

Escola Politècnica Superior  
Universitat de Girona

# Aplicatiu per la gestió de centres educatius

TREBALL FINAL DE GRAU  
GEINF

*Document:* Memòria  
*Autor:* Pau Recacha Borell  
*Convocatòria:* Setembre 2020

# Index

1. Introducció .....	4
1.1.- Origen de la idea.....	4
1.2.- Objectius del projecte.....	4
1.3.- Necessitats des del punt de vista de cada usuari .....	5
2. Estudi de viabilitat .....	7
3. Metodologia .....	9
4. Planificació .....	11
5. Marc de treball i conceptes previs .....	13
6. Requisits del sistema.....	16
6.1.- Requisits funcionals .....	16
6.2.- Requisits no funcionals .....	19
6.2.1.- Requisits d'eficiència .....	19
6.2.2.- Requisits d'ús.....	19
6.2.3.- Requisits de suport.....	19
7. Anàlisi i disseny del sistema.....	20
7.1.- Anàlisi del sistema.....	20
7.1.1.- Actors.....	20
7.1.2.- Diagrames de casos d'ús .....	22
7.1.3.- Fitxes de casos d'ús .....	25
7.2.- Model de dades.....	34
7.3.- Disseny de les interfícies d'usuari de la app .....	42
7.3.1.- Interfícies d'alumne i professor .....	42
7.3.2.- Interfícies d'alumne.....	45
7.3.3.- Interfícies de professor .....	47
8. Estudi i decisions .....	51
9. Implementació i proves .....	56
9.1.- WAMPServer.....	56
9.1.1.- Configuració del servidor Apache .....	57
9.1.2.- Configuració del servei MySQL .....	59
9.2.- HeidiSQL .....	60
9.2.1.- Creació d'una nova base de dades.....	60
9.3.- Postman.....	62
9.4.- Volley .....	64
9.4.2.- Proves .....	65
9.5.- Visual Studio i ASP.NET MVC en C# .....	66

9.5.1.- Proves .....	68
9.6.- 000webhost .....	70
9.6.1.- Obtenir un domini a través de 000webhost .....	70
9.6.2.- Exportar l'aplicació al servei .....	70
9.6.3.- Exportar la base de dades al servidor .....	71
9.7.- Generació de vistes i rutines .....	73
10. Implantació i resultats.....	78
10.1.- Procés de desenvolupament.....	78
10.2.- Resultats obtinguts .....	78
10.2.1.- Aplicació Android .....	78
10.2.2.- Imatges dels alumnes/professors.....	92
10.2.3.- Aplicació ASP.NET .....	92
11. Conclusions .....	96
11.1.- Planificació original i desviacions.....	96
11.2.- Aprenentatge .....	96
11.3.- Conclusions personals .....	97
12. Treball futur.....	98
13. Bibliografia.....	99

# 1. Introducció

## 1.1.- Origen de la idea

Des de sempre m'ha agradat l'entorn del desenvolupament d'aplicacions per mòbil i ja des de 3er d'ESO i 4rt d'ESO sabia que voldria estudiar i dedicar-me. Per això quan va ser el moment de pensar un projecte per el treball de recerca a 2on de Batxillerat no m'ho vaig pensar ni dues vegades: una aplicació per a dispositius mòbils que tingués algun tipus de relació amb l'institut, útil pels alumnes i professors i que actues a mode d'eina acadèmica per a facilitar la feina a les dues parts.

El problema va ser que jo en aquell moment tenia molts pocs coneixements d'informàtica (alguns de pàgines web) i tot aquest món era desconegut per mi.

Després de setmanes aprenent de tutorials a la web i fent proves amb l'entorn Eclipse, vaig elaborar una aplicació Android senzilla de dos funcions que mitjançant fitxers PHP d'intermediari entre la base de dades i l'aplicatiu permetia: guardar cadenes de caràcters en registres d'una taula i obtenir un llistat de totes les entrades guardades en la taula.

La aplicació resultant era molt petita i no feia ni un petit percentatge del que al començar el treball tenia en ment, per la qual cosa sempre he volgut recuperar aquell projecte i ampliar-lo d'alguna manera per a que s'assembla a la idea que en un principi volia desenvolupar.

Per tant, el meu propòsit en aquest Treball Final de Grau, és el de programar un aplicatiu que implementi totes les funcionalitats de la idea que tenia fa uns anys, aplicant els coneixement que he anat adquirint al llarg de la carrera d'informàtica per tal d'obtenir un producte final més elaborat que l'anterior.

Aquest treball està pensat per portar la gestió i control d'un centre escolar tipus ensenyament primari o secundari. En aquest tipus d'aplicacions intervenen tres tipus d'usuaris: els alumnes, els mestres/professors i l'administrador del sistema que sol ser un professor del centre amb més coneixements informàtics que la resta.

## 1.2.- Objectius del projecte

L'objectiu d'aquest projecte és el desenvolupar l'anàlisi, el disseny i la implementació d'un aplicatiu multiplataforma per portar el control diari d'un centre educatiu.

## 1.3.- Necessitats des del punt de vista de cada usuari

Les necessitats de l'aplicació són semblant a les del sector de gestió de centres educatius i dins d'ella podem identificar tres tipus d'usuaris: l'administrador del sistema encarregat de gestionar tant la base de dades com l'aplicatiu per a possibles errors i queixes en l'aplicació, l'alumne que està identificat i pot accedir a la part de la app creada només per ell i el professor que semblant a l'alumne pot accedir només a la part de la app creada per ell.

Necessitats de l'administrador:

- Iniciar sessió en la base de dades.
- Gestionar alumnes.
- Gestionar professors.
- Gestionar notícies.
- Gestionar incidències.
- Gestionar assistències.
- Gestionar classes.
- Gestionar preguntes del fòrum.
- Gestionar respostes del fòrum.

Necessitats de l'alumne:

- Iniciar sessió.
- Tancar sessió.
- Veure les assignatures que cursa en l'any actual.
- Accedir a una assignatura que cursa actualment.
- Consultar notícies de assignatures
- Enviar mails.
- Consultar notes de tasques realitzades.
- Penjar preguntes al fòrum.
- Contestar preguntes al fòrum.

Necessitats del professor:

- Iniciar sessió.
- Tancar sessió.
- Consultar assignatures que imparteix.
- Afegir notícies per una assignatura que imparteix.
- Gestionar la classe que imparteix en cert dia durant una certa hora.
- Enviar mails.
- Afegir tasques.
- Puntuar tasques.
- Veure notes de tasques puntuades.
- Penjar preguntes al fòrum.
- Contestar preguntes al fòrum.

## 2. Estudi de viabilitat

Després d'unes hores de recerca he trobat alguns exemples semblants al projecte que tinc en ment.

Podem dir que tecnològicament es viable desenvolupar la idea actual del treball per a dispositius mòbils ja que es tractarà d'un software semblant a moltes aplicacions ja existents en el mercat.

Alguns dels exemples trobats són:

- **Attendance:** és una app que a partir de dades introduïdes per l'usuari, crea una base de dades i permet guardar alumnes, grups d'alumnes, classes, imatges dels alumnes, ... tot això per crear una interfície adient per a que un professor pugui passar llista a classe. A més ofereix la possibilitat de afegir notes sobre com a anat la classe sobre un alumne o certes activitats que s'han realitzat. És semblant a la idea del sistema d'incidències que es vol implementar en aquest projecte.

Url de la app: <https://play.google.com/store/apps/details?id=com.aor.attendance>

- **iDoceo:** és una aplicació de gestió de activitats pels professors. Bàsicament és com un assistent pels professors. A més, permet crear fòrums de notícies a compartir amb els alumnes. Desenvolupa una idea que també té en ment aquest projecte que és la de notificació de notícies als alumnes per part dels professors així com la implementació d'un fòrum per afavorir la comunicació entre la classe i el professorat. Ofereix també varies funcionalitats com per exemple la de un bloc de alumnes on apuntar notes, un control de assistència i incidències, una agenda, ... (Només per ipad).

Url de la app: <https://www.idoceo.es/index.php/es/>

- **Additio:** Additio és un programa molt popular en l'àmbit educatiu. Ofereix més de 150 funcionalitats entre elles les de gestió de classes com notes i assistències, pujar tasques a fer i les seves possibles solucions, els alumnes i les seves famílies poden accedir al quadern del professor on es veuen les notes de cada un d'ells, autoavaluacions amb rubriques, ... Es troba per a dispositius mòbils però també per a ordinadors.

Url de la app: <https://www.additioapp.com/es/>

- **TeacherKit:** TeacherKit és una app dedicada a la gestió del alumnat per als professors. Permet posar assistències a classe, incidències i notes a més d'oferir un sistema d'estadístiques per veure el rendiment de cada alumne al llarg del curs. Només es troba per dispositius iOS.

Url de la app: <https://apps.apple.com/es/app/teacherkit/id389584618>

- **Edmodo:** Edmodo és una aplicació que ofereix un servei de xarxa social semblant a Facebook però enfocat al ús acadèmic fomentant la comunicació entre alumnes i professors. No només ofereix serveis de comunicació si no que permet pujar documents, penjar tasques i les seves possibles solucions, crear exàmens a fer, ...). A més, els pares del alumnat poden entrar per veure les notes dels seus fills així com veure quin contingut estan aprenent.

Url de la app: <https://new.edmodo.com/?go2url=%2Fhome>

Econòmicament el projecte és viable ja que els costos del seu desenvolupament són pràcticament inexistents. En un principi la màquina encarregada de fer de servidor serà un ordinador personal i utilitzo varis mòbils per fer proves. La base de dades així com els fitxers PHP s'han pujat a un servei de hosting gratuït i que no afegeix cost al producte final. Les eines utilitzades per el desenvolupament de la aplicació són de software lliure i no suposen cap cost extra.

Si es considerés ampliar la base de dades per fer-la d'un tamany semblant a un cas real (amb centenars d'alumnes, les seves respectives notes, assignatures, classes, ...) probablement el cost augmentaria (tot i així els serveis de hosting són relativament econòmics en algunes plataformes).

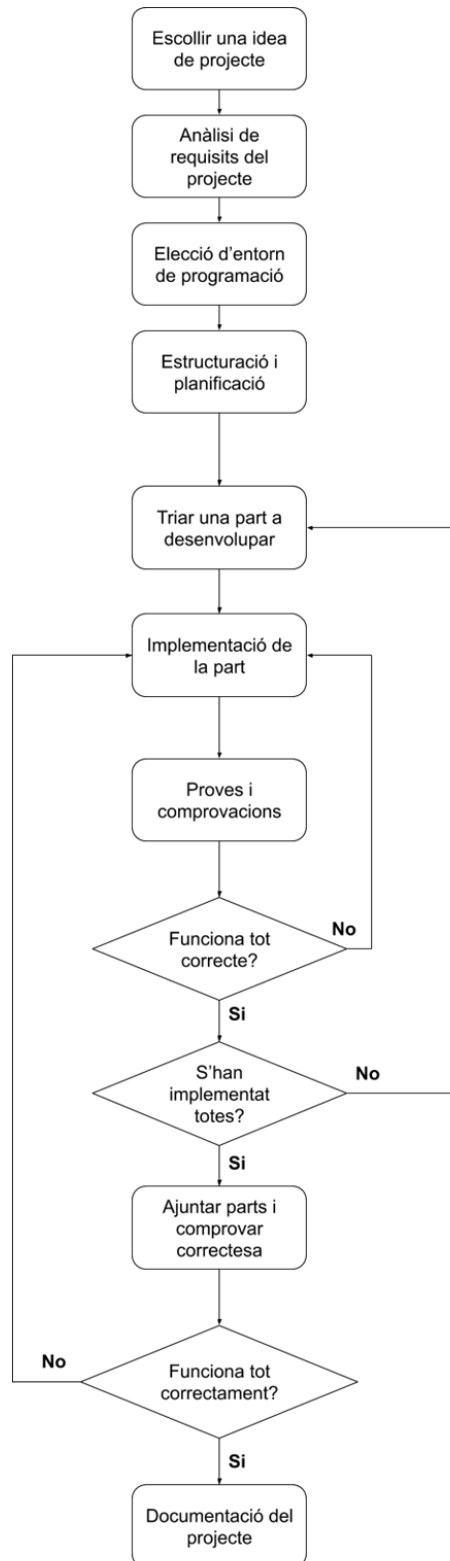
## 3. Metodologia

Pel desenvolupament d'aquest treball, al ser projecte petit, no s'ha seguit cap metodologia en concret, però s'han seguit les passes habituals del desenvolupament d'un projecte informàtic.

Els passos que s'han seguit han estat:

1. Triar la idea a desenvolupar: primer de tot es tria una idea o concepte d'aplicatiu i s'estudia per veure si es viable i adient per fer com a Treball Final de Grau.
2. Anàlisi de requisits del projecte: amb la idea de projecte en ment, es realitza un anàlisi de requisits per veure quines necessitats té l'aplicació i quines funcionalitats principals són importants d'implementar.
3. Es tria l'entorn de programació: una vegada decidit la idea i amb els requisits clars, es tria l'entorn en que es programarà l'aplicatiu més adequat per desenvolupat el projecte.
4. S'estructura i es planifiquen les parts del projecte: és important saber, abans de posar-se a programar, les parts a implementar en el projecte. El projecte s'estructura en diferents parts, cada una més o menys complexa i important d'on cada una cal fer-ne un anàlisi i disseny.
5. Implementació: arribar a aquest apartat, es comença a desenvolupar el projecte, apartat per apartat segons s'ha separat en el punt número 4.
6. Proves i comprovacions: una vegada s'acaba el punt 5, es necessari comprovar que la implementació ha sigut correcte i que tot funciona amb normalitat. També cal comprovar que les implementacions compleixen les especificacions i requisits prèviament triats. Si hi ha algun error o algun requisit no es compleix, es torna a implementar (és a dir, es torna a l'apartat 5).
7. Una vegada implementades i comprovades totes les seccions, és fa una prova conjunta de les parts per verificar que totes juntes funcionen correctament. Igual que amb el apartat 6, si alguna part no compleix amb el que s'esperava d'ella, es torna al punt 5.
8. Documentació.

Per tal d'entendre més a fons la metodologia seguida, un esquema a continuació:



Esquema 3.1: Procés de desenvolupament

## 4. Planificació

El projecte es va començar a finals del principi de maig de 2019, i vaig aprofitar que un membre de la meva família és professor d'institut per preguntar quines funcionalitats considera ella importants per un aplicatiu com el que desenvolupo en aquest projecte. Amb les seves idees i les meves, vaig fer-me un llistat de requisits del sistema.

Una vegada vaig tenir clar quines funcionalitats havia d'implementar en l'aplicació, es varen definir els casos d'ús per a cada usuari així com els seus corresponent diagrames per facilitar-me la feina en un futur.

Després d'aquest procés, vaig dissenyar el model de dades necessari per a que l'aplicació pogués fer totes les funcionalitats que volia afegir-li en el futur. Amb el disseny creat, vaig procedir a crear-la i afegir dades per a poder realitzar proves més endavant.

Amb la base de dades ja creada, vaig configurar l'ordinador per a que actués de host amb l'aplicació i vaig crear i pujar al programa de hosting, els fitxers PHP per a les diferents consultes que necessitaria en un futur en l'aplicació. Per cada fitxer creat, utilitzava la eina POSTMAN per comprovar que els fitxers funcionessin correctament.

Amb tot els elements del host creats, només faltava centrar-se en l'aplicació. Per desenvolupar la app em centrava primer en crear un esbós de cada pantalla com seria i com estaria estructurada per a que fos el més clara i entenedora possible. Una vegada creades, vaig consultar amb el professor de la família que li semblava i després d'uns canvis varen quedar preparades per ser implementades. Amb una idea de com estarien col·locats els elements en l'aplicació, vaig escriure cada fitxa de cas d'ús basant-me en el disseny de la aplicació i els diagrames fets en el passat.

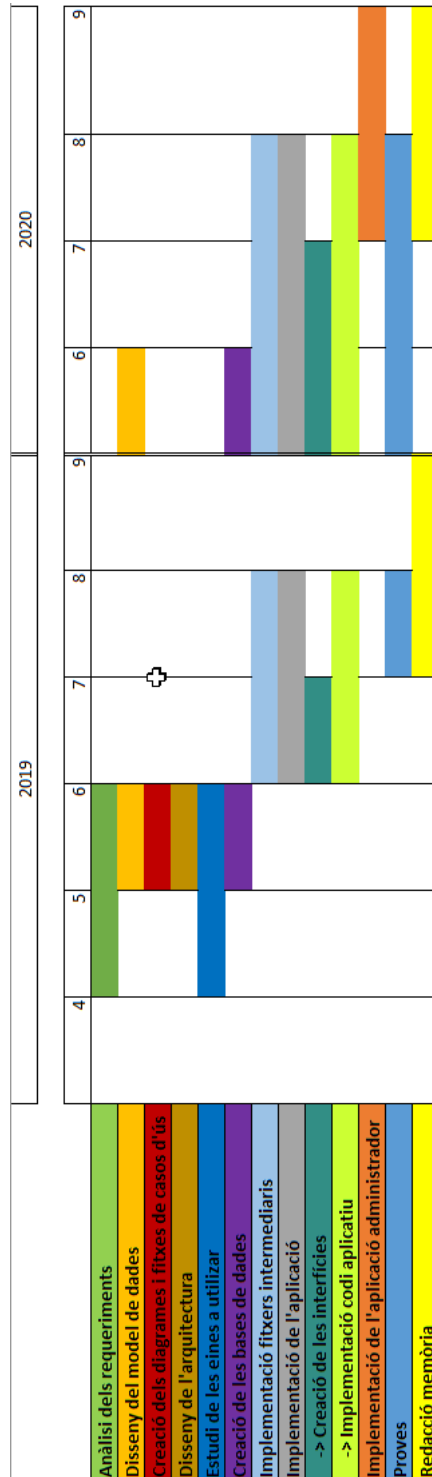
Una vegada tot va quedar clar, primer es varen implementar les interfícies dels alumnes i quan es va comprovar que tot funcionava, es varen seguir amb les dels professors.

Un cop acabades, es va ajuntar tot i es va comprovar que funcionessin les interaccions entre professor i alumne i es varen solucionar problemes que sorgissin. També es varen tenir presents tasques que no havia implementat per falta de temps (i que apareixen en full de projecte) i que quedarien per desenvolupar en un futur (així com

possibles millores).

La memòria s'ha redactada tota al final del projecte quan la aplicació s'ha considerat acabada. En el seu procés s'ha intentat documentar com va ser tot el desenvolupament del projecte.

A continuació es mostra el diagrama de Gantt amb la planificació temporal del desenvolupament del projecte:



Imatge 4.1: Diagrama de Gantt del projecte

## 5. Marc de treball i conceptes previs

Abans de posar-me a dissenyar i desenvolupar el projecte vaig estar uns dies pensant en quines necessitats tindria un alumne d'institut en una aplicació que busqués simplificar-li la feina en els seus dies d'estudi. Com que també hi intervindrien professors en el sistema, amb l'ajuda del professor de la família vaig recollir informació prèvia a les necessitats que tindrien aquest tipus d'usuaris en la aplicació.

Amb la informació que podria utilitzar per desenvolupar els requisits del sistema, vaig pensar en fer un estudi previ sobre quines eines em permetrien implementar el projecte i en quines diferents formes podria fer-ho.

El primer punt era la necessitat d'un host encarregat d'emmagatzemar la base de dades així com la de gestionar les diferents consultes entrants dels clients. Per solucionar aquest apartat, podria utilitzar programes gratuïts que es troben en la web, que ofereixen diferents serveis per a què un ordinador qualsevol actui com a host. Alguns exemples trobat eren XAMPP o WAMPServer.



Imatge 5.1: WAMPServer i serveis que ofereix



Imatge 5.2: XAMPP i els serveis que ofereix

Tot i que WAMPServer o XAMPP ja ofereixen un client phpMyAdmin per configurar la base de dades, també es poden configurar per a que el servei gestor de base de dades corri en un programa diferent (en un port diferent). Per tal d'ampliar el nombre de programes així com l'experiència guanyada en altres àmbits, em vaig decantar per la segona opció per també ampliar els coneixements en altres eines del camp.



Imatge 5.3: Logo MariaDB



Imatge 5.4: Logo MySQL



Imatge 5.5: Logo PostgreSQL

## Aplicatiu per la gestió de centres educatius

A part de la base de dades, les aplicacions necessitaran d'algun llenguatge intermediari que s'encarregui de processar consultes enviades per les aplicacions i respondre-les amb un llenguatge que elles entenguin. El cas de PHP era el més adient i aquí sí que no tenia cap dubte en que la eina a utilitzar que més familiar tenia era el Notepad++. A més, necessitaria d'una eina que em permetés fer crides GET i POST per comprovar que els fitxers PHP fessin la seva feina. Des de sempre he treballat amb POSTMAN i tampoc vaig dubtar en aquest cas.



Imatge 5.6: Logo Notepad++



POSTMAN

Imatge 5.7: Logo POSTMAN

Per dur a terme el desenvolupament de l'aplicació, hi han molts entorn de desenvolupament que permet programar amb Android. Tots i cada un d'ells ofereixen característiques interessants i d'altres semblants. En aquest cas dubtava sobre dos entorn en utilitzar, Android Studio i IntelliJ Idea.



Imatge 5.8: Logo Android Studio



Imatge 5.9: Logo IntelliJ Idea

Per facilitar la gestió de la base de dades, existeixen varies tipologies d'aplicacions que faciliten el control d'una forma fàcil i sense necessitats de coneixements informàtics elevats. En aquest àmbit no hi he treballat mai i per recomanacions m'he decidit per ASP.NET MVC en C#:



Imatge 5.10: Logo ASP.NET MVC

## Aplicatiu per la gestió de centres educatius

Finalment i si tenia temps, pensava en pujar la base de dades i els fitxers PHP en un pàgina web de hosting per provar la aplicació des de qualsevol lloc. Hi han moltes opcions que vaig trobar, algunes d'elles són:



**000webhost**

Imatge 5.11: Logo 000webhost



**HOSTINGER**

Imatge 5.12: Logo HOSTINGER



Imatge 5.13: Logo FreeHosting.com

## 6. Requisits del sistema

### 6.1.- Requisits funcionals

Els requisits funcionals defineixen les funcions del sistema de l'aplicació o dels seus components sense descriure com han de ser implementats.

S'han classificat en apartats per tal de facilitar-ne la comprensió:

#### Requisits funcionals en quant a la gestió d'usuaris

# Requisit	Descripció
1	L'aplicació ha de permetre als usuaris donats d'alta en l'aplicatiu autenticar-se
2	L'aplicació ha de permetre als usuaris que s'han autenticat, tancar la sessió.
3	El sistema ha de permetre al administrador donar d'alta a nous usuaris en l'aplicatiu.
4	El sistema ha de permetre al administrador donar de baixa nous usuaris en l'aplicatiu.
5	El sistema ha de permetre al administrador modificar les dades d'un usuari.

#### Requisits funcionals en quant a la gestió de assignatures

# Requisit	Descripció
6	Els alumnes han de poder consultar les assignatures que cursen en l'any actual.
7	Els professors han de poder veure les assignatures que imparteixen en l'any actual.
8	El sistema ha de permetre al administrador afegir noves assignatures.
9	El sistema ha de permetre al administrador modificar assignatures ja existents.
10	El sistema ha de permetre al administrador eliminar assignatures.

#### Requisits funcionals en quant a la gestió de notícies

# Requisit	Descripció
11	Els alumnes han de poder consultar les notícies d'una assignatura que cursen en l'any actual.
12	Els professors han de poder afegir noves notícies per assignatures que imparteixen l'any actual.
13	El sistema ha de permetre al administrador eliminar notícies.

**Requisits funcionals en quant a la gestió d'incidències**

# Requisit	Descripció
14	El professor ha de poder afegir faltes d'assistència als alumnes d'una classe.
15	El professor ha de poder afegir incidències als alumnes d'una classe.
16	El professor ha de poder treure faltes d'assistència als alumnes d'una classe que prèviament havia posat.
17	El sistema ha de permetre al administrador eliminar incidències.

**Requisits funcionals en quant a la gestió de tasques**

# Requisit	Descripció
18	L'alumne ha de poder consultar la nota de les tasques realitzades.
19	El professor ha de poder afegir noves tasques
20	El professor ha de poder consultar les tasques afegides
21	El professor ha de poder valorar les tasques per alumnes que cursen la assignatura
22	El professor ha de poder consultar les notes dels alumnes de certa tasca

**Requisits funcionals en quant a la gestió del fòrum**

# Requisit	Descripció
23	L'alumne ha de poder afegir noves preguntes al fòrum.
24	L'alumne ha de poder afegir noves respostes al fòrum.
25	El professor ha de poder afegir noves preguntes al fòrum.
26	El professor ha de poder afegir noves respostes al fòrum.
27	L'administrador ha de poder eliminar preguntes del fòrum.
28	L'administrador ha de poder eliminar respostes del fòrum.

**Requisits funcionals en quant a la gestió de altres elements**

# Requisit	Descripció
29	El sistema ha de permetre al administrador afegir alumnes nous.
30	El sistema ha de permetre al administrador modificar alumnes ja existents.
31	El sistema ha de permetre al administrador eliminar alumnes.
32	El sistema ha de permetre al administrador afegir professors nous.
33	El sistema ha de permetre al administrador modificar professors ja existents.
34	El sistema ha de permetre al administrador eliminar professors ja existents.
35	El sistema ha de permetre al administrador afegir classes.

36	El sistema ha de permetre al administrador modificar classes.
37	El sistema ha de permetre al administrador eliminar classes.

**Requisits funcionals relacionats amb missatgeria electrònica**

# Requisit	Descripció
38	Els alumnes han de poder utilitzar un client extern de missatgeria electrònica a partir d'un destinatari i assumpte entrats en l'aplicatiu.
39	Els professors han de poder utilitzar un client extern de missatgeria electrònica a partir d'un destinatari i assumpte entrats en l'aplicatiu.

## 6.2.- Requisits no funcionals

Els requisits funcionals del sistema són requisits que especifiquen criteris a valorar sobre el comportament i el disseny de la aplicació però no aspectes a dissenyar en l'aplicatiu, ja que aquests corresponen a requisits funcionals (són els atributs de qualitat del sistema).

### 6.2.1.- Requisits d'eficiència

# Requisit	Descripció
1	L'aplicació ha de ser capaç de realitzar el procés de autenticació en un temps màxim de 5 segons amb una velocitat mínima de internet del 3 mb/s.
2	L'aplicació ha de ser capaç de tancar la sessió del usuari en un temps màxim de 5 segons amb una velocitat mínima de internet del 3 mb/s.
3	L'aplicació ha de ser capaç d'enviar informació a la base de dades en un temps màxim de 3 segons amb una velocitat mínima de internet del 3 mb/s.
4	L'aplicació ha de ser capaç de obtenir informació de la base de dades en un temps màxim de 3 segons amb una velocitat mínima de internet del 3 mb/s.

### 6.2.2.- Requisits d'ús

# Requisit	Descripció
3	Els botons i imatges de l'aplicació han de ser prou visibles i clarament diferenciats de la resta del contingut.
4	El text de l'aplicació ha de ser fàcilment visible des de una distància màxim de 70 cm.
5	Un usuari no experimentat ha de poder utilitzar sense cap problema l'aplicació després de 2 minuts d'utilitzar-la.

### 6.2.3.- Requisits de suport

# Requisit	Descripció
6	Els dispositius amb una versió d'Android 8.1 (Oreo) o superior, han de poder instal·lar i utilitzar l'aplicació sense cap problema.

## 7. Anàlisi i disseny del sistema

Durant aquest apartat s'analitzarà el sistema en qüestió per entendre els factors que intervenen en la aplicació i ajudar d'aquesta manera al desenvolupament posterior del codi.

L'aplicatiu té 2 tipus d'interfícies segons el tipus d'usuari que inicia sessió:

- **Alumne:** és l'usuari que consta com alumne del institut dins la base de dades
- **Professor:** és l'usuari que consta com a professor dins la base de dades.

### 7.1.- Anàlisi del sistema

En aquest apartat es centrarà en analitzar el sistema per trobar cada usuari que interactuarà amb ell (actor) així com cada acció que podrà realitzar. Els diagrames i casos d'ús s'han dissenyat una vegada s'han tingut clares les interaccions.

#### 7.1.1.- Actors

##### **Alumne**

Aquest usuari requereix autenticar-se abans de res i pot:

- Consultar les assignatures que està cursant aquest any.
- Accedir a una assignatura de les que està cursant aquest any.
- Consultar notícies d'una assignatura.
- Accedir a un calendari.
- Escriure un assumpte i un destinatari per enviar mail.
- Consultar notes de tasques puntuades pel professor.
- Crear noves preguntes al fòrum.
- Respondre a les preguntes del fòrum.

##### **Professor**

Aquest usuari requereix autenticar-se abans de res i pot:

- Crear noves notícies per aquelles assignatures que imparteix.
- Escriure un assumpte i un destinatari per enviar mail.

## Aplicatiu per la gestió de centres educatius

- Gestionar les assistències i incidències d'una classe durant una hora de la setmana.
- Crear noves tasques per aquelles assignatures que imparteix.
- Consultar tasques creades.
- Afegir notes de les tasques creades als alumnes d'una assignatura (totes les classes).
- Consultar notes de les tasques puntuades (totes les classes).
- Crear noves preguntes al fòrum.
- Respondre preguntes al fòrum.

### **Administrador**

L'administrador no consta com a usuari de la aplicació però és l'únic amb accés a la base de dades i l'encarregat de gestionar cada taula en ella. La seva feina consisteix en fer que el sistema es mantingui estable i estar atent a possibles errors així com peticions de professors per modificar elements en els registres de les taules. Pot:

- Gestionar alumnes.
- Gestionar professors.
- Eliminar notícies.
- Eliminar incidències.
- Gestionar usuaris.
- Gestionar classe.
- Gestionar tasques.
- Gestionar el fòrum.

## 7.1.2.- Diagrames de casos d'ús

En aquest subapartat trobarem cada cas d'ús creat a partir dels requisits definits de cada un dels actors que intervenen en l'aplicatiu.

### 7.1.2.1.- Diagrama casos d'ús visió general per l'actor alumne

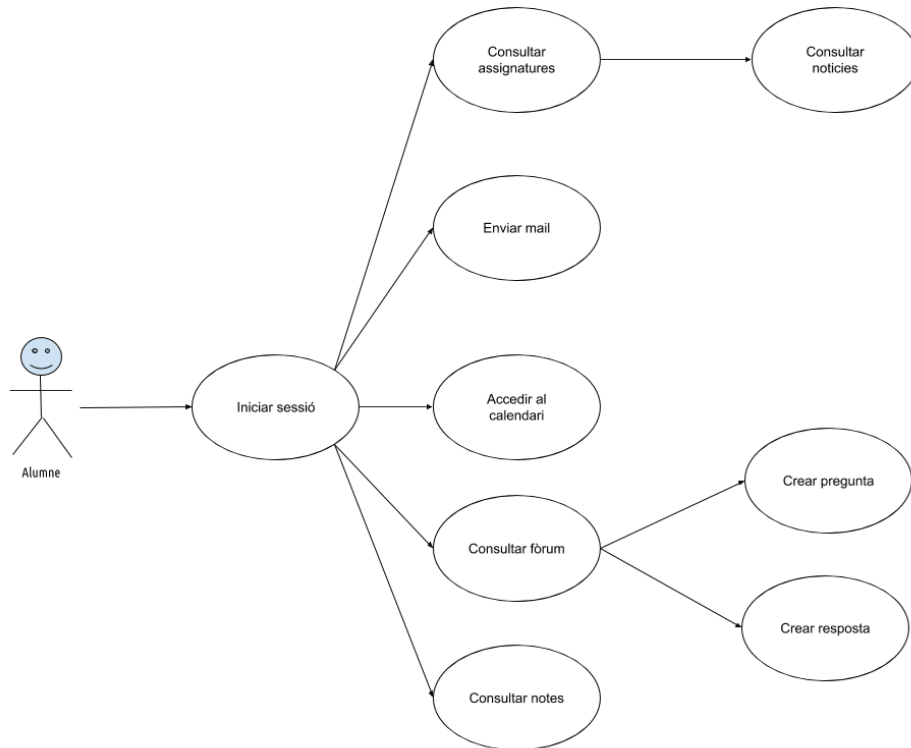


Diagrama 7.1: Casos d'ús visió general alumne

L'alumne pot iniciar sessió a través de la pantalla de login de l'aplicatiu. A partir d'aquí podrà accedir a cada una de les pantalles amb una funcionalitat diferent.

Dins de les seves assignatures haurà de poder accedir a les notícies que li afegeix els professors.

També haurà de poder accedir al calendari, consultar les seves notes i accedir al fòrum des d'on podrà crear preguntes i respostes així com mirar el contingut d'aquest.

7.1.2.2.- Diagrama casos d'ús visió general per l'actor professor



Diagrama 7.2: Casos d'ús visió general professor

El professor disposa de més funcionalitats que l'usuari. Pot accedir a la funcionalitat d'enviar mails i visualització del fòrum. De totes maneres disposa de eines per crear, visualitzar i puntuar tasques així com afegir assistències, retards i faltes.

També ha de poder afegir notícies.

### 7.1.2.3.- Diagrama casos d'ús visió general per l'administrador

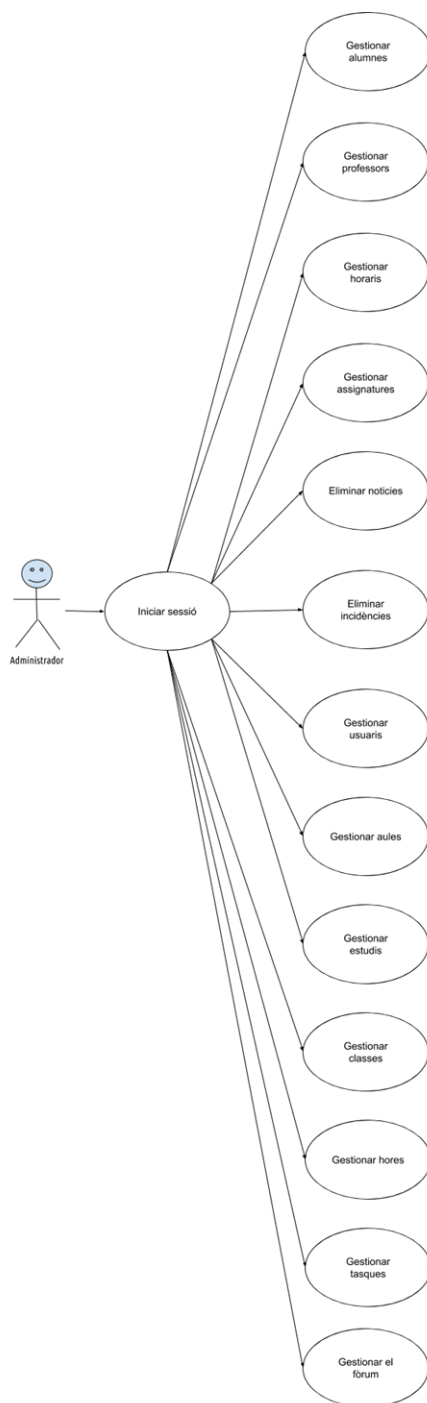


Diagrama 7.3: Casos d'ús visió general administrador

L'administrador podrà gestionar els registres de la base de dades segons les necessitats que sorgeixin. Té total disponibilitat per afegir, modificar i eliminar registres.

7.1.2.4.- Diagrama de casos d'ús: gestió d'objectes

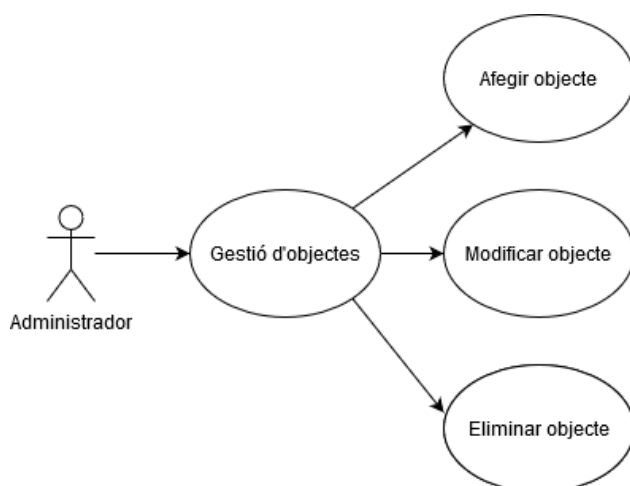


Diagrama 7.4: Cas d'ús gestió d'objectes

L'administrador ha de disposar d'eines que li permetin modificar, afegir i eliminar objectes de la base de dades.

7.1.3.- Fitxes de casos d'ús

En aquest apartat hi consten les fitxes de casos d'ús per aquells casos d'ús dels actors alumne i professor. Donat que l'administrador sempre actua de la mateixa forma i sense escenaris alternatius (afegir, modificar i eliminar registres), no s'ha considerat necessari la creació de les seves fitxes de casos d'ús. A més, tot i ser un element necessari del sistema, no té interfícies o funcionalitats a part dins l'aplicació.

**Fitxa de cas d'ús: Iniciar sessió**

<b>Nom</b>	Iniciar sessió
<b>Autor</b>	Pau Recacha Borrell
<b>Data</b>	26/08/19
<b>Descripció</b>	L'usuari introdueix les credencials per iniciar sessió.
<b>Actors</b>	Alumne, Professor
<b>Precondició</b>	L'usuari ha de existir en la base de dades.
<b>Postcondició</b>	L'usuari s'ha autenticat.
<b>Escenari Principal</b>	1.- L'usuari obre la aplicació i apareix la pantalla d'inici de sessió 2.- L'usuari entra les seves credencials i prem el botó per autenticar-se. 3.- Es mostra la pantalla de carga i es carrega el menú principal de l'aplicació.
<b>Escenari Alternatiu 1</b>	1.- L'usuari entra dades errònies i la aplicació mostra un missatge per pantalla.
<b>Escenari Alternatiu 2</b>	1.- L'usuari fa menys de dos dies que s'havia autenticat la última vegada amb el seu mòbil i l'aplicació carrega automàticament la pantalla principal.

**Fitxa de cas d'ús: Tancar sessió**

<b>Nom</b>	Tancar sessió
<b>Autor</b>	Pau Recacha Borrell
<b>Data</b>	26/08/19
<b>Descripció</b>	L'usuari tanca la sessió.
<b>Actors</b>	Alumne, Professor
<b>Precondició</b>	L'usuari s'ha d'haver autenticat prèviament.
<b>Postcondició</b>	L'usuari ha tancat la seva sessió
<b>Escenari Principal</b>	1.- L'usuari prem el botó per tancar sessió situat en el menú principal. 2.- L'aplicació borra les dades de la sessió i carrega la pantalla de login.
<b>Escenari Alternatiu</b>	-

**Fitxa de cas d'ús: Consultar assignatures**

<b>Nom</b>	Consultar assignatures
<b>Autor</b>	Pau Recacha Borrell
<b>Data</b>	26/08/19
<b>Descripció</b>	L'alumne pot veure quines assignatures fa
<b>Actors</b>	Alumne
<b>Precondició</b>	L'alumne s'ha d'haver autenticat prèviament.
<b>Postcondició</b>	L'alumne veu per pantalla les assignatures que cursa.
<b>Escenari Principal</b>	1.- L'usuari pot veure en un llistat les assignatures que cursa.
<b>Escenari Alternatiu</b>	-

**Fitxa de cas d'ús: Consultar notícies**

<b>Nom</b>	Consultar notícies
<b>Autor</b>	Pau Recacha Borrell
<b>Data</b>	26/08/19
<b>Descripció</b>	L'alumne pot veure les notícies que hi han associades en una assignatura que cursa
<b>Actors</b>	Alumne
<b>Precondició</b>	L'alumne s'ha autenticat prèviament i ha fet clic a una assignatura.
<b>Postcondició</b>	La aplicació ha carregat les notícies d'aquella assignatura per pantalla si es que n'hi havien.
<b>Escenari Principal</b>	1.- L'alumne prem un assignatura de les que se li mostren en la pantalla principal. 2.- La aplicació carrega per pantalla totes les notícies que hi han guardades per aquella assignatura. 3.- L'usuari prem una notícia per veure el seu contingut i el seu autor.
<b>Escenari Alternatiu</b>	1.- L'alumne prem un assignatura de les que se li mostren en la pantalla principal. 2.- L'aplicació mostra per pantalla un error i torna al menú principal quan no hi han notícies de aquella assignatura.

**Fitxa de cas d'ús: Afegir notícia**

<b>Nom</b>	Afegir notícia.
<b>Autor</b>	Pau Recacha Borrell
<b>Data</b>	26/08/19
<b>Descripció</b>	El professor afegeix una notícia nova d'una assignatura de les que n'és encarregat.
<b>Actors</b>	Professor
<b>Precondició</b>	El professor s'ha autenticat prèviament.
<b>Postcondició</b>	Una nova notícia s'afegeix a la base de dades.
<b>Escenari Principal</b>	1.- El professor fa clic al desplegable d'assignatures i en selecciona una. 2.- El professor entra un títol de la notícia i el contingut en els seus camps corresponents. 3.- El professor fa clic a sobre el botó de guardar.
<b>Escenari Alternatiu 1</b>	1.- El professor no selecciona una assignatura i la aplicació mostra per pantalla un error.
<b>Escenari Alternatiu 2</b>	1.- El professor es deixa de emplenar un dels dos camps de la notícia i la aplicació mostra un error.

**Fitxa de cas d'ús: Afegir falta**

<b>Nom</b>	Afegir falta
<b>Autor</b>	Pau Recacha Borrell
<b>Data</b>	26/08/19
<b>Descripció</b>	El professor afegeix una falta d'assistència a un alumne de la classe que està impartint en aquell moment.
<b>Actors</b>	Professor
<b>Precondició</b>	El professor s'ha autenticat prèviament.
<b>Postcondició</b>	El professor ha afegir una falta d'assistència a un alumne.
<b>Escenari Principal</b>	<ol style="list-style-type: none"> <li>1.- El professor fa clic a la pestanya de "Classe actual" o mou el dit en la pantalla de dreta a esquerre.</li> <li>2.- La aplicació carrega la classe que li toca fer al professor.</li> <li>3.- El professor busca l'alumne que no ha assistit a classe.</li> <li>4.- El professor fa clic al "switch" d'assistència per afegir una falta al alumne.</li> </ol>
<b>Escenari Alternatiu</b>	1.- L'aplicació no carrega cap alumne en la pestanya de "Classe actual" perquè el professor no té classe en aquell moment.

**Fitxa de cas d'ús: Treure falta**

<b>Nom</b>	Treure falta
<b>Autor</b>	Pau Recacha Borrell
<b>Data</b>	26/08/19
<b>Descripció</b>	El professor treu la falta que prèviament havia posat a un alumne.
<b>Actors</b>	Professor
<b>Precondició</b>	El professor s'ha autenticat prèviament i ha posat una falta a un alumne.
<b>Postcondició</b>	S'ha tret la falta al alumne.
<b>Escenari Principal</b>	<ol style="list-style-type: none"> <li>1.- El professor fa clic al "switch" d'assistència de aquell alumne que prèviament havia desactivat.</li> <li>2.- La aplicació treu la falta de alumne i el switch torna a estar activat.</li> </ol>
<b>Escenari Alternatiu</b>	-

**Fitxa de cas d'ús: Afegir incidència**

Nom	Afegir incidència
Autor	Pau Recacha Borrell
Data	26/08/19
Descripció	El professor afegeix una incidència a un alumne de la classe que imparteix en aquell moment.
Actors	Professor
Precondició	El professor s'ha autenticat prèviament.
Postcondició	La aplicació guarda la nova incidència del alumne.
Escenari Principal	<ol style="list-style-type: none"> <li>1.- El professor fa clic a la pestanya de "Classe actual" o mou el dit en la pantalla de dreta a esquerre.</li> <li>2.- La aplicació carrega la classe que li toca fer al professor.</li> <li>3.- El professor busca aquell alumne al qual vol afegir una incidència.</li> <li>4.- El professor fa clic al botó d'afegir.</li> <li>5.- L'aplicació mostra una pantalla on introduir el contingut de la incidència.</li> <li>6.- El professor fa clic a guardar.</li> </ol>
Escenari Alternatiu 1	1.- L'aplicació no carrega cap alumne en la pestanya de "Classe actual" perquè el professor no té classe en aquell moment.
Escenari Alternatiu 2	<ol style="list-style-type: none"> <li>1.- El professor fa clic a per tancar la pantalla d'afegir incidència.</li> <li>2.- La pantalla es tanca.</li> </ol>

**Fitxa de cas d'ús: Afegir tasca**

Nom	Afegir tasca
Autor	Pau Recacha Borrell
Data	26/08/19
Descripció	El professor afegeix una nova tasca a la base de dades
Actors	Professor
Precondició	El professor s'ha autenticat prèviament.
Postcondició	La aplicació guarda la nova tasca afegida pel professor.
Escenari Principal	<ol style="list-style-type: none"> <li>1.- El professor fa clic al boto de la pantalla principal per afegir una nova tasca.</li> <li>2.- El professor selecciona del desplegable la assignatura a la qual vol associar la tasca</li> <li>3.- El professor afegeix un títol a la tasca així com un contingut.</li> <li>4.- El professor fa clic al botó de guardar.</li> </ol>
Escenari Alternatiu	1.- El professor no selecciona assignatura i l'aplicació mostra un error.

**Fitxa de cas d'ús: Consultar tasques**

Nom	Consultar tasques
Autor	Pau Recacha Borrell
Data	26/08/19
Descripció	El professor consulta els tasques que ell ha guardat
Actors	Professor
Precondició	El professor s'ha autenticat prèviament.
Postcondició	El professor ha pogut veure les tasques creades
Escenari Principal	1.- El professor fa clic a la pestanya de tasques o mou el dit de dreta a esquerre fins arribar-hi. 2.- El professor pot veure per pantalla quines tasques ha afegit
Escenari Alternatiu	1.- El professor no ha afegit cap tasca i la pantalla es mostra buida. 2.- L'aplicació mostra per pantalla un missatge assenyalant el problema.

**Fitxa de cas d'ús: Puntuar tasca**

Nom	Puntuar tasca
Autor	Pau Recacha Borrell
Data	26/08/19
Descripció	El professor puntua una tasca que ell havia creat prèviament.
Actors	Professor
Precondició	El professor s'ha autenticat prèviament.
Postcondició	La aplicació guarda la nota de la tasca entrada pel professor.
Escenari Principal	1.- El professor fa clic a la pestanya de tasques o mou el dit de dreta a esquerre fins arribar-hi. 2.- El professor selecciona una assignatura de la llista. 3.- El desplegable de tasques d'aquella assignatura s'obre mostrant les tasques de la assignatura. 4.- El professor clica alguna tasca a puntuar. 5.- S'obra una pantalla amb el nom de cada alumne que participa en l'assignatura per a que el professor pugui posar notes. 6.- El professor posa les notes dels alumnes i prem guardar. 7.- Es torna a la pantalla anterior.
Escenari Alternatiu	1.- El professor fa clic a una tasca que ja ha avaluat. 2.- L'aplicació mostra una pantalla de notes.

**Fitxa de cas d'ús: Consultar notes professor**

Nom	Consultar notes professor
Autor	Pau Recacha Borrell
Data	26/08/19
Descripció	El professor consulta les notes de les tasques
Actors	Professor
Precondició	El professor s'ha autenticat prèviament.
Postcondició	La aplicació guarda la nova incidència del alumne.
Escenari Principal	<ol style="list-style-type: none"> <li>1.- El professor fa clic a la pestanya de tasques o mou el dit de dreta a esquerre fins arribar-hi.</li> <li>2.- El professor selecciona una assignatura de la llista.</li> <li>3.- El desplegable de tasques d'aquella assignatura s'obre mostrant les tasques de la assignatura.</li> <li>4.- El professor clica alguna tasca.</li> <li>5.- S'obra una nova pantalla amb un llistat d'alumnes amb la nota de la tasca.</li> </ol>
Escenari Alternatiu 1	1.- El professor no ha avaluat la tasca clicada i s'obra la pantalla d'avaluació.

**Fitxa de cas d'ús: Consultar notes alumne**

Nom	Consultar notes alumne
Autor	Pau Recacha Borrell
Data	26/08/19
Descripció	L'alumne consulta la nota de les tasques avaluades.
Actors	Alumne
Precondició	El professor s'ha autenticat prèviament.
Postcondició	La aplicació ha mostrat per pantalla les notes del alumne.
Escenari Principal	<ol style="list-style-type: none"> <li>1.- L'alumne fa clic a notes o mou el dit en la pantalla de dreta a esquerre fins arribar a la pantalla.</li> <li>2.- L'aplicació mostra per pantalla les notes de les diferents tasques avaluades pel professor.</li> </ol>
Escenari Alternatiu 1	1.- L'alumne no té notes avaluades i es mostra per pantalla un error.

**Fitxa de cas d'ús: Crear pregunta fòrum**

Nom	Crear pregunta fòrum
Autor	Pau Recacha Borrell
Data	26/08/19
Descripció	L'usuari afegeix una nova pregunta al fòrum.
Actors	Professor, Alumne
Precondició	L'usuari s'ha autenticat prèviament.
Postcondició	La aplicació guarda la pregunta del fòrum.
Escenari Principal	<ol style="list-style-type: none"> <li>1.- L'usuari clica al botó de fòrum situat en la pantalla principal.</li> <li>2.- Es carrega la pantalla de fòrum de l'aplicació amb les diferents preguntes guardades segons l'estudi.</li> <li>3.- L'usuari prem el botó per afegir pregunta.</li> <li>4.- Es mostra una nova pantalla per afegir un títol i un contingut de pregunta.</li> <li>5.- L'usuari entra les dades i prem el botó per guardar.</li> </ol>
Escenari Alternatiu 1	1.- L'usuari és un professor i pot veure totes les preguntes guardades al fòrum.

**Fitxa de cas d'ús: Crear resposta fòrum**

Nom	Crear resposta fòrum
Autor	Pau Recacha Borrell
Data	26/08/19
Descripció	L'usuari ha respòs a una pregunta del fòrum.
Actors	Professor, Alumne
Precondició	L'usuari s'ha autenticat prèviament
Postcondició	La aplicació guarda la nova resposta a la pregunta del fòrum.
Escenari Principal	<ol style="list-style-type: none"> <li>1.- L'usuari clica al botó de fòrum situat en la pantalla principal.</li> <li>2.- Es carrega la pantalla de fòrum de l'aplicació amb les diferents preguntes guardades segons l'estudi.</li> <li>3.- L'usuari clica a una pregunta del fòrum.</li> <li>4.- Es carrega la pantalla de respostes de la pregunta.</li> <li>5.- L'usuari prem al botó per afegir una nova resposta.</li> <li>6.- Es mostra una pantalla per afegir una resposta a la pregunta.</li> <li>7.- L'usuari emplena els camps de la resposta i prem guardar.</li> </ol>
Escenari Alternatiu 1	1.- L'usuari és un professor i pot veure totes les preguntes guardades al fòrum.

**Fitxa de cas d'ús: Enviar mail**

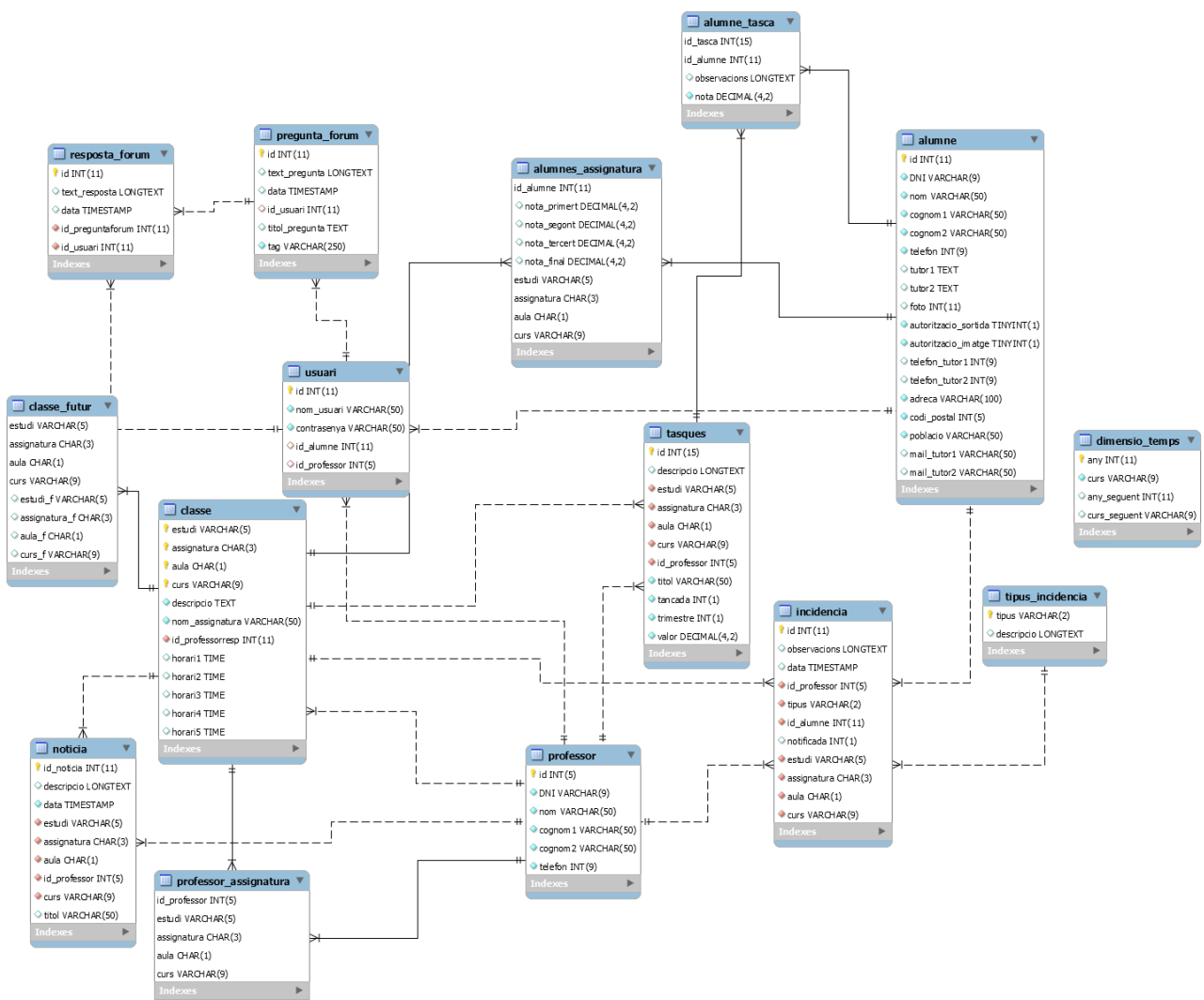
<b>Nom</b>	Enviar mail
<b>Autor</b>	Pau Recacha Borrell
<b>Data</b>	26/08/19
<b>Descripció</b>	L'usuari escull un assumpte i destinatari per utilitzar un client de mails
<b>Actors</b>	Alumne, Professor
<b>Precondició</b>	L'usuari s'ha d'haver autenticat prèviament.
<b>Postcondició</b>	S'obra un client de correu electrònic per enviar un mail
<b>Escenari Principal</b>	<ol style="list-style-type: none"> <li>1.- L'usuari prem el botó amb el símbol de carta</li> <li>2.- Una vegada clicat el botó, l'aplicació obre una nova pantalla.</li> <li>3.- L'usuari entra un destinatari i un assumpte en els camps corresponents.</li> <li>4.- L'usuari prem "Enviar" i el mòbil preguntarà al usuari quina aplicació per enviar mails vol obrir.</li> <li>5.- S'obra la aplicació triada i una vegada carregada, els camps assumpte i destinatari estaran omplerts.</li> </ol>
<b>Escenari Alternatiu</b>	<ol style="list-style-type: none"> <li>1.- L'usuari clica per tancar la nova pantalla.</li> <li>2.- La pantalla es tanca.</li> </ol>

## 7.2.- Model de dades

Una vegada s'han analitzat els requisits del sistema i s'han creat les diferents fitxes de casos d'ús, s'ha definit el model de dades.

La base de dades que s'ha utilitzat per aquest projecte s'ha creat amb la interfície HeidiSQL que permet de forma senzilla i simple crear l'estructura de les taules d'aquesta, les relacions entre elles així com les seves corresponents claus primàries, foranes i índexs.

La estructura de la base de dades és la següent:



Imatge 7.1 : Model de la base de dades

Les classes es classifiquen segons l'estudi (per exemple 1rESO), l'assignatura abreviada (MAT), l'aula on es realitza dins del estudi (A) i el curs (2020-2021). De la classe també en sabem el seus horaris segons el dia de la setmana.

Cada classe hi ha un grup d'estudiants. De cada estudiant en volem saber quina nota trimestral ha tingut per aquella assignatura i la final.

De cada classe n'és responsable un professor el qual pot penjar notícies per informar de certs assumptes als alumnes.

Quan un professor esta a la classe fent la hora, pot apuntar qui ha assistit a classe i qui no, així com afegir incidències als alumnes. Les incidències poden ser de varis tipus, com per exemple; retard, falta, falta greu o expulsió.

Els professor, per puntuar els alumnes d'una classe, crea tasques. Les tasques poden ser des de la resolució d'uns exercicis fins a un examen trimestral i estan relacionades a un trimestre. Quan un professor puntuar una tasca i la guarda a la base de dades aquesta queda tancada.

Tant professors com alumnes disposen d'un usuari al qual accedir a l'aplicatiu. Dins el fòrum de l'aplicació un usuari pot crear preguntes i respondre a d'altres. L'usuari només visualitzarà aquelles preguntes relacionades amb les assignatures que fa.

Per tal de millorar la gestió del temps de la base de dades s'inclou una taula de temps on trobarem per cada curs, a quin any correspon i els seus corresponents anys futurs.

També, per donar una relació als cursos entre anys (per exemple, quan un alumne aprova una assignatura, quina assignatura li toca fer l'any vinent), s'inclou una altre taula on s'especifica aquesta relació.

Per entendre millor el model, a continuació es mostren les dades de les taules al detall. En negreta es troben les claus primàries i les foranes subratllades.

**Taula alumnes:** conté les dades sobre els alumnes.

*alumne(id, DNI, nom, congom1, congom2, telefon, tutor1, tutor2, foto, autorització\_sortida, autorització\_imatge, telèfon\_tutor1, telèfon\_tutor2, adreça, codi\_postal, població, mail\_tutor1, mail\_tutor2)*

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
id	INT(11)	NO	PRIMÀRIA	AUTO_INCREMENT
DNI	VARCHAR(9)	NO	-	-
nom	VARCHAR(50)	NO	-	-

cognom1	VARCHAR(50)	NO	-	-
cognom2	VARCHAR(50)	NO	-	-
telefon	VARCHAR(9)	NO	-	-
tutor1	TEXT	SI	-	NULL
tutor2	TEXT	SI	-	NULL
foto	INT(11)	SI	-	NULL
autoritzacio_sor tida	TINYINT	NO	-	
autoritzacio_ima tge	TINYINT	NO	-	
telefon_tutor1	INT(9)	SI	-	NULL
telefon_tutor2	INT(9)	SI	-	NULL
adreca	VARCHAR(100)	NO	-	-
codi_postal	INT(5)	NO	-	-
poblacio	VARCHAR(50)	NO	-	-
mail_tutor1	VARCHAR(50)	SI	-	NULL
mail_tutor2	VARCHAR(50)	SI	-	NULL

**Taula classe:** Conté la informació sobre les classes que es fan en el col·legi.

*classe(estudi, assignatura, aula, curs, descripció, nom\_assignatura, id\_professorresp, horari1, horari2, horari3, horari4, horari5)*

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
estudi	VARCHAR(5)	NO	PRIMÀRIA	-
assignatura	CHAR(3)	NO	PRIMÀRIA	-
aula	CHAR(3)	NO	PRIMÀRIA	-
curs	VARCHAR(9)	NO	PRIMÀRIA	-
descripció	TEXT	NO	-	-
nom_assignatura	VARCHAR(50)	NO	-	-
id_professorresp	INT(11)	NO	-	-
horari1	TIME	SI	-	NULL
horari2	TIME	SI	-	NULL
horari3	TIME	SI	-	NULL
horari4	TIME	SI	-	NULL
horari5	TIME	SI	-	NULL

horari'X': Determina la hora en que es fa la classe. El número estableix el dia de la setmana (1 correspon al dilluns, 2 al dimarts, ...).

**Taula alumnes\_assignatura:** Conté la informació sobre les assignatures que fa cada alumne juntament amb les seves notes trimestrals i finals.

*alumnes\_assignatura(id\_alumne, estudi, assignatura, aula, curs, nota\_primert, nota\_segont, nota\_tercert, nota\_final)*

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
id_alumne	INT(11)	NO	PRIMÀRIA/FORANA	-
estudi	VARCHAR(5)	NO	PRIMÀRIA/FORANA	-
assignatura	VARCHAR(3)	NO	PRIMÀRIA/FORANA	-
aula	VARCHAR(1)	NO	PRIMÀRIA/FORANA	-
curs	VARCHAR(9)	NO	PRIMÀRIA/FORANA	-
nota_primert	DECIMAL(4,2)	SI	-	NULL
nota_segont	DECIMAL(4,2)	SI	-	NULL
nota_tercert	DECIMAL(4,2)	SI	-	NULL
nota_final	DECIMAL(4,2)	SI	-	NULL

**Taula tasques:** conté el conjunt de tasques que els professors han afegit a la base de dades.

*tasques(id, estudi, assignatura, aula, curs, id\_professor, descripcio, títol, tancada, trimestre, valor)*

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
id	INT(15)	NO	PRIMÀRIA	AUTO_INCREMENT
estudi	VARCHAR(5)	NO	FORANA	-
assignatura	CHAR(3)	NO	FORANA	-
aula	CHAR(1)	NO	FORANA	-
curs	VARCHAR(9)	NO	FORANA	-
id_professor	INT(5)	NO	FORANA	-
descripció	LONGTEXT	SI	-	NULL
títol	VARCHAR(50)	SI	-	NULL
tancada	INT(1)	NO	-	0
trimestre	INT(1)	SI	-	NULL
valor	DECIMAL(4,2)	NO	-	0.0

tancada: Flag per establir quan una tasca esta tancada o no. Si està tancada el professor ja ha entrat les notes.

valor: Determina el % de pes que se li dona a la tasca pel trimestre en qüestió.

**Taula alumne\_tasca:** conté les notes dels alumnes respecte les tasques tancades.

*alumne\_tasca(id\_tasca, id\_alumne, observacions, nota)*

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
Id_tasca	INT(15)	NO	PRIMÀRIA	-
Id_alumne	INT(11)	NO	PRIMÀRIA/FORANA	-
observacions	LONGTEXT	SI	-	NULL
nota	DECIMAL(4,2)	NO	-	-

**Taula classe\_futur:** conté la relació de classes d'un curs al següent (determina quina classe ha d'anar un alumne quan ha aprovat la actual).

*classe\_futur (estudi, assignatura, aula, curs, estudi\_f, assignatura\_f, aula\_f, curs\_f)*

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
estudi	VARCHAR(5)	NO	PRIMÀRIA/FORANA	-
assignatura	CHAR(3)	NO	PRIMÀRIA/FORANA	-
aula	CHAR(1)	NO	PRIMÀRIA/FORANA	-
curs	VARCHAR(9)	NO	PRIMÀRIA/FORANA	-
estudi_f	VARCHAR(5)	SI	-	NULL
assignatura_f	CHAR(3)	SI	-	NULL
aula_f	CHAR(1)	SI	-	NULL
curs_f	VARCHAR(9)	SI	-	NULL

**Taula dimensio\_temps:** conté informació relacionada amb la temporalitat.

*dimensio\_temps(any, curs, any\_seguent, curs\_seguent)*

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
any	INT(11)	NO	PRIMÀRIA	-
curs	VARCHAR(9)	NO	-	-
any_seguent	INT(11)	SI	-	NULL
curs_seguent	VARCHAR(9)	SI	-	NULL

**Taula incidències:** conté informació sobre les incidències que els professors han anat inserint al llarg del curs.

*incidencies(id, id\_professor, tipus, id\_alumne, estudi, assignatura, aula, curs, observacions, data, notificada)*

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
id	INT(11)	NO	PRIMÀRIA	AUTO_INCREMENT
id_professor	INT(5)	NO	FORANA	-
tipus	VARCHAR(2)	NO	FORANA	-
id_alumne	INT(11)	NO	FORANA	-
estudi	VARCHAR(5)	NO	FORANA	-
assignatura	CHAR(3)	NO	FORANA	-
aula	CHAR(1)	NO	FORANA	-
curs	VARCHAR(9)	NO	FORANA	-
observacions	LONGTEXT	SI	-	NULL
data	TIMESTAMP	SI	-	NULL
notificada	INT(1)	SI	-	0

**Taula tipus\_incidència:** conté informació sobre el tipus de cada incidència.

*tipus\_incidencia(tipus, descripcio)*

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
tipus	VARCHAR(2)	NO	PRIMÀRIA	-
descripció	LONGTEXT	SI	-	NULL

**Taula notícies:** conté informació sobre el conjunt de notícies que els professor han anat penjant a la base de dades.

*noticies(id, estudi, assignatura, aula, curs, id\_professor, títol, descripcio, data)*

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
id	INT(11)	NO	PRIMÀRIA	AUTO_INCREMENT
estudi	VARCHAR(5)	NO	FORANA	-
assignatura	CHAR(3)	NO	FORANA	-
aula	CHAR(1)	NO	FORANA	-
curs	VARCHAR(9)	NO	FORANA	-
id_professor	INT(5)	NO	FORANA	-
títol	VARCHAR(50)	SI	-	NULL
descripció	LONGTEXT	SI	-	NULL
data	TIMESTAMP	NO	-	CURRENT_TIMESTAMP

**Taula pregunta\_forum:** conté informació sobre les preguntes creades en el fòrum de l'aplicació.

*pregunta\_forum(id, id\_usuari, títol\_pregunta, text\_pregunta, tag, data)*

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
id	INT(11)	NO	PRIMÀRIA	AUTO_INCREMENT
id_usuari	INT(11)	NO	FORANA	-
títol_pregunta	TEXT	SI	-	NULL
text_pregunta	LONGTEXT	SI	-	NULL
tag	VARCHAR(250)	NO	-	-
data	TIMESTAMP	SI	-	CURRENT_TIMESTAMP

tag: "Concepte" de la pregunta. Associat a les assignatures que fa el usuari amb id 'id\_usuari' o concepte "General" (pot veure-ho tothom).

**Taula resposta\_forum:** conté informació sobre les respostes a les preguntes creades en el fòrum de l'aplicació.

*resposta\_forum(id, id\_usuari, títol\_pregunta, text\_pregunta, tag, data)*

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
id	INT(11)	NO	PRIMÀRIA	AUTO_INCREMENT
id_usuari	INT(11)	NO	FORANA	-
Id_preguntaforum	INT(11)	NO	FORANA	-
text_resposta	LONG_TEXT	SI	-	-
data	TIMESTAMP	SI	-	CURRENT_TIMESTAMP

**Taula professor:** conte la informació sobre els professors del col·legi.

*professor(id, DNI, nom, cognom1, cognom2, telefon)*

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
id	INT(5)	NO	PRIMÀRIA	AUTO_INCREMENT
DNI	VARCHAR(9)	SI	-	NULL
nom	VARCHAR(50)	SI	-	NULL
cognom1	VARCHAR(50)	SI	-	NULL
cognom2	VARCHAR(50)	SI	-	NULL
telefon	INT(9)	SI	-	NULL

**Taula professor\_assignatura:** conté informació sobre les assignatures que fan els professors.

professor\_assignatura (*id\_professor, estudi, assignatura, aula, curs*)

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
id_professor	INT(5)	NO	PRIMÀRIA/FORANA	-
estudi	VARCHAR(5)	NO	PRIMÀRIA/FORANA	-
assignatura	CHAR(3)	NO	PRIMÀRIA/FORANA	-
aula	CHAR(1)	NO	PRIMÀRIA/FORANA	-
curs	VARCHAR(9)	NO	PRIMÀRIA/FORANA	-

**Taula usuari:** conté informació sobre els usuaris que utilitzen l'aplicació.

usuari (*id, id\_alumne, id\_professor, nom\_usuari, contrasenya*)

#Nom	#Tipus	#NULL	#Clau	#Predeterminat
id	INT(11)	NO	PRIMÀRIA	AUTO_INCREMENT
id_alumne	INT(11)	NO	FORANA	-
id_professor	INT(5)	NO	FORANA	-
nom_usuari	VARCHAR(50)	SI	-	NULL
contrasenya	VARCHAR(50)	SI	-	NULL

## 7.3.- Disseny de les interfícies d'usuari de la app

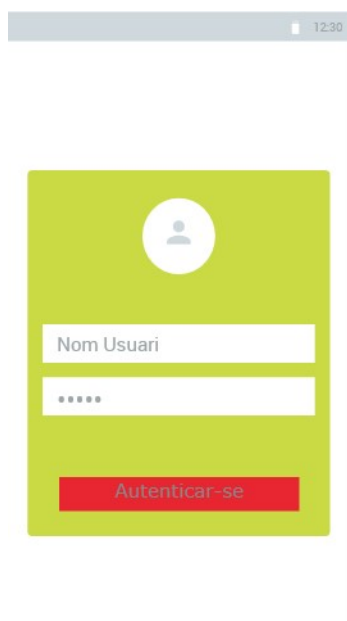
Un vegada ja tenim clars els requisits i necessitats del sistema, així com creats els diagrames i les fitxes de casos d'ús, podem passar a pensar i dissenyar les interfícies que tindrà la aplicació per a cada usuari.

En aquest apartat es separaran cada una de les interfícies per els usuaris que la utilitzin. En cada interfície s'explicaran quines funcionalitats es tenen pensada per a cada una i com es comportarà la aplicació segons la interacció del usuari (en algunes s'ha considerat innecessari donada la seva simplicitat).

El disseny s'ha fet amb la pàgina web FluidUI i alguns dissenys no són el resultat final de les interfícies de l'aplicatiu.

### 7.3.1.- Interfícies d'alumne i professor

#### 7.3.1.1.- Interfície de login

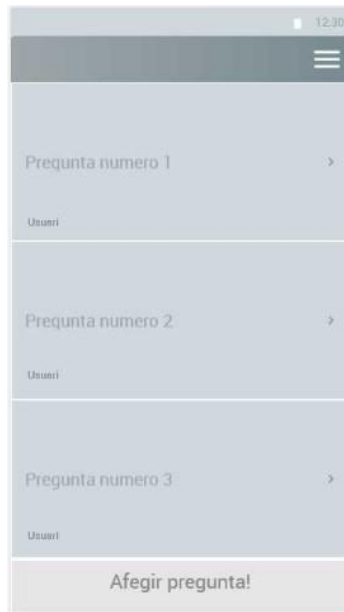


Imatge 7.2: Interfície login

La pantalla de login és una pantalla molt comuna i s'assembla a moltes aplicacions que hi han al mercat. Disposa de dos entrades de text, una pel nom d'usuari i un altre per la contrasenya.

En la pantalla de login, si l'usuari s'oblida d'entrar algun camp i prem el botó d'autenticar-se, l'aplicació automàticament mostrarà un error per pantalla per fer-li veure que s'ha oblidat d'un camp. Quan l'usuari premi autenticar-se, l'aplicació mostrarà una pantalla de carga i si les credencials són correctes, iniciarà sessió sense cap problema. Si l'usuari s'ha equivocat, mostrarà un error al respecte.

### 7.3.1.2.- Interfície principal de fòrum.



Imatge 7.3: Interfície principal fòrum

La interfície principal del fòrum es el mes senzilla possible per evitar confusió en l'usuari. Cada pregunta es situarà en un bloc de llistat de preguntes guardades en el fòrum (que es relacionin amb l'estudi del alumne si l'usuari és d'aquest tipus).

Quan l'usuari premi una pregunta de les que apareguin, automàticament s'obrirà la pantalla 7.4.1.3. Per altre banda, si decideix afegir una nova pregunta clicarà sobre "Afegir pregunta!" i s'obrirà una pantalla 7.4.1.4. Si a l'usuari li interessa comprovar si hi han noves preguntes, des de dalt de tot de la pantalla, mourà el dit de dalt cap avall i es tornarà a generar una consulta a la base de dades per comprovar si hi han noves preguntes.

### 7.3.1.3.- Interfície de respostes de pregunta.



Imatge 7.4: Interfície respostes pregunta

En aquesta pantalla es veurà la pregunta en la seva totalitat. Es veurà quin usuari la ha penjat, quina és la pregunta (aquestes dos coses ja es veien en la pantalla 7.4.1.2) i el contingut de la pregunta. Sota d'ella es situaran les respostes a la pregunta. Si el usuari vol afegir una nova resposta, només haurà de clicar al botó "Afegir resposta!" i s'obrirà la pantalla 7.4.1.4.

### 7.3.1.4.- Interfície de crear nou registre al fòrum.



Imatge 7.5: Interfície crear nou registre

Una interfície simple amb una sola funció: guardar una nova pregunta/resposta del usuari. Si el usuari vol guardar una resposta, el bloc de text de "Posa aquí el títol" desapareixerà per mostrar només el "Posa aquí el

contingut”. Quan l'usuari cregui que ja està llesta la pregunta o resposta, clicarà “Guardar!”.

### 7.3.2.- Interfícies d'alumne

#### 7.3.2.1.- Interfície d'assignatures

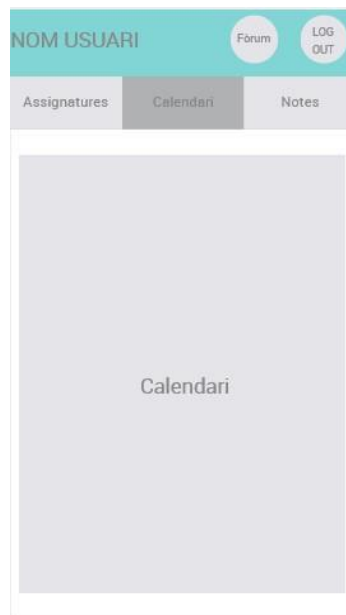


Imatge 7.6: Interfície d'assignatures del alumne

Aquesta interfície és el “Menú principal” del usuari alumne. En ella es segueix un patró que s'ha utilitzar per crear totes les interfícies que estan lligades d'alguna manera al menú principal. Quan l'alumne entra en aquesta pantalla a través del login, es carreguen en la llista de “Meves Assignatures” les assignatures que està cursant aquest any. Si prem alguna assignatura de les llistades, la aplicació carregarà la interfície de l'apartat 7.4.2.3.

Com es pot veure, a la “toolbar” de la interfície veiem que s'hi carregarà el nom de l'alumne que s'autentifiqui i a la dreta hi hauran dos botons; un per tancar sessió i l'altre per entrar al fòrum. A sota la toolbar hi veurem un conjunt de pestanyes que serviran per accedir a la resta de pantalles. La pestanya on es trobi l'usuari quedarà marcada amb un color diferent a la resta.

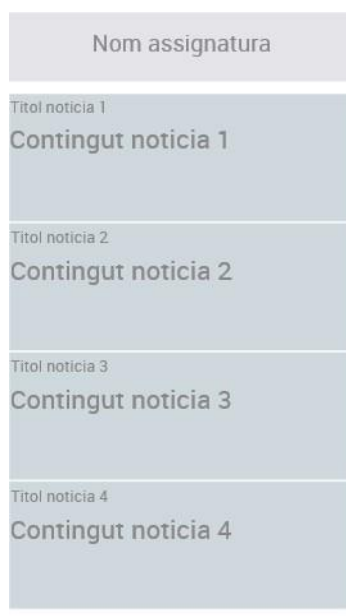
### 7.3.2.2.- Interfície de calendari



Imatge 7.7: Interfície del calendari

La interfície del calendari utilitza un calendari que ocupa tota la pantalla. La idea de aquesta interfície era que l'alumne disposés d'un calendari que funcionés a mode d'agenda.

### 7.3.2.3.- Interfície de notícies



Imatge 7.8: Interfície de les notícies del alumnat

La interfície de notícies mostra les notícies que el professor d'una assignatura ha anat afegint. Cada notícia conté un títol i un contingut.

### 7.3.2.4.- Interfície de notes



Imatge 7.9: Interfície de les notes del alumne

La última interfície dels alumnes els permet consultar notes que han realitzat de les diferents assignatures que cursen. Quan un alumne prem un desplegable dels que porten noms d'assignatures, el desplegable mostra cada una de les tasques guardades per aquella assignatura. Cada tasca conté el títol d'aquesta així com la nota treta en ella.

### 7.3.3.- Interfícies de professor

#### 7.3.3.1.- Interfície de creació de notícies/tasques



Imatge 7.10: Interfície de nou registre

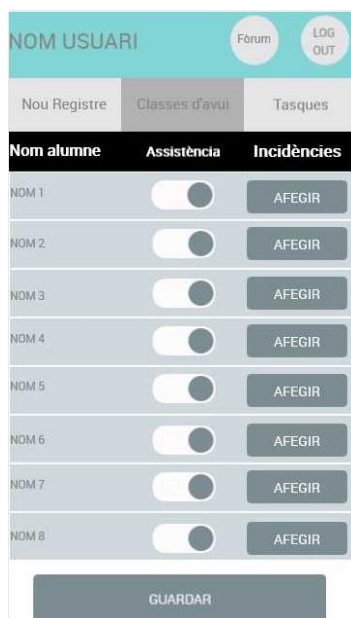
## Aplicatiu per la gestió de centres educatius

El patró d'aquesta interfície és el mateix que la resta de les del alumne, però el nom i contingut de cada pantalla canvia. En la toolbar tornem a trobar el nom del professor i dos botons; un per accedir al fòrum de la aplicació i l'altre per tancar sessió.

En aquesta pantalla el professor pot afegir un nou registre de notícia o tasca a la base de dades. Per afegir una notícia ha de activar el switch i per afegir una tasca, desactivar-lo.

A sota del switch hi ha un desplegable amb les assignatures que imparteix el professor. A sota del desplegable s'hi troben dos caixes on el professor haurà de posar el títol i el contingut del registre. Si el professor no selecciona una assignatura o no entra dades en una de les dos caixes (o les dos), la aplicació mostrarà un error per pantalla.

### 7.3.3.2.- Interfície de gestió de assistències/incidències

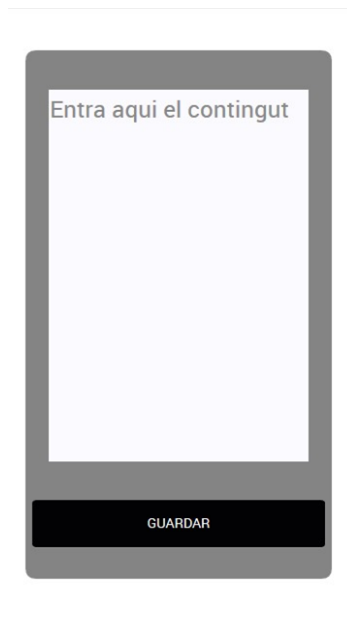


Imatge 7.11: Interfície de gestió de classe

Aquesta pantalla carrega els alumnes d'una classe que té el professor segons l'horari establert a la base de dades. Cada un dels elements del desplegable conté: el nom de l'alumne, un switch d'assistència i un botó per afegir incidències al alumne.

Preferiblement, el professor hauria de mantenir aquesta pantalla en l'aplicació mentre duri la classe. D'aquesta manera podrà seguir afegint incidències o canviar les assistències. Una vegada cregui que ja pot guardar les dades inserides, només ha de clicar al botó "GUARDAR" per enviar les dades a la base de dades.

### 7.3.3.3.- Interfície per afegir incidència



Imatge 7.12: Interfície per afegir incidència

### 7.3.3.4.- Interfície de consultar tasques



Imatge 7.13: Interfície consultar tasques

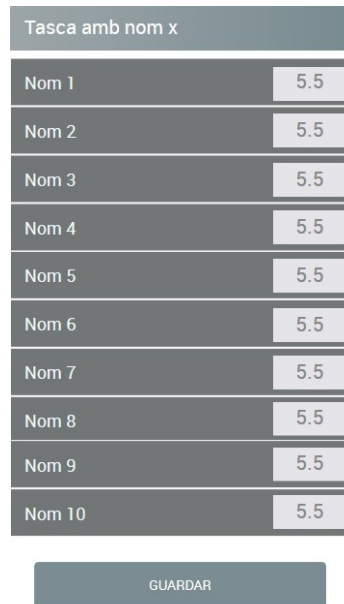
En aquesta pantalla es mostraran en un desplegable, les assignatures que tinguin tasques associades. Si no hi han tasques guardades per aquell professor, la pantalla estarà buida i l'aplicació mostrarà un error.

Si hi han assignatures i el professor fa clic a una d'elles, el desplegable mostrarà les tasques guardades per aquella assignatura. Si una tasca ja ha estat puntuada, llavors tindrà un color diferent a la resta.

## Aplicatiu per la gestió de centres educatius

Si un professor fa clic a una tasca no puntuada, l'aplicació carregarà la interfície de l'apartat 7.4.3.4. Per altre banda, si ja s'ha puntuat, l'aplicació mostrarà la pantalla de l'apartat 7.4.3.4 però aquesta vegada amb les notes guardades per cada alumne que el professor havia puntuat en algun moment passat.

### 7.3.3.5.- Interfície de puntuar/consultar notes tasca



Tasca amb nom x	
Nom 1	5.5
Nom 2	5.5
Nom 3	5.5
Nom 4	5.5
Nom 5	5.5
Nom 6	5.5
Nom 7	5.5
Nom 8	5.5
Nom 9	5.5
Nom 10	5.5

GUARDAR

Imatge 7.14: Interfície per puntuar/consultar notes tasca

Aquesta pantalla és on el professor puntuarà les notes de les tasques. Cada tasca es puntua per estudis, de manera que quan un professor vulgui puntuar una tasca, es carregaran els alumnes de totes les classes del estudi (1er ESO, 2on ESO, ...).

Si el professor entra una nota superior a 10, l'aplicació mostrarà per pantalla un error i modificarà la nota a 0.

Una vegada el professor ha puntuat la tasca, clicarà al botó "GUARDAR" per enviar les dades a la base de dades.

Aquesta pantalla té una altre funció. Si la tasca ja ha sigut puntuada, a la pantalla es veuran les notes que va treure cada alumne del estudi per la tasca determinada. Quan el professor vulgui tornar, només haurà de fer clic al botó enrere del seu mòbil o prémer el botó "TORNAR" (és el mateix botó que "GUARDAR" però amb una funció diferent).

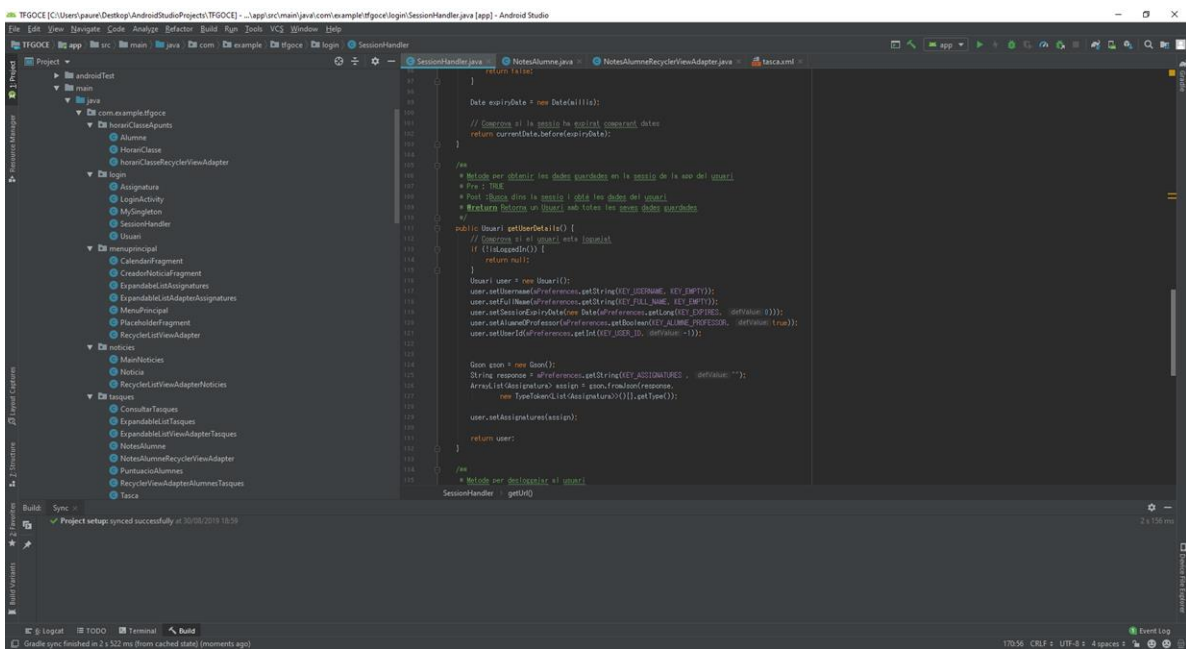
## 8. Estudi i decisions

Amb tota la informació sobre el model de dades, disseny i requisits necessaris per implementar el projecte, es va realitzar un estudi sobre quines eines serien més adients per fer-ho.

Els llenguatges escollits per desenvolupar el projecte han sigut:

- Android: per desenvolupar les funcionalitats de l'aplicatiu.
- XML: per dissenyar les interfícies de l'aplicatiu.
- PHP: per gestionar les crides a la base de dades de l'aplicatiu.
- MySQL: per realitzar les consultes i insercions a la base de dades.
- C# amb ASP i MVC per l'aplicació del administrador.

Per tal de implementar l'aplicació es va escollir el programa Android Studio, un programa amb llicència gratuïta i molt fàcil d'utilitzar. El punt bo d'aquesta programa es que l'utilitza molta gent i es fàcil trobar solucions a possibles problemes que sorgeixin amb aquest.

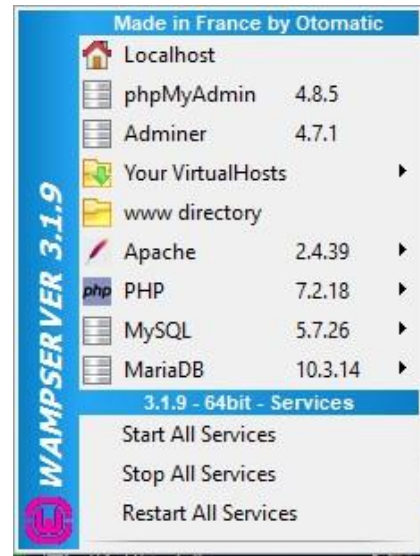


Imatge 8.1: Interfície Android Studio

Per tenir un servidor local a casa, he triat el programa WAMPServer (gratuït també). Aquest programa va ser el que vaig utilitzar fa uns anys per desenvolupar l'aplicatiu del treball de recerca i vaig pensar en el moment de triar-lo que em seria fàcil de configurar. Tot i que vaig tenir uns problemes que varen comportar dies de proves i errors, al final m'ha anat molt bé.

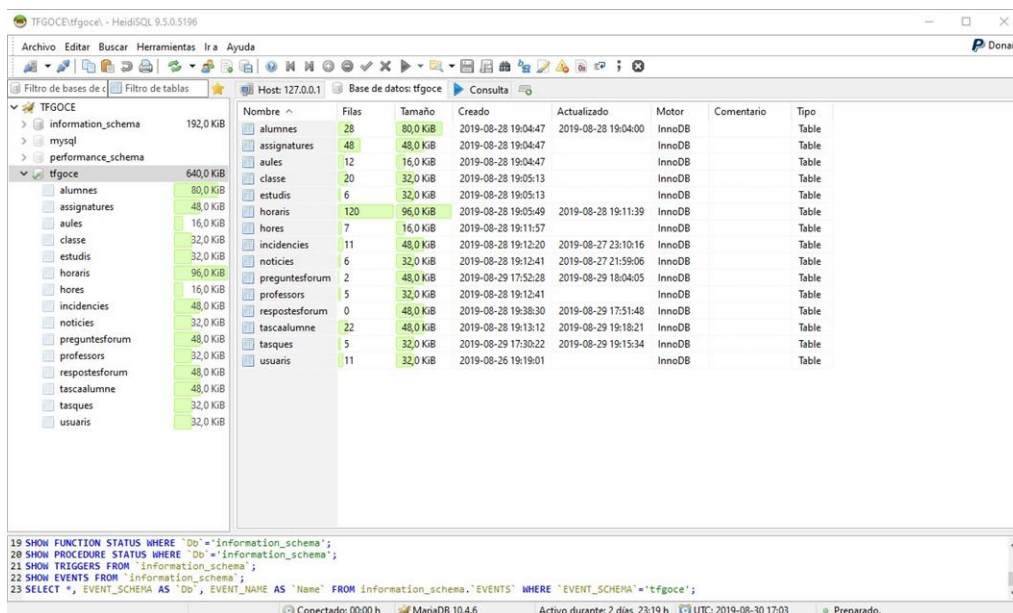


Imatge 8.2: WAMP a la barra d'eines



Imatge 8.3: Desplegable amb eines de WAMP

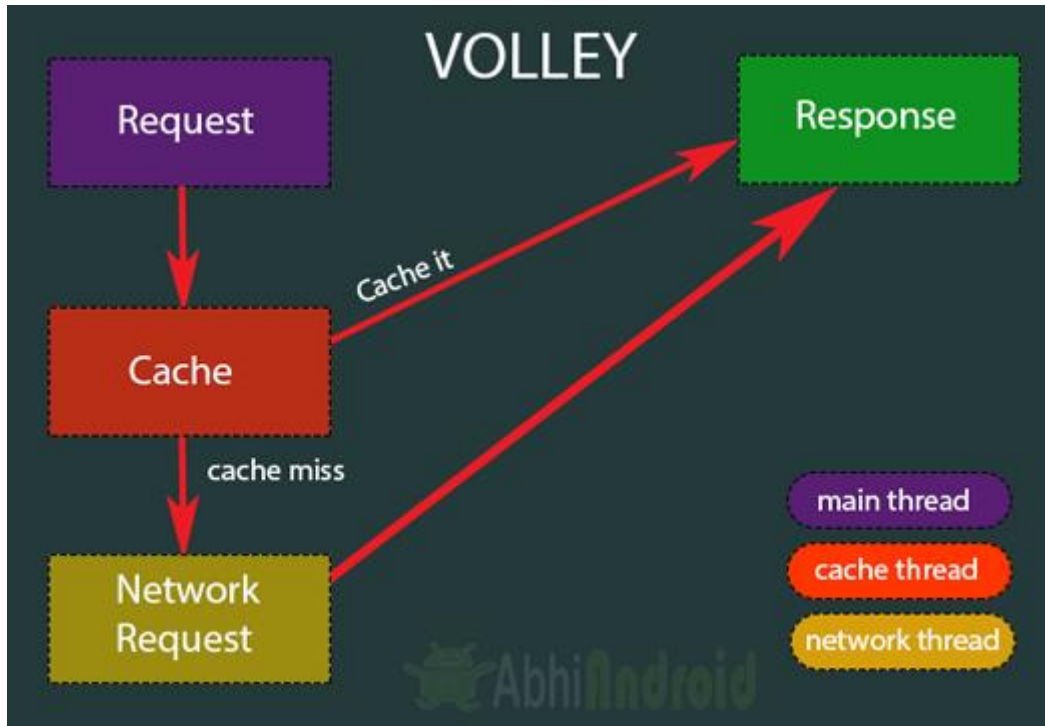
Tot i que WAMPServer té el seu client phpMyAdmin per crear i gestionar bases de dades, vaig optar per provar el MariaDB per aquest projecte. Una vegada instal·lat, ofereix una interfície amb nom HeidiSQL, semblant al phpMyAdmin que fa les mateixes funcions (tot i que HeidiSQL es més fàcil d'entendre que phpMyAdmin).



Imatge 8.4: Interfície HeidiSQL



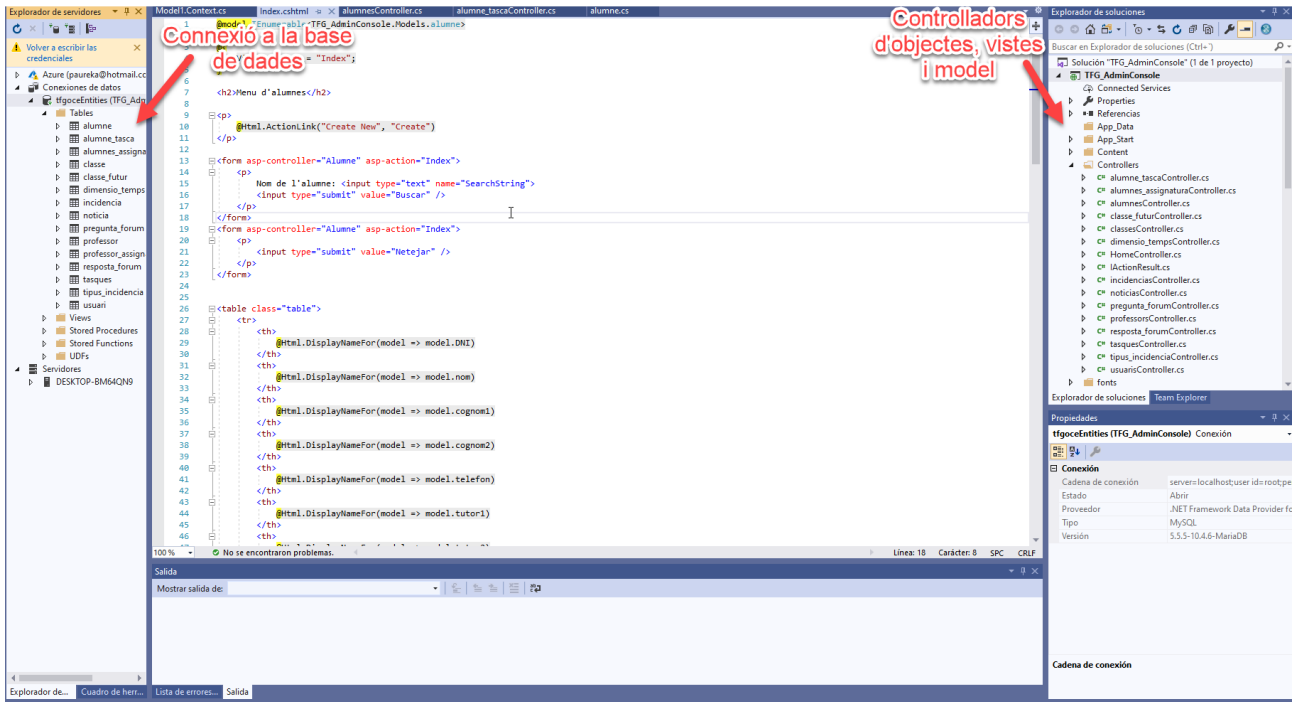
Finalment, per tal de elaborar crides amb objectes JSON als fitxers PHP des d'Android, necessitaria d'una llibreria que em facilités la feina. Vaig trobar-ne una fàcil d'utilitzar i d'entendre i amb forces tutorials per internet: Volley.



Imatge 8.7: Llibreria per Android Volley

Per últim, ens queda decidir en quin entorn programar l'aplicació que utilitzarà l'administrador de l'aplicatiu per gestionar les dades dins la base de dades.

L'IDE Visual Studio proporciona un ampli ventall d'opcions i eines als programadors per codificar en una gran varietat de llenguatges i plataformes. En aquest cas, si volem programar una aplicació ASP.NET MVC (en C#) és una gran opció.



Imatge 8.8: Interfície Visual Studio

Quan s'ha comprovat que tot funcionava, he triat la plataforma de hosting gratuïta 000webhost per pujar la base de dades i els fitxers PHP a internet. D'aquesta manera podria provar la aplicació sense necessitat de estar a la mateixa xarxa que el host.

Una vegada vaig tenir clar quines eines utilitzaria vaig posar-me a programar. Durant aquest temps vaig adonar-me que era tediós per l'usuari proporcionar les credencials sempre que volgués accedir a l'aplicació. Per mirar de solucionar una mica el tema, vaig triar utilitzar les Shared Preferences de l'aplicació per guardar les dades del usuari i realitzar un procés d'autenticació automàtic cada vegada que l'usuari obrís l'aplicatiu.

Les preferències són dades que una aplicació ha de guardar per personalitzar l'experiència del usuari (per exemple: informació personal, dades d'autenticació, ...). Android ofereix un mètode per administrar aquest tipus de dades i aquest són les Shared Preferences. Cada preferència guardada està composta per un identificador únic i un valor associat a ell. Cada dada s'emmagatzema en fitxers XML.

## 9. Implementació i proves

Amb l'anàlisi de les eines fet i decidit quines utilitzar, podia començar a implementar l'aplicació i fer les proves adients a cada part per comprovar el bon funcionament del codi. Com s'ha comentat abans, s'han utilitzat els llenguatges de programació Android, PHP, MySQL i XML.

Alguns dels programes utilitzats en els que aprofundiré en aquest apartat són: WAMPServer, HeidiSQL i Postman. També parlaré de la llibreria Volley i l'aplicatiu de Visual Studio.

En cada apartat s'ha descrit un exemple de proves realitzades per comprovar que cada element funciona correctament.




### 9.1.- WAMPServer

WAMPServer instal·la tot un conjunt de serveis a Windows per funcionar: PHP, Apache i MySQL. També proporciona una eina que es situa a la barra d'eines per tal de gestionar aquests serveis.

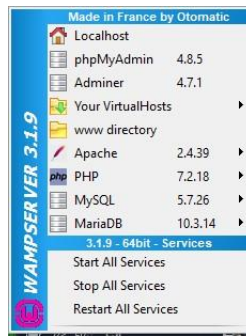


Imatge 9.1: Icona de WAMPServer

La icona d'aquesta eina pot adoptar 3 tipus de colors:

-  Indica que tots els serveis que ofereix (PHP, Apache i MySQL) estan operatius.
-  Indica que algun servei dels que ofereix no està operatiu.
-  Indica que el servidor està parat i cap servei està en funcionament.

Al fer clic sobre la icona apareix un menú per gestionar el programa:

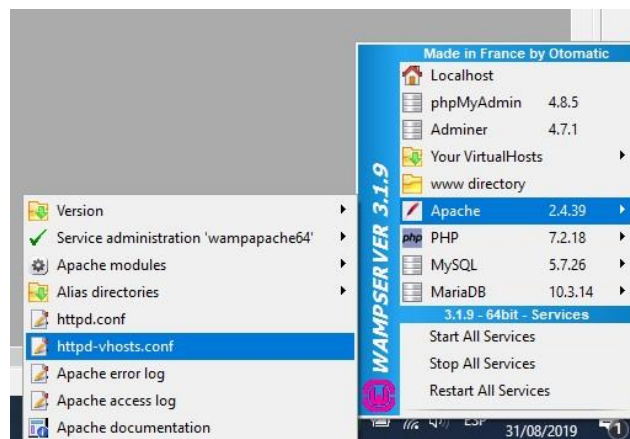


Imatge 9.2: Menú gestor de WAMPServer

D'aquestes opcions només parlaré de les configuracions del servidor, posat que és l'únic apartat amb el que he treballat.

### 9.1.1.- Configuració del servidor Apache

El servidor apache de WAMPServer només escolta consultes que es realitzen en una determinada IP en un cert port. Per tal de fer funcionar el servidor, cal que la IP dels fitxers de configuració del servidor, "apuntin" a la màquina interessada en fer de host. Per fer-ho cal modificar dos fitxers: httpd.conf i httpd-vhosts.conf.



Imatge 9.3: Paràmetres de configuració del servei Apache

### 9.1.1.1.- Fitxer httpd.conf

El fitxer httpd.conf és el principal fitxer de configuració del servidor Apache. Conté varies línies, però pel cas que ens pertoca, només ens interessa la configuració "Listen". Fent una prèvia comprovació de la IP que té el ordinador que actuarà de host dins la xarxa en que es troba, només cal col·locar-la en la línia sense el símbol #. El port no cal tocar-lo, sempre serà el 80.

```
#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
# Change this to Listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#
Listen 192.168.1.102:80
```

### 9.1.1.2.- Fitxer httpd-vhosts.conf

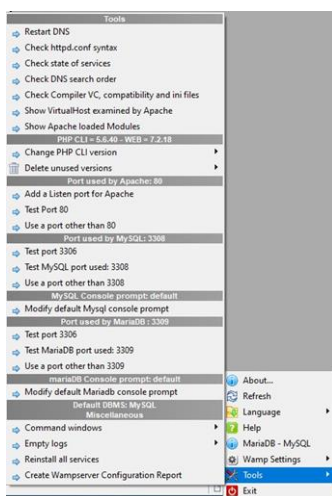
Aquest fitxer també cal modificar-lo per fer funcionar el servidor. Conté paràmetres que serviran per configurar cada un dels servidors virtuals que corrin en la màquina. Cal posar-li la IP trobada en l'apartat anterior i si escau modificar els paràmetres dins el tag <VirtualHost> però no es recomanable (podem canviar el nom del servidor, on es troben els documents, el directori on es troba, permisos, ...).

```
<VirtualHost 192.168.1.102:80>
    ServerName localhost
    ServerAlias localhost
    DocumentRoot "${INSTALL_DIR}/www/usuarios/"
    <Directory "${INSTALL_DIR}/www/">
        Options +Indexes +Includes +FollowSymLinks +MultiViews
        AllowOverride All
        Require all granted
    </Directory>
</VirtualHost>
```

## 9.1.2.- Configuració del servei MySQL

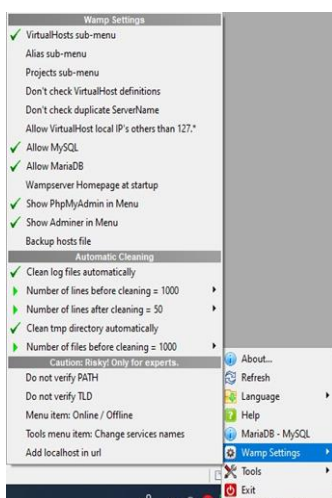
Donat que havia triat utilitzar MariaDB i HeidiSQL en comptes del servei que donava WAMPServer, també calia modificar alguns paràmetres dels serveis MySQL per evitar confrontacions entre els dos. WAMPServer detecta quins dels sistemes anteriors tens instal·lats en el ordinador i permet operar amb els dos (tot i que només necessites un!).

Tant MariaDB com MySQL operen en el mateix port. Per canviar-los cal fer clic dret al icona de WAMP, i seleccionar “Tools”>“Use a port other than X” en alguns dels dos serveis. Quan canviem el port al adient, els serveis estaran totalment operatius i llestos per fer servir.



Imatge 9.4: Configuració MariaDB

Per tal de evitar que la màquina on fem funcionar el host engegui els dos serveis, caldrà especificar quin servei vol que engegui només. Altre vegada, cal fer clic dret al icona de WAMP i seleccionar “WAMP Settings”>“AllowMySQL” o “Allow MariaDB”.



Imatge 9,5: Configuració MariaDB 2

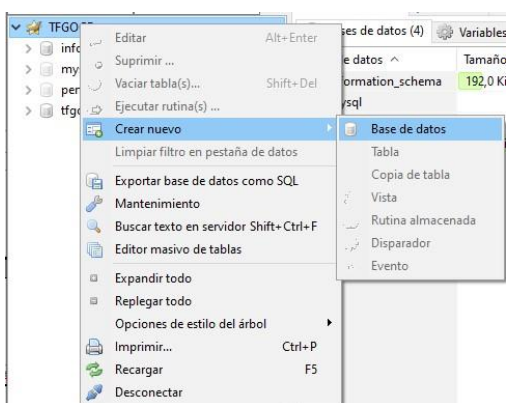
Una vegada s'ha configurat, MariaDB funcionarà amb WAMPServer sense cap problema.

## 9.2.- HeidiSQL

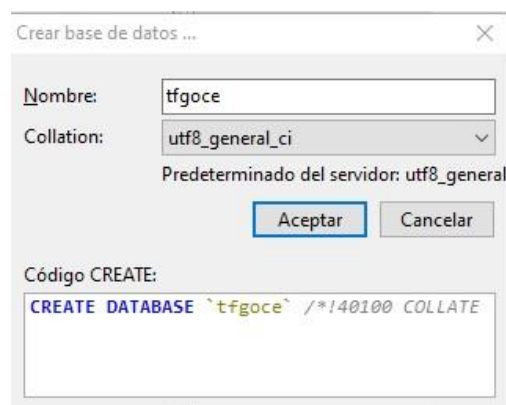
La intenció d'aquest subapartat és la d'explicar el funcionament de la interfície HeidiSQL per crear bases de dades, les seves corresponents taules així com claus primàries o foranes i afegir nous registres a cada una d'elles.

### 9.2.1.- Creació d'una nova base de dades

Un cop s'ha iniciat sessió dins de HeidiSQL, veurem a la dreta les base de dades ja creades en el servei MariaDB. Per crear una nova base de dades, farem clic dret al apartat que conté el nom de la sessió amb la qual hem iniciat, "Crear nuevo" i "Base de datos".



Imatge 9.6: Crear nova base de dades

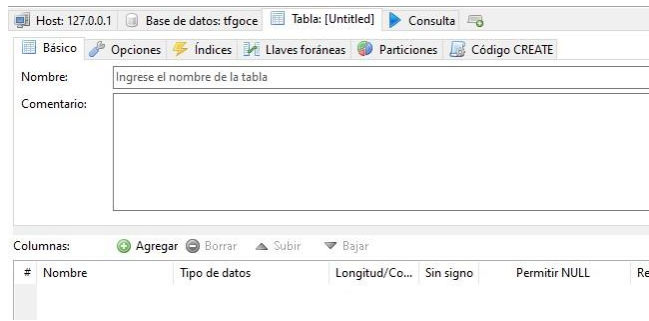


Imatge 9.7: Nom i ordenació de la base de dades

Dins la pantalla de creació, donarem un nou nom a la nostre base de dades i una ordenació (en el meu cas tfgoce i utf8\_general\_ci).

#### 9.2.1.1.- Creació d'una nova taula

Per crear una nova taula a la base de dades, cal seguir un procediment semblant al anterior. Farem clic dret a la nova base de dades, "Crear nuevo" i "Tabla".



Imatge 9.8: Creació nova taula

Tot seguit, cal donar un nom a la taula i si escau, podem posar un comentari per afegir una descripció a aquesta.

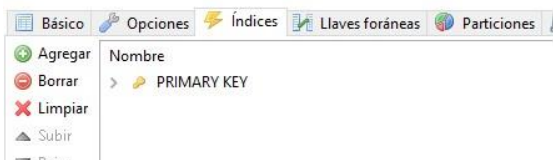
Per afegir un nou camp a la taula caldrà fer clic al botó “Agregar” i automàticament se'n afegirà un.



Imatge 9.9: Afegir nou camp a la taula

Una vegada s'ha afegit, podem modificar-ne els paràmetres (nom, tipus, allargada, amb/sense signe, Null o no, un comentari i un valor predeterminat com per exemple AUTO\_INCREMENT).

Amb els nous valors afegits a la taula caldrà especificar una clau primària per la taula. Per fer-ho ens mourem en el submenú de la taula en el apartat “Índices” i farem clic a “Agregar”. Quan s'hagi afegit l'índex, especificarem que és de tipus PRIMARY i amb el ratolí, mourem el camp de la taula que ens interessa per ser clau primària a sobre l'índex. D'aquesta manera, aquell camp serà guardat com a clau primària de la taula.



Imatge 9.10: Creació nou índex 1



Imatge 9.11: Creació nou índex 2

### 9.2.1.2.- Afegir claus foranes a les taules

Per afegir claus foranes a les taules, cal seleccionar una taula i en els submenús d'aquesta, seleccionar la pestanya “Llaves foráneas”.

En aquest apartat, farem clic “Agregar” i escollirem un nom per a la taula (pot ser qualsevol, tot i que no pot coincidir amb d'altres que també tinguin claus foranes amb la mateixa taula), triarem quin camp de la nostra taula és el que actua de clau forana, escollirem la taula de referència i el camp a que ens referim d'aquella taula.



Imatge 9.12: Afegir claus foranes a les taules

Nombre	Tipo / Longit
PRIMARY KEY	PRIMARY KEY
Clau Foranes	KEY
tutor_id	KEY
Estudi	KEY
estudi_id	KEY
UsuariID	KEY

Imatge 9.13: Nous indexs afegits

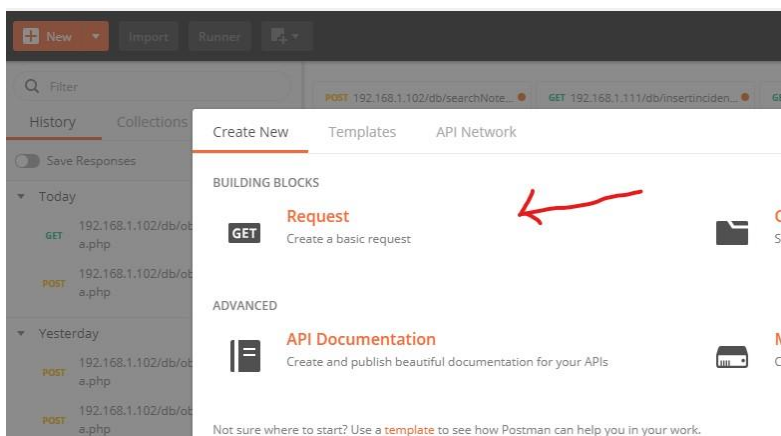
Si tot s'ha fet correctament, la clau forana s'afegirà sense cap problema i la veurem en la pestanya de “Índices”.

## 9.3.- Postman

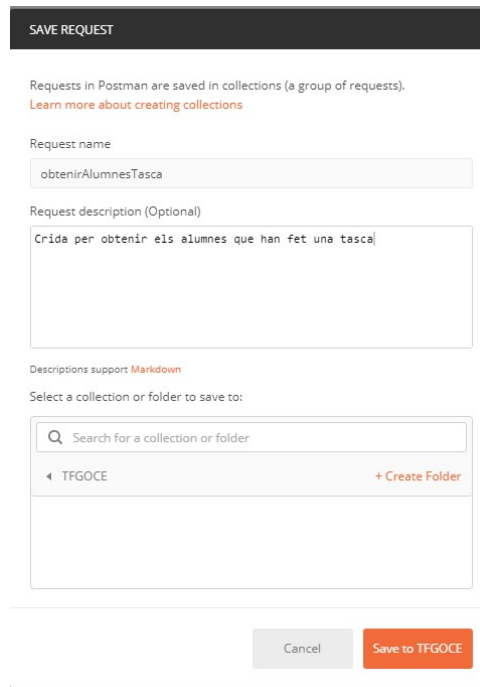
Postman no requereix de configuració prèvia. Una vegada s'han creat i codificat els fitxers PHP per realitzar les tasques associades a cada un d'ells, es comprovava si funcionaven correctament amb Postman.

Per tal de fer-ho, una vegada obert el programa, s'han de seguir els següents passos:

1. Clicar al botó taronja amb les lletres “New” i seleccionar el tipus Request.



Imatge 9.14: Creació nova request al postman



**SAVE REQUEST**

Requests in Postman are saved in collections (a group of requests).  
[Learn more about creating collections](#)

Request name  
obtenirAlumnesTasca

Request description (Optional)  
Crída per obtenir els alumnes que han fet una tasca

Descriptions support [Markdown](#)

Select a collection or folder to save to:

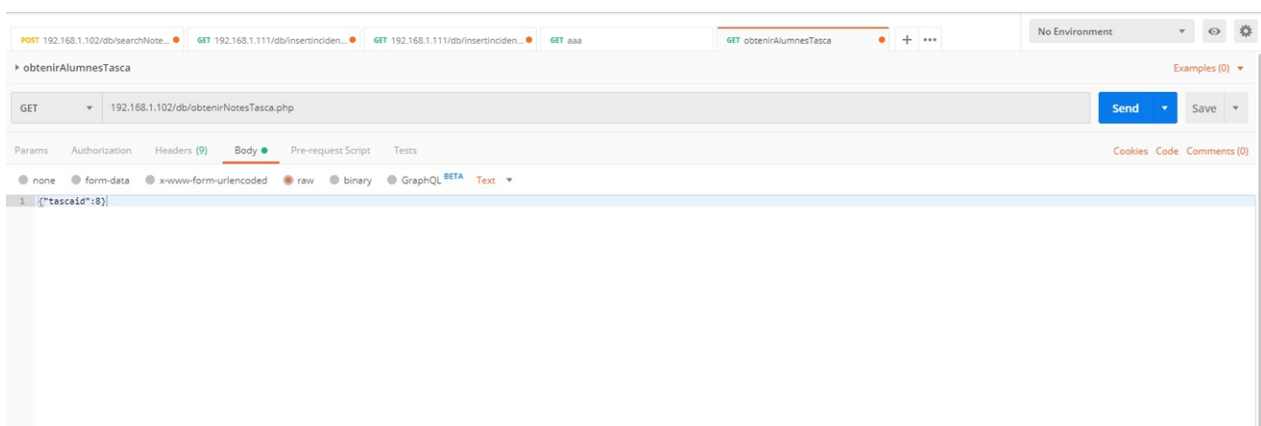
Search for a collection or folder

TFGOCE [+ Create Folder](#)

Cancel [Save to TFGOCE](#)

Imatge 9.15: Paràmetres de la nova request

- Una vegada dins aquesta finestreta, afegirem un nom a la request i una descripció si escau. També caldrà seleccionar una carpeta on guardar la request.
- Una vegada s'ha creat, ja podem treballar amb ella. Ara posarem la url del fitxer a provar dins la caixa amb nom "Enter request URL" i seleccionarem el tipus de request (GET, POST, PUT, ...). Com que treballem amb JSONs, de les pestanyetes d'avall seleccionarem "Body" i afegirem el JSON que actuarà d'entrada al fitxer PHP.



Imatge 9.16: Enviar nova petició al servidor

4. Si tot ha anat correcte veurem com a la caixa de “Response” obtenim el JSON de la resposta.



Imatge 9.17: Resposta del servidor

## 9.4.- Volley

La forma de funcionar de Volley amb les peticions JSON al servidor és la mateixa per a cada crida. Tot i que cada objecte JSON que s'envia pot ser construït diferent (i això si que ha calgut seguir un escenari de prova i error), cada procés de petició és sempre el mateix.

El codi utilitzat per les crides és el següent:

```
JsonObjectRequest jsArrayRequest = new JsonObjectRequest
(Request.Method.POST, session.getUrl(7), request, new Response.Listener<JSONObject>() {
    @Override
    public void onResponse(JSONObject response) {
        try {
            if (response.getInt(KEY_STATUS) == 0) { // Per una resposta correcte
            }
            else {
                Toast.makeText(getApplicationContext(), TEXT_DEL_ERROR, Toast.LENGTH_SHORT).show();
            }
        } catch (JSONException e) {
            e.printStackTrace();
        }
    }
}),
new Response.ErrorListener() {
    @Override
    public void onErrorResponse(VolleyError error) {
        if (error instanceof TimeoutError) {
            Toast.makeText(getApplicationContext(), "TimeOut de la connexiot", Toast.LENGTH_LONG).show();
        } else if (error instanceof NoConnectionError) {
            System.out.println(error.getMessage());
            Toast.makeText(getApplicationContext(), "No hi ha connexio", Toast.LENGTH_LONG).show();
        } else if (error instanceof AuthFailureError) {
            Toast.makeText(getApplicationContext(), "Error d'autenticacio", Toast.LENGTH_LONG).show();
        } else if (error instanceof ServerError) {
            Toast.makeText(getApplicationContext(), "No Hi ha connexio amb el servidor", Toast.LENGTH_LONG).show();
        } else if (error instanceof NetworkError) {
            Toast.makeText(getApplicationContext(), "Error de xarxa", Toast.LENGTH_LONG).show();
        } else if (error instanceof ParseError) {
            System.out.println(error.getMessage());
            Toast.makeText(getApplicationContext(), "Error de traduccio", Toast.LENGTH_LONG).show();
        }
    }
});
// Estableix un timeout més elevat per si li costa contestar una mica
jsArrayRequest.setRetryPolicy(new DefaultRetryPolicy(
    10000,
    DefaultRetryPolicy.DEFAULT_MAX_RETRIES,
    DefaultRetryPolicy.DEFAULT_BACKOFF_MULT));
// Accedeix a la RequestQueue fent servir la classe MySingleton
MySingleton.getInstance(getApplicationContext()).addToRequestQueue(jsArrayRequest);
```

## Aplicatiu per la gestió de centres educatius

La crida consisteix en dos apartats: el tractament de la resposta i el tractament dels errors que poguessin sorgir amb la crida. Cada fitxer PHP retorna un número del estat de la resposta i sempre que val 0 suposa una execució correcte.

Cada crida utilitza una url que es guarda dins l'objecte Sessió i que amb el mètode getUrl() se'n obté un de diferent segons el enter que se li passa.

Per dur un control més exhaustiu amb els errors de l'aplicatiu, cada possible error amb la crida es mostra amb un missatge per pantalla que ajuda a entendre'l i arreglar-lo en un futur.

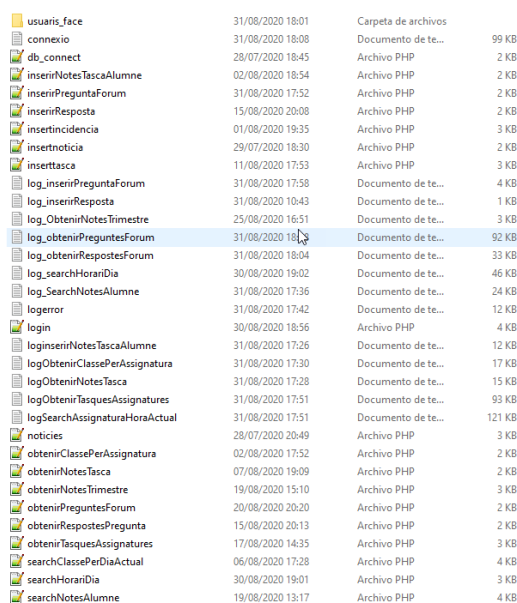
Per acabar, la classe MySingleton és una classe que s'utilitza per guardar les possibles crides que es generin en l'aplicació en una cua per anar executant-les una per una.

### 9.4.2.- Proves

Les proves realitzades sobre la llibreria de Volley s'han anat fent a mesura que s'anaven creant les noves pantalles, on cada una necessitava una consulta diferent.

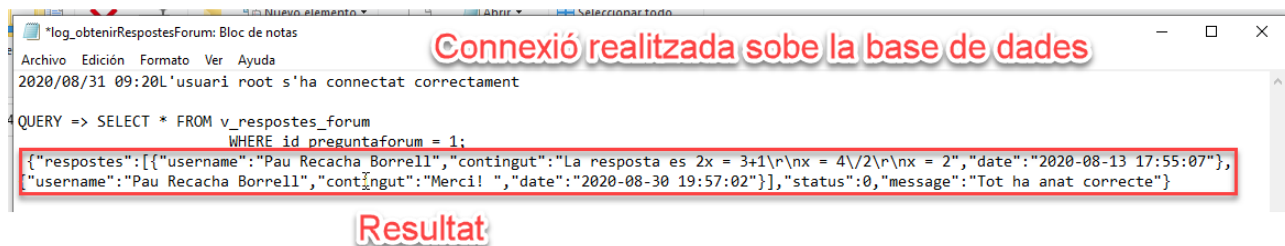
Per veure com es comunica Volley amb els fitxers .php, es fa ús del bloc try-catch i dels missatges per consola.

Per comprovar les comunicacions entre els fitxers .php i la base de dades, cada fitxer .php genera un log amb les consultes que es realitzen i el resultat que es retorna a l'aplicació:



Fitxer	Data i hora	Tipus	Mida
usuarios_face	31/08/2020 18:01	Carpeta de archivos	
connexio	31/08/2020 18:08	Documento de te...	99 KB
db_connect	28/07/2020 18:45	Archivo PHP	2 KB
insertarNotaAlumne	02/08/2020 18:54	Archivo PHP	2 KB
insertarPreguntaForum	31/08/2020 17:52	Archivo PHP	2 KB
insertarResposta	15/08/2020 20:08	Archivo PHP	2 KB
insertarCidencia	01/08/2020 19:35	Archivo PHP	3 KB
insertarNoticia	29/07/2020 18:30	Archivo PHP	2 KB
insertarTasca	11/08/2020 17:53	Archivo PHP	3 KB
log_insertarPreguntaForum	31/08/2020 17:58	Documento de te...	4 KB
log_insertarResposta	31/08/2020 10:43	Documento de te...	1 KB
log_ObtenirNotesTrimestre	25/08/2020 16:51	Documento de te...	3 KB
log_obtenirReguntesForum	31/08/2020 18:14	Documento de te...	92 KB
log_obtenirRespostesForum	31/08/2020 18:04	Documento de te...	33 KB
log_searchHorariDia	30/08/2020 19:02	Documento de te...	46 KB
log_searchNotesAlumne	31/08/2020 17:36	Documento de te...	24 KB
logerror	31/08/2020 17:42	Documento de te...	12 KB
login	30/08/2020 18:56	Archivo PHP	4 KB
logininsertarNotaAlumne	31/08/2020 17:26	Documento de te...	12 KB
logObtenirClassePerAssignatura	31/08/2020 17:30	Documento de te...	17 KB
logObtenirNotesTasca	31/08/2020 17:28	Documento de te...	15 KB
logObtenirTasquesAssignatures	31/08/2020 17:51	Documento de te...	93 KB
logSearchAssignaturaHoraActual	31/08/2020 17:51	Documento de te...	121 KB
noticies	28/07/2020 20:49	Archivo PHP	3 KB
obtenirClassePerAssignatura	02/08/2020 17:52	Archivo PHP	2 KB
obtenirNotesTasca	07/08/2020 19:09	Archivo PHP	2 KB
obtenirNotesTrimestre	19/08/2020 15:10	Archivo PHP	3 KB
obtenirPreguntesForum	20/08/2020 20:20	Archivo PHP	2 KB
obtenirRespostesPregunta	15/08/2020 20:13	Archivo PHP	2 KB
obtenirTasquesAssignatures	17/08/2020 14:35	Archivo PHP	3 KB
searchClassePerDiaActual	06/08/2020 17:28	Archivo PHP	4 KB
searchHorariDia	30/08/2020 19:01	Archivo PHP	3 KB
searchNotesAlumne	19/08/2020 13:17	Archivo PHP	4 KB

Imatge 9.18: Fitxers logs i .php al servidor

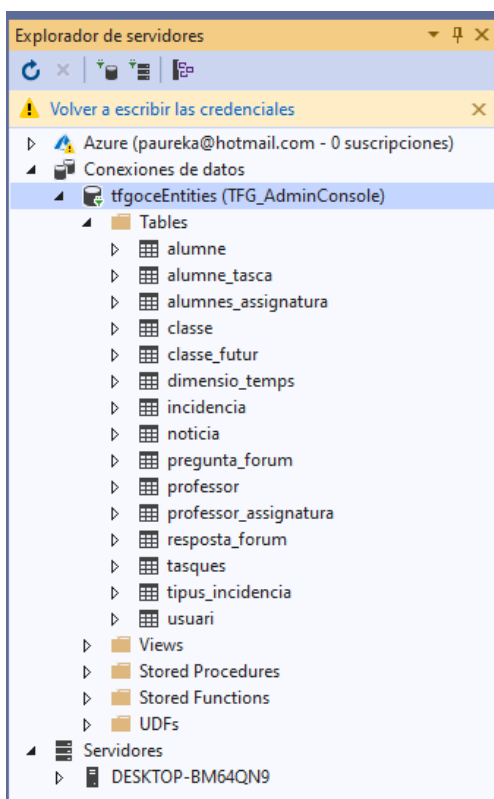


Imatge 9.19: Exemple de resultat d'una consulta

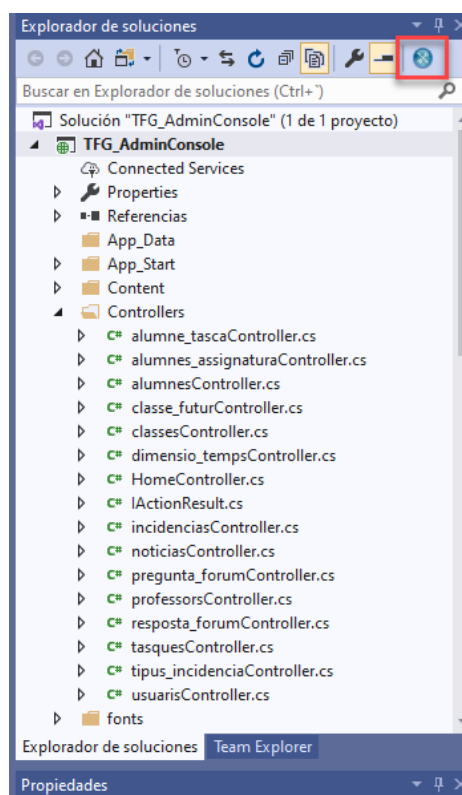
## 9.5.- Visual Studio i ASP.NET MVC en C#

La configuració d'una connexió entre una aplicació .NET amb un servidor MySQL es pot fer tediosa. Per facilitar la comunicació entre els dos, MySQL ofereix un "Wizard" en Visual Studio, que fa tota la feina necessària per tenir un entorn preparat per la programació d'una aplicació que dialogui amb la base de dades.

Al configurar una connexió MySQL en la finestra de connexions a bases de dades de l'entorn, apareix un botó a sobre dels fitxers de la "solució" del programa:



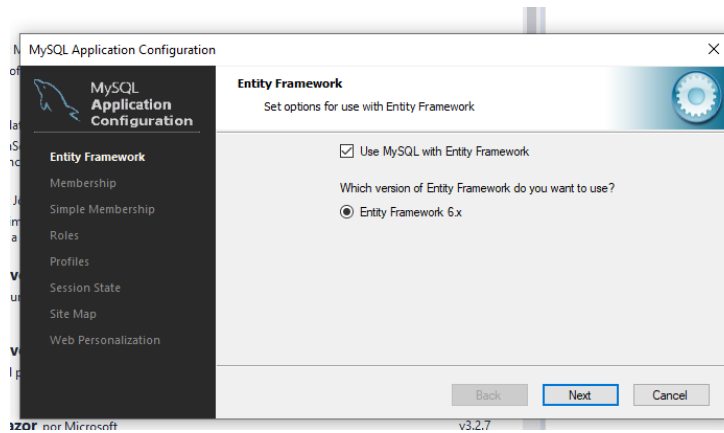
Imatge 9.20: Explorador de servidors del Visual Studio



Imatge 9.21: Explorador de solucions del Visual Studio

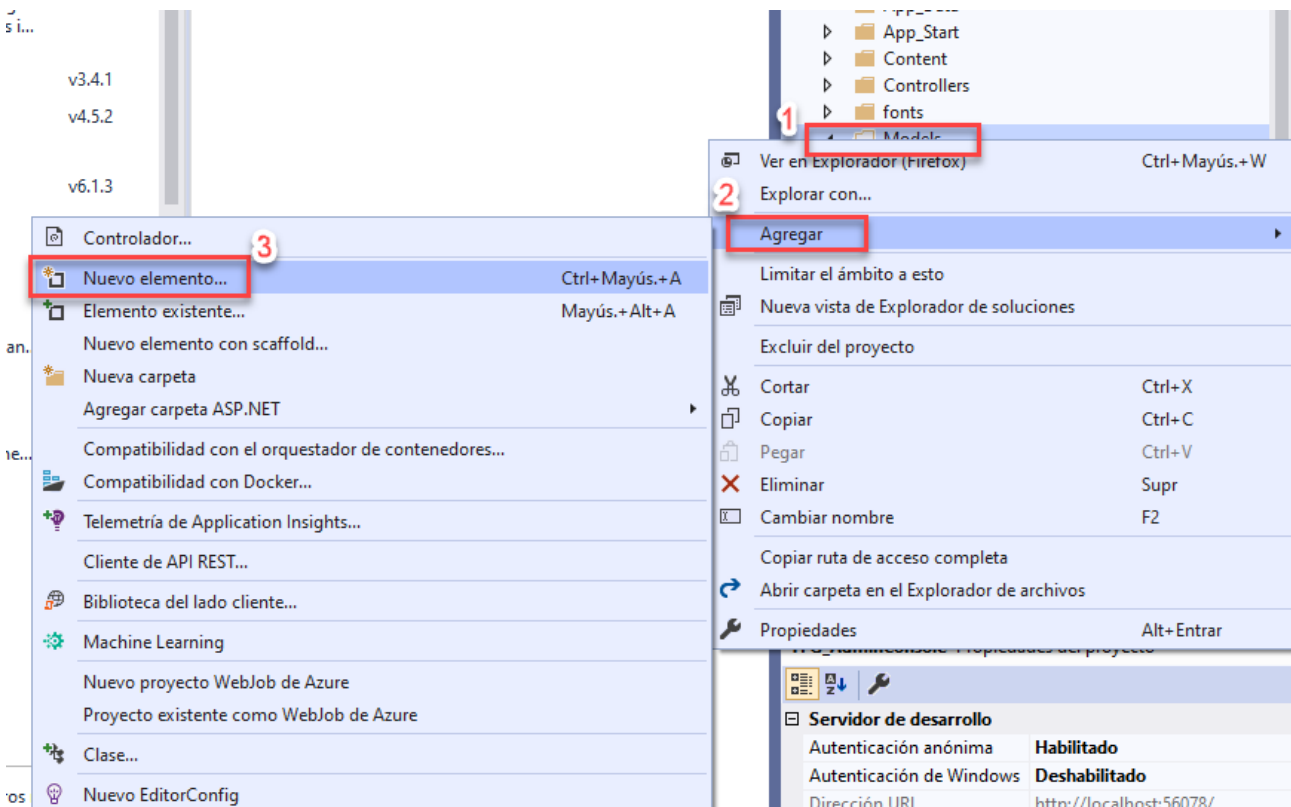
## Aplicatiu per la gestió de centres educatius

Des d'aquest Wizard, s'instal·laran totes les extensions i paquets "NuGet" necessaris per treballar amb una base de dades MySQL. En el meu cas, només fent click a "Next" sense configurar res més, ha funcionat perfectament. De totes maneres, proporciona configuracions extres si l'usuari les necessita.

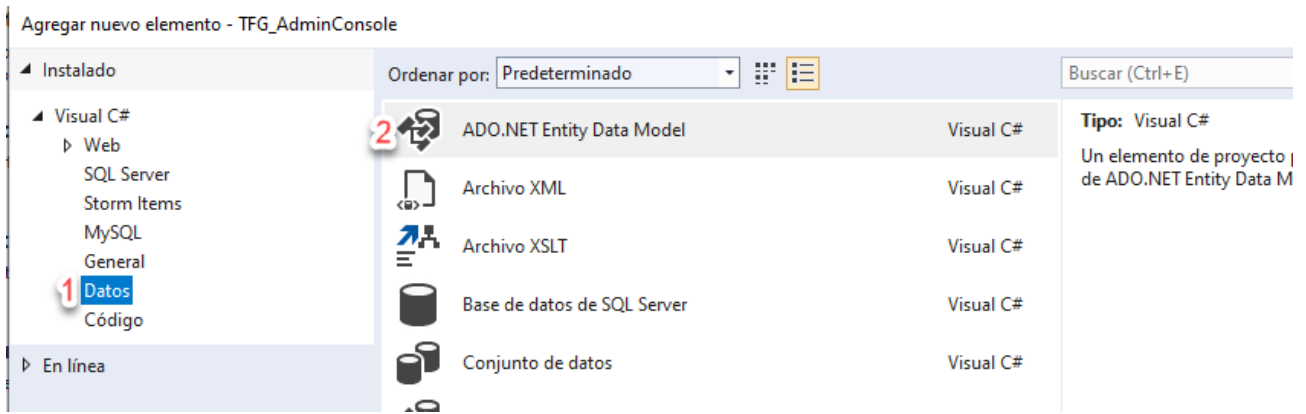


Imatge 9.22: Wizard MySQL per configurar una solució per aplicacions que es connecten a MySQL

Una vegada configurada aquesta part, ens queda incorporar el model a la nostre solució. Per fer-ho, cal seguir els següents passos:



Imatge 9.23: Com afegir un model a la solució (1)



Imatge 9.24: Com afegir un model a la solució (2)

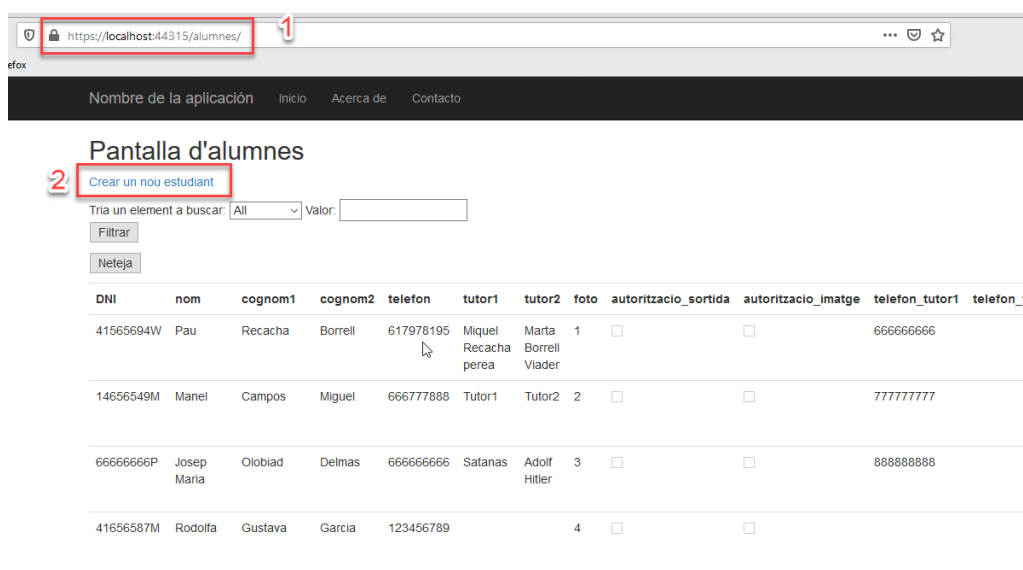
Només caldrà anar fent “Next” en cada pas. Amb això importarem les nostres taules (el model) al nostre projecte (solució).

A partir d’aquí tocarà programar cada part per fer una aplicació robusta sobre el model de la base de dades (afegir, modificar i eliminar dades).

### 9.5.1.- Proves

Per testejar les connexions de l’aplicatiu i el seu funcionament contra la base de dades, s’utilitzen les classes controladors i vistes. Per les proves, s’ha comprovat per cada vista referent a un objecte del model, com funcionava el “Create”, “Delete” i “Modify”.

Una vegada l’aplicació s’està executant, ens dirigim a la URL de la vista que ens interressi, en aquest cas alumnes, i farem click a sobre el link de crear un nou estudiant:



Imatge 9.25: Prova d’afegir alumne a la base de dades

En aquesta pantalla, es poden introduir les dades noves de l’alumne:

The screenshot shows a web form for creating a new student. The form contains several input fields and checkboxes. The fields are: cognom1, cognom2, telefon, tutor1, tutor2, foto, autoritzacio\_sortida (checkbox), autoritzacioimatge (checkbox), telefon\_tutor1, telefon\_tutor2, adreca, codi\_postal, poblacio, mail\_tutor1, and mail\_tutor2. All text input fields contain the placeholder text 'ProvaMemoria'. The 'telefon' and 'telefono\_tutor' fields contain the number '617978195'. The 'codi\_postal' field contains '17257'. The 'foto' field contains '1'. A red rectangular box highlights the 'Crear nou alumne' button at the bottom of the form, with a yellow arrow pointing to it from the left. Below the form, there is a blue link labeled 'Tornar enrere'.

Imatge 9.26: Pantalla creació alumne

Una vegada hem acabat, fem click a sobre el botó de crear. Si anem a la plataforma de HeidiSQL veurem que s'ha creat un nou alumne:

The screenshot shows a table in HeidiSQL with the following data:

9	56172890P	Miguel	Losantos	Rodriguez	111.111.111	(NULL)	(NULL)	(NULL)	0	0	(NULL)	(NULL)	C/
11	41565694W	ProvaMemoria	ProvaMemoria	ProvaMemoria	617.978.195	ProvaMemoria	ProvaMemoria	1	0	1	617.978.195	617.978.195	Pr

Imatge 9.27: Nou alumne a la base de dades

Per les noves funcionalitats afegides a l'aplicatiu (com per exemple, un link a una nova vista), s'ha comprovat que aquesta funció és correctament i no provoques cap parada de codi.

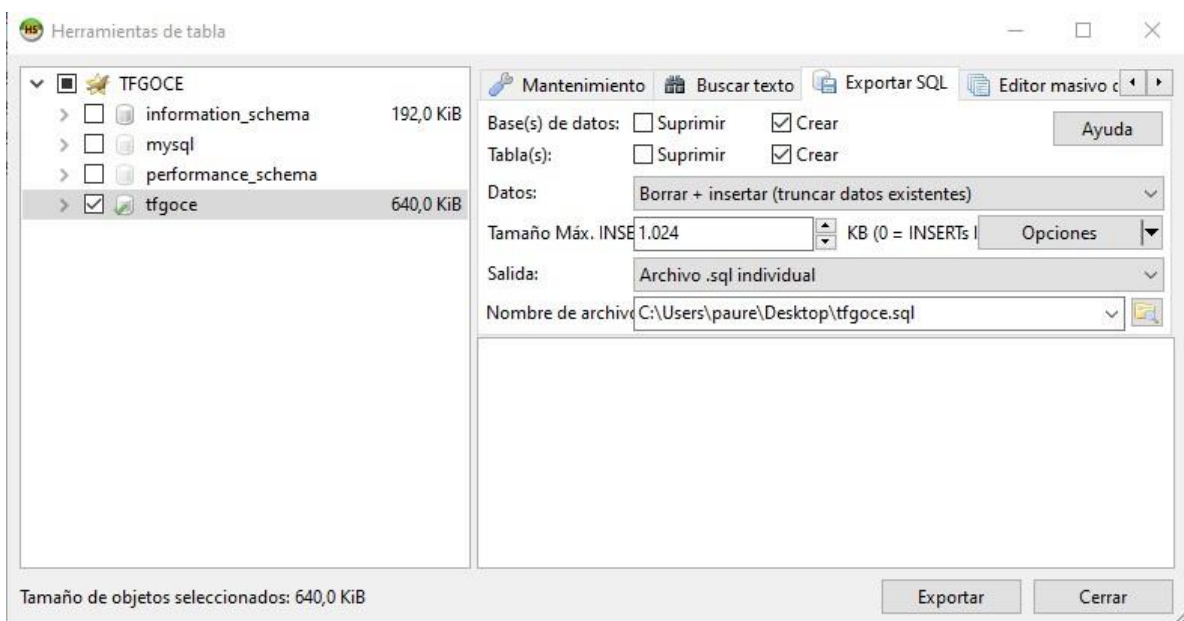


Una vegada pujats els fitxers PHP al servidor, caldrà fer el mateix amb la base de dades. Per fer-ho, tornarem anar al menú principal de la nostra aplicació, farem clic a “+ New database” i proporcionarem un nom per la base de dades, un nom d'usuari i una contrasenya.

Una vegada creada, només quedarà importar la base de dades local i ja tindrem el servidor preparat.

### 9.6.3.- Exportar la base de dades al servidor

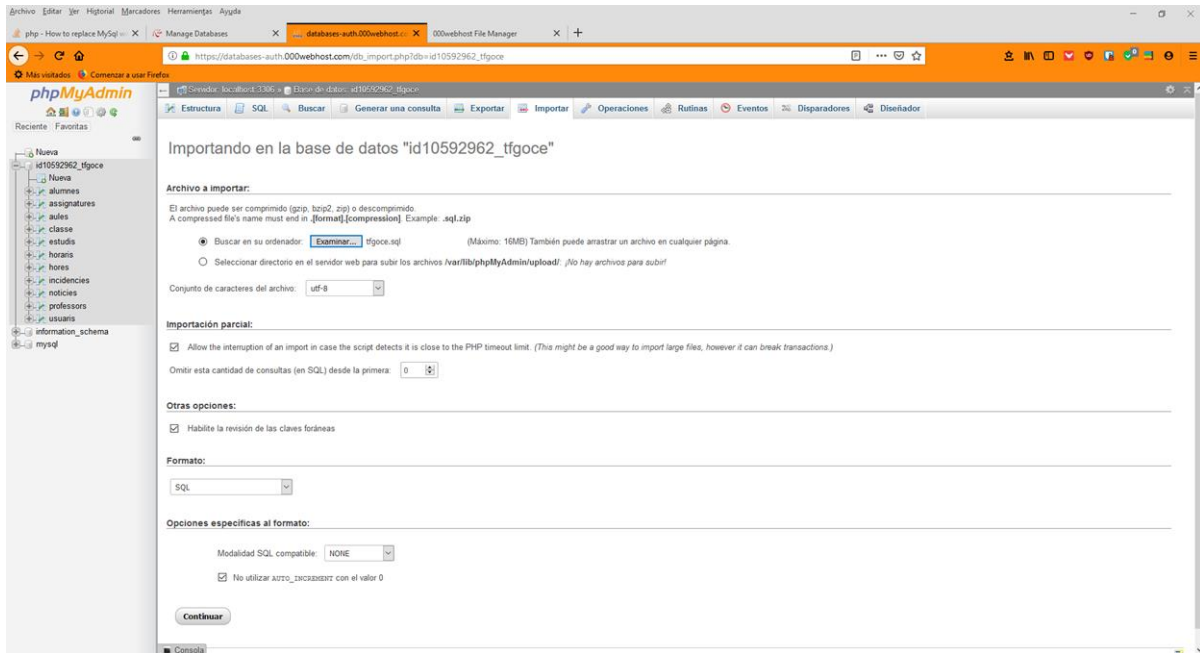
Per tal d'exportar la base de dades, obrirem la interfície HeidiSQL i farem clic dret a la base de dades, “Exportar” i “Exportar base de datos como SQL”.



Imatge 9.29: Exportar base de dades des de HeidiSQL

Dins aquesta pantalla, seleccionarem “Crear” en les caixes de ticks i “Insertar”. Com a sortida seleccionarem “Archivo .sql individual” i per acabar el directori on es guardarà el fitxer .sql resultant.

Ara, tornarem anar a 000webhost, farem clic a “Manage”>”phpMyAdmin” per obrir phpMyAdmin i importar la base de dades. Dins aquesta interfície crearem una nova base de dades i importarem la base de dades a partir del fitxer .sql generat.



Imatge 9.30: Pantalla d'importació d'una base de dades

Si tot ha anat bé, es mostrarà un missatge de confirmació i podrem veure totes les taules carregades a la base de dades nova.

## 9.7.- Generació de vistes i rutines

Per facilitar la feina de l'administrador i fer consultes més simplifiades en els fitxers PHP, s'han afegit una sèrie de vistes i rutines dins la base de dades

### 9.7.1.1.- Vistes

Les vistes creades són les següents:

- v\_alumnes\_assignatures: Vista per recollir les assignatures que fa cada alumne:

personaid	nom	cognoms	estudi	assignatura	aula	curs	nom_assignatura
1	Pau	Recacha Borrell	1rESO	BIO	A	2020-2021	Biologia

Imatge 9.31: Contingut de v\_alumnes\_assignatures

- v\_alumnes\_notes\_trimestrals: Vista per recollir les notes trimestrals de cada alumne-assignatura:

id	nom	nota_primert	nota_segont	nota_tercert	estudi	assignatura	aula	curs
1	Pau Recacha Borrell	(NULL)	(NULL)	(NULL)	1rESO	BIO	A	2020-2021
1	Pau Recacha Borrell	(NULL)	(NULL)	(NULL)	1rESO	MAT	A	2020-2021

Imatge 9.32: Contingut de v\_alumnes\_notes\_trimestrals

- v\_alumnes\_tasca: Vista on es recull la informació sobre les tasques de cada alumne:

id_tasca	id_alumne	titul	descripcio	nota	full_name	estudi	assignatura	aula	curs	observacions	trimestre	valor	tancada
3	1	Prova 2 - Trimestral	Prova 2; Prova trimestral	0,00	Pau Recacha Borrell	1rESO	MAT	A	2020-2021		1	40,00	1

Imatge 9.33: Contingut de v\_alumnes\_tasca

- v\_horari\_alumne: Vista per recollir l'horari dels alumnes:

alumne_id	estudi	assignatura	aula	curs	nom_assignatura	horari
1	1rESO	BIO	A	2020-2021	Biologia	08:15:00
1	1rESO	MAT	A	2020-2021	Matematiques	09:15:00

Imatge 9.34: Contingut de v\_horari\_alumne

- v\_horari\_professor: Vista per recollir l'horari dels professors:

professor_id	estudi	assignatura	aula	curs	nom_assignatura	horari
1	1rESO	MAT	A	2020-2021	Matematiques	09:15:00

Imatge 9.35: Contingut de v\_horari\_professor

- v\_noticies\_assignatura: Vista per recollir les notícies de les assignatures:

data	titol	descripcio	nom	cognoms	estudi	assignatura	curs	aula
2020-07-28 20:36:09	Prova1Titol	Prova1	Marta	Borrell Viader	1rESO	MAT	2020-2021	A

Imatge 9.36: Contingut de v\_noticies\_assignatura

- v\_preguntes\_forum: Vista per recollir les preguntes del fòrum, juntament amb el nom del creador i alguna informació extra:

data	titol	descripcio	nom	cognoms	estudi	assignatura	curs	aula
2020-07-28 20:36:09	Prova1Titol	Prova1	Marta	Borrell Viader	1rESO	MAT	2020-2021	A

Imatge 9.37: Contingut de v\_preguntes\_forum

- v\_preguntes\_forum\_filtrat: Vista on es recullen per cada usuari, les preguntes que pot visualitzar:

user_id	id	text_pregunta	data	id_usuari	titol_pregunta	tag	num_respostes	name	concepte
1	2	PreguntaTest2	2020-08-13 13:15:34	2	PreguntaTest2Titol	General	0	Marta Borrell Viader	General
2	2	PreguntaTest2	2020-08-13 13:15:34	2	PreguntaTest2Titol	General	0	Marta Borrell Viader	General

Imatge 9.38: Contingut de v\_preguntes\_forum\_filtrat

- v\_professors\_assignatures: Vista per recollir les assignatures que fa cada professor:

personaid	nom	cognoms	estudi	assignatura	aula	curs	nom_assignatura
1	Marta	Borrell Viader	1rESO	MAT	A	2020-2021	Matematiques
1	Marta	Borrell Viader	2nESO	BIO	B	2019-2020	Biologia

Imatge 9.39: Contingut de v\_professors\_assignatures

- v\_respostes\_forum: Vista per recollir les respostes a preguntes del fòrum, juntament amb el nom i alguna informació extra:

personaid	nom	cognoms	estudi	assignatura	aula	curs	nom_assignatura
1	Marta	Borrell Viader	1rESO	MAT	A	2020-2021	Matematiques
1	Marta	Borrell Viader	2nESO	BIO	B	2019-2020	Biologia

Imatge 9.40 Contingut de v\_respostes\_forum

- v\_usuari\_assignatures: Vista per recollir les assignatures que fa cada usuari:

personaid	nom	cognoms	estudi	assignatura	aula	curs	nom_assignatura
1	Marta	Borrell Viader	1rESO	MAT	A	2020-2021	Matematiques
1	Marta	Borrell Viader	2nESO	BIO	B	2019-2020	Biologia

Imatge 9.41: Contingut de v\_usuari\_assignatures

### 9.7.1.2.- Rutines

Les rutines creades són les següents:

- **update\_nota\_trimestre**(*estudi, assignatura, aula, curs, id\_alumne, trimestre, nota*): La rutina actualitza la nota trimestral d'un alumne-assignatura segons els paràmetres passats.
- **r\_usuari\_alumne\_professor**(*id\_usuari*): La rutina retorna 0 si l'usuari passat no és alumne ni professor, 1 quan és professor i 2 quan és alumne.
- **r\_passar\_alumnes\_curs**(*any\_de\_calcul*): La rutina s'encarrega, a partir del any que se li ha passat i la taula de mapeig "classe\_futur", assignar les noves assignatures als alumnes que han superat les antigues. Si l'alumne no ha superat l'assignatura, se li torna a assignar la mateixa però pel curs vinent.
- **r\_get\_n\_alumnes\_classe**(*estudi, assignatura, aula, curs*): La rutina retorna el número d'alumnes que pertanyen a la classe en qüestió.
- **r\_end\_curs**(): Rutina pensada per gestionar el final de curs. Primer llença la rutina "r\_assignatura\_final" i després la de "r\_passar\_alumnes\_curs". D'aquesta manera no cal llençar les 2 a mà.
- **r\_assignatures\_trimestre\_nota**(*estudi, assignatura, aula, curs, trimestre*): La rutina s'encarrega d'actualitzar les notes trimestrals d'una classe en funció del trimestre passat per paràmetre. Per fer-ho agafa les notes de les tasques ja tancades per aquell trimestre i calcula les notes en funció del pes de cada una.
- **r\_assignatura\_final**(): La rutina s'encarrega de trobar per cada classe dins el sistema, la nota final per cada alumne que la estigui fent.
- **get\_horari\_assignatura**(*estudi, assignatura, aula, curs*): La rutina retorna el camp "horariX" de la taula "classe" en funció del dia en que es crida i l'assignatura passada.
- **get\_horari\_dia**(*estudi, assignatura, aula, curs, weekday\_name*): Igual que la rutina "get\_horari\_assignatura" però en aquest cas el dia és passat per paràmetre.
- **r\_assignatura\_nota**(*estudi, assignatura, aula, curs*): Calcula les notes de tots els trimestres donada una assignatura.

### 9.7.1.3.- Proves

Les vistes afegides, s'han provat amb simples consultes SELECT.

Per altre banda, les rutines s'ha comprovat que tenien un comportament adequat des de la interfície del HeidiSQL, mitjançant el botó que permet executar-les (també es pot realitzar un "CALL" amb els paràmetres que toquin).

## Aplicatiu per la gestió de centres educatius

Per exemple, per testejar la rutina `r_assignatura_final()`, partim d'un escenari principal amb 3 tasques puntuades per cada trimestre d'una assignatura per l'alumne "Pau Recacha Borrell":

tfgoce.tasques: 4 filas en total (aproximadamente)

	descripcio	estudi	assignatura	aula	curs	id_professor	titol	tancada	trimestre	valor
1	Examen 1r trimestre	1rESO	MAT	A	2020-2021	1	Examen 1	1	1	50,00
2	Examen 2n trimestre	1rESO	MAT	A	2020-2021	1	Examen 2	1	2	50,00
3	Examen 3r trimestre	1rESO	MAT	A	2020-2021	1	Examen 3	1	3	50,00
8	Examen 1r trimestre 2	1rESO	MAT	A	2020-2021	1	Examen 1 (2)	1	1	50,00

Imatge 9.42: Tasques per l'assignatura 1rESO MAT A 2020-2021

tfgoce.alumne\_tasca: 4 filas en total (aproximadamente)

id_tasca	id_alumne	observacions	nota
1	1	(NULL)	6,00
3	1	(NULL)	7,00
8	1	(NULL)	1,59
2	1	(NULL)	3,00

Imatge 9.43: Avaluació per les tasques per l'alumne amb id 1

La rutina `r_end_curs()` serveix per provar les 2 rutines principals: `r_assignatura_final()` i `r_passar_alumnes_curs()`. De totes maneres, per poder executar aquesta rutina, les notes dels trimestres han d'estar entrades.

Per fer-ho executem la rutina `r_assignatura_notas()` per la assignatura en qüestió:

id_alumne	nota_primert	nota_segont	nota_tercert	nota_final	estudi	assignatura	aula	curs
1	(NULL)	(NULL)	(NULL)	(NULL)	1rESO	BIO	A	2020-2021
1	3,80	1,50	3,50	(NULL)	1rESO	MAT	A	2020-2021
2	(NULL)	(NULL)	(NULL)	(NULL)	1rESO	BIO	A	2020-2021
2	0,00	0,00	0,00	(NULL)	1rESO	MAT	A	2020-2021
3	0,00	0,00	0,00	(NULL)	1rESO	MAT	A	2020-2021
4	0,00	0,00	0,00	(NULL)	1rESO	MAT	A	2020-2021
5	0,00	0,00	0,00	(NULL)	1rESO	MAT	A	2020-2021
6	0,00	0,00	0,00	(NULL)	1rESO	MAT	A	2020-2021
7	0,00	0,00	0,00	(NULL)	1rESO	MAT	A	2020-2021
8	0,00	0,00	0,00	(NULL)	1rESO	MAT	A	2020-2021
9	0,00	0,00	0,00	(NULL)	1rESO	MAT	A	2020-2021

Imatge 9.44: Notes trimestrals per l'alumne en qüestió

Ara, si executem la rutina `r_end_curs()`, veurem que s'assigna una nota final al alumne per cada assignatura que ha cursat, i se li assigna una nova classe en funció de la taula `classe_futur` (aquelles notes no entrades o amb valor de NULL, no afectaran al sistema).

## Aplicatiu per la gestió de centres educatius

ID Alumne	Nota_Primer	Nota_Segon	Nota_Tercer	Nota_Final	Estudi	Assignatura	Nota	Curs
1	(NULL)	(NULL)	(NULL)	(NULL)	1rESO	BIO	A	2020-2021
1	3.80	1.50	3.50	2.64	1rESO	MAT	A	2020-2021
1	(NULL)	(NULL)	(NULL)	(NULL)	1rESO	MAT	A	2021-2022

Imatge 9.45: Nova assignatura per l'alumne "Pau Recacha Borrell"

Veiem com en Pau tornarà a fer Matemàtiques de 1rESO l'any que ve.

## 10. Implantació i resultats

### 10.1.- Procés de desenvolupament

Primer de tot es va estudiar quines necessitats tindrien els usuaris que utilitzessin l'aplicació, es varen establir uns objectius del projecte i es va estudiar la viabilitat del conjunt.

Tot seguit es va estudiar i dissenyar (a partir dels casos d'ús i requisits del sistema) les pantalles que tindria l'aplicació així com les funcionalitats de cada una d'elles.

Amb les necessitats es va fer un petit estudi sobre alguna de les eines a utilitzar en el projecte així com els llenguatges de programació més adients.

Una vegada es va considerar acabat l'estudi, es va començar a implementar l'aplicació amb la metodologia mostrada en l'apartat 3. Quan una interfície era acabada s'utilitzava el mòbil de casa i el Postman per comprovar que tot funcionés correctament.

Un cop es va considerar que les funcionalitats de l'aplicació estaven fetes, es va dedicar uns dies a polir el disseny d'aquestes.

Per acabar es va dedicar unes setmanes per redactar la memòria.

### 10.2.- Resultats obtinguts

#### 10.2.1.- Aplicació Android

##### 10.2.1.1.- Pantalla principal

Al engegar l'aplicació sense haver fet login anteriorment en un plaç d'uns dies, s'obre la pantalla de login:



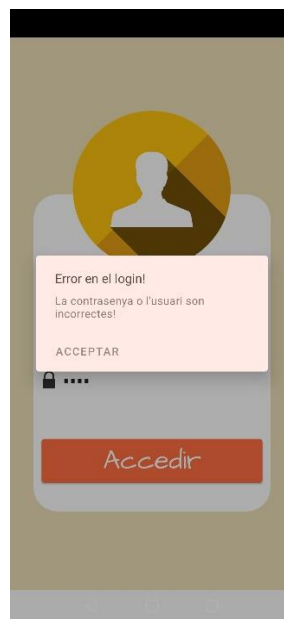
Imatge 10.1: Pantalla login



Imatge 10.2: Pantalla carga

## Aplicatiu per la gestió de centres educatius

Dins aquesta pantalla, podrem insertar les nostres credencials. Si són les correctes es carregarà la pantalla principal. En cas d'haver introduït unes credencials errònies, es mostrarà per pantalla el següent error:



Imatge 10.3: Error en el login!

Internament, l'aplicació recull les assignatures relacionades amb l'usuari per guardar-les i evitar una segona consulta a la base de dades.

### 10.2.1.2.- Pantalla principal (alumne)

La pantalla principal de l'alumne es carreguen les assignatures que fa en una llista clicable. A la part superior es mostra el seu nom complet i un botó per fer "logout". A la part inferior esquerra, es mostren dos botons; un per accedir al fòrum i un altre per introduir informació per enviar en un mail:



Imatge 10.4: Pantalla inicial alumne

Si fem "click" al botó amb el símbol de carta apareix el següent popup:



Imatge 10.5: Pantalla inicial alumne (popup email)

Si fem "click" al botó amb una graella, accedirem al fòrum.

### 10.2.1.3.- Pantalla principal (professor)

La pantalla de professor es força semblant a la d'alumne però difereix en el contingut inicial que es mostra. El professor veurà dos botons per crear noves notícies o noves tasques. Depenent del botó que premi, accedirà a una pantalla on haurà d'introduir dades diferents.

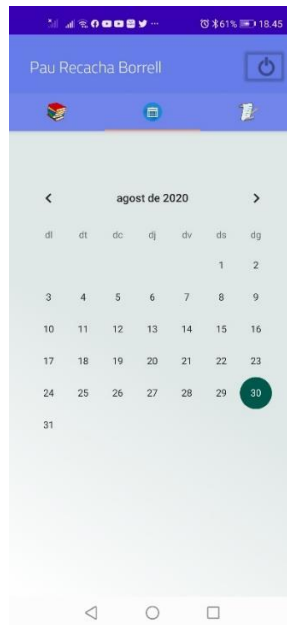


Imatge 10.6: Pantalla inicial professor

El funcionament dels dos botons situats a la part inferior esquerra de la pantalla és el mateix que pel cas de l'alumne.

### 10.2.1.4.- Pantalla calendari (alumne/professor)

La pantalla de calendari és igual pels dos tipus d'usuaris. Es mostra un calendari simple des d'on l'usuari pot navegar. Si prem a un dia en concret, es mostrarà l'horari que correspon a aquell dia:



Imatge 10.7: Pantalla calendari

#### 10.2.1.5.- Pantalla horari dia (alumne/professor)

Dins aquesta pantalla es carregaran en una llista ordenada per hora, les assignatures que toca fer aquell dia seleccionat en la pantalla de calendari:



Imatge 10.8: Pantalla horari dia 27/8

Si l'usuari desitja moure's pels dies, ho pot fer des de la barra inferior de la pantalla:



Imatge 10.9: Pantalla horari dia 28/08

#### 10.2.1.6.- Pantalla horari de classe (professor)

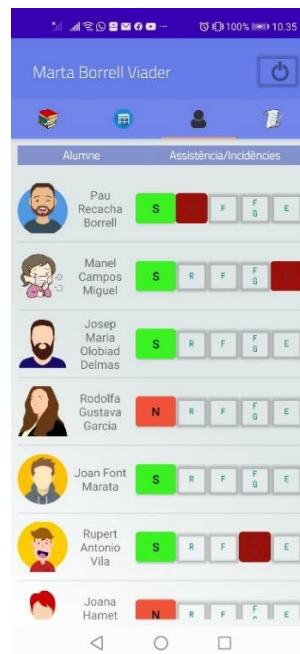
Aquesta pantalla es exclusiva pels professors i funciona segons la hora actual. Si la hora actual coincideix amb un rang d'hores d'una classe que imparteix, es carregaran els alumnes en pantalla:



Imatge 10.10: Pantalla horari assistència

## Aplicatiu per la gestió de centres educatius

Com podem veure, es una llista amb cada alumne de la classe. En ella hi visualitzem la foto del alumne, el nom de l'alumne i una sèrie de botons. En aquest ordre, la funcionalitat dels botons són: canviar assistència a classe, afegir retard, afegir falta, afegir falta greu i afegir expulsió:



Imatge 10.11: Pantalla horari assistència

Si el professor prem a un botó de les incidències, s'obrirà una "popup" per a que aquest pugui entrar una observació a la incidència que esta afegint al alumne:



Imatge 10.12: Pantalla horari assistència (observacions)

## Aplicatiu per la gestió de centres educatius

Si el professor necessita ampliar la imatge del alumne per veure millor la foto d'aquest, només ha de prémer-la:



Imatge 10.13: Pantalla horari assistència

### 10.2.1.7.- Pantalla tasques (alumne)

En aquesta pantalla l'alumne podrà veure (per les assignatures que esta cursant actualment, no les que ja hagi fet), un llistat de les tasques puntuades que hi ha per cada assignatura. Si no hi ha tasques, no se'n mostrarà cap.

El llistat es una triple llista, on el pare és l'assignatura, el fill1 és el trimestre i el fill2 és la tasca. Cada fill tindrà la nota assignada a la seva dreta si el professor ha puntuat el fill en qüestió:



Imatge 10.14: Pantalla tasques (visió alumne)

#### 10.2.1.8.- Pantalla tasques (professor)

La pantalla de tasques pel professor és similar a la de l'alumne, però el professor (a part de no tenir notes en cada fill) té la funcionalitat de afegir notes en cada tasca:

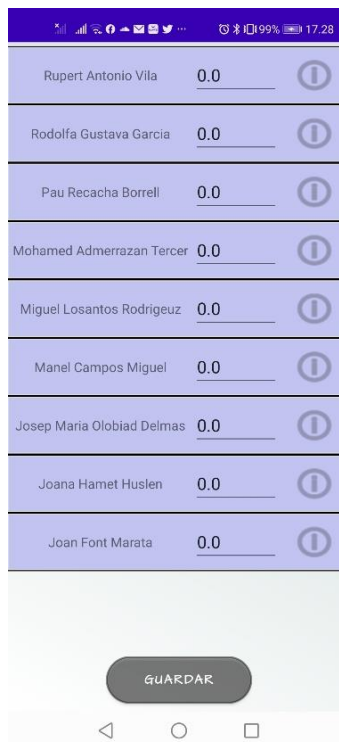


Imatge 10.15: Pantalla tasques (visió professor)

Si prem sobre una tasca, s'obrirà la pantalla per puntuar-la. Si manté el dit sobre un trimestre, es mostraran les notes trimestrals per aquell trimestre en qüestió. En els dos casos, la pantalla que es carrega es la mateixa però les dades que es mostren corresponen a conceptes diferents.

### 10.2.1.9.- Pantalla puntuació (professor)

Segons l'element clickat en la pantalla de tasques, es carregaran un llistat d'alumnes en pantalla per a que el professor pugui puntuar-les:



Imatge 10.16: Pantalla puntuació

En cas que la tasca o trimestre ja estigui puntuat/puntuada, es mostraran les notes assignades en el seu moment.

Si el professor vol afegir un comentari a la nota, només ha de fer click a la “i”:

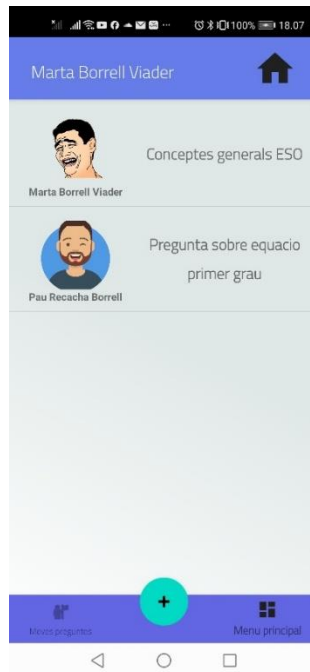


Imatge 10.17: Pantalla puntuació (observacions)

Quan es prem el botó de guardar, la tasca es tanca i es guarden les notes a la base de dades. És un pas fràgil i perillós però que es pot solucionar amb l'administrador en cas que el professor s'equivoqui. Mentre les notes siguin borrades per aquella tasca i la tasca s'obri (el flag de "tancada" passi a 1), l'aplicació permetrà tornar a puntuar-la.

#### 10.2.1.10.- Pantalla principal del fòrum (alumne/professor)

Les pantalles del fòrum són iguals per cada usuari. En la pantalla inicial es mostren les preguntes que l'usuari té capacitat de veure (segons el concepte de la pregunta):

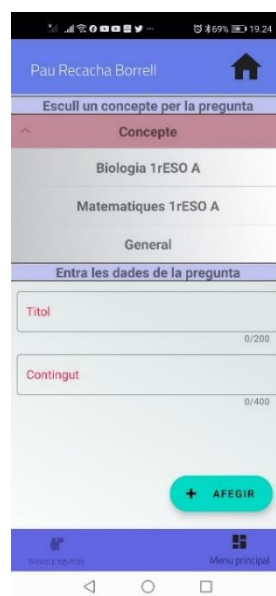


Imatge 10.18: Pantalla principal fòrum

A la part inferior hi ha una barra que permet desplaçar-se entre les pantalles del fòrum. Per accedir a una pregunta, simplement s'ha de fer click a sobre al títol d'aquesta.

#### 10.2.1.11.- Pantalla afegir pregunta (alumne/professor)

La pantalla d'afegir pregunta al fòrum consta de 2 fragments; en un s'escull el concepte de la pregunta i en l'altre el contingut d'aquesta:

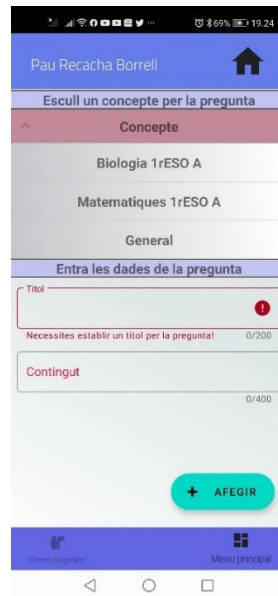


Imatge 10.19: Pantalla afegir pregunta fòrum

## Aplicatiu per la gestió de centres educatius

Cal saber que el concepte és un element que farà que uns alumnes/professors vegin o no la pregunta (el concepte es basa en les assignatures que fa l'usuari. També hi ha un concepte de "General").

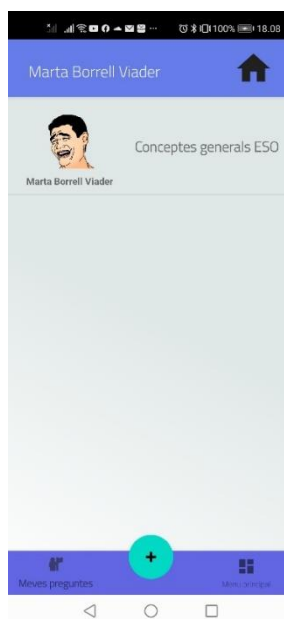
Una vegada entrades les dades, si premem "+Afegir" s'afegirà la pregunta a la base de dades. En cas que l'usuari s'hagi deixat algun camp per emplenar, l'aplicació li mostrarà en quin camp falten dades:



Imatge 10.20: Pantalla afegir pregunta fòrum (error)

### 10.2.1.12.- Pantalla meves preguntes (alumne/professor)

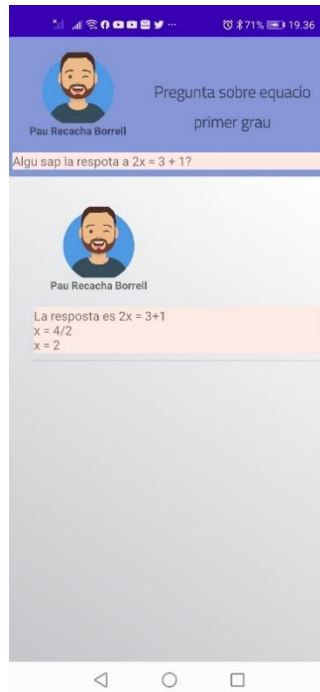
Aquesta pantalla és igual que la principal del fòrum però es filtren les preguntes per les creades pel usuari:



Imatge 10.21: Pantalla fòrum (preguntes usuari)

### 10.2.1.13.- Pantalla respostes (alumne/professor)

En aquesta pantalla es mostraran en una llista les respostes que ha rebut la pregunta a la qual s'ha accedit des de la pantalla principal del fòrum:



Imatge 10.22: Pantalla respostes

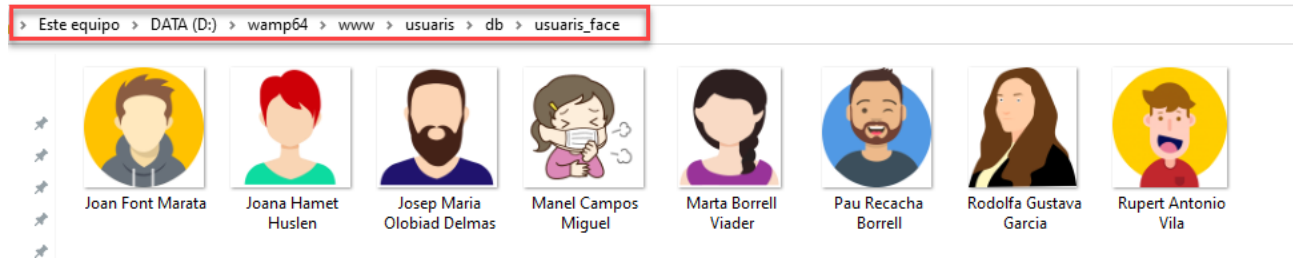
Si es vol afegir una resposta, només s'haurà de prémer el botó amb un "+" i apareixerà una "popup" amb els camps a emplenar:



Imatge 10.23: Pantalla respostes (nova resposta)

## 10.2.2.- Imatges dels alumnes/professors

Per tal de carregar les cares dels usuaris en l'aplicació, les fotos es guarden a la màquina del servidor i l'aplicació les busca en funció del nom de la persona:



Imatge 10.24: Localització de les imatges dels usuaris

## 10.2.3.- Aplicació ASP.NET

L'aplicació .NET es molt bàsica i permet modificar, afegir i borrar registres de la base de dades. Quan obrim l'aplicació veiem la següent pantalla:



Imatge 10.25: Pantalla principal de l'aplicatiu

Quan es prem un link, la pantalla és igual per cada objecte. En el cas de la llista d'estudiants trobem el següent:

Pantalla d'alumnes

[Crear un nou estudiant](#)

Tria un element a buscar: All Valor:

DNI	nom	cognom1	cognom2	telefon	tutor1	tutor2	foto	autoritzacio_sortida	autoritzacio_imatge	telefon_tutor1	telefon_tutor2	adreca	codi_postal	poblacio	mail_tutor1	mail_tutor2
41565694W	Pau	Recacha	Borrell	617978195	Miquel Recacha perea	Marta Borrell Viader	1	<input type="checkbox"/>	<input type="checkbox"/>	666666666		C/Andreu Sabat, 59	17257	Torroella de Montgri	mrecacha@gmail.com	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
14656549M	Manel	Campos	Miguel	666777888	Tutor1	Tutor2	2	<input type="checkbox"/>	<input type="checkbox"/>	777777777		C/Independencia Molt Aviat, 77	14567	Girona		<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
66666666P	Josep Maria	Olobiad	Delmas	666666666	Satanas	Adolf Hitler	3	<input type="checkbox"/>	<input type="checkbox"/>	888888888		C/infern 122	17666	Infern	infern@gmail.com	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
41656587M	Rodolfo	Gustava	Garcia	123456789			4	<input type="checkbox"/>	<input type="checkbox"/>			C/Plaça Madrid	17654	Madrid		<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
54861953I	Joan	Font	Marata	897899123			5	<input type="checkbox"/>	<input type="checkbox"/>			C/Sabatia, 11	893123	Tarrasa		<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
13987452K	Rupert	Antonio	Vila	972456789				<input type="checkbox"/>	<input type="checkbox"/>			C/IDK, 13	18999	IDK		<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
78945621A	Joana	Hamet	Huslen	123987456				<input type="checkbox"/>	<input type="checkbox"/>			C/Mesquita, 90	17257	Torroella de Montgri		<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>

Imatge 10.26: Pantalla principal per els estudiants

Dins d'aquesta pantalla l'usuari administrador pot visualitzar els alumnes emmagatzemats a la base de dades i pot fer ús del filtre per buscar el que li interessi:

Pantalla d'alumnes

[Crear un nou estudiant](#)

Tria un element a buscar: cognom1 Valor:

DNI	nom	cognom1	cognom2	telefon	tutor1	tutor2	foto	autoritzacio_sortida	autoritzacio_imatge	telefon_tutor1	telefon_tutor2	adreca	codi_postal	poblacio	mail_tutor1	mail_tutor2
41565694W	Pau	Recacha	Borrell	617978195	Miquel Recacha perea	Marta Borrell Viader	1	<input type="checkbox"/>	<input type="checkbox"/>	666666666		C/Andreu Sabat, 59	17257	Torroella de Montgri	mrecacha@gmail.com	

Imatge 10.27: Pantalla principal per els estudiants (búsqueda alumne)

Dins d'aquesta pantalla, pot fer click a sobre "Crear" per crear un nou alumne, "Modificar" per editar les dades del alumne de la fila o "Eliminar" per eliminar el registre:

### Crear estudiant

DNI	<input type="text"/>
nom	<input type="text"/>
cognom1	<input type="text"/>
cognom2	<input type="text"/>
telefon	<input type="text"/>
tutor1	<input type="text"/>
tutor2	<input type="text"/>
foto	<input type="text"/>
autoritzacio_sortida	<input type="checkbox"/>
autoritzacio_imatge	<input type="checkbox"/>
telefon_tutor1	<input type="text"/>
telefon_tutor2	<input type="text"/>
adreca	<input type="text"/>
codi_postal	<input type="text"/>

Imatge 10.28: Pantalla crear nou alumne

Si fem click a "Detalls" en una de les files de la pantalla principal d'alumnes, se'ns obra una pantalla on es veu el valor de cada camp. Si fem click a "Edit" podrem modificar-ne els camps i guardar-ne el resultat:

### Detalls de l'estudiant

<b>DNI</b>	41565694W
<b>nom</b>	Pau
<b>cognom1</b>	Recacha
<b>cognom2</b>	Borrell
<b>telefon</b>	617978195
<b>tutor1</b>	Miquel Recacha perea
<b>tutor2</b>	Marta Borrell Viader
<b>foto</b>	1
<b>autoritzacio_sortida</b>	<input type="checkbox"/>
<b>autoritzacio_imatge</b>	<input type="checkbox"/>
<b>telefon_tutor1</b>	666666666
<b>telefon_tutor2</b>	
<b>adreca</b>	C/Andreu Sabat, 59
<b>codi_postal</b>	17257
<b>poblacio</b>	Torroella de Montgri
<b>mail_tutor1</b>	mrecacha@gmail.com
<b>mail_tutor2</b>	

[Modificar](#) | [Tornar al principi](#)

Imatge 10.29: Pantalla detalls de l'alumne

### Modifica els camps del estudiant

DNI	<input type="text" value="41565694W"/>
nom	<input type="text" value="Pau"/>
cognom1	<input type="text" value="Recacha"/>
cognom2	<input type="text" value="Borrell"/>
telefon	<input type="text" value="617978195"/>
tutor1	<input type="text" value="Miquel Recacha perea"/>
tutor2	<input type="text" value="Marta Borrell Viader"/>
foto	<input type="text" value="1"/>
autoritzacio_sortida	<input type="checkbox"/>
autoritzacio_imatge	<input type="checkbox"/>
telefon_tutor1	<input type="text" value="666666666"/>
telefon_tutor2	<input type="text"/>
adreca	<input type="text" value="C/Andreu Sabat, 59"/>
codi_postal	<input type="text" value="17257"/>
poblacio	<input type="text" value="Torroella de Montgri"/>

Imatge 10.30: Pantalla modificar alumne

## Aplicatiu per la gestió de centres educatius

La pantalla principal també té un link que porta a la pantalla de modificar un alumne. Si l'administrador desitja borrar un registre, pot fer-ho des de la pantalla principal:

Segur que vols borrar aquest alumne?

<b>DNI</b>	41566694W
<b>nom</b>	Pau
<b>cognom1</b>	Recacha
<b>cognom2</b>	Borrell
<b>telefon</b>	617978195
<b>tutor1</b>	Miquel Recacha perea
<b>tutor2</b>	Marta Borrell Viader
<b>foto</b>	1
<b>autoritzacio_sortida</b>	<input type="checkbox"/>
<b>autoritzacio_imatge</b>	<input type="checkbox"/>
<b>telefon_tutor1</b>	666666666
<b>telefon_tutor2</b>	
<b>adrecia</b>	C/Andreu Sabat, 59
<b>codi_postal</b>	17257
<b>poblacio</b>	Torroella de Montgri
<b>mail_tutor1</b>	mrecacha@gmail.com
<b>mail_tutor2</b>	

[Esborrar](#) | [Tornar](#)

Imatge 10.31: Pantalla esborrar alumne

# 11. Conclusions

L'objectiu d'aquest projecte era el de desenvolupar l'anàlisi, el disseny i la implementació d'un aplicatiu multiplataforma per portar el control diari d'un centre educatiu i un cop acabat el projecte, puc dir que tot i que els requisits que s'havien establert per l'aplicació han estat implementats, m'hauria agradat aprofundir més en alguns aspectes.

## 11.1.- Planificació original i desviacions

En un principi esperava tenir més temps per poder dedicar-me a estudiar i veure com funcionaven algunes aplicacions en el mercat i agafar possibles idees per utilitzar en l'aplicació. El problema va ser no l'hi vaig dedicar molt de temps ja que vaig posar-me amb el projecte uns dies tard per aprovar dos assignatura i tenir menys de 15 crèdits per acabar la carrera i poder fer el treball final de grau.

També volia dedicar més temps al disseny de les interfícies així com la depuració del codi i la seva documentació. Algunes classes poden estendre's per fer funcions que realitzen d'altres i així evitar duplicar codi (sobretot amb els adaptadors de les llistes).

Una idea que tenia en ment des del principi era la de contactar amb l'institut amb el qual vaig fer el projecte de recerca per poder portar el treball final de grau a un entorn real per recollir més necessitats, requisits i idees. A més, podria haver compartit el producte final amb els alumnes i professors de l'institut per recollir opinions sobre el producte final.

Un altre apartat dins la planificació original era la de proporcionar pantalles i eines a l'usuari administrador agradables a la vista i que no suposessin un problema per gestionar per part de l'administrador. Tot i que l'aplicació permet fer operacions bàsiques sobre la base de dades, ha quedat una aplicació pobre en contingut.

## 11.2.- Aprenentatge

Els principals coneixements adquirits en el desenvolupament d'aquest Projecte de Final de Grau han sigut:

- Millora de coneixements PHP.
- Millora de coneixements SQL.
- Gran millora de coneixements Android.

- Descobrint aplicacions web.
- Millora coneixements C# i aplicacions .NET.
- Millora de coneixements XML.
- Desenvolupar una aplicació amb funcionalitats de crides a un servidor local/real.
- Ús d'altres sistemes gestors de bases de dades així com la millora de coneixements en la gestió de bases de dades.
- Transferir un sistema local a un de real.

### 11.3.- Conclusions personals

Tot i que no estic satisfet puc veure la diferència abismal entre l'aplicació que he creat en aquest projecte de final de grau respecte el projecte de recerca. Tot i que els meus coneixements d'Android no eren molt grans, la capacitat de programació així com la de buscar solucions a problemes adquirits en la carrera, m'ha ajudat bastant en aquest treball.

## 12. Treball futur

En un futur es considera implementar les següents millors a l'aplicació:

- Fer l'aplicació multiplataforma.
- Implementar interfícies per l'administrador que li permetin fer una gestió semblant a la que fa ara però des de dins l'aplicatiu.
- Millorar el disseny d'algunes pantalles.
- Utilitzar animacions o altres tecnologies semblants dins l'aplicació per fer-la més amena i interactiva.
- Dissenyar un client i servidor REST que s'encarreguin de la gestió de crides a la base de dades. També cau la possibilitat de guardar la base de dades en l'aplicació i guardar els canvis que es facin cada 'x' temps o quan es perd la connexió.
- Fer que l'aplicació sigui multiidioma per estudiants que no entenen bé el català.
- Implementar un sistema de notificacions per avisar al alumne quan se'l ha puntuat una tasca o quan una nova notícia és creada per un professor.

## 13. Bibliografia

- [1] Geeks for Geeks, <https://www.geeksforgeeks.org/>
- [2] StackOverflow, <https://stackoverflow.com/>
- [3] Android Developers, References, <https://developer.android.com/reference/>
- [4] Andalucía Digital, Apps para profesores, <https://www.blog.andaluciaesdigital.es/apps-para-profesores/>
- [5] Educacion 3.0, Apps, plataformas y herramientas para evaluar a los estudiantes, <https://www.educaciontrespuntocero.com/recursos/apps-notas-alumnos-estudiantes/40413.html>
- [6] Softonic, 16 apps para maestros y profesores de cualquier nivel educativo, <https://www.softonic.com/articulos/apps-para-maestros-y-profesores-tic-aula>
- [7] Canva, Las 27 mejores apps para profesores, [https://www.canva.com/es\\_mx/aprende/apps-para-profesores/](https://www.canva.com/es_mx/aprende/apps-para-profesores/)
- [8] Camp Tecnológico, 8 herramientas de animación para utilizar en clase, <https://camptecnologico.com/8-apps-para-alumnos-y-profesores-en-el-aula/>
- [9] WAMPServer, página principal, <http://www.wampserver.com/en/>
- [10] WAMPServer, fòrum de dubtes, <http://forum.wampserver.com/list.php?2>
- [11] Postman, página principal, <https://www.getpostman.com/>
- [12] Notepad++, Descàrrega Notepad++, <https://notepad-plus-plus.org/download/v7.7.1.html>

[13] MariaDB, descàrrega MariaDB, <https://mariadb.org/download/>

[14] 000webhost, pàgina principal, <https://www.000webhost.com/>

[15] Visual Studio, pàgina principal, <https://visualstudio.microsoft.com/>

[16] Tutorial Microsoft, Introducció a ASP.NET MVC 5, <https://docs.microsoft.com/es-es/aspnet/mvc/overview/getting-started/introduction/getting-started>

[17] Glide, Getting Started, <https://bumptech.github.io/glide/doc/getting-started.html>