificació dels casos d'ús, obtenim una visió del sistema força acurada.

8.1. Requeriments funcionals

8.1.1 Descripció bàsica del funcionament.

Aquest projecte vol aprofitar les noves tecnologies per dur-les a l'educació primària. Primer de tot, s'ha d'haver registrat un usuari com a "mestre/tutor". Tot seguit, ha de definir quins seran els seus alumnes. (Un mestre podrà assignar diferents exercicis als seus alumnes, i posteriorment l'alumne haurà de realitzar els exercicis corresponents).

8.1.2 Requeriments funcionals entorn web

En l'aplicació PracticaClic es poden dividir els requeriments en dos blocs:

- La part de l'alumne.
- La part del mestre.

Alumne

- Ha de poder realitzar els exercicis assignats pel seu mestre.
- També ha de poder modificar la seva contrasenya.

Mestre

- Ha de poder assignar exercicis als seus alumnes.
- Veure estadístiques dels resultats dels exercicis realitzats.
- També ha de poder modificar la seva contrasenya.

8.1.3 Requeriments funcionals entorn mòbil

Amb un dispositiu mòbil amb accés a internet, podrà realitzar les mateixes tasques que des de l'entorn Web. Per a dur a terme el projecte, he utilitzat l'eina PhoneGap , per realitzar una aplicació específica per a dispositius mòbils i així facilitar als alumnes la realització dels exercicis. Aquesta aplicació té els següents requeriments:

- Ha de poder realitzar els exercicis assignats per algun dels seus mestres.
- També ha de tenir accés a l'ajuda.

8.2. Requeriments no funcionals

Hi ha d'haver una coordinació/comunicació per part del mestre i l'alumne. És a dir, hi ha d'haver una comunicació verbal entre mestre i alumne, per tal que l'alumne conegui que se l'hi han assignat diferents exercicis.

S'ha tingut en compte la simplicitat/facilitat d'ús tant en l'aplicació mòbil com en la plataforma web.

8.3. Funcionalitats del sistema

8.3.1 Plataforma Web

Des de l'entorn web es poden realitzar les següents operacions :

Usuari

Petició registrar-se com a mestre: Un usuari demana enregistrar-se al sistema com a tutor.

Usuari registrat

Inici sessió: Comprova l'usuari i contrasenya. Si l'autenticació falla, es mostrarà una pàgina d'error. Si l'autenticació és correcta, entrarà al sistema segons el tipus d'usuari.

Contrasenya oblidada: Si un usuari *tutor* oblida la seva contrasenya, pot demanar al sistema la recuperació de la contrasenya enviant un e-mail a l'usuari (Restaurar contrasenya). En cas de ser l'alumne qui hagi oblidat la contrasenya, aquest li haurà de demanar al seu tutor que li modifiqui i la memoritzi.

Usuari Identificat

Tancar sessió: Tanca la sessió oberta.

Canvi de contrasenya: Qualsevol usuari pot modificar la contrasenya un cop ha entrat dins del sistema.

Mestre Identificat

Assignar-se un alumne: Un mestre pot donar d'alta un alumne i assignar-se'l.

Assignar feina als alumnes: Permet assignar diferents exercicis a l'alumne/s que tingui assignats.

Veure resultats en global: Veure els resultats dels tots els alumnes assignats.

Veure resultats alumne: Pot veure els resultats d'un alumne concret.

Alumne Identificat

Realització dels exercicis assignats: L'alumne pot realitzar els exercicis que té assignats.

8.3.2 Plataforma mòbil

Des de l'entorn mòbil (si s'utilitza el navegador que porta incorporat) es podran realitzar les mateixes operacions definides en el punt 8.3.1 *Plataforma web*.

En el cas d'utilitzar l'aplicació mòbil, es podran realitzar les següents operacions:

Inici de sessió: Comprova l'usuari i contrasenya. Si l'autenticació falla, es mostrarà una pàgina d'error. Si l'autenticació és correcta, entra en el sistema.

Realització dels exercicis assignats: L'alumne pot realitzar els exercicis que té assignats.

Ajuda: Veure l'ajuda de l'aplicació.

Tancar sessió: Tanca la sessió oberta.

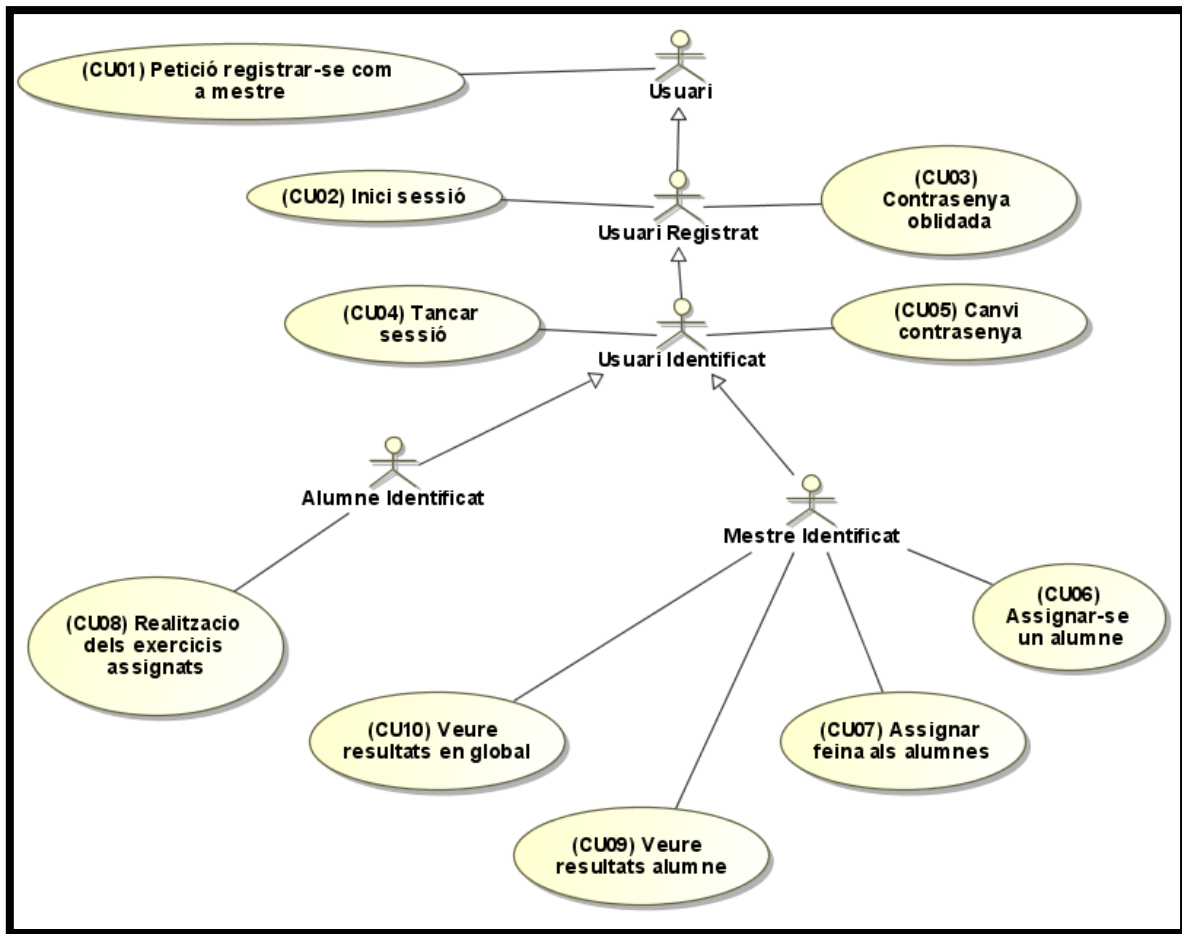
8.4. Usuaris del sistema

En els sistema s'identifiquen 2 usuaris :

- Mestre: usuari que pot assignar diferents exercicis als seus alumnes.
- Alumne: L'usuari que realitzarà els exercicis assignats pel seu mestre.

8.5. Casos d'ús

En següent imatge es poden veure els actors que intervenen en el sistema i quines funcionalitats pot tenir cada actor.



En la imatge anterior, es pot veure que l'actor *usuari* pot executar el cas d'ús "Petició registrar-se com a mestre", aquest cas es donarà quan un mestre/tutor vulgui donar-se d'alta en el sistema.

Un *usuari registrat*, en cas d'oblidar la seva contrasenya pot demanar al sistema, la seva recuperació mitjançant l'enviament d'un correu electrònic, sempre hi quan sigui un usuari del tipus mestre. En cas de ser un alumne, aquest li haurà de demanar al seu mestre la nova contrasenya.

Un *usuari registrat*, pot entrar al sistema i en funció de quin tipus d'usuari, tindrà accés a uns determinats programes, definint com a programa les diferents "pantalles" a les que té accés.

A continuació, es descriuen els diferents **casos d'ús** per veure'ls amb més detall.

8.5.1 Cas d'ús: Petició registrar-se com a mestre.

Identificador	CU01
Nom	Petició registrar-se com a mestre
Resum	Representa la petició de registrar-se al sistema com a mestre.
Actors	Usuari.
Precondicions	L'usuari no està registrat en el sistema.
Postcondicions	L'usuari realitza la petició de registre. L'usuari cancel·la la petició.
Flux normal	<ol style="list-style-type: none"> 1- L'usuari introdueix les dades del formulari de la pantalla principal i valida el formulari. (e-mail, nom, cognoms) 2- El sistema comprova si les dades introduïdes són correctes. 3- Si la informació és correcta, la petició queda registrada i s'envia un e-mail amb la contrasenya d'accés.
Fluxos alternatius	<p>2b- Si les dades subministrades no són correctes es mostra un missatge d'error i el flux torna al pas 1.</p> <p>3b- En cas que el sistema no tingui connexió a internet o es produeixi un error, l'aplicació mostra un missatge d'error i finalitza el cas d'ús.</p>
Inclusions	Cap
Extensions	Cap.

8.5.2 Cas d'ús: Inici sessió.

Identificador	CU02
Nom	Inici sessió
Resum	Representa com s'identifica un usuari al sistema.
Actors	Usuari registrat
Precondicions	L'usuari no ha iniciat sessió prèviament.
Postcondicions	L'usuari inicia la sessió - L'usuari tanca l'aplicació.
Flux normal	<ol style="list-style-type: none"> 1- L'usuari inicia l'aplicació. 2- L'usuari introdueix el nom d'usuari i la contrasenya del formulari de la pantalla principal i valida el formulari. 3- El sistema comprova si les dades introduïdes són correctes. 4- Si la informació és correcta, la sessió queda iniciada i es mostra el menú principal de l'aplicació, mostrant les opcions disponibles.
Fluxos alternatius	<p>4b- Si la informació no és correcta es mostra un missatge d'error i el flux torna al pas 2.</p> <p>4c- En cas que el sistema no tingui connexió a internet o es produeixi un error, l'aplicació mostra un missatge d'error i finalitza el cas d'ús.</p>
Inclusions	Cap
Extensions	Cap

8.5.3 Cas d'ús: Contrasenya oblidada.

Identificador	CU03
Nom	<i>Contrasenya oblidada</i>
Resum	Un usuari registrat oblida la seva contrasenya i la vol recuperar.
Actors	Usuari registrat i del tipus mestre.
Precondicions	L'usuari no ha iniciat sessió prèviament i no recorda la seva contrasenya.
Postcondicions	L'usuari recupera la seva contrasenya.
Flux normal	<ol style="list-style-type: none"> 1- L'usuari prem el botó contrasenya oblidada. 2- El sistema sol·licita el seu nom usuari (e-mail). 3- L'usuari introdueix les dades sol·licitades i valida el formulari. 4- El sistema cerca l' e-mail, si el troba envia un e-mail amb la contrasenya a l'usuari.
Fluxos alternatius	4b- Si el sistema no troba el usuari registrat envia un error.
Inclusions	Cap
Extensions	Cap

8.5.4 Cas d'ús: Tancar sessió.

Identificador	CU04
Nom	<i>Tancar sessió</i>
Resum	Un usuari finalitza la seva sessió.
Actors	Usuari identificat.
Precondicions	L'usuari ha iniciat una sessió amb l'aplicació.
Postcondicions	No hi ha sessió d'usuari activa.
Flux normal	<ol style="list-style-type: none"> 1- L'usuari prem el botó de tancar sessió. 2- El sistema dona per finalitzada la sessió en curs.
Fluxos alternatius	Cap
Inclusions	Cap
Extensions	Cap

8.5.5 Cas d'ús: Canvi de contrasenya.

Identificador	CU05
Nom	Canvi de contrasenya
Resum	Un usuari identificat vol modificar la seva contrasenya.
Actors	Usuari identificat.
Precondicions	L'usuari ha iniciat la sessió.
Postcondicions	L'usuari ha modificat la seva contrasenya.
Flux normal	<ol style="list-style-type: none"> 1- Selecciona en el menú la petició de canvi de contrasenya. 2- El sistema sol·licita les dades de la nova contrasenya. 3- L'usuari les introdueix i valida el formulari. 4- Si les dades són correctes, el sistema modifica la contrasenya.
Fluxos alternatius	4b- Si les dades no són correctes, dona un avís d'error.
Inclusions	Cap.
Extensions	Cap.

8.5.6 Cas d'ús: Assignar-se un alumne.

Identificador	CU06.1
Nom	Gestió Assignar-se un alumne.
Resum	Donar d'alta un alumne i al fer-ho li quedarà assignat al tutor.
Actors	mestre.
Precondicions	L'usuari vol assignar-se un nou alumne.
Postcondicions	El alumne queda donat d'alta i assignat al tutor.
Flux normal	<ol style="list-style-type: none"> 1- L'usuari selecciona del menú Gestió Assignar-se alumne. 2- El sistema sol·licita les dades de l'alumne. 3- L'usuari introdueix les dades de l'alumne. 4- El sistema comprova si existeix, en cas d'existir, carrega les dades. 5- L'usuari valida les dades i prem el boto guardar. 6- Si les dades són correctes, a l'alumne li queda assignat. Si l'alumne existia s'actualitzen les dades.
Fluxos alternatius	Cap
Inclusions	Cap
Extensions	Cap

8.5.7 Cas d'ús: Esborrar alumne assignat (Esborrar alumne assignat).

Identificador	CU06.2
Nom	<i>Esborrar alumne assignat.</i>
Resum	Esborrar un alumne que un mestre té assignat.
Actors	mestre.
Precondicions	L'usuari vol esborrar un alumne que té assignat.
Postcondicions	El alumne queda des associat del mestre i esborrat del sistema.
Flux normal	<ol style="list-style-type: none"> 1- L'usuari selecciona del menú Assignar-se alumne. 2- El sistema sol·licita les dades de l'alumne. 3- L'usuari introdueix les dades de l'alumne. 4- El sistema comprova si existeix, en cas d'existir, carrega les dades. 5- L'usuari valida les dades i prem el boto esborrar. 6- Si les dades són correctes, l'alumne queda des associat del tutor i esborrat del sistema.
Fluxos alternatius	Cap
Inclusions	Cap
Extensions	Cap

8.5.8 Cas d'ús: Assignar feina als alumnes

Identificador	CU07
Nom	<i>Assignar feina a alumne</i>
Resum	Permet assignar diferents exercicis als alumnes associats
Actors	Mestre
Precondicions	L'usuari mestre ha iniciat la sessió.
Postcondicions	El tutor ha assignat exercicis.
Flux normal	<ol style="list-style-type: none"> 1- L'usuari selecciona del menú Assignar feina als alumnes. 2- L'usuari selecciona els exercicis que vol que realitzin els seus alumnes. 3- L'usuari prem el botó guardar, si totes les dades són correctes els exercicis queden assignats als alumnes que té associats.
Fluxos alternatius	2b- L'usuari realitza un filtre dels exercicis.
Inclusions	Cap.
Extensions	Cap.

8.5.9 Cas d'ús: Realització dels exercicis assignats

Identificador	CU08
Nom	Realització dels exercicis assignats
Resum	L'alumne pot realitzar els exercicis que té assignats.
Actors	Alumne
Precondicions	L'usuari alumne ha iniciat la sessió. Tenir exercicis assignats no realitzats.
Postcondicions	Finalitzar exercici assignat.
Flux normal	<ol style="list-style-type: none"> 1- L'usuari selecciona del menú Realització d'exercicis. 2- El sistema mostra la pantalla inicial de realització d'exercicis, esperant que l'usuari confirmi iniciar els exercicis. 3- L'usuari fa clic en el boto següent. 4- Si hi ha un exercici pendent es mostra en la pantalla. 5- L'alumne realitza l'exercici i el confirma. 6- El sistema guarda quan s'ha fet l'exercici i quina nota n'ha obtingut.
Fluxos alternatius	5b- L'alumne cancel·la la realització d'exercicis.
Inclusions	Cap.
Extensions	Cap.

8.5.10 Cas d'ús: Veure resultats alumne

Identificador	CU09
Nom	Veure resultats alumne
Resum	Pot veure els resultats d'un alumne en concret.
Actors	Mestre
Precondicions	L'usuari mestre ha iniciat la sessió.
Postcondicions	Gràfic amb els resultats d'exercicis aprovats, suspesos i no realitzats.
Flux normal	<ol style="list-style-type: none"> 1- L'usuari selecciona del menú Estadístiques d'exercicis. 2- El sistema sol·licita data d'inici, data fi i les dades de l'alumne. 3- L'usuari introdueix les dades. 4- Prem el botó mostrar. El sistema comprova que tot sigui correcte, si ho és, mostra el gràfic amb la quantitat d'exercicis suspesos, aprovats i no realitzats.
Fluxos alternatius	2b- L'usuari cancel·la la consulta.
Inclusions	Cap.
Extensions	Cap.

8.5.11 Cas d'ús: Veure resultats en global

Identificador	CU10
Nom	<i>Veure resultats en global</i>
Resum	Veure els resultats dels tots els alumnes assignats.
Actors	Mestre
Precondicions	L'usuari mestre ha iniciat la sessió.
Postcondicions	Gràfic amb els resultats d'exercicis aprovats, suspesos i no realitzats.
Flux normal	<ol style="list-style-type: none"> 1- L'usuari selecciona del menú Estadístiques d'exercicis. 2- El sistema sol·licita data d'inici, data fi. 3- L'usuari introdueix les dades. 4- Prem el botó mostrar. 5- El sistema comprova que tot sigui correcte, si ho és, mostra el gràfic amb la quantitat d'exercicis suspesos, aprovats i no realitzats.
Fluxos alternatius	2b- L'usuari cancel·la la consulta.
Inclusions	Cap.
Extensions	Cap.

9. Disseny tècnic

Com arquitectura general del projecte PracticaClic es basa en un model Client-Servidor.

A continuació es descriu l'arquitectura que assumeix cada part :

9.1. Arquitectura plataforma web (part servidora)

La plataforma web conté la base de dades que utilitza un sistema gestor MySQL. Aquesta base de dades conté tota la informació del sistema.

El servidor web, és un servidor **Apache**.

També el servidor contindrà les interfícies web (pantalles), desenvolupades seguint el Standard **HTML5**. També s'utilitza el **javascript** per interactuar amb el client.

Pel que fa les funcionalitats principals del sistema estan desenvolupades amb **php**, també és la tecnologia encarregada de comunicar-se amb la base de dades MySQL.

9.2. Arquitectura plataforma mòbil (part client)

L'aplicació mòbil és una part de l'aplicació web, que aprofitant que esta desenvolupada amb el estàndard html5 i JavaScript, podem generar la aplicació “*nativa*” per les diferents plataformes mòbils utilitzant l'eina de Phonegap 3.0.

Aquesta aplicació únicament està desenvolupada per donar un accés més ràpid a l'alumne a l'hora de realitzar els diferents exercicis.

9.3. Arquitectura física

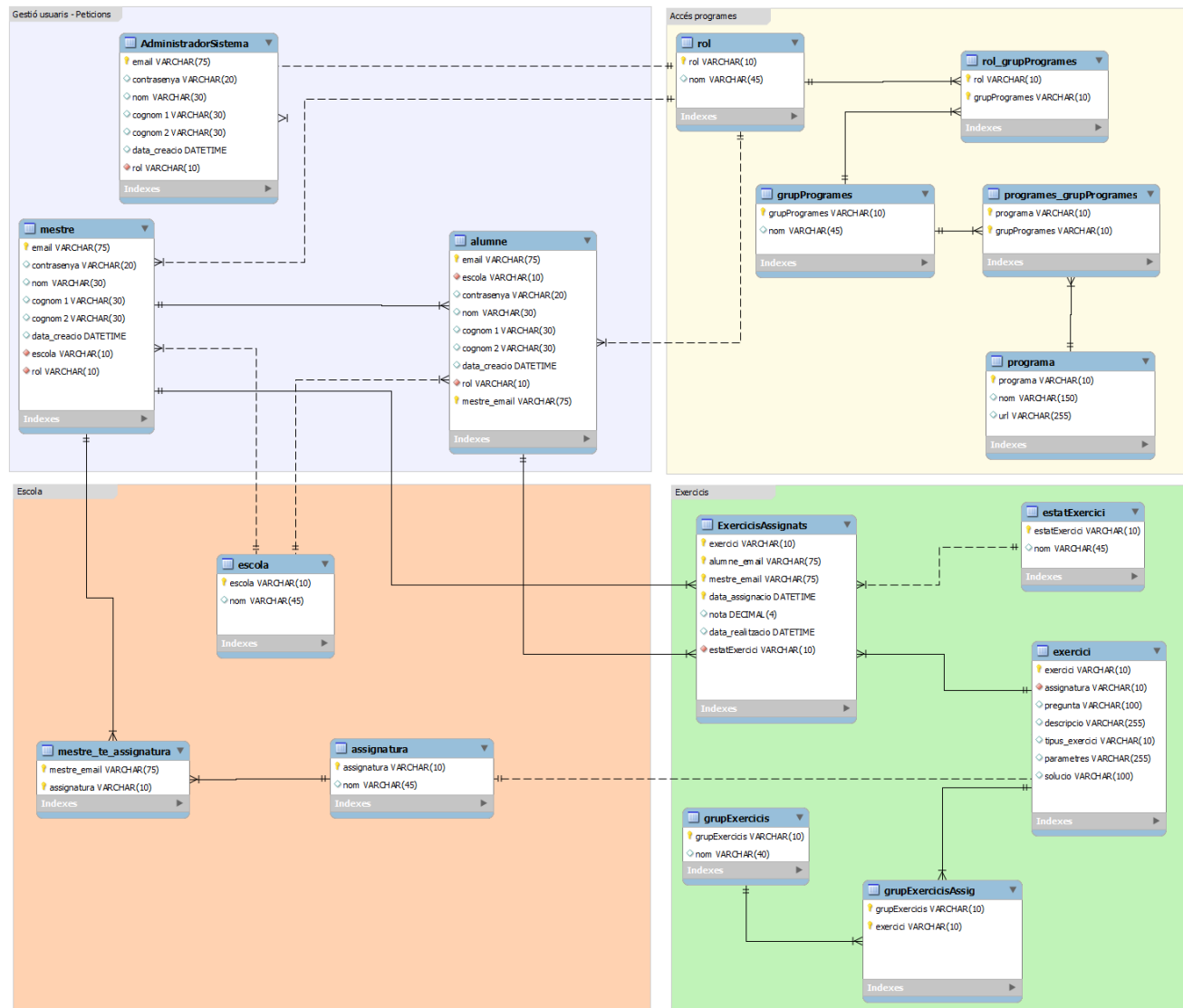
Per tal de fer el desenvolupament del projecte, he utilitzat una xarxa local que conté:

- Un ordinador amb el sistema operatiu Windows i com a servidor web l'Apache.
- Per a realitzar test de les pantalles d'interfície web, he utilitzat el mateix servidor i un portàtil connectat via Wi-fi a la xarxa.
- Per a realitzar test de les pantalles i el correcte funcionament de l'aplicació mòbil, he utilitzat un dispositiu mòbil : el model Nexus 5.

9.4. Arquitectura de base de dades

9.4.1 Model relacional de la base de dades.

A continuació es pot veure el disseny relacional de la base de dades :



9.4.2 Entitat Rol

Aquesta taula guarda els diferents rols de l'aplicació. En el sistema tenim dos usuaris diferents on cadascun pot accedir a uns determinats programes/pantalles segons el seu rol. Aquesta taula conté la definició d'aquests rols.

Per exemple, per l'usuari *mestre* es pot definir amb un codi RU_MESTRE que correspondrà a l'atribut *rol* de la taula , i per l'atribut *nom* es pot definir amb el text *Mestre*.

Rol				
Atribut	Descripció	Tipus	Longitud	Clau primària
rol	Guarda l'identificador del rol d'un usuari.	Text	10	SI
nom	Guarda un nom del rol d'un usuari.	Text	45	NO

9.4.3 Entitat grupProgrames

Aquesta entitat defineix un nom per agrupar una sèrie de programes/pantalles de l'aplicació, per tant es poden definir diferents grups de programes, que després es poden utilitzar per indicar quins rols el poden utilitzar. És a dir, es pot definir un grup de programes que després el pugui utilitzar tant el rol de mestre com el rol d'administrador del sistema.

grupProgrames				
Atribut	Descripció	Tipus	Longitud	Clau primària
grupProgrames	Guarda l'identificar d'un grup de programes.	Text	10	SI
nom	Defineix un nom pel grup de programes.	Text	45	NO

9.4.4 Entitat Rol_grupProgrames

Aquesta taula indica per cada rol d'usuari a quin grup de programes té accés.

Rol_grupProgrames				
Atribut	Descripció	Tipus	Longitud	Clau primària
rol	Guarda l'identificar del rol d'un usuari.	Text	10	SI
grupProgrames	Guarda l'identificador d'un grup de programes.	Text	10	SI

9.4.5 Entitat Programes_grupProgrames.

Aquesta entitat conté els programes que pertanyen a un grup de programa.

Programes_grupProgrames				
Atribut	Descripció	Tipus	Longitud	Clau primària
programa	Guarda l'identificar d'un programa.	Text	10	SI
grupProgrames	Guarda l'identificador d'un grup de programes.	Text	10	SI

9.4.6 Entitat Programa.

Conté la definició d'un programa/pantalla i la seva ubicació (url) dins del servidor web.

Quan un usuari ingressa en el sistema, es consulta quin és el seu rol per veure a quins programes pot accedir.

Programa				
Atribut	Descripció	Tipus	Longitud	Clau primària
programa	Guarda l'identificador del programa.	Text	10	SI
Nom	Guarda una descripció de la utilitat del programa.	Text	150	NO
url	Guarda la ubicació del programa.	Text	255	NO

9.4.7 Entitat Mestre.

Conté les dades d'un usuari mestre.

Mestre				
Atribut	Descripció	Tipus	Longitud	Clau primària
e-mail	Adreça de correu que s'utilitza com identificador d'un mestre.	Text	75	SI
Escola	Identificador de l'escola que pertany el mestre.	Text	10	NO
contrasenya	Guarda la contrasenya d'accés al sistema.	Text	20	NO
nom	Guarda el nom real de l'usuari.	Text	30	NO
Cognom1	Guarda el primer cognom de l'usuari.	Text	30	NO
Cognom2	Guarda el segon cognom de l'usuari.	Text	30	NO
Data_creacio	Dia que s'ha afegit l'usuari.	Data	-	NO
rol	Guarda el rol de l'usuari.	Text	10	NO

9.4.8 Entitat AdministradorSistema.

Conté les dades de l'administrador de tot el sistema.

AdministradorSistema				
Atribut	Descripció	Tipus	Longitud	Clau primària
email	Adreça de correu que s'utilitza com identificador d'un administrador del sistema.	Text	75	SI
contrasenya	Guarda la contrasenya d'accés al sistema.	Text	20	NO
nom	Guarda el nom real de l'usuari.	Text	30	NO
Cognom1	Guarda el primer cognom de l'usuari.	Text	30	NO
Cognom2	Guarda el segon cognom de l'usuari.	Text	30	NO
Data_creacio	Dia que s'ha afegit l'usuari.	Data	-	NO
rol	Guarda el rol de l'usuari.	Text	10	NO

9.4.9 Entitat Alumne.

Conté les dades dels usuaris del tipus Alumnes.

Alumne				
Atribut	Descripció	Tipus	Longitud	Clau primària
email	Nom d'usuari que identifica l'alumne. No té perquè ser un correu electrònic.	Text	75	SI
email_mestre	Adreça de correu que identifica el mestre de l'alumne.	Text	75	SI
Escola	Identificador de l'escola que pertany l'alumne	Text	10	NO
contrasenya	Guarda la contrasenya d'accés al sistema.	Text	20	NO
nom	Guarda el nom real de l'usuari.	Text	30	NO
Cognom1	Guarda el primer cognom de l'usuari.	Text	30	NO
Cognom2	Guarda el segon cognom de l'usuari.	Text	30	NO
Data_creacio	Dia que s'ha afegit l'usuari.	Data	-	NO
rol	Guarda el rol de l'usuari.	Text	10	NO

9.4.10 Entitat assignatura

Conté les diferents assignatures. Com ja he esmentat, aquest projecte només contindrà l'assignatura de matemàtiques, ja que, només contindrà exercicis de matemàtiques.

assignatura				
Atribut	Descripció	Tipus	Longitud	Clau primària
assignatura	Identificador de l'assignatura.	Text	10	SI
nom	Nom de l'assignatura.	Text	45	NO

9.4.11 Entitat Mestre_te_assignatura.

En aquesta taula es guardaran quines assignatures té assignades un mestre. En funció de les assignatures assignades, podrà posar uns determinats exercicis als seus alumnes o uns altres.

Mestre_te_assignatura				
Atribut	Descripció	Tipus	Longitud	Clau primària
Mestre_email	Identificador del mestre.	Text	75	SI
assignatura	Identificador de l'assignatura.	Text	10	SI

9.4.12 Entitat Escola.

Entitat que guarda les diferents escoles.

Escola				
Atribut	Descripció	Tipus	Longitud	Clau primària
Escola	Identificador de l'escola.	Text	10	SI
Nom	Nom de l'escola.	Text	45	NO

9.4.13 Entitat exercici

Conté la descripció d'un exercici i la seva ubicació (url) en el servidor.

exercici				
Atribut	Descripció	Tipus	Longitud	Clau primària
exercici	Identificador de l'exercici.	Text	10	SI
Assignatura	Assignatura a la que pertany l'exercici.	Text	10	NO
Pregunta	Text que conté la pregunta de l'exercici.	Text	100	NO
Descripcio	Descripció de l'exercici.	Text	255	NO
Tipus_exercici	Defineix quin tipus d'exercici pertany.	Text	10	NO
Paràmetres	Paràmetres que conté l'exercici.	Text	255	NO
Solucio	Resposta correcta de l'exercici	Text	100	NO

9.4.14 Entitat grupExercicis

Conté el nom d'un determinat grup d'exercicis.

grupExercicis				
Atribut	Descripció	Tipus	Longitud	Clau primària
grupExercicis	Identificador d'un grup d'exercicis.	Text	10	SI
nom	Nom descriptiu del grup.	Text	40	NO

9.4.15 Entitat grupExercicisAssig

Conté quins exercicis té un grup d'exercicis i a quina assignatura pertanyen.

grupExercicisAssig				
Atribut	Descripció	Tipus	Longitud	Clau primària
grupExercicis	Identificador d'un grup d'exercicis.	Text	10	SI
exercici	Identificador de l'exercici.	Text	10	SI

9.4.16 Entitat estatExercici

Defineix els diferents estats que pot tenir un exercici assignat a un alumne. Per exemple, pendent de realitzar, realitzat, etc.

EstatExercici				
Atribut	Descripció	Tipus	Longitud	Clau primària
estatExercici	Identificador de l'estat d'un exercici.	Text	10	SI
Nom	Descripció de l'estat de l'exercici.	Text	45	NO

9.4.17 Entitat exercicisAssignats.

Conté els exercicis que un mestre ha assignats als alumnes. En aquesta taula es guardarà la nota que ha tret l'alumne d'un exercici.

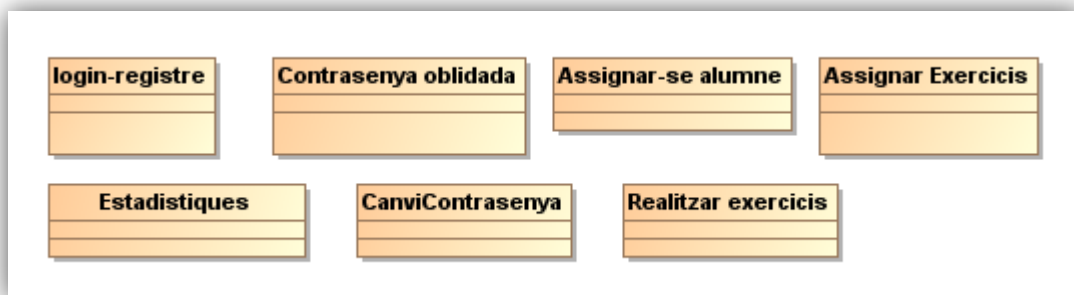
ExercicisAssignats				
Atribut	Descripció	Tipus	Longitud	Clau primària
exercici	Identificador de l'exercici assignat.	Text	10	SI
Alumne_email	Identificador de l'alumne.	Text	75	SI
Mestre_email	Identificador del mestre que ha assignat l'exercici.	Text	75	SI
Data_assignacio	Data que s'ha assignat l'exercici.	Data	-	SI
Nota	Nota que ha tret l'alumne de l'exercici.	Decimal	4	NO
Data_realitzacio	Data en que es realitza l'exercici.	Data	-	NO
estatExercici	Estat en que es troba l'assignació (Pendent, realitzat)	Numèric	-	NO

9.5. Diagrama de classes

9.5.1 Classes de la plataforma web (capa de presentació)

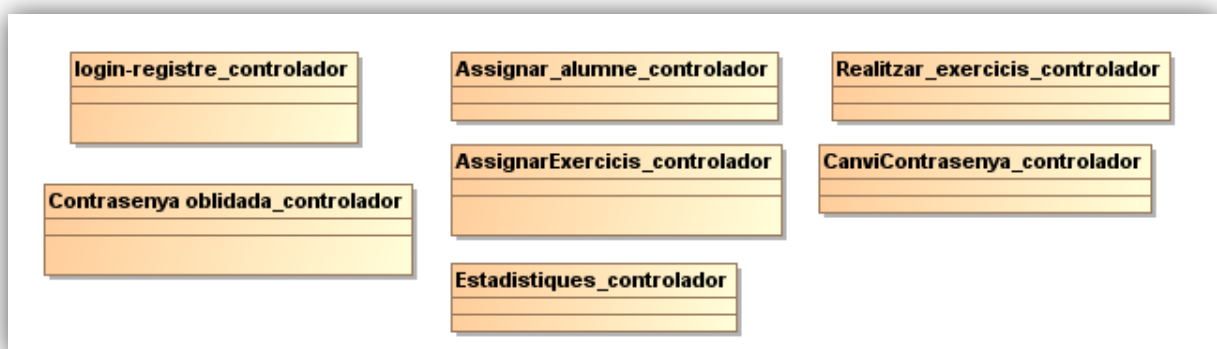
La capa de presentació de la plataforma web, està definida per interfícies amb tecnologia HTML5.

A continuació hi ha les pantalles que estan definides.



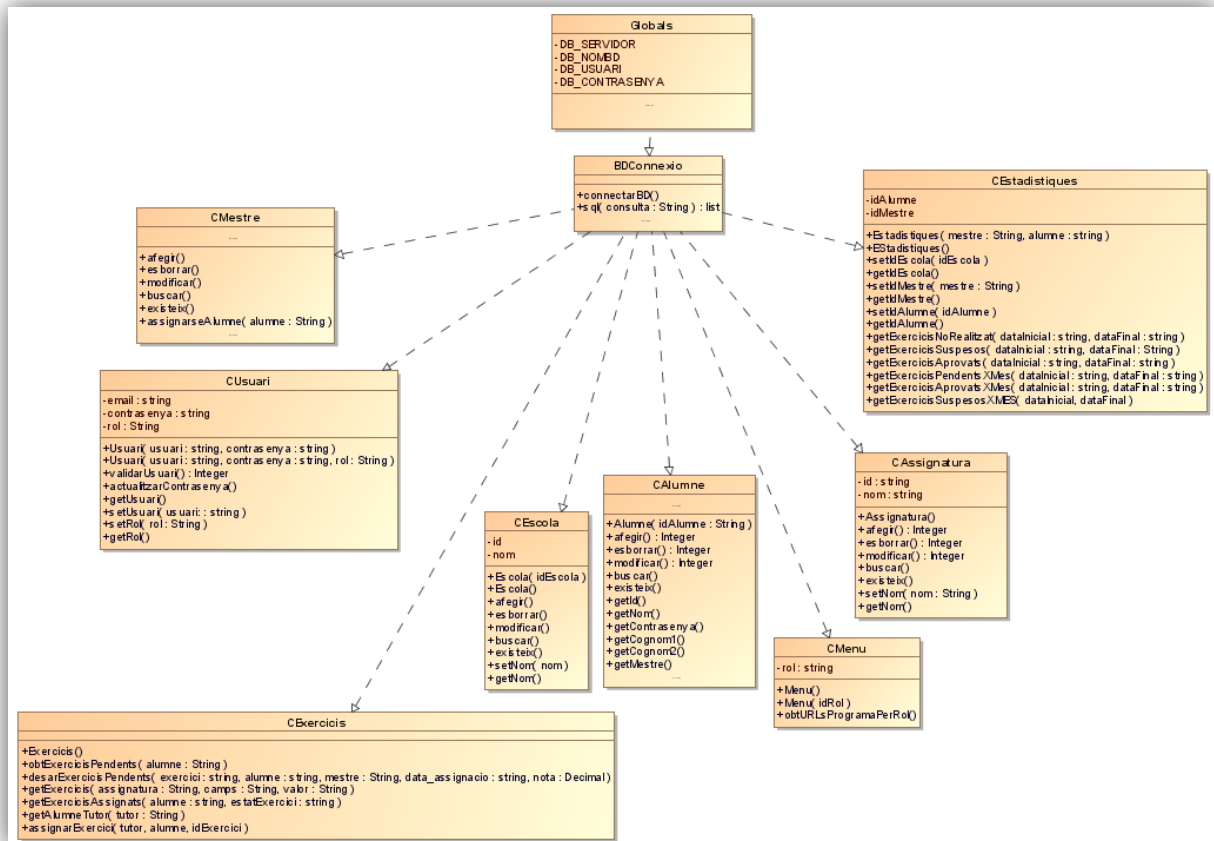
9.5.2 Classes de la plataforma web (capa de negoci)

Les següents classes s'han definit per interactuar amb les classes de la capa de presentació i comunicar amb la capa d'accés.



9.5.3 Classes de la plataforma web (capa d'accés)

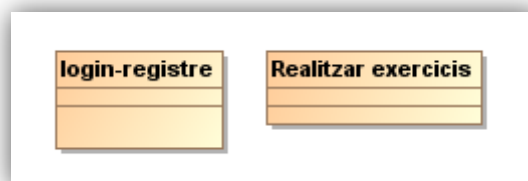
Es defineixen les classes per interactuar amb la connexió a la base de dades MySQL.



9.5.4 Classes de la plataforma mòbil (capa de presentació)

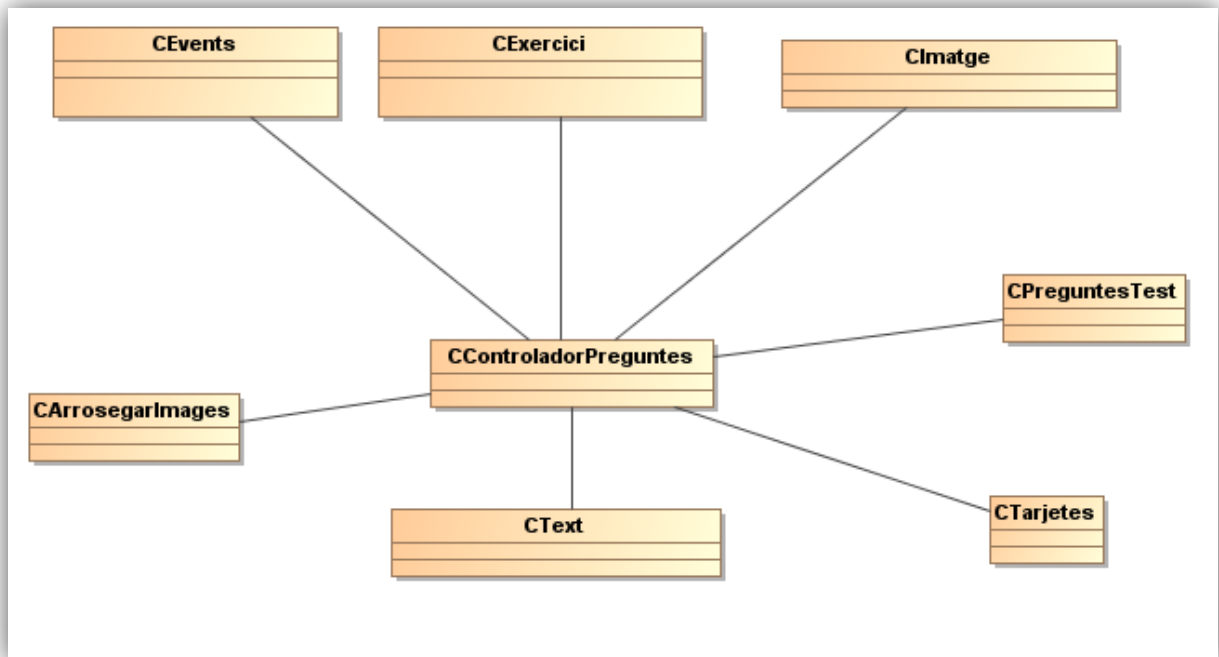
Pel que fa la capa de presentació de l'aplicació mòbil és bastant simple. Les classes corresponen a les pantalles de la interfície:

- Quan s'obre l'aplicació demana el nom d'usuari i contrasenya.
- Quan ha accedit es mostra la pantalla d'exercicis a realitzar.



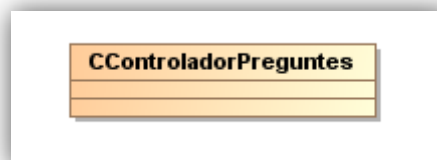
9.5.5 Classes de la plataforma mòbil (capa de negoci)

La classe “CControladorPreguntes”, és comunica amb la capa de presentació per mostrar els exercicis que cal realitzar.



9.5.6 Classes de la plataforma mòbil (capa d'accés)

Pel que fa la capa d'accés conté una classe que s'encarrega de fer les crides al servidor, quan arriba la resposta del servidor (una pàgina web amb HTML5) , la càrrega en la pantalla.



10. Prototip

La plataforma consta de dues parts d'interfícies visuals:

- La interfície visual de la plataforma web.
- La interfície visual de la plataforma mòbil.

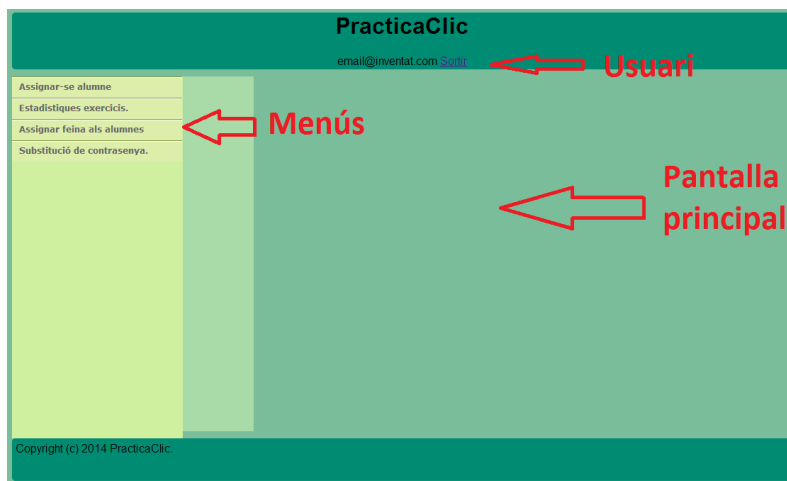
Aquestes pantalles serveixen com a referència, durant la fase de desenvolupament. És possible que un cop finalitzat el projecte, hi hagi diferències en les pantalles mostrades en el prototip i en les de l'aplicació final.

Tot i que hi ha moltes eines per a dissenyar pantalles d'aplicacions, he decidit realitzar-les directament amb *Eclipse*, tant per la plataforma web com per la plataforma mòbil. He realitzat el disseny utilitzant HTML5. Potser, no ha sigut el camí més ràpid per dissenyar les pantalles, però sí ha sigut útil per adquirir coneixements de HTML5, a més a més de tenir en compte que les pantalles ja estan realitzades.

A continuació es presenten els prototips d'algunes pantalles de la interfície visual de la plataforma web i la plataforma mòbil.

10.1. Característiques de les interfícies

La interfície consta de quatre apartats: la capçalera, el peu, el bloc de menús i la pantalla principal.



L'usuari pot veure amb quin identificador s'ha autenticat i també es podrà desconnectar del sistema amb el botó sortir.

En el bloc de menús, l'usuari veurà els accessos als programes depenent del seu rol.

La pantalla principal és on apareix el *programa* que ha seleccionat l'usuari.

10.2. Pantalla d'inici.

PracticaClic

Et vols donar d'alta?

email

Nom

Primer cognom

Segon cognom

Petició

Alumne

Nom Usuari

contrasenya

Inici

Mestre

email

contrasenya

[Contrasenya oblidada?](#)

Inici

Copyright (c) 2014 PracticaClic

Si accedim a l'aplicació amb un navegador web, s'inicia amb la pantalla d'identificació.

En el cas de ser un usuari registrat la pantalla ens sol·licita el nom d'usuari i contrasenya.

Al prémer el botó *Inici* el servidor valida les dades . Si són correctes, en funció del tipus d'usuari, tindrà accés als *programes* que li correspongui el seu rol.

Si un usuari registrat ha oblidat la contrasenya, el sistema li permetrà recuperar-la mitjançant l'enviament d'un e-mail sempre que sigui un usuari : *mestre/tutor*. En el cas de ser un usuari tipus : *alumne* haurà de demanar al seu tutor la nova contrasenya.

Per últim, si un usuari que no es troba registrat en el sistema però vol donar-se d'alta com a tutor/mestre, haurà d'introduir les dades sol·licitades. Al prémer el botó *Petició* , el sistema li enviarà un e-mail amb la contrasenya.

10.3. Pantalla assignar feina alumnes.

exercici	Grup	assignatura	pregunta
	Volum 1	Matemàtiques	Calcula la suma: 3 + 2
<input type="checkbox"/> MAT10	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 0?
<input type="checkbox"/> MAT100	Volum 1	Matemàtiques	Arossega 8 dofins dins el requadre.
<input type="checkbox"/> MAT11	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 1?
<input type="checkbox"/> MAT12	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 2?
<input type="checkbox"/> MAT13	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 3?
<input type="checkbox"/> MAT14	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 4?
<input type="checkbox"/> MAT15	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 5?
<input type="checkbox"/> MAT16	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 6?

En aquesta pantalla un mestre pot assignar diferents exercicis.

Podrà fer filtres dels possibles exercicis que hi ha a la base de dades.

Haurà de marcar el “check” per indicar que vol assignar l’exercici en concret i després prémer el botó guardar els exercicis quedaran assignats.

Si el tutor primer selecciona uns exercicis i després fa un filtre, els exercicis que havia seleccionat inicialment queden desmarcats.

10.4. Pantalla Assignar-se alumnes.

Aquesta pantalla ens permet al mestre/tutor inserir, esborrar alumnes. També ens permet modificar les seves dades.

El mestre inserirà un nou alumne al sistema i li quedarà associat.

Haurà de definir el nom d’usuari que tindrà l’alumne i la contrasenya.

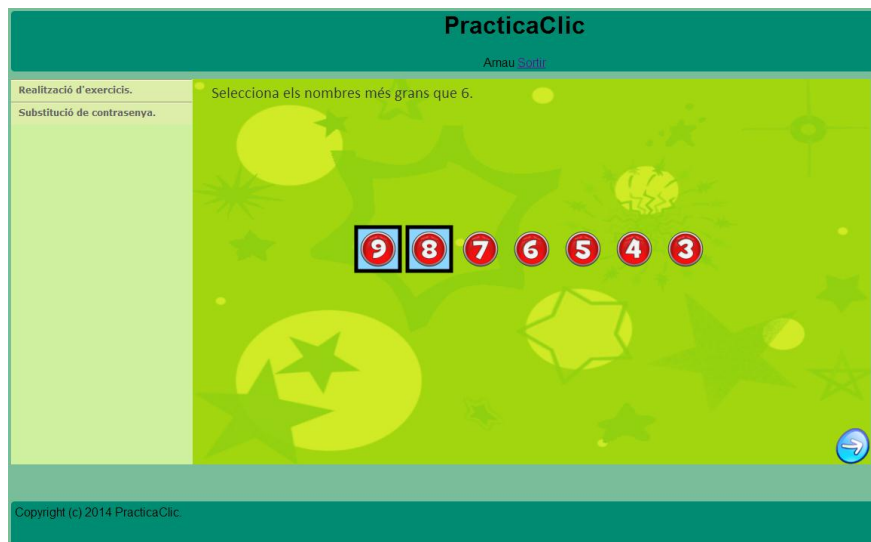
Per esborrar-lo o modificar-lo, cal que primer introdueixi el nom d’usuari de l’alumne, després el sistema buscarà en la base de dades l’usuari i si el troba el carregarà, llavors el podrà esborrar o bé modificar les seves dades.

10.5. Pantalla exercicis (interfície web).

En el projecte he definit 3 tipus d'exercicis:

- Exercicis tipus test.
- Exercici d'arrossegar amb el dit.
- Exercicis on s'ha de seleccionar una tarja (imatge).

Anem a veure alguns **exemples d'exercicis del tipus Test**. Aquest exercicis poden ser preguntes on la resposta es un valor numèric. Poden ser sumes, restes, etc. També es pot donar el cas de que la resposta tingui més d'una opció.



Els **exercicis del tipus seleccionar tarja** són semblants al del tipus test, però les respostes són imatges amb grups d'objectes.



Els **exercicis del tipus arrossegar objectes**, consisteixen amb una pregunta que demana inserir dins el requadre blau, una quantitat determinada dels objectes.



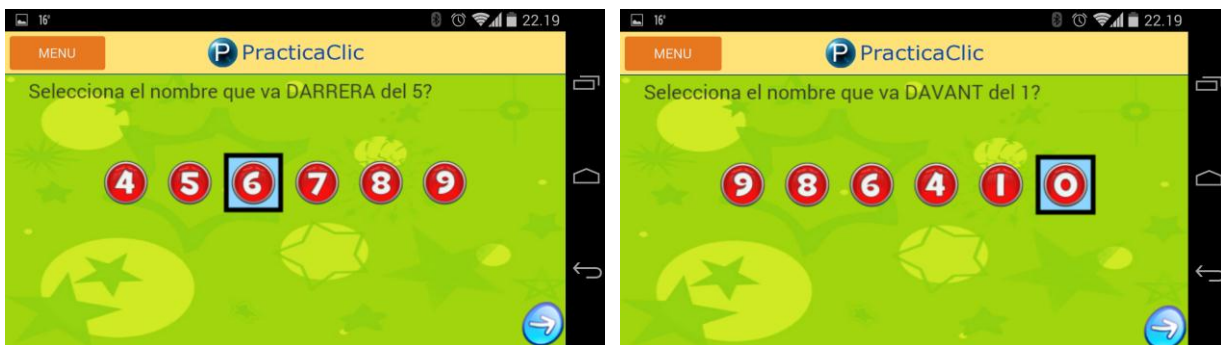
10.6. Pantalla exercicis (aplicació mòbil)

A continuació es mostra com queden els exercicis quan l'alumne realitza els exercicis en una aplicació mòbil.

Exemple d'exercicis del tipus tarja.



Exemple dels exercicis tipus test.



Exemple dels exercicis tipus arrossegar.



11. Implementació

Tot seguit es comenten els criteris que he seguit per dur a terme la implementació, així com comentaré també els detalls més importants.

11.1. Premisses de la implementació

Per la realització de la implementació s'han tingut en compte diferents aspectes per tal de garantir un correcte funcionament del sistema i també s'ha tingut en compte que tant l'aplicació web com l'aplicació mòbil siguin fàcils d'utilitzar per l'usuari final.

S'ha seguit uns criteris a l'hora de desenvolupar l'aplicació per tal d'obtenir un codi clar i ordenat, així doncs en el moment que s'hagin d'incorporar noves funcionalitats, es pugui dur a terme de manera àgil evitant al màxim els possibles errors.

11.2. Interfície

S'ha dissenyat una interfície, amb un menú a l'esquerra, on hi ha totes les opcions possibles segons l'usuari.

Quan un alumne vol realitzar els exercicis els pot realitzar des de l'entorn web, mitjançant un ordinador i amb accés a internet, però també ho pot fer amb un dispositiu mòbil ja sigui una tablet o un telèfon mòbil.

La pantalla dels exercicis es la mateixa tant per l'entorn web com la del dispositiu mòbil, de fet, es el mateix codi font, aquesta es una de les gran avantatges d'utilitzar les tecnologies html5, css, i javascript.

No existeix una aplicació mòbil pel tutor/mestre però si vol accedir amb un dispositiu mòbil haurà d'utilitzar el navegador del dispositiu.

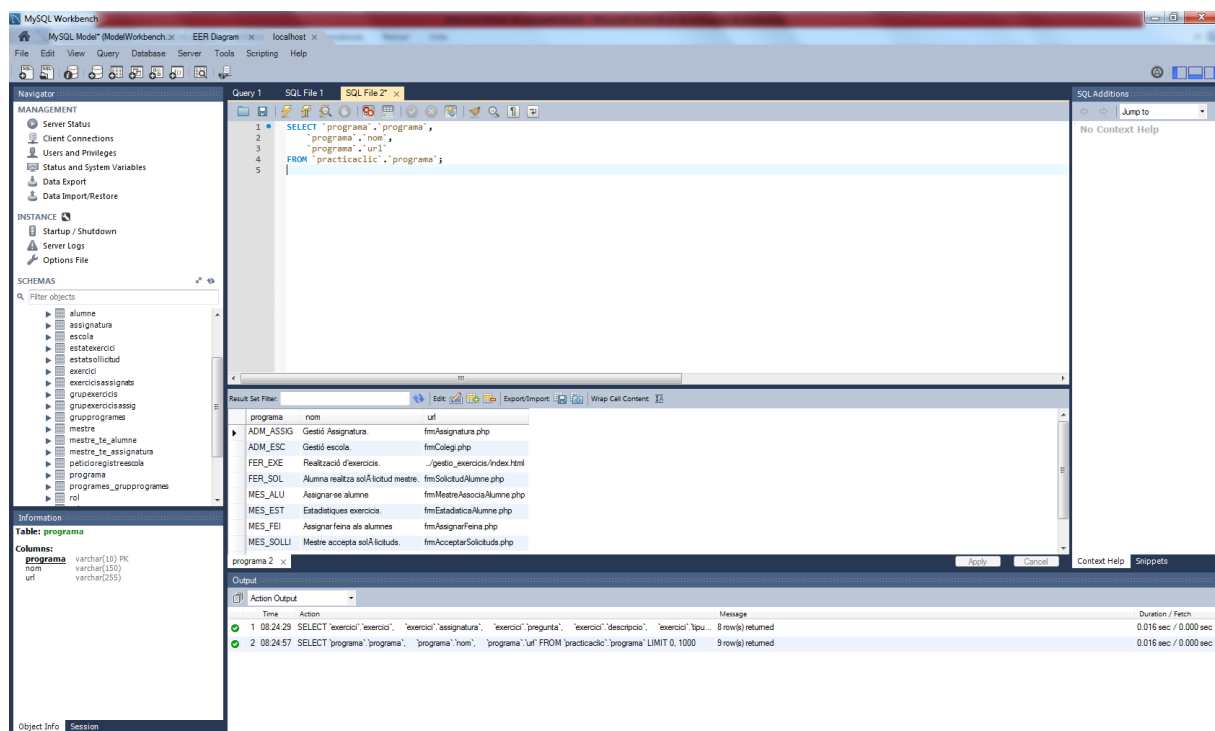
S'ha implementat l'aplicació mòbil pels alumnes, per obtenir un accés més ràpid, i més intuïtiu, envers a utilitzar un ordinador.

11.3. Implementació de base de dades servidor

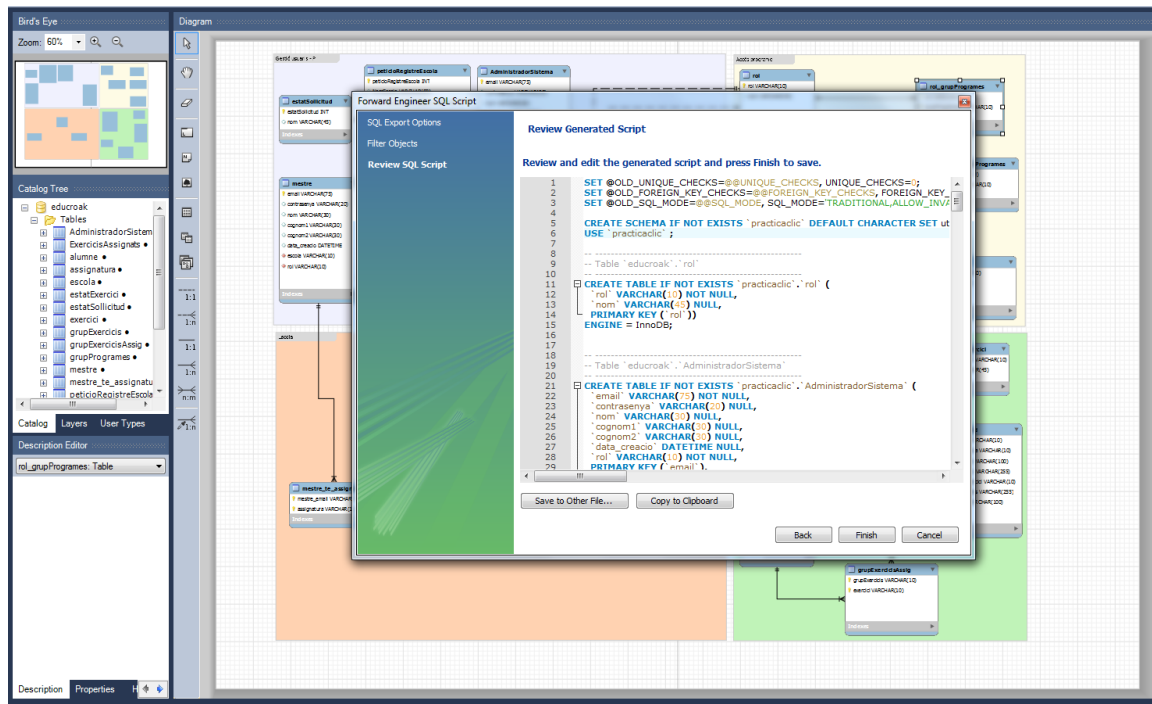
Per dissenyar la base de dades s'ha utilitzat l'eina *MySQL Workbench*. Aquesta eina permet dissenyar la base de dades gràficament. Un cop finalitzat el disseny permet generar l'script per una base de dades MySQL.

Amb *MySQL Workbench* també permet connectar-se amb el gestor de base de dades MySQL i realitzar consultes a la base de dades i així validar-les.

En la següent imatge podem veure com s'executa una consulta SQL.

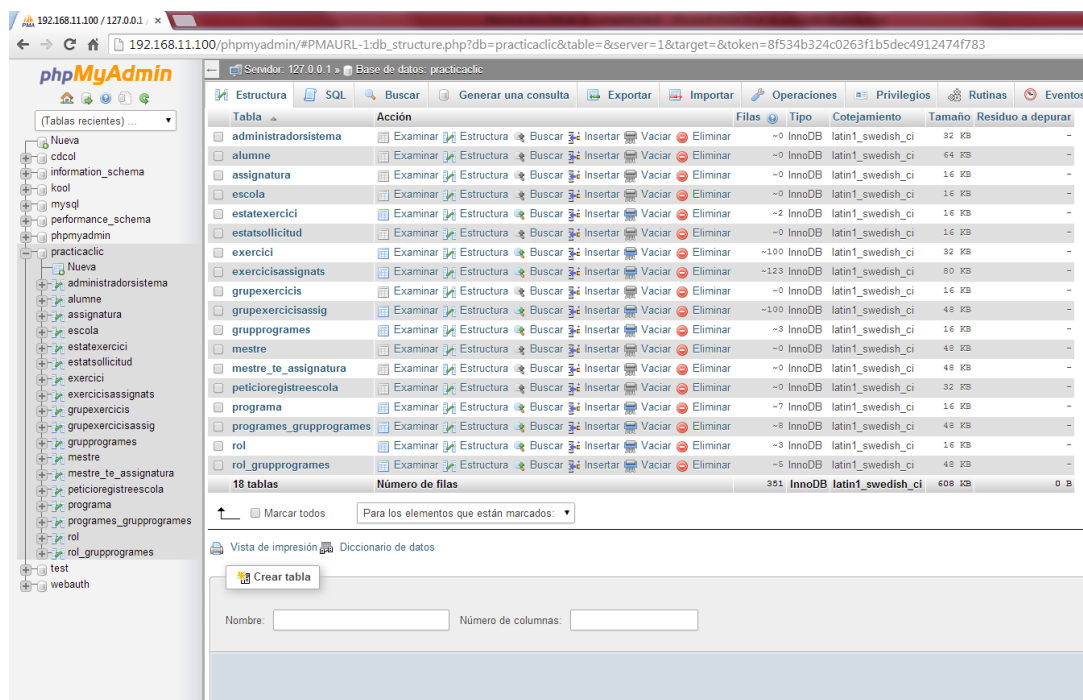


En la següent imatge, ens mostra com *MySQL Workbench* genera l'script per importar les taules a la base de dades.



Un cop s'ha generat l'script, s'importa al gestor de base de dades MySQL. Per a realitzar aquesta tasca s'ha utilitzat l'eina *phpMyAdmin*.

En la següent imatge, podem veure la base de dades carregada en el phpMyAdmin

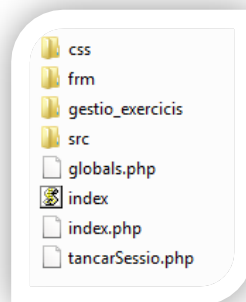


11.4. Implementació de la plataforma web

Per implementar la plataforma web, bàsicament he utilitzat les tecnologies HTML5, JavaScript i estils CSS.

A continuació es comenten alguns dels aspectes més importants de la implementació.

En la carpeta principal del servidor, ens trobarem els següent arxius:



D'entrada podem veure les carpetes `css`, `frm`, `gestió_exercicis`, i `src` anem a definir que contenen cadascuna.

- Carpeta “`css`” conté els diferents fulls d'estils CSS de l'entorn web.
- Carpeta “`frm`” conté els arxius de la capa de presentació i capa de negoci, més endavant els comentarem amb més detall.
- Carpeta “`gestió_exercicis`” conté tots els arxius relacionats amb la realització dels exercicis. Aquesta carpeta conté tots els arxius necessaris per “compilar” l'aplicació mòbil.
- Carpeta “`src`” conté els arxius de la capa d'accés de l'aplicació web.

També ens trobem amb els següents arxius :

- “`globals.php`” conté les variables globals definides del “php”.
- “`index.php`” és la pagina inicial de la plataforma web.
- “`index.js`”, aquest arxiu conté les funcions *JavaScript* necessàries per validar l'usuari quan vol iniciar una sessió. Aquest arxiu l'utilitza `index.php`. Per fer la validació es connecta mitjançant webservice utilitzant JSON.
- `tancarSessio.php`, quan un usuari tanca la sessió s'executa aquest arxiu per desconnectar-lo.

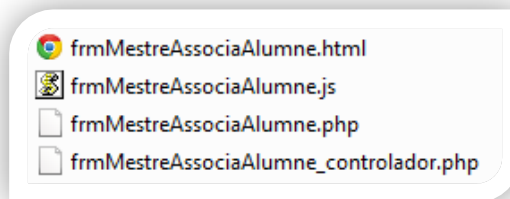
11.4.1. Capa de presentació i capa de negoci

Com s'ha comentat amb anterioritat la carpeta “frm” conté els arxius destinats a la capa de presentació i capa de negoci.

Per definir els noms dels arxius s'ha seguit el següent procediment:

- Els arxius començaran amb els 3 caràcters “frm”.
- Tots es diran igual menys l'arxiu capa de negoci que al final del nom de l'arxiu s'hi afegirà el text “_controlador”.
- Es distingiran per la seva extensió.

Per tant la majoria de les pantalles contenen 4 arxius:



- L'arxiu que conté l'interfície que interactua amb l'usuari. Esta implementat amb codi HTML5 i té l'extensió “php”.
- L'arxiu “js” conté les funcions necessàries per la pantalla de cares al client. (funcions web services, validació de dades, etc.).
- L'arxiu que finalitza per “_controlador.php”, és el controlador de la pantalla, es ha dir actuarà en funció de l'acció que l'usuari faci.

A continuació es mostra un fragment de codi de l'arxiu "frmMestreAssociaAlumne.php" que correspon al seu formulari :

```
<form class="contact_form" onSubmit="return false;" name="form1" id="form1" method="post">
<ul>
<li>
<div id="barraGuardar">
<input type="submit" value="Guardar" onclick="enviarDades(this);return
false;" id="afegir" name="afegir" />
<input type="submit" value="Esborrar" onclick="enviarDades(this);return
false;" id="esborrar" name="esborrar" />
<input type="submit" value="Netejar" onclick="formReset('form1');return
false;" />
<input type="submit" value="Ajuda"
onclick="obrirFinestra('frmMestreAssociaAlumne.html',400,300);return false;" />
</div>
<h2><?php echo utf8_encode("Gestió Alumnes") ?></h2>
</li>
<li>
</li>
<li>
<label for="codiAlumne">Usuari alumne:</label>
<input type="text" name="codi" id="codi" onBlur="agregarDades(this);return false;"
placeholder="Usuari alumne" required />
</li>
<li>
<label for="contrasenyaAlumne">Contrasenya:</label>
<input type="password" name="contrasenya" id="contrasenya" placeholder="*****"
required />
</li>
<li>
<label for="nomAlumne">Nom:</label>
<input type="text" name="nom" id="nom" placeholder="Nom" required />
</li>
<li>
<label for="cognomAlumne">Cognom:</label>
<input type="text" name="cognom" id="cognom" placeholder="Cognom" required />
</li>
<li>
<label for="cognom2Alumne">Segon cognom:</label>
<input type="text" name="cognom2" id="cognom2" placeholder="Segon cognom" />
</li>
</ul>
</form>
```

Quan l'usuari introdueix el nom d'un alumne, es connecta mitjançant JSON amb el servidor per comprovar si aquest alumne existeix, si es així omple el formulari amb les dades de l'alumne que ha trobat. Aquest tipus de funcions les trobem en els arxius amb extensió "js" .

Aquest exemple correspon a l'arxiu "frmMestreAssociaAlumne.js" :

```
function agregarDades(codi) {
    $.ajax({
        type: "POST",
        url:
"http://" + CVarGlobal.urlServior + "/frm/frmMestreAssociaAlumne_controlador.php",
        data: {"codi": codi.value},
        async: false,
        dataType: 'json',
        success: tractarDades,
        error: function (obj, error, objError) {
            alert(error);
        }
    });
}

function tractarDades(dades) {
    var dataJson = eval(dades);
    for(var i in dataJson) {
        document.form1.contrasenya.value = dataJson[i].CONTRASENYA;
        document.form1.nom.value = dataJson[i].NOM;
        document.form1.cognom.value = dataJson[i].COGNOM;
        document.form1.cognom2.value = dataJson[i].COGNOM2;
    }
}
```

A continuació es defineix el controlador.

Si es crida al controlador i no s'ha premut cap botó, voldrà dir que l'usuari acaba d'introduir el nom de l'alumne; llavors el controlador comprova si existeix, si es així li retorna les dades. En cas que l'usuari hagi fet clic en algun dels botons, el controlador valida que l'usuari hagi enviat totes les dades i comprova quina acció vol executar. Si és afegir un nou alumne o bé esborrar-lo.

El controlador es comunicarà amb la capa d'accés a dades per donar d'alta l'alumne i associar-lo al tutor o bé eliminar-lo segons l'acció pressa pel mestre.

```
<?php
    session_start();

    require_once("../src/CAlumne.php");
    require_once("../src/CMestre.php");

    if(empty($_POST['btn'])){
        $idAlumne = $_POST['codi'];
        $alumne = new CAlumne($idAlumne);
        if ($alumne->existeix()){
            $arr[] = array('CONTRASENYA' => utf8_encode($alumne->getContrasenya()),
                          'NOM' => utf8_encode($alumne->getNom()),
                          'COGNOM' => utf8_encode($alumne->getCognom1()),
                          'COGNOM2' => utf8_encode($alumne->getCognom2()));
            echo '' . json_encode($arr) . '';
        } else {
            echo '' . json_encode('') . '';
        }
    }
    else {
        $sBotoPres = $_POST['btn'];
        if (isset($_POST["codi"]) && isset($_POST["nom"]) && isset($_POST["contrasenya"]) &&
            isset($_POST["cognom1"]) && isset($_POST["cognom2"])){

            $idAlumne = $_POST['codi'];
            $nom = addslashes($_POST['nom']);
            $contrasenya = addslashes($_POST['contrasenya']);
            $cognom1 = addslashes($_POST['cognom1']);
            $cognom2 = addslashes($_POST['cognom2']);
            if ($sBotoPres != '') {
                $alumne = new CAlumne($idAlumne,$contrasenya,$nom,$cognom1,$cognom2);
                switch ($sBotoPres) {
                    case 'afegir': afegirActualitzar($alumne,$SESSION['userid']); break;
                    case 'esborrar': $alumne->esborrar(); echo "esborrat"; break;
                }
            }
        }
    }

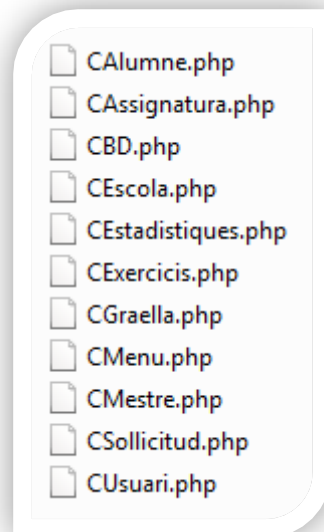
    function afegirActualitzar($alumne,$strMestre){
        if ($alumne->existeix()){
            $alumne->modificar();
            echo "Actualitzat";
        } else {
            $alumne->afegir();
            echo "Afegit";
        }
        $mestre = new CMestre($strMestre);
        $mestre->assignarseAlumne($alumne->getId());
    }

    ?>
```

L'arxiu "frmMestreAssociaAlumne.html", conté una breu explicació del funcionament de la pantalla.

11.4.2. Capa d'accés

En la carpeta “src” del directori arrel del servidor, trobem els arxius referents a la capa d'accés. Tots aquest arxius estan definits amb el llenguatge php. Tots comencen amb el caràcter “C” indicant que conté la definició d'una classe.



La classe CBD, ens permet connectar amb la base de dades i amb el mètode *sql*, es podem realitzar consultes.

```
<?php
class CBD {
    private $bd_servidor;
    private $bd_nomBD;
    private $bd_usuari;
    private $bd_contrasenya;
    private $bd_link;

    function __construct() {
        require_once("../globals.php");
        $this->bd_servidor = DB_SERVER;
        $this->bd_nomBD = DB_NAME;
        $this->bd_usuari = DB_USER;
        $this->bd_contrasenya = DB_PASS;

        $this->connectarBD();
    }

    private function connectarBD(){
        $this->bd_link = mysql_connect($this->bd_servidor, $this->bd_usuari, $this->bd_contrasenya)
        or die('No es pot connectar amb la base de dades: ' . mysql_error());
        mysql_select_db($this->bd_nomBD) or die('No s\'ha pogut seleccionar la base de dades.' . $this->bd_nomBD);
    }

    public function sql($consulta){
        $result = mysql_query($consulta) or die('Consulta fallida: ' . mysql_error());
        return $result;
    }
}
?>
```

A partir d'aquí s'implementen les classes necessàries utilitzant la classe CBD.

Si observem la classe CALumne ,d'entrada, podem veure els seus atributs i la definició de les seves constants que fan referència a la taula alumnes de la base de dades.

```
private $email; //Camp clau.
private $contrasenya;
private $nom;
private $cognom1;
private $cognom2;
private $data_creacio;
private $rol;
private $escola;
private $mestre;

const CAMP_CLAU = 'email'; //CAMP CLAU
const NOM = 'nom';
const CONTRASENYA='contrasenya';
const COGNOM1='cognom1';
const COGNOM2='cognom2';
const DATA_CREACIO='data_creacio';
const ROL='rol';
const ESCOLA='escola';
const MESTRE_EMAIL='mestre_email';

const NOM_TAULA = 'alumne';
const ROL_ALUMNE = 'ALUMNE';
```

També es defineix una variable per la gestió de consultes amb la base de dades.

```
private $bd;
```

Com que en “php” no es poden tenir més d'un constructor. Ho solucionem de la següent forma:

```
function __construct(){
    require_once("CBD.php");
    $this->bd = new CBD();
    $params = func_get_args();
    $num_params = func_num_args();
    $function_constructor = '__construct' . $num_params;
    if (method_exists($this,$function_constructor)) {
        call_user_func_array(array($this,$function_constructor), $params);
    }
}

function __construct0(){
    $this->email="";
    $this->contrasenya="";
    $this->nom="";
    $this->cognom1="";
    $this->cognom2="";
    $this->data_creacio="";
    $this->rol=self::ROL_ALUMNE;
    $this->escola="";
}

function __construct1($email){
    $this->email=$email;
    $this->buscar();
}
```

Per exemple per definir el mètode esborrar, cal fer la crida a la funció *sql* de la classe CBD.

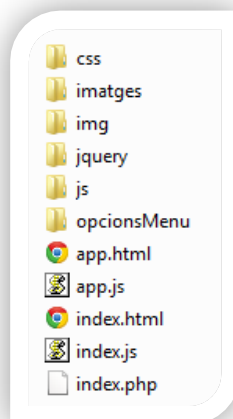
```
public function esborrar() {
    $sql = $this->bd->sql("DELETE FROM `".self::NOM_TAUOLA."` WHERE
        ".self::CAMP_CLAU."='".$this->email."'");
}
```

Per definir mètodes getters i setters es tan fàcil com en l'exemple que es mostra a continuació.

```
public function getNom() {
    return $this->nom;
}
public function setNom($nom) {
    $this->nom = $nom;
}
```

11.4.3. Implementació comuna tant del servidor com aplicació mòbil.

La part que s'explica a continuació tant pertany al servidor com a l'aplicació mòbil, gràcies a la utilització de HTML5, JavaScript i estils CSS. Tota aquesta part d'implementació la podem trobar dins la carpeta "gestió_exercicis", dins la carpeta arrel del servidor.

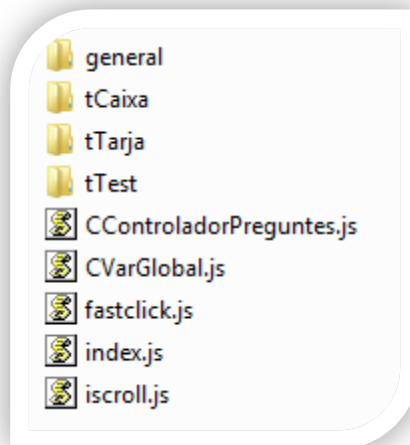


En aquesta carpeta hi ha dos arxius: un anomenat "index.html" i l'altre "index.php".

El primer correspon a l'aplicació mòbil i el segon a l'aplicació web. La diferència està en que l'aplicació mòbil s'ha d'identificar, mentre que en l'aplicació web no cal perquè ja ho ha fet amb anterioritat.

Per tant per "compilar" l'aplicació mòbil, l'arxiu "index.php" no ens cal, ja que s'utilitza l'arxiu "index.html". Aquest arxiu i els fulls d'estils són l'única diferència entre l'aplicació web i l'aplicació mòbil. Tota la resta del contingut d'aquesta carpeta s'utilitza per les dues plataformes.

Anem a comentar la distribució d'aquestes carpetes i com està implementada.



L'arxiu "*CControladorPreguntes.js*" conté la classe encarregada de mostrar els exercicis que s'han de realitzar per part de l'alumne.

Els paràmetres del constructor d'aquesta classe, són l' identificador del *Canvas* (element html5) i l' identificador de l'alumne.

Per tant, en la pàgina web html5 hi ha d'haver un Canvas on es mostraran els diferents exercicis a realitzar. I com que els exercicis els realitza un alumne en concret, la classe ha de saber de quin alumne es tracta.

El constructor de la classe d'entrada dibuixarà els fons dels exercicis i el botó següent en el *Canvas*.

Aquesta classe també controla els events del *Canvas* amb l'ajuda de la classe *CEvents.js* que es troba dins la carpeta *general*.

Quan l'alumne fa clic en el botó "*següent*" del canvas, la classe *CControladorPreguntes.js* es connecta amb el servidor mitjançant JSON per comprovar si l'alumne té exercicis pendents.

Com a paràmetres de la crida del WebService s'indica que la "*tasca*" té el valor de següent, això indica al controlador del WebService que cal comprovar si l'alumne té més exercicis. També s'indica de quin alumne es tracta.

```
function ajaxSeguentPregunta (alumne) {
    $.ajax({
        type: "POST",
        url: urlWsExercicis,
        data: { "tasca": "següent", "alumne": alumne },
        async: false,
        dataType: 'json',
        success: tractarDades,
        error: function (obj, error, objError) {
            //avisar que ocorrió un error
            alert(error);
        }
    });
}
```

Un cop s'ha rebut la resposta del servidor, es carreguen les dades en l'objecte *CExercicis.js*. L'arxiu corresponent el trobem en la carpeta general.

```
function tractarDades(dades) {
    exercici = null;
    exercici = new CExercici();

    var dataJson = eval(dades);
    for(var i in dataJson){
        exercici.tipus = dataJson[i].TIPUS;
        exercici.pregunta = dataJson[i].PREGUNTA;
        exercici.parametres = dataJson[i].PARAM;
        exercici.exercici = dataJson[i].EXERCICI;
        exercici.solucio = dataJson[i].SOLUCIO;
        exercici.dataAssignacio = dataJson[i].DATA_ASSIGNACIO;
    }
}
```

Els paràmetres són:

- Tipus d'exercici: Amb aquest paràmetre sap quin tipus d'exercici ha de mostrar.
- Pregunta: És la pregunta que es mostra en l'exercici.
- Parametres: Són els possibles paràmetres que pot contenir segons el tipus d'exercici.
- Solucio: Conté el valor de la solució del exercici.
- dataAssignacio: El dia que va assignar l'exercici a l'alumne.

L'objecte exercicis indicarà a l'objecte *CControladorPreguntes*, quin exercici s'ha de mostrar en el Canvas.

CControladorPreguntes, quan detecti que l'alumne ha fet clic en el botó següent, haurà de connectar-se en el servidor mitjançant JSON per comprovar si hi ha algun exercici pendent i en el cas de ser així, comprovarà de quin tipus d'exercici es tracta.

Pot ser del tipus "tTest", "tTarja" o "tCaixa".

```
switch(exercici.tipus){
    case "tTest": prv = new CPreguntesTest(this,exercici,jo);break;
    case "tTarja": prv = new CTarjetas(this,exercici,jo); break;
    case "tCaixa": prv = new ArrosegatImages(this,exercici,jo);break;
    default;;
}
```

En funció del tipus d'exercici, crea un objecte o un altre i aquest serà l'encarregat de dibuixar per pantalla l'exercici i de gestionar els events. Si l'alumne fa clic en el botó següent i ja havia resolt un exercici, primer el guardarà, per tant farà una crida JSON, indicant:

- De quin exercici es tracta.
- De l'alumne que l'ha realitzat.
- La data d'assignació.
- I la nota que ha tret del exercici.

```
function ajaxDesarPregunta(exercici, alumne, data_assignacio, nota) {
    $.ajax({
        type: "POST",
        url: urlWsExercicis,
        data:
        { "tasca": "desarResultatExercici", "exer": exercici, "alumne": alumne, "nota": nota, "data_assignacio": data_assignacio },
        async: false,
        success: tractarDadesDesar,
        error: function (obj, error, objError) {
            alert(error);
        }
    });
}
```

El controlador d'aquestes dues crides WebService, es mostra a continuació:

```
<?php

require_once("../src/CAlumne.php");
require_once("../src/CExercicis.php");

$accio = $_POST['tasca'];

switch ($accio) {
    case "seguent":
        seguentExercici($_POST['alumne']);
        break;
    case "desarResultatExercici":
        $exercici = $_POST['exer'];
        $alumne = $_POST['alumne'];
        $nota = $_POST['nota'];
        $data_assignacio = $_POST['data_assignacio'];
        desarResultatExercici($exercici, $alumne, $data_assignacio, $nota);

        break;
    default: break;
}

function seguentExercici($alumne) {
    $e = new CExercicis();
    echo $e->obtExercicisPendants($alumne);
}

function desarResultatExercici($exercici, $alumne, $data_assignacio, $nota) {
    $calumne = new CAlumne($alumne);
    $mestre = $calumne->getMestre();
    $e = new CExercicis();
    $e->desarExercicisPendants($exercici, $alumne, $mestre, $data_assignacio, $nota);
}

?>
```

Si el paràmetre “tasca” conté el valor de *seguent*, crea un objecte *CExercici*, i crida al mètode *obtExercicisPendants*; el qual retorna un exercici (al més antic), si n’hi ha.

En cas de que l’acció sigui desar el resultat de l’exercici, primer crearà l’objecte *CAlumne*, per consultar quin es el seu tutor/mestre, i després amb l’objecte de la classe *CExercicis* cridarà al mètode *desarExercicisPendants*, per enregistrar que l’alumne ja l’ha realitzat l’exercici i amb quina nota (també s’enregistrarà el dia i hora que l’ha realitzat.).

Amb anterioritat s'ha comentat que en funció del tipus d'exercici, es crea un objecte o un altre. Quan es vulgui afegir un nou tipus d'exercici al sistema, podem definir un nou objecte amb les funcionalitats que siguin necessàries per aquest nou tipus d'exercici i afegir-lo en el *switch* que s'ha mostrat abans.

Els objectes corresponents al tipus d'exercici estan guardats cadascun en la seva carpeta.

- Carpeta "*tCaixa*", conté l'objecte del tipus d'exercici que ha d'arrossegar les imatges dins del quadre.
- Carpeta "*tTarja*", conté l'objecte que mostra els exercicis del tipus que hi ha imatges que es poden seleccionar en funció de la pregunta.
- Carpeta "*tTest*", conté l'objecte que mostra els exercicis tipus test.

Els exercicis s'enregistren en la base de dades en la taula exercicis de la següent forma:

exercici	assignatura	pregunta	descripcio	tipus_exercici	parametres	solucio
MAT1	MATES	Calcula la suma: 3 + 2		tTest	5,6,9,1,2;	5
MAT10	MATES	Selecciona el nombre que va DARRERA del 0?		tTest	1,2,6,5,0;	1
MAT100	MATES	Arrossega 8 dofins dins el requadre.		tCaixa	dolphin,dolphin,dolphin,dolphin,dolphin,dolphin,do...	dolphin,dolphin,dol
MAT11	MATES	Selecciona el nombre que va DARRERA del 1?		tTest	1,2,6,5,0;	2
MAT12	MATES	Selecciona el nombre que va DARRERA del 2?		tTest	2,3,4,5,0;	3
MAT13	MATES	Selecciona el nombre que va DARRERA del 3?		tTest	1,2,4,5,0;	4
MAT14	MATES	Selecciona el nombre que va DARRERA del 4?		tTest	7,4,6,5,0;	5
MAT15	MATES	Selecciona el nombre que va DARRERA del 5?		tTest	4,5,6,7,8,9;	6
MAT16	MATES	Selecciona el nombre que va DARRERA del 6?		tTest	6,5,7,3,9;	7
MAT17	MATES	Selecciona el nombre que va DARRERA del 7?		tTest	4,8,7,3,9;	8
MAT18	MATES	Selecciona el nombre que va DARRERA del 8?		tTest	7,5,8,9,3;	9

Així quan l'aplicació demana el següent exercici, es consulta a la base de dades per saber de quin tipus d'exercici es tracta, (ja sigui tTest, tCaixa, tTarja), obtenir la pregunta que ha de realitzar i quina es la solució.

El camp paràmetres indica quines imatges ha de carregar. Per exemple, si ens fixem amb el primer exercici, es pot veure que el camp paràmetres val 5,6,9,1,2; això està indicant a l'objecte "*tTest*" que ha de mostrar com a possibles respostes les imatges 5, 6, 9, 1 i 2.

Per tant l'objecte *tTest* anirà a buscar-les en la seva carpeta d'imatges. Per fer-ho afegirà l'extensió (png) (5.png,6.png,etc..) el punt i coma del final dels paràmetres és per indicar que s'han finalitzat les imatges, si tot seguit hi hagués el valor 1 indicaria que l'exercici pot tenir més d'una resposta.

En funció del tipus d'objecte el camp paràmetres tindrà uns valors o uns altres.

Afegir nous exercicis és molt senzill, ja que només caldrà afegir nous registres.

Actualment he afegit 100 exercicis per poder testejar l'aplicació.

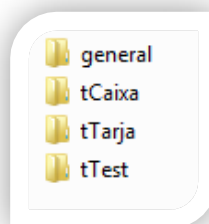
Per inserir els exercicis, primer s'han definit Microsoft excel i després s'ha importat a la base de dades de MySQL.

Si continuem observant el contingut de la carpeta `gestio_exercicis\js` es poden veure les següent carpetes:

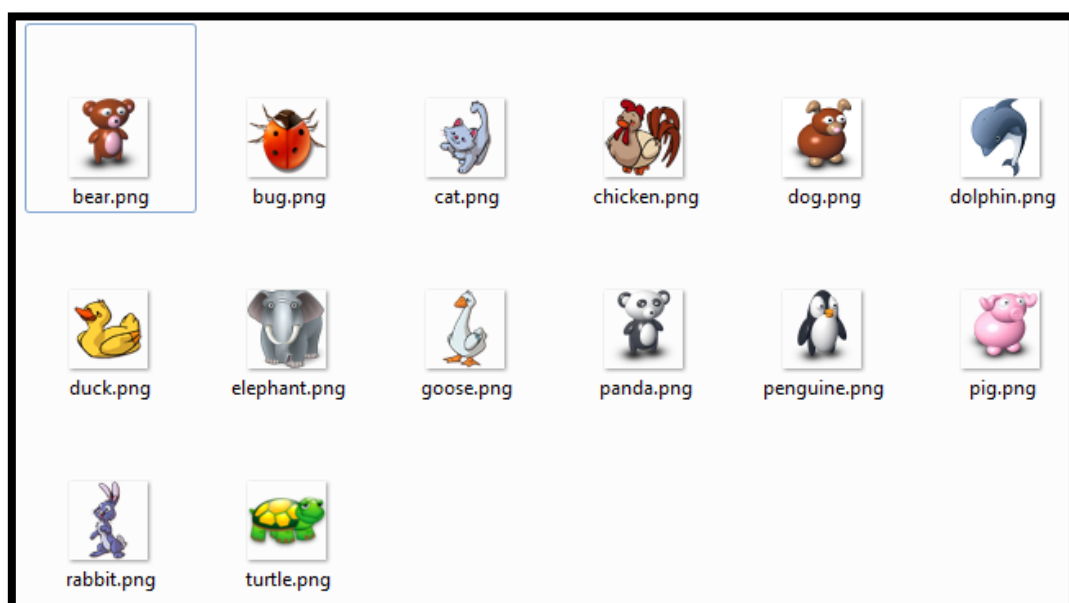
- “css” conté els fulls d'estils, tant els de l'aplicació web com els de l'aplicació mòbil.
- En la carpeta imatges, hi trobem la carpeta “general” que conté les imatges de l'aplicació, el fons i el botó de confirmació d'exercicis.

També ens trobem 3 carpetes (**tCaixa**, **tTarja**, **tTest**), les quals, cadascuna d'elles conté les imatges d'un tipus d'exercici.

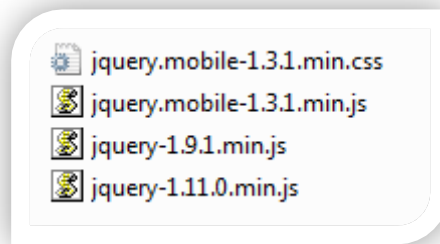
Per exemple, dins la carpeta “**tCaixa**” hi ha totes les imatges que poden aparèixer en els exercicis d'aquest tipus.



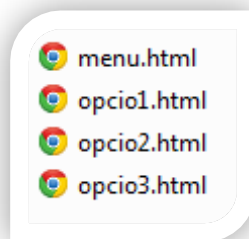
En la carpeta “**tCaixa**”, ens trobarem imatges com les que es mostren a continuació :



En la carpeta “**jquery**” trobem les llibreries jquery :



En la carpeta “**opcionsMenu**” trobem les opcions del menú de l’aplicació mòbil .:



11.5. Implementació de la plataforma mòbil

En el punt anterior s’ha comentat la part comuna de l’entorn web i aplicació mòbil, hem vist que hi ha alguns arxius que únicament són utilitzats per l’aplicació mòbil. Però s’han mantingut en aquest carpeta per facilitar la *compilació* de l’aplicació mòbil i evitar tenir arxius duplicats.

Aquesta aplicació s’ha “compilat” amb **Phonegap 3.0** (en realitat amb **cordova**). Amb aquesta eina es pot “compilar” per diferents plataformes mòbils, com poden ser Android, IOS, Windows Phone, etc.

Per compilar l’aplicació mòbil primer cal crear el projecte amb **cordova**. Això ho realitzo amb la consola del sistema operatiu.

En aquest cas com s’està implementat amb us sistema operatiu Windows utilitzo la consola *cmd*. En aquest punt, hem d’haver instal·lat les SDK de la plataforma mòbil en la qual volem fer la compilació, i afegir la plataforma a **cordova**.

```

C:\Users\pc>cordova create PracticaClic com.jrv.practicaclic PracticaClic
Creating a new cordova project with name "PracticaClic" and id "com.jrv.practicaclic" at location "C:\Users\pc\PracticaClic"
C:\Users\pc>

```

La pantalla del mòbil s'ha bloquejat perquè només es puguin visualitzar els exercicis en horitzontal.



Per tal que es pugui mantenir la pantalla horitzontal, cal accedir a la carpeta del projecte de l'aplicació i modificar l'arxiu `\PracticaClic\platforms\android\AndroidManifest.xml`, i afegir el valor `android:screenOrientation="landscape"` en l'activitat principal. Després copiarem tots els arxius de la carpeta "gestio_exercicis" menys `index.php`, dins la carpeta `PracticaClic\www`.

Ara ja podem compilar per l'aplicació mòbil per Android.

```

C:\Users\pc\PracticaClic>cordova build android_

```

Un cop compilat amb **cordova** per la plataforma Android, he realitzat el test de l'aplicació.

Nota : També es podria haver executat la comanda "cordova run android", i el que faríem amb això, seria executar l'aplicació amb l'emulador d'Android.

11.6 Eines de validació i debug.

Per validar que s'ha definit correctament l'html5, existeix una eina online.
(<http://validator.w3.org/>)

Per validar el codi html5, podem :

- indicar l'url d'on es troba el document,
- pujar un document del nostre escriptori,
- o simplement copiar el codi.

Un cop hem triat una d'aquestes tres opcions, li podem demanar que faci la comprovació.

En la següent imatge es pot veure com s'ha volgut validar un document; però aquest ens avisa de que conté 2 errors.

The screenshot shows the W3C Markup Validation Service interface. At the top, it says "Markup Validation Service" and "Check the markup (HTML, XHTML, ...) of Web documents". Below this, there are tabs for "Jump To: Notes and Potential Issues" and "Validation Output". The main content area has a red header that says "Errors found while checking this document as HTML5!". Below this, a table shows the validation results:

Result:	2 Errors, 1 warning(s)
Address:	<input type="text" value="http://practicaclio/"/>
Encoding:	utf-8 (detect automatically)
Doctype:	HTML5 (detect automatically)
Root Element:	html

Below the table, there is a section for "W3C VALIDATOR Suite" with a link to "Try now the W3C Validator Suite™". It also mentions that the W3C validators are developed with assistance from the Mozilla Foundation and supported by community donations. There is a "Donate" link and a "Flat" button.

Validation Output: 2 Errors

✖ Line 19, Column 51: Bad value for attribute action on element form: Must be non-empty.

```
<form action="" method="post" class="registreNou" >
```

Syntax of IRI reference:

Any URL. For example: /hello, #canvas, Or <http://example.org/>. Characters should be represented in [NFC](#) and spaces should be escaped as %20.

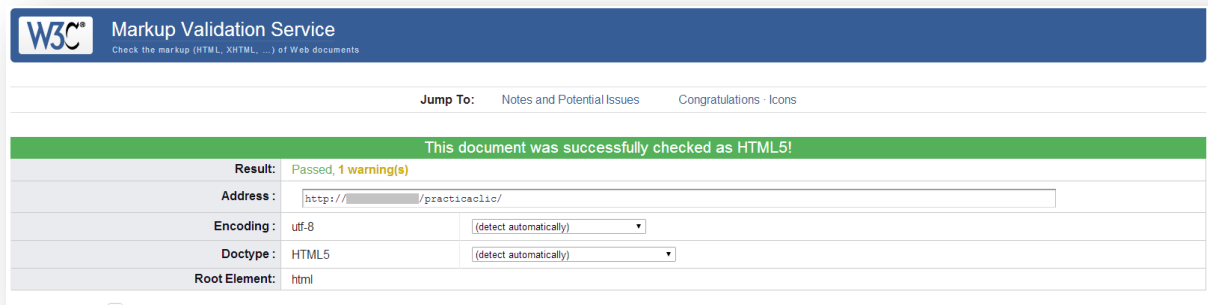
✖ Line 28, Column 77: Bad value for attribute action on element form: Must be non-empty.

```
<form class="login" action="" name="frmMestre" id="frmMestre" method="post" >
```

Syntax of IRI reference:

Any URL. For example: /hello, #canvas, Or <http://example.org/>. Characters should be represented in [NFC](#) and spaces should be escaped as %20.

Un cop corregits els errors, tornem a provar de validar el document. Si ho hem fet bé, ens ho indicarà tal com es mostra en la imatge següent :

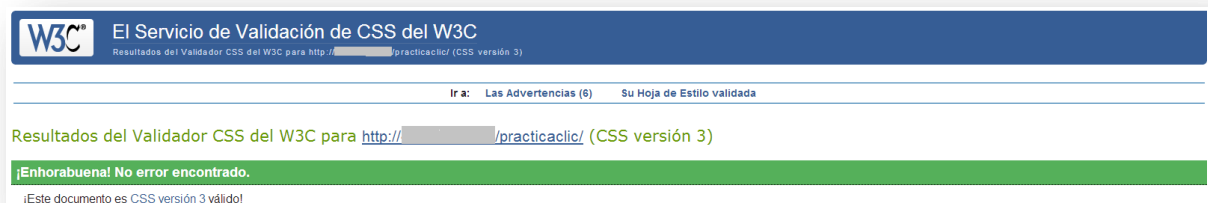


Per validar els arxius d'estils, també existeix una eina online que funciona igual que l'anterior però per documents d'estils (<http://jigsaw.w3.org/css-validator/>).

A continuació es pot veure una imatge on hi ha un exemple que ens indica que hi ha un error en el full d'estils.



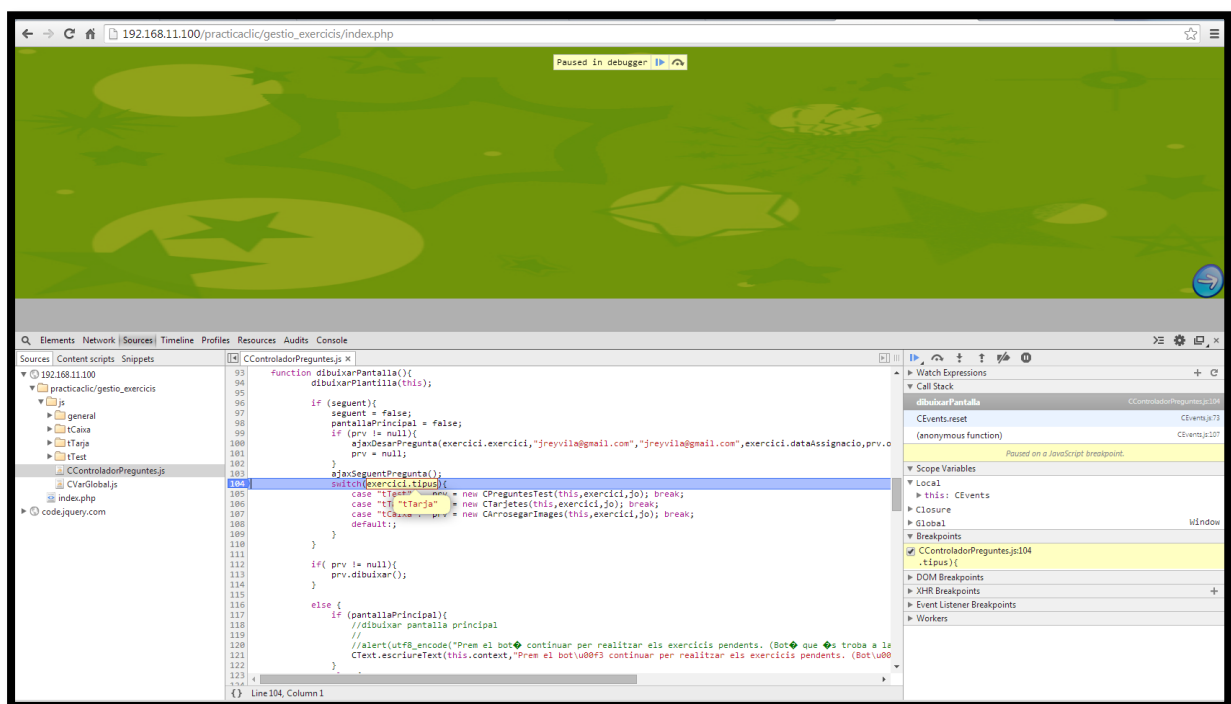
Un cop corregit es torna a provar i ens indica que l'hem corregit adequadament.



L'eina de depuració que s'ha utilitzat és la que porta incorporada el navegador **Chrome**.

Des de el propi navegador hi podem accedir per mitjà del menú “eines” i seleccionar “consola JavaScript”, des d'on podrem veure els arxius fonts, parar l'execució (breakpoint) i veure els valors de les variables.

Quan s'executa una part de codi on hi ha un error, ens el marca en vermell indicant l'error.



12. Funcionament de l'aplicació

Com ja he comentat amb anterioritat, el projecte consta d'una plataforma web on hi tenen accés tant els tutors/mestres com els alumnes. Un tutor/mestre quan accedeix a la plataforma web donarà d'alta els seus alumnes, i a partir d'aquí podrà seleccionar una sèrie d'exercicis, els quals quedaran assignats a tots els seus alumnes.

Quan sigui un alumne qui accedeixi a la plataforma web, podrà realitzar els exercicis que el seu tutor/mestre li hagi assignat.

Anem a veure tot això amb una mica més de detall.

12.1. Funcionament aplicació web.

Quan un usuari arriba a la pantalla inicial pot ser que encara no estigui donat d'alta en el sistema, en aquest cas pot decidir si enregistrar-se com a mestre o evidentment no fer-ho i marxar.

Quan l'usuari s'enregistra rep un correu electrònic amb la contrasenya d'accés.

A continuació es pot veure la pantalla d'inici de sessió.

The screenshot shows the login page of the PracticaClic application. The page has a green header with the title "PracticaClic". Below the header, there are three main sections: "Et vols donar d'alta?" (Sign up), "Alumne" (Student), and "Mestre" (Teacher). Each section contains input fields for email and password, and a "Petició" or "Inici" button. There are also links for "Contrasenya oblidada?" (Forgot password?). The footer shows "Copyright (c) 2014 PracticaClic".

Si un tutor ha oblidat la seva contrasenya la podrà recuperar, fent un clic en el vincle "contrasenya oblidada" i haurà d'introduir el seu correu electrònic, per tal de rebre la nova contrasenya a la seva bústia de correu personal.

Quan un tutor està enregistrat al sistema i s'identifica en la web, d'entrada veurà la següent pantalla.

The screenshot shows the dashboard of the PracticaClic application. The page has a green header with the title "PracticaClic" and the email "correu@inventat.com" and a "Logout" link. Below the header, there is a sidebar with a menu containing: "Assignar-se alumne", "Estadístiques exercicis.", "Assignar feina als alumnes", and "Substitució de contrasenya.". The main content area is empty. The footer shows "Copyright (c) 2014 PracticaClic".

Si és la primera vegada que entra en el sistema, la primera cosa que ha de fer és associar-se els alumnes per posteriorment assignar-los-hi els exercicis.

En la següent pantalla es pot veure el formulari que ha d'introduir un mestre per assignar-se un nou alumne.

The screenshot displays the PracticaClic web application interface. At the top, a green header bar contains the title "PracticaClic" and the email "correu@inventat.com" with a "Sortir" link. Below the header, a light green sidebar on the left lists four menu items: "Assignar-se alumne", "Estadístiques exercicis.", "Assignar feina als alumnes", and "Substitució de contrasenya.". The main content area is titled "Gestió Alumnes" and features a form with five input fields: "Usuari alumne:", "Contrasenya:", "Nom:", "Cognom:", and "Segon cognom:". Above the form, there are four buttons: "Guardar", "Esborrar", "Netejar", and "Ajuda". At the bottom of the page, a green footer bar contains the text "Copyright (c) 2014 PracticaClic."

Ara el tutor/mestre ja podrà assignar exercicis als seus alumnes.

Per fer-ho ho farà en l'apartat "Assignar feina als alumnes".

En aquesta pantalla es poden veure els exercicis disponibles en l'aplicació, i quan el mestre vulgui assignar els exercicis, únicament haurà de marcar el checks dels exercicis triats i prémer el botó de guardar. (El sistema ens avisarà de que els està desant i quan finalitzi, també ens ho indicarà).

The screenshot shows the 'PracticaClic' web application interface. The header is green with the title 'PracticaClic' and the email 'email@inventat.com'. On the left, there is a sidebar with links: 'Assignar-se alumne', 'Estadístiques exercicis.', 'Assignar feina als alumnes', and 'Substitució de contrasenya.'. The main content area has a title 'Assignar feina aula' and buttons for 'Guardar', 'Netejar', and 'Ajuda'. Below the title, there is a search bar with a dropdown menu set to 'tots' and a 'Cerca' button. The main table lists exercises with columns: 'exercici', 'Grup', 'assignatura', and 'pregunta'. The table contains 10 rows of exercises, all from 'Matemàtiques' and 'Volum 1'. The first two exercises, 'MAT1' and 'MAT10', have their checkboxes selected. Each row also has a small icon in the last column. At the bottom, there is a copyright notice: 'Copyright (c) 2014 PracticaClic.'.

exercici	Grup	assignatura	pregunta
<input checked="" type="checkbox"/> MAT1	Volum 1	Matemàtiques	Calcula la suma: $3 + 2$
<input checked="" type="checkbox"/> MAT10	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 0?
<input checked="" type="checkbox"/> MAT100	Volum 1	Matemàtiques	Arossega 8 dofins dins el requadre.
<input type="checkbox"/> MAT11	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 1?
<input type="checkbox"/> MAT12	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 2?
<input type="checkbox"/> MAT13	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 3?
<input type="checkbox"/> MAT14	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 4?
<input type="checkbox"/> MAT15	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 5?
<input type="checkbox"/> MAT16	Volum 1	Matemàtiques	Selecciona el nombre que va DARRERA del 6?

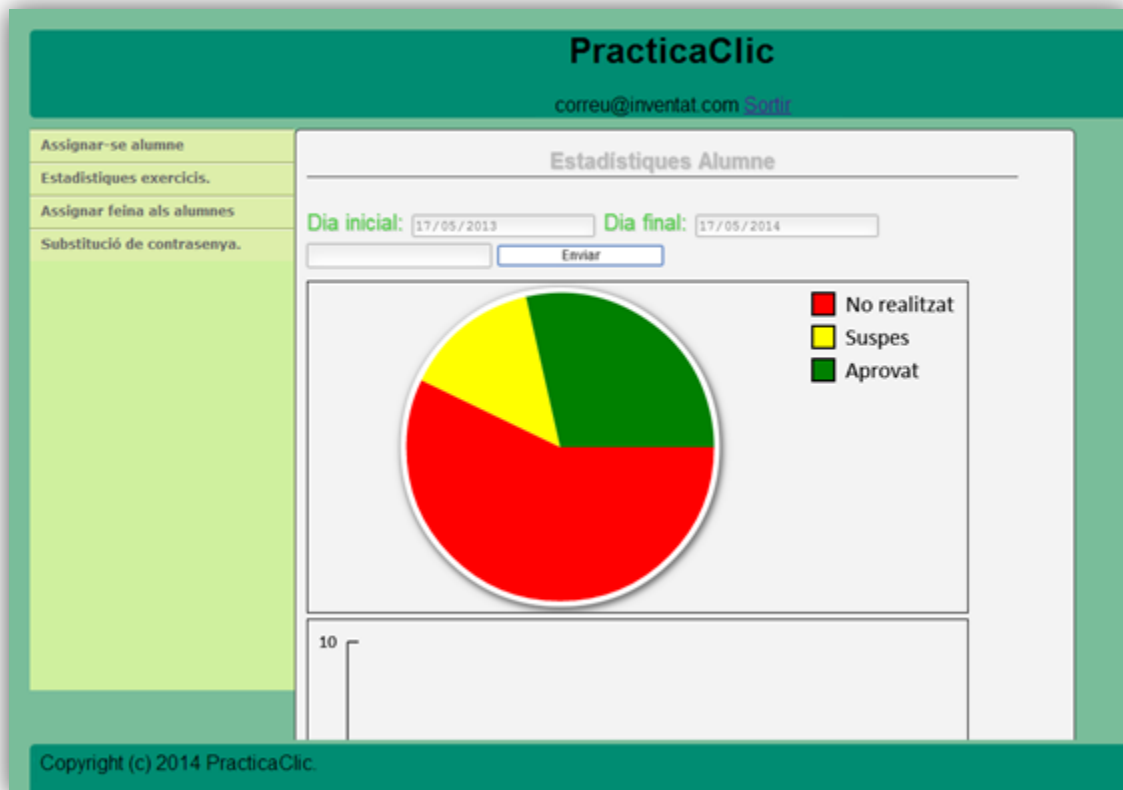
Un cop els alumnes hagin realitzat o no els exercicis, el mestre/tutor podrà veure les estadístiques per saber si un alumne ha realitzat els exercicis, si els ha aprovat o bé si els ha suspès.

De la mateixa manera que podrà veure les estadístiques d'un alumne, també podrà veure el global de tots els alumnes que el mestre tingui assignats.

El funcionament és simple.

- Si es vol saber el resultat d'un alumne concret, cal indicar el nom d'usuari d'alumne i indicar quin dia o dies se li van assignar els exercicis.
- Si es vol veure el resultat de tots els alumnes, només haurà d'indicar la data d'inici i data fi del dia - dies que se'ls hi van assignar els exercicis.

A continuació es pot veure en el següent gràfic :

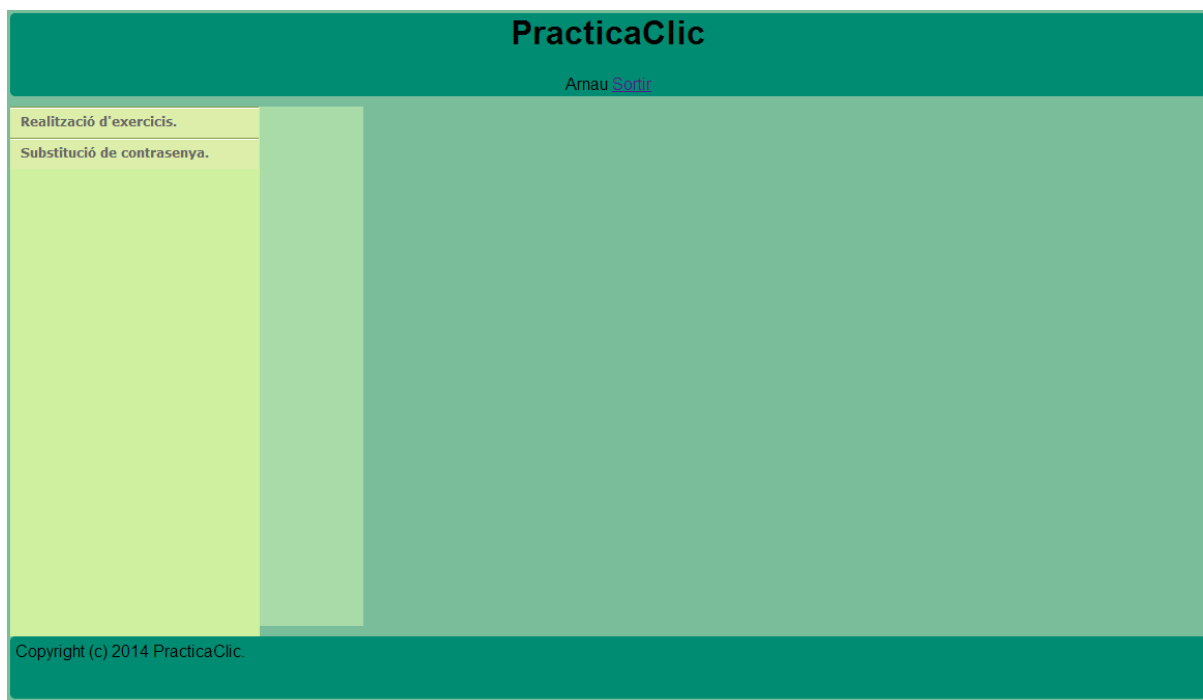


I per acabar, la última opció que pot realitzar un mestre/tutor, és la de modificar la seva contrasenya.

The screenshot shows the 'Canvi contrasenya' (Change Password) page in the PracticaClic application. The page has a green header with the title 'PracticaClic' and the email 'correu@inventat.com' with a 'Sortir' (Exit) link. On the left, there is a sidebar menu with options: 'Assignar-se alumne', 'Estadístiques exercicis.', 'Assignar feina als alumnes', and 'Substitució de contrasenya.'. The main content area displays a form for changing the password. At the top of the form, there are three buttons: 'Guardar' (Save), 'Netejar' (Clear), and 'Ajuda' (Help). Below these buttons, the form has three input fields: 'Contrasenya actual:' (Current password), 'Nova contrasenya:' (New password), and 'Repetir nova contrasenya:' (Repeat new password). The footer of the page states 'Copyright (c) 2014 PracticaClic.'.

COM UN ALUMNE POT REALITZAR ELS EXERCICIS DES DE L'ENTORN WEB.

Quan un alumne vulgui entrar al sistema, veurà la següent pantalla :



Quan vulgui realitzar els exercicis que li ha proposat el seu tutor/mestre, cal que seleccioni l'opció "Realització d'exercicis", i tot seguit li apareixerà la següent pantalla.



Al prémer el botó de següent (fletxa), li apareixerà un exercici pendent de realitzar.

Si l'alumne va deixant pendents els exercicis i a mesura que passen els dies el mestre/tutor va assignant més exercicis, l'ordre que li apareixeran els exercicis és de més antic a més actual.

A continuació hi ha diferents il·lustracions dels possibles d'exercicis.

Les següents imatges corresponen al tipus d'**exercicis "tipus test"**, on hi ha una pregunta, i l'alumne ha de seleccionar una o vàries respostes. Després per confirmar la resposta ha de fer clic al botó "següent" (fletxa).

En la següent imatge és pot veure com a l'alumne se li demana que faci la suma de $3 + 2$, i l'alumne ha triat com a opció el número 5.



En el següent exercici l'alumne, ha de triar el valor que va després del 0, i en aquest cas, es pot veure que l'alumne ha seleccionat el valor 1.



Un altre exemple d'exercici del tipus test que es pot veure a continuació : A l'alumne se li demana quin és el valor que va darrera del 5, i ens selecciona l'opció del número 6.



En el següent exercici a l'alumne se li demana seleccionar els nombres més grans que 7 i veiem que n'ha seleccionat 3, un d'incorrecte i dos de correctes.

Nota : En els exercicis de tipus test, hi pot haver preguntes on només es pugui seleccionar una opció o com en el següent cas que es puguin seleccionar varies possibles opcions.



Les preguntes de tipus “Test”, poden ser molt diverses.

Des dels exemples que hem vist, fins a tenir exercicis de restes, multiplicacions, etc.

En definitiva tots els que se'ns puguin acudir.

A continuació es poden veure alguns **exercicis** del tipus “**Tarja**”. Els exercicis de tipus “tarja”, hi apareixen diverses targetes o requadres, les quals contenen diferents grups de dibuixos, i l’alumne n’haurà de seleccionar els que cregui que són els correctes, en funció de la pregunta que se li formulí.

En el següent exercici, per exemple, a l’alumne se li pregunta Quins targetes tenen 8 animals? I veiem que l’alumne, ha seleccionat dues “targetes”.



En aquest cas, la pregunta formulada és Quines targetes tenen 6 animals? I l’alumne en selecciona també dues.



Els exercicis de tipus “Tarja” no tenen perquè ser tots de comptar animals. També es podrien fer preguntes del tipus : Quina de les següents imatges sumen 5?, i tenir diverses imatges amb sumes o restes, és a dir, tot depèn de la nostra imaginació.

A continuació es poden veure **exercicis** del tipus “**caixa**”. Aquest tipus d'exercicis consisteixen en arrossegar els objectes correctes dins del requadre blau.

En la següent imatge es pot veure com encara l'alumne no ha arrossegat cap conill dins del requadre.



En la següent imatge, l'alumne ja ha arrossegat els 6 conills cap a dins del requadre.



Un altre exercici on ja ha arrossegat 5 gats dins del requadre.



En la següent imatge es pot veure un altre exercici, on l'alumne ha arrossegat 8 pingüins dins el requadre i només li en manca un per tal que l'exercici estigui completat de manera correcta.



Quan l'alumne ja no té més exercicis assignats, li apareix el següent missatge per pantalla.



Un alumne també pot modificar la seva contrasenya al igual que ho pot fer un mestre/tutor. Però en aquest cas si l'alumne s'oblida la contrasenya, serà el tutor qui li modifiqui.

12.2. Funcionament aplicació mòbil.

L'aplicació mòbil està destinada a l'alumne. El qual podrà o haurà de realitzar els exercicis que el seu tutor/mestre li hagi assignat.

Inicialment, l'aplicació demana : nom d'usuari i una contrasenya per tal d'iniciar la sessió.



Un cop l'alumne s'hagi identificat, o li hagi fet el seu pare/mestre/tutor, l'estructura de la pantalla inicial consta d'una capçalera i la pantalla principal on es realitzen els exercicis.

Com es pot veure en la següent imatge la pantalla principal es idèntica a la de l'entorn web, i seu funcionament és exactament igual.



L'alumne, al prémer el botó li apareixerà el primer exercici pendent de realitzar. És a partir d'aquest moment que ja podrà començar a resoldre l'exercici o exercicis.

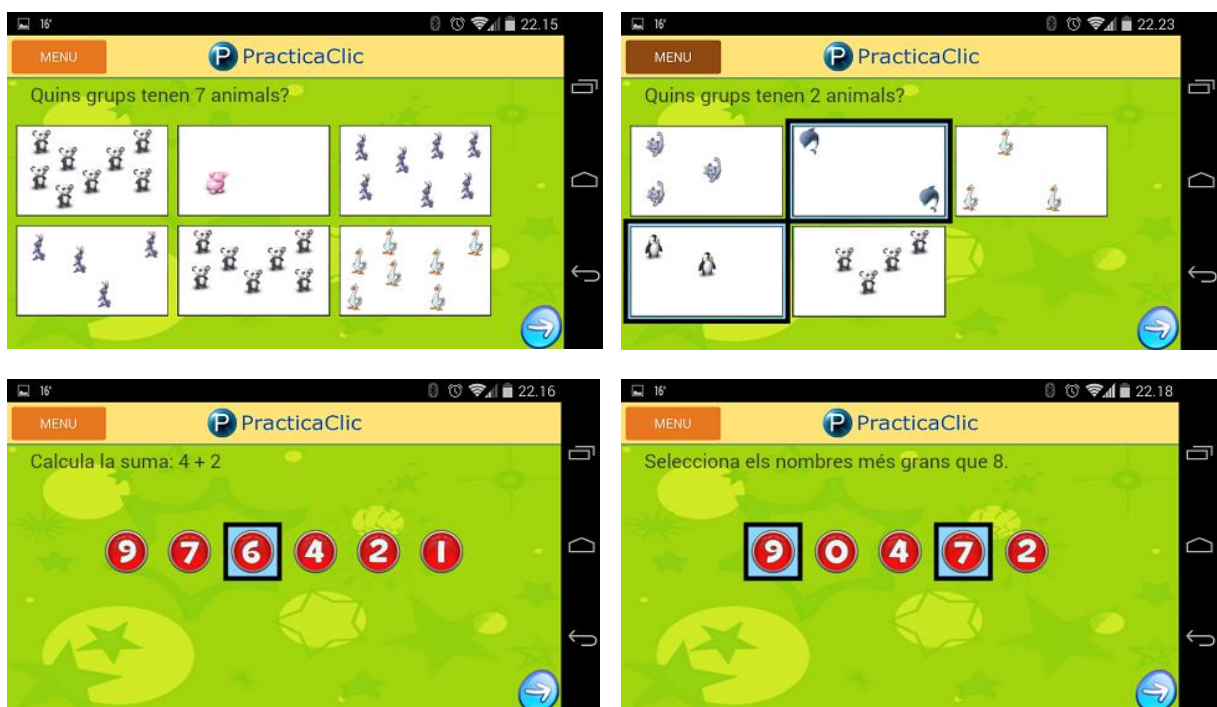
Quan hagi finalitzat l'exercici, haurà de prémer el botó següent (fletxa).

En la següent imatge es pot veure com l'alumne havia de resoldre la suma $3 + 2$. Com a solució, ha indicat el valor 5 i posteriorment ha clicat damunt del botó següent. (Veiem un cas on a l'alumne ja no li quedaven més exercicis per a realitzar, per això al clicar damunt del botó següent, li ha aparegut el missatge de : “No hi ha exercicis pendents”.



Com es pot veure per la realització dels exercicis en l'aplicació mòbil és idèntica que en l'entorn web.

Tot seguit, podem veure algunes imatges de diferents exercicis realitzats en l'aplicació mòbil.





DETALL A TENIR EN COMPTE :

En la capçalera, hi ha un botó que dona accés a un menú amb 3 opcions.

MENÚ

- Fer exercicis. Continuar realitzant els exercicis pendents.
- Ajuda. Conté una explicació del funcionament de l'aplicació
- Sortir. Tanca l'aplicació.

En la següent imatge es pot veure el menú desplegat.



13. Conclusions

13.1. Assoliment d'objectius

13.2. Variacions del producte final respecte el disseny inicial previst

13.3. Valoració personal

13.4. Futures millores

Afegir diferents tipus d'exercicis.

Oportunitat de millora, és que l'aplicació mòbil pugui enregistrar l'inici de sessió.

Entorn per afegir nous exercicis.

14. Fonts d'informació

14.1. Bibliografia

14.2. Formació Online

14.3. Consultes Android

14.4. Consultes PHP

14.5. Consultes generals