

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MOHAMED BOUDIAF - M'SILA

FACULTE DE MATHEMATIQUE ET DE
L'INFORMATIQUE

DEPARTEMENT D'INFORMATIQUE

N° :.....



DOMAINE : MATHEMATIQUE ET
INFORMATIQUE

FILIERE : INFORMATIQUE

OPTION : SIGL

**Mémoire présenté pour l'obtention
Du diplôme de Master Académique**

Par:

BAKRI FATIHA

LAICHI IBTISSAM

Intitulé

**Conception et Réalisation d'une Application
Smartphone Dédiée aux Parents pour Suivre leurs
Enfants élèves.**

Soutenu devant le jury composé de :

TAHRI ZOHIR

Université de M'sila

Président

.....

Université de M'sila

Rapporteur

.....

Université de M'sila

Examineur

Année universitaire : 2019 /2020



Dédicace

À mes chers parents ma mère et mon père , pour tous leurs patience , leur amour, leur tendresse, leur soutien et leurs prières tout au long de mes études.

À mes grandes- parents qui sont le symbole de bonté et de gentillesse.

À mes frères.

À mes sœurs.

À ma famille, mes chers proches et à ce qui me donnent de l'amour et de la vivacité.

À ma binôme Ibtissem , et tous mes chers amis proches

À tous le personne que connaisse et que je n'ai pas citées.

À tous ceux que j'aime.

Je dédie le fruit de mes efforts.

Fatiha





Dédicace

Je dédie ce modeste travail aux êtres qui me sont les plus chers ,

je cite :

Les parents les plus chers au monde , papa et maman , que dieu

les garde et protégé.

Mes chers frères

Toute ma famille

ma binôme Fatiha et Toutes mes amies , toute la promotion

2019-2020

Ibtissem 

Remerciements



Au nom d'Allah le clément le miséricorde



Nous ne savons pas comment exprimer notre gratitude envers Dieu, le tout puissant, qui nous a donnés la santé, le courage et la volonté à réaliser ce projet et pour toutes ces faveurs innombrables ;

Je voudrais dans un premier temps remercier, mon encadrant de mémoire Z.TAHRJ, pour sa patience, sa disponibilité et surtout ses judicieux conseils, qui ont contribué à alimenter ma réflexion.

Nous remercions également tous les enseignants de la faculté des mathématiques et de l'informatique pour toutes les connaissances qu'ils nous ont inculquées ;

Ainsi que les membres du jury ;

Merci pour tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail ;

Que Dieu accepte notre travail.

Table des matières

Table des matières.....	iv
Table des figures	vi
Liste des tableaux	vii
Introduction générale.....	01

Chapitre 1 Généralités sur les applications mobiles sous Android

1. Introduction	02
2. L'informatique mobile	02
2.1 Les terminaux mobiles	02
3. Les applications mobiles	04
3.1 Définition	04
3.2 Histoire des applications mobiles.....	05
3.3 Les différents types d'application mobile.....	06
3.4 Développement d'application mobile	06
3.5 Cycle de vie d'application mobile	08
4. Le système d'exploitation Android	09
4.1 Historique	09
4.2 Versions Android.....	12
4.3 Une application Android.....	12
4.4 Caractéristiques des applications Android.....	13
4.5 Architecture.....	13
4.6 Les outils de développement Android.....	15
5. Les avantages d'android	18
6. Conclusion.....	19

Chapitre 2 Le suivi parental des élèves

1. Introduction	20
2. Principe	20
3. Objectifs de suivi	21
4. Outils de communication	22
5. Les effets de la collaboration famille-école Sur les élèves	25
6. Efficacité de communication	25
7. Obstacles et Défis.....	27
8. Applications Web pour la gestion de la vie scolaire.....	28
9. Systèmes existants de suivi.....	30
10. Perspective de l'absence parentale.....	34
11. Conclusion	35

Chapitre 3 Analyse et conception du système

1. Introduction.....	36
2. Architecture général de notre application	36
3. Analyse et spécification des besoins.....	36
3.1 Expression initiale des besoins	36
3.2 Expression initiale des besoins	36
3.3 Besoins non fonctionnels détaillés.....	37
3.4 Besoins techniques.....	37
4. Présentation d’UML.....	37
5. Diagramme de cas d’utilisation.....	38
5.1 Définition.....	38
5.2 Acteurs.....	38
5.3 les cas d’utilisation.....	38
5.4 description des cas d’utilisation.....	40
6. diagramme des séquence.....	45
6.1 Authentification	45
6.2 Inscription.....	46
6.3 Consulter les notes.....	47
6.4 Ajouter les appréciations.....	47
7. diagramme de classe	48
8. conclusion.....	48

Chapitre 4 Implémentation

1. Introduction.....	49
2. Architecture.....	49
3. Présentation de l’environnement et des outils de développement.....	50
3.1 Environnement matériel.....	50
3.2 Environnement logiciel	50
4. Implémentation de base de données.....	51
4.1 NoSql.....	51
4.2 Firebase.....	52
4.2.1 Firebase realtime data base.....	57
4.2.2 Firebase storage.....	58
5. Présentation de l’application.....	59
6. Conclusion.....	66
Conclusion générale.....	67
Bibliographie.....	68
Résumé.....	70

Table des Figures

FIGURE 1.1	Un assistant personnel.....	03
FIGURE 1.2	Les smartphones.....	03
FIGURE 1.3	Les tablettes.....	04
FIGURE 1.4	Organigramme de Cycle de vie d'une application Android.	07
FIGURE 1.5	l'architecture d'un système Android.....	14
FIGURE 1.6	Sdk android.....	16
FIGURE 1.7	l'IDE Eclipse.....	17
FIGURE 1.8	l'IDE android studio.....	17
FIGURE 2.1	agenda scolaire.....	22
FIGURE 2.2	communication par téléphone.....	22
FIGURE 2.3	communication par courriel.....	23
FIGURE 2.4	Bulletin.....	23
FIGURE 2.5	classroom app logo.....	30
FIGURE 2.6	la page de classes et exercice.....	31
FIGURE 2.7	le logo de class dojo app.....	31
FIGURE 2.8	classroom app.....	32
FIGURE 2.9	Page de login et inscription de site.....	33
FIGURE 2.8	Page principale de site.....	33
FIGURE 3.1	Diagramme des cas d'utilisation.....	39
FIGURE 3.2	cas d'utilisation gestion des comptes.....	40
FIGURE 3.3	cas d'utilisation gestion des entités.....	42
FIGURE 3.4	Diagramme de séquence de l'authentification.....	45
FIGURE 3.5	Diagramme de séquence de l'inscription.....	46
FIGURE 3.6	Diagramme de séquence de Consulter les notes.....	47
FIGURE 3.7	Diagramme de séquence de ajouter les appréciations...	47
FIGURE 3.8	Diagramme des classes.....	48
FIGURE 4.1	Architecture client /serveur.....	49
FIGURE 4.2	Logo android studio.....	51
FIGURE 4.3	Logo de jdk.....	51
FIGURE 4.4	Interface initial de firebase.....	52
FIGURE 4.5	ajouter un projet dans le firebase.....	53
FIGURE 4.6	Première étape de l'implémentation.....	53
FIGURE 4.7	deuxième étape de l'implémentation.....	54
FIGURE 4.8	app explorer.....	55
FIGURE 4.9	google-services.json dans le dossier app.....	55

FIGURE 4.10	troisième étape de l'implémentation.....	56
FIGURE 4.11	Fichier build.gradle au niveau du projet.....	56
FIGURE 4.12	Fichier build.gradle au niveau du projet.....	57
FIGURE 4.13	Fichier build.gradle au niveau de l'application.....	57
FIGURE 4.14	réglage de base de données.....	58
FIGURE 4.15	Firestore data base.....	58
FIGURE 4.16	Firestore storage.....	58
FIGURE 4.17	Firestore storage en détail.....	59
FIGURE 4.18	logo d'application.....	59
FIGURE 4.19	Splashscreen.....	60
FIGURE 4.20	Fenêtre de login.....	60
FIGURE 4.21	Fenêtre d'inscription.....	60
FIGURE 4.22	Fenêtre de l'agenda.....	60
FIGURE 4.23	page de parent.....	61
FIGURE 4.24	page de l'enseignant.....	61
FIGURE 4.25	page d'élève.....	61
FIGURE 4.26	menu de l'admin.....	61
FIGURE 4.27	gestion des enseignants.....	62
FIGURE 4.28	gestion des parents.....	62
FIGURE 4.29	gestion des élèves.....	62
FIGURE 4.30	gestion des rendez-vous.....	62
FIGURE 4.31	les appréciations des enfants.....	63
FIGURE 4.32	notes des enfants.....	63
FIGURE 4.33	les enfants de parent.....	63
FIGURE 4.34	entre des appréciations.....	64
FIGURE 4.35	entre des absences	64
FIGURE 4.36	entre des notes.....	64
FIGURE 4.37	demande de rendez- vous.....	64
FIGURE 4.38	Fenêtre de devoirs.....	65
FIGURE 4.39	rendez-vous de parent.....	65
FIGURE 4.40	Fenêtre des messages.....	65

Liste des Tableaux

Table 1.1	historique du système Android.....	09
Table 1.2	tableau comparatif entre Eclipse et Android Studio.....	18
Table 3. 1	Cas d'utilisation Consulter un compte.....	40
Table 3. 2	Cas d'utilisation créer un compte.....	41
Table 3. 3	Cas d'utilisation modifier un compte.....	41
Table 3. 4	Cas d'utilisation consulter des entités.....	42
Table 3. 5	Cas d'utilisation saisir une entités.....	43
Table 3.6	Cas d'utilisation authentification.....	44
Table 3.7	Cas d'utilisation inscription.....	44
Table 4. 1	Présentation de l'environnement de travail utilisé.....	50

Introduction Générale

Le suivi scolaire présente un grand défi tant pour les parents que pour l'établissement de l'élève où les enseignants ont besoin d'une coopération plus rapprochée des parents afin de mieux mener leur tâche d'enseignement.

Or, ce suivi s'avère être une opération qui nécessite beaucoup de soin au niveau de l'assistance et des outils à utiliser à cet effet, chose qui souffre d'un manque considérable au niveau de nos communautés.

Pour résoudre ce problème de manque, nous tentons de présenter dans ce travail, la réalisation d'une application mobile qui permet d'assurer cette opération de suivi d'une manière aisée et efficace et contribuer ainsi, au développement de la vie scolaire dans nos établissements. Cette application devrait répondre aux objectifs suivants :

- Faciliter la communication entre les parents et les enseignants
- Avoir des suivis efficaces des enfants
- Bénéficier de la rapidité et l'efficacité des applications Android

Pour présenter notre travail, nous avons subdivisé notre mémoire en quatre chapitres :

↳ Le premier chapitre consiste à présenter des notions et généralités sur les applications Android afin d'initier le lecteur à ce domaine pour mieux comprendre l'importance de notre travail.

↳ Le deuxième chapitre entamera le sujet du suivi parental des élèves où on détaillera quelques notions telles que, le principe de suivi, ses objectifs, ainsi que quelques outils de communication entre l'école, les parents et les enseignants.

↳ Le troisième chapitre est le chapitre qui va servir à présenter, d'une manière détaillée, la conception du système d'information à travers lequel nous précisons la solution informatique qui s'adapte au mieux à l'atteinte des objectifs.

↳ Dans le dernier chapitre nous exposerons le côté pratique de la réalisation de notre application où nous parlerons des outils utilisés en présentant quelques-unes de ses interfaces.

↳ Finalement, une conclusion clôturera ce document pour évaluer et estimer le travail réalisé.

CHAPITRE 01

GÉNÉRALITÉS SUR LES APPLICATIONS MOBILES SOUS ANDROID

- L'informatique mobile
- Les applications mobiles
- Le système d'exploitation Android

1. Introduction :

Dans ce chapitre, nous parlons en bref sur les notions générale des applications mobile sous android , nous commencerons par présenter l'informatique mobile, puis nous verrons quelques terminologies mobiles après cela, nous expliquerons le système d'exploitation Android . Enfin, nous terminerons par les avantages d' android.

2. L'informatique mobile :

L'informatique mobile est un domaine où l'interaction homme-ordinateur nécessite le transport de l'ordinateur d'un emplacement à un autre au cours d'une utilisation normale, tout en permettant la transmission de données, voix et vidéo. Parler de l'informatique mobile entraîne nécessairement l'évocation de la communication mobile, le matériel mobile et les logiciels mobiles.

2.1. Les terminaux mobiles

Un terminal mobile est un petit appareil informatique ou de communication qu'on peut transporter avec soi dans ses déplacements et l'utiliser comme terminal donnant accès sans fil à un ou plusieurs réseaux.

Parmi les terminaux mobiles, on trouve des assistants numériques personnels (PDA), téléphones intelligents (Smartphone), des tablettes, etc...

➤ *Les assistants personnels (PDA)*

Un assistant personnel est un périphérique portable qui fonctionne comme un gestionnaire d'informations personnelles. Les PDA sont utilisés pour la navigation sur le Web, les applications bureautiques, les vidéos, les photos ou les téléphones mobiles. Les caractéristiques du modèle PDA varient, mais les fonctionnalités communes courantes incluent les écrans tactiles, la connectivité Bluetooth et Wifi, etc. Les PDA contiennent des logiciels pour synchroniser les informations.[1]



Figure 1.1 Un assistant personnel

➤ *Les smartphones*

Un smartphone (téléphone intelligent) est un téléphone portable doté de fonctionnalités très avancées. Un smartphone typique dispose d'un écran tactile haute résolution, de la connectivité WIFI, des capacités de navigation Web et de la capacité d'accepter des applications sophistiquées. La plupart de ces appareils fonctionnent sur l'un de ces systèmes d'exploitation mobiles populaire : Android, Symbian, IOS, BlackBerry OS et Windows. [2]



Figure 1.2 Les smartphones

➤ *Les tablettes*

Tablette (de l'anglais tablet, plaque) est le nom donné à une famille d'ordinateurs portables dépourvus de clavier à touches et munis d'un écran tactile, de la même dimension qu'une feuille A4 ou plus petits. L'écran tactile est toujours multipoints, donc capable de détecter plusieurs touches simultanés.

Ces ordinateurs sont essentiellement tournés vers l'utilisation d'Internet : consultation de pages Web, lecture de journaux en ligne ou de livres électroniques ou messageries. Leurs dimensions excluent l'intégration d'une mémoire de stockage mécanique, comme un lecteur-graveur de DVD ou un disque dur. [3]



Figure 1.3 Les tablettes

3. Les applications mobiles :

3.1. Définition

L'application mobile se définit comme étant un logiciel téléchargeable qu'on installe facilement sur son smartphone comme nous le faisons avec tout logiciel sur notre PC portable.

Le téléchargement de l'application mobile se fait suivant deux options :

- Sur téléphone par le biais de connexion internet.
- Sur PC en le branchant avec le téléphone mobile. [4]

3.2. Histoire des applications mobiles

La naissance des applications mobiles remonte à la fin du XXe siècle, leurs industries a commencé quand l'iPhone d'Apple et leur magasin iTunes ont été introduits en 2007. C'était aussi le moment où les utilisateurs ont commencé à utiliser leurs appareils mobiles pour surfer sur internet, seules quelques personnes regardaient leurs courriers électroniques, mais avec l'iPhone, le taux de consommateurs a augmenté et les téléphones mobiles ne sont plus utilisés que pour des appels téléphoniques, mais pour beaucoup plus que cela ; puisque le nombre de nouvelles applications mobiles s'accroît de jour en jour et ces applications nous offrent plusieurs services tels la consultation du courrier électronique, la téléphonie IP etc.....[5]

3.3. Les différents types d'applications mobiles

Il existe plusieurs types d'applications mobiles qui sont les applications natives, web et hybrides .

- *Application native :*

Bien que le terme d'application native ne soit pas très connu du grand public, cela représente la grande majorité des applications que nous téléchargeons chaque jour. Quand on pense à une application, on pense inconsciemment à une application native. C'est une application qui est développée spécifiquement pour un système d'exploitation donné. [6]

- *Applications Web (WebApps) :*

La version mobile d'un site web, c'est une application web. À l'opposé de l'application native, se trouve l'application web car c'est une application qui fonctionne comme un template de site web, sans aucune différence de plateforme, de système d'exploitation, de code... Cela veut dire que les utilisateurs n'auront pas à installer l'application sur leurs smartphones, le stockage reste donc intact. Ce qui est un très gros point fort quand on voit à quel point la mémoire de nos téléphones se remplit vite.

En général, le but d'une application web est de rendre du contenu disponible, ou du moins fonctionnel sur mobile. Elle se comporte comme une application normale, ressemble à une application normale, mais son accessibilité et son prix n'ont rien à voir avec les applications mobiles. En effet, une application web coûte jusqu'à trois fois moins cher qu'une application native.[6]

- *Les applications hybrides :*

Entre l'application native et l'application web se trouve l'application hybride. Comparée à une application native, elle est plus rapide et moins chère à développer. Elle offre également plus de possibilités qu'une application web.

Un des principaux avantages de l'application hybride est qu'elle est plus facile et plus rapide à développer qu'une application native. La maintenance de l'application sera également plus facile puisqu'il n'y a qu'une seule version à revoir pour plusieurs plateformes.

Cependant, toutes ces facilités ont un prix : les performances de l'application sont moins bonnes et moins stables puisque le système est moins adapté à chaque plateforme.[6]

3.4. Développement d'applications mobiles

La plupart des smartphones et tablettes non-Apple fonctionnent avec certaines versions du système d'exploitation Android. Les appareils les plus populaires qui utilisent Android sont les suivants : Nexus, Samsung, Sony, Huawei, etc...

Lors du développement d'applications mobiles pour ce système d'exploitation, nous devons prendre en compte tous ces appareils. Il est possible que chacun ait des dimensions différentes. Nous devons adapter notre développement d'applications mobiles au standard. C'est le seul moyen de s'assurer que l'application est visible et utilisable.

La façon la plus simple et la plus directe de commencer à développer des applications mobiles pour Android est de télécharger le SDK Android et l'IDE Eclipse ou L'IDE Android Développer Tools et Android Studio .

Le développement Android peut se faire à partir d'ordinateurs Microsoft® Windows®, Mac OS X ou Linux, étant donné sa nature open source.

Il est fortement recommandé d'avoir un émulateur Android à proximité si nous sommes dédiés au développement d'applications mobiles pour ce système d'exploitation. Avec lui, nous pourrions voir les progrès que nous faisons dans le développement, les erreurs, les corrections.[7]

3.5. Cycle de vie d'une application mobile

Les technologies mobiles prennent de plus en plus de place sur le marché. Les Smartphones sont considérés comme des petits ordinateurs et dotés d'un système d'exploitation s'appuyant sur un noyau Linux. Cependant, ils diffèrent des ordinateurs classiques par le cycle de vie d'une application. Sous ANDROID, une application est composée d'une ou plusieurs activités. Une activité est la base d'un composant pour la création d'interfaces utilisateur. Afin de faciliter la cinématique de l'application, il est préconisé de n'avoir qu'une interface visuelle par activité.[8]

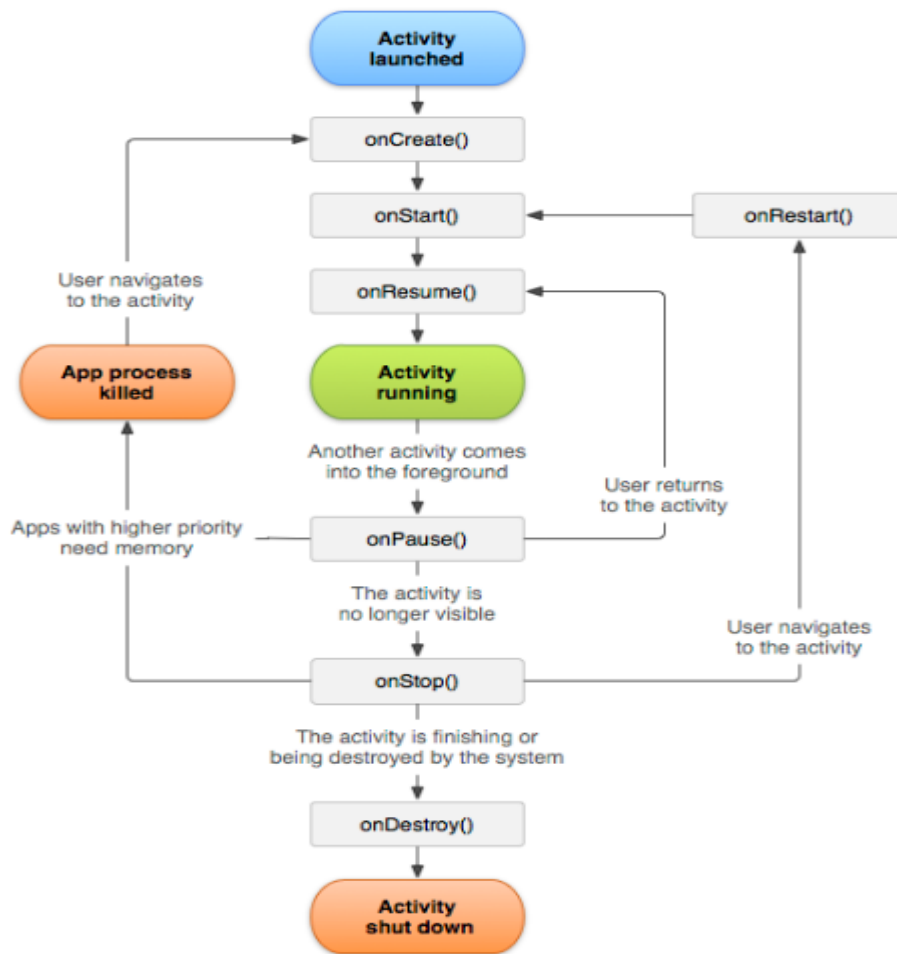


Figure 1.4 Organigramme de Cycle de vie d'une application Android.

Les différentes étapes de l'organigramme précédent sont :

La méthode onCreate()

Cette méthode, qu'on connaît déjà, est appelée lorsque notre activité est créée par le système et qu'elle entre dans l'état Created.

Généralement, les opérations effectuées dans cette méthode servent à mettre en place l'interface graphique, à initialiser les variables, à configurer les listeners ou à se connecter au modèle.

La méthode onStart()

Cette méthode est appelée par le système lorsque l'activité entre dans l'état Started. L'interface graphique devient visible à l'utilisateur, mais il ne peut pas encore interagir avec les différents éléments.

La méthode onResume()

Cette méthode est appelée lorsque l'activité entre dans l'état Resumed. L'activité devient entièrement opérationnelle. L'utilisateur peut utiliser l'application et cliquer sur les différents éléments graphiques.

La méthode onPause()

Cette méthode est appelée lorsque l'activité entre dans l'état Paused. Cette méthode est le pendant de la méthode onResume() : tout ce qui est initié dans onResume() doit être mis en pause dans cette méthode. Par exemple, une animation présentée à l'utilisateur est démarrée dans onResume() puis stoppée dans onPause().

Les traitements effectués dans cette méthode doivent être les plus courts possibles.

La méthode onStop()

Cette méthode est appelée lorsque l'activité entre dans l'état Stopped. Par exemple, lorsqu'une nouvelle activité est démarrée, l'activité appelante va se retrouver dans cet état. Elle n'est donc plus visible à l'utilisateur. Les traitements liés à la mise à jour de l'interface graphique peuvent être arrêtés.

La méthode onDestroy()

Cette méthode est appelée lorsque l'activité est arrêtée. Par exemple, ce peut être après avoir appelée la méthode finish(), ou si le système décide d'arrêter l'activité pour libérer de la mémoire.[9]

4. Le Système d'exploitation Android :

Android est une plateforme intégrée pour la première fois dans un smartphone (téléphone intelligent) sorti en France en mars 2009, Android s'est depuis émancipé très rapidement pour conquérir de nombreux appareils, mobile ou non, tels que les netbooks (mini-ordinateurs), les

tablettes tactiles et les télévisions connectées, au point de devenir aujourd'hui l'un des systèmes d'exploitation majeurs dans le monde.[10]

4.1. Historique

L'histoire d'Android commence en octobre 2003, lorsqu'Andy Rubin, Rich Miner, Nick Sears et Chris White créent la société Android à Palo Alto (Californie). Google a racheté la société en août 2005. Deux ans plus tard, l'Open Handset Alliance est annoncée et Android devient officiellement open source.

La première version du SDK Android 1.0 est sortie en 2008 avec le premier téléphone sous Android (HTC DRAM). En avril 2009, la version 1.5 (API 3) d'Android sort. Cette version baptisée Cupcake (petit gâteau) inaugure les nouveaux noms des versions d'Android ce qui donnera pour les futures versions comme le montre la table ci-dessous. [11]

Nom	N version	Date	Niveau API
N/A	1.0	Septembre 2008	1
	1.1	Février 2009	2
Cupcake	1.5	Avril 2009	3
Donut	1.6	Septembre 2009	4
Eclair	2.0-2.1	Octobre 2009	5-7
Froyo	2.2-2.2.3	Mai 2010	8
Gingerbread	2.3-2.3.7	Décembre 2010	9-10
Honeycomb[a]	3.0-3.2.6	Février 2011	11-13
Ice Cream Sandwich	4.0-4.0.4	Octobre 2011	14-15
Jelly Bean	4.1-4.3.1	Juin 2012	16-18
Kitkat	4.4.x	Octobre 2013	19-20
Lollipop	5.0-5.1.1	Octobre 2014	21-22
Marshmallow	6.0-6.0.1	Mai 2015	23
Nougat	7.0-7.1.1	Septembre 2016	24-25

Table 1.1 historique du système Android

4.2. Versions Android

- *Android 1.0 et 1.1* : À la date du 22 octobre 2008, le premier smartphone sous Android débarque chez l'opérateur T-Mobile en Amérique, le HTC Dream (ou HTC G1). Il n'y avait pas encore de clavier ou la gestion du multitouche. La grande innovation pour cette

première version d'Android fut la barre de notifications, au travers d'un menu déroulant qui regroupe plusieurs affichages pratiques et rapidement accessibles. [12]

- *Android 1.5 CupCake* : La mise à jour 1.5 « CupCake » a été publiée le 30 avril 2009, améliorant l'interface graphique d'Android. À partir de celle-ci, il a été décidé de nommer les différentes versions du système avec des noms de gâteaux (peut-être que les entreprises étaient gourmandes), en suivant l'ordre alphabétique. CupCake était donc la troisième itération de l'O.S, mais surtout la venue (enfin) du premier clavier tactile, et du presse-papier. [12]

- *Android 1.6 Donut* : Si la version 1.5 était une mise à jour majeure, celle qui a suivi l'était moins. Android 1.6 « Donut » offrait tout de même quelques nouvelles fonctionnalités comme la prise en charge du réseau CDMA, le support d'autres définitions d'écran. Cette version apportait également la fonction de recherche directement en local sur votre appareil Android, en comptant également sur les recherches internet grâce à Google Search. [12]

- *Android 2.0 et 2.1* : Eclair Google a frappé fort à la fin du mois d'octobre 2009, en publiant la version Android 2.0 « Eclair », tandis que Motorola annonçait son téléphone Droid. Il aura surtout été possible grâce à cette mise à jour d'associer plusieurs comptes Google au même appareil. Celle-ci aura été majeure et aura apporté beaucoup de fonctionnalités intéressantes, comme les suivantes : Amélioration du clavier tactile, Fonds d'écran animés, Prise en charge du Bluetooth 2.1, Arrivée de Google Maps, Nouvelles fonctions pour l'appareil photo (flash, zoom digital, balance des blancs, effets de couleurs et focus macro) et la fonction « speech-to-text » fait également son entrée et introduit la synthèse vocale au sein du système Android. [12]

- *Android 2.2 Froyo, et 2.3 Gingerbread* : Le 20 mai 2010 arrivait la version Android 2.2 « Froyo », qui a profité au Nexus One tout d'abord (Google oblige) et ce fut à nouveau de belles améliorations pour le système opensource. De nouveaux raccourcis ont été ajoutés aux côtés du lanceur d'applications : le téléphone et le navigateur natif. La mise à jour proposait également du nouveau pour la galerie avec une meilleure interface pour regarder vos photos, des effets de transition, ce qui rendait l'O.S. [12]

- *Android 3.0 Honeycomb* : Dans toute l'histoire de l'OS, il existe une mise à jour, la 3.0 « Honeycomb », qui fut réservée aux tablettes Android. À cette époque, la société Motorola détenait l'exclusivité avec son smartphone Droid et la version 2.0 du système. La tablette Xoom est née avec cette mise à jour Honeycomb et nous avons pu découvrir le nouveau virement de bord de la part de Google sur l'interface. [12]

- *Android 4.0 IceCreamSandwich* : C'est le 19 octobre 2011 que l'histoire d'Android va atteindre son apogée (ou presque). L'arrivée de la version 4.0 « Ice Cream Sandwich », compatible avec les smartphones et les tablettes, nous laisse rêveur avec beaucoup de nouveautés tant attendues par les utilisateurs, et quelques-unes que je vous cite : Ajout de dictionnaires dans le clavier virtuel, Application photo améliorée, rapidité de capture quasiment en mode rafale, Capture d'écran native, Consultation de son trafic data, pour ne plus avoir de hors forfait à votre prochaine facture, Création plus facile de dossiers d'applications, Écran de verrouillage repensé, sous forme d'un cadenas qui permet de lancer des applications en raccourci, Mise à jour du navigateur (ouvertures de plusieurs pages, navigation privée) et enfin la Prise en charge de l'enregistrement vidéo Full HD 1080p. [12]

- *Android 4.1, 4.2, 4.3 Jelly Bean, et 4.4 KitKat « Jelly Bean »* : est le nom d'Android, annoncé le 27 juin 2012 lors du Google I/O. Le premier appareil équipé de la mise à jour 4.1 fut la Nexus 7, la tablette de Google commercialisée à partir du 13 juillet 2012 aux États-Unis. C'est une mise à jour qui fut très attendue, et pour preuve, à l'heure actuelle, nos appareils sont encore équipés d'une version Jelly Bean de l'O.S. Le 13 novembre 2012 sortait la version suivante, Android 4.2, toujours une Jelly Bean. Avec cela, Google a fait l'annonce de trois nouveaux dispositifs : le Nexus 4, la Nexus 10 et une prochaine Nexus 7. Une mise à jour qui a comme d'habitude, apporté un lot de nouvelles fonctionnalités. Le 24 juillet 2013 sera la date de la dernière mise à jour Android 4.3 qui est disponible sur de très nombreux appareils grâce à la mise à jour en OTA. [12]

- *Android 5.0, 5.0.1, 5.0.2 et 5.1 Lollipop* : Android 5.0 Lollipop a été officiellement nommé ainsi le 15 octobre 2014 par Google. De nombreuses mises à jour sont depuis en cours vers cette nouvelle version majeure qui fut disponible publiquement le 3 novembre de la même année pour les appareils Nexus dans un premier temps. Les Nexus 6 et 9 en furent d'ailleurs équipés dès leur commercialisation.[12]

- *Android 6.0 Marshmallow* : Le surnom de Marshmallow a été officialisé le 18 août 2015. Côté nouveautés apportées, Google a souligné 6 fonctions principales qui porteront Android 6.0, mais en réalité, rien que la version développeur en comptabilise déjà une cinquantaine, que ce soit par des ajouts de fonctionnalités inédites ou d'améliorations de certaines déjà existantes. [12]

- *Android 7.0 Nougat* : Android Nougat, c'est la 7ème version du système d'exploitation Android. Annoncé en grande pompe par Google lors d'un événement exceptionnel à Montélimar, cette mise à jour propose plusieurs nouvelles fonctionnalités. On commence par un des plus gros changements le multi-fenêtre disponible nativement sur Android Nougat, cela permet d'avoir deux applications en même temps sur l'écran. Il y a les mises à jour dites « transparentes », il faut comprendre par là qu'elles se font sans que l'utilisateur ne s'en aperçoive, il faut juste redémarrer l'appareil. [12]

- *Android 8.0 Oreo* : Nommée à partir du célèbre biscuit Oreo, cette 8ème version a été dévoilée par Google fin du mois d'août 2017. Malgré quelques bugs de lancement plutôt ennuyeux, cette nouvelle mise à jour propose d'intéressantes nouveautés, comme 70 nouveaux émojis, le Picture-in-Picture, la gestion multitâche et le démarrage rapide, une meilleure gestion des notifications, [12]

- *Android 9.0 Pie* : Android Pie (en français, « Tarte ») est la version 9.x du système d'exploitation mobile Android développé par Google. Cette version du système est disponible depuis le 6 août 2018 pour les smartphones de la gamme Google Pixel, l'Essential Phone2, Nokia et OnePlus. Elle est décrite par le numéro d'API 28.[12]

- *Android 10* : est la dix-septième version d'Android et la première à ne pas profiter d'un sobriquet. Officialisée au Google I/O 2019 sous le nom d'Android Q, Android 10 a vu sa version finale présentée en août 2019 avec la nouvelle « stratégie de marque » voulue par Google. En effet, compte tenu de la popularité d'Android, Google a souhaité simplifier la dénomination des versions d'Android, comme iOS ou Windows.[12]

4.3. Une application Android

Une application Android est une application mobile spécifiquement développée pour les smartphones et les tablettes utilisant le système Android. Elles sont de nature très variable telles que les applications jeux, mobile commerce, utilitaire, service d'information. [13]

Sous Android, une application est composée d'une ou plusieurs activités. Une activité est la base d'un composant pour la création d'interfaces utilisateur. Afin de faciliter la cinématique de l'application, il est préconisé de n'avoir qu'une interface visuelle par activité.

4.4. Caractéristiques des applications Android

Plusieurs caractéristiques spécifient les applications Android :

- *Une application mobile est plus rapide* : car elle va exploiter au mieux les capacités du téléphone sauf si elle est très mal programmée.
- *Elle offre plus de possibilités* : Accès aux photos, à l'agenda, aux contacts etc...
- Plus difficile à développer.
- *Il faut développer pour chaque plateforme* : Pour s'adapter à l'ergonomie de chaque plateforme et exploiter au mieux les fonctionnalités.
- *On gagne plus avec une application mobile* : plus aisé qu'avec une Web App de vendre l'application sur l'App Store ou l'Android Market.
- Le déploiement est plus compliqué, plus long.
- Pas besoin d'être connecté pour utiliser mon application. [14]

4.5. Architecture

Cette figure illustre les différentes couches au niveau du système Android.

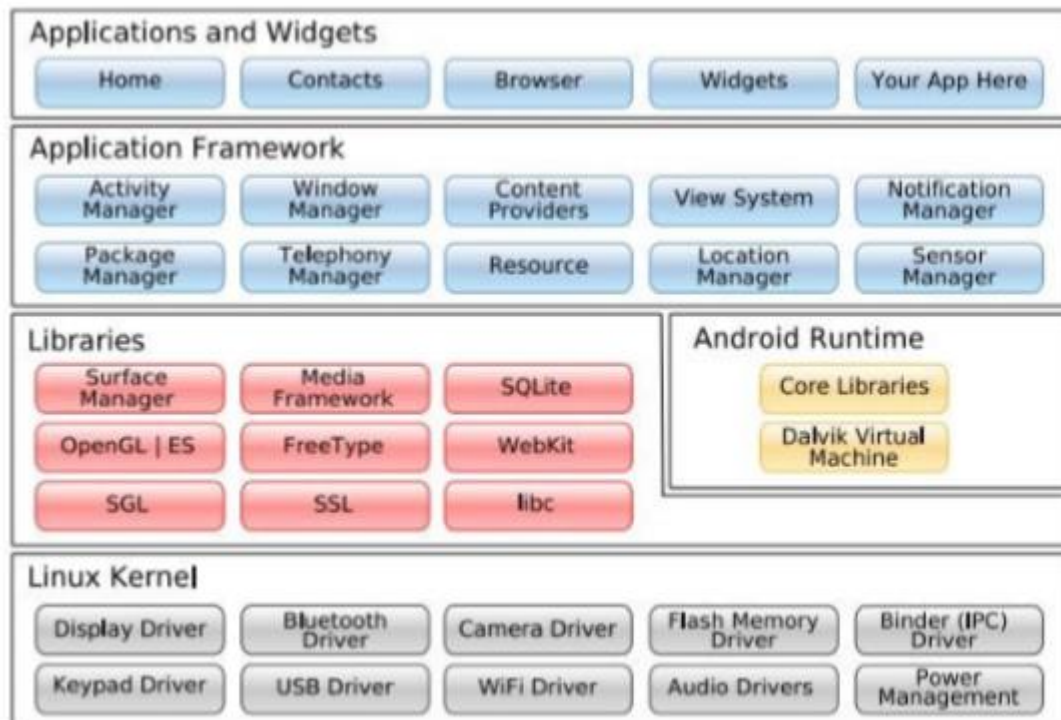


Figure 1.5 l'architecture d'un système Android

Comme le montre la figure ci-dessus, Le système d'exploitation Android est basé sur une architecture à quatre niveaux.

- *Linux Kernel* : Le système d'exploitation Android s'appuie sur un noyau Linux, régulièrement mis à jour selon les versions du système. Cette première couche prend en charge la gestion des couches basses (gestion des processus, de la mémoire, de la couche matérielle) ainsi que les droits utilisateurs.

Au-dessus du noyau se trouve une couche d'espace utilisateur native, constituée par le binaire init (le premier processus démarré, qui fait tourner tous les autres processus), plusieurs démons et quelques centaines de bibliothèque natives sont utilisées dans tout le système. [15]

- *Les librairies*

A ce niveau, figure la couche des bibliothèques principales du système fournies par des tiers. Celles-ci, de bas niveau, sont écrites en C et/ou C++. Elles fournissent des services essentiels tels que la bibliothèque d'affichage en 2D (SGL), la bibliothèque d'affichage

graphique 3D (OpenGL), la bibliothèque de base de données (SQLite), la lecture et l'enregistrement audio et vidéo (Media Framework), un moteur de navigation Web (Web Kit) ...[10]

- *Android Runtime (Le moteur d'exécution linux)*

Au même niveau que cet ensemble de bibliothèques, se trouve l'environnement fonctionnel d'Android basé sur une technologie Java avec une spécificité unique : sa machine virtuelle optimisée pour les applications mobiles. Cette machine virtuelle, dénommée Dalvik, est initialisée par chaque application lors de son lancement et ouvre, pour chacune, une Dalvik spécifique, ce qui rend l'OS complètement multitâche. Dalvik est fournie avec son kit de développement spécifique ou Application Programming Interface (API) écrit en Java ; les applications Android sont donc écrites en Java, mais une java spécifique à Android. Le Java Development Kit (JDK) Android est, en fait, un sous-ensemble du JDK de SUN et intègre une partie des classes JDK SUN et JDK SWING, plus des classes spécifiques à Android. [16]

- *Framework Android*

Les fonctionnalités offertes par ces bibliothèques sont ensuite reprises et utilisées par cette couche sous forme de bibliothèques Java. Celles-ci fournissent des bibliothèques et composants réutilisables spécifiques à des domaines particuliers. On y retrouve par exemple les bibliothèques de base de données, de téléphone, localisation géographique, de communication en champ proche. [10]

- *Les applications*

Enfin, une dernière couche parachève cette structure : les applications. Ces applications peuvent être, bien sûr, les applications tierces téléchargées sur le magasin d'application officiel, mais également des applications installées par défaut, telles que l'application d'accueil (aussi appelée Launcher), le navigateur web, les applications de SMS et téléphonie, etc. Toutes ces applications sont communément développées en Java. [15]

4.6. Les outils de développement Android

Les différents outils de développement Android sont :

- *Le SDK Android* : Le Kit de développement logiciel Android (Android SDK) contient les outils nécessaires pour créer, compiler et déployer les applications Android. La plupart de ces outils sont en ligne de commande. [17]

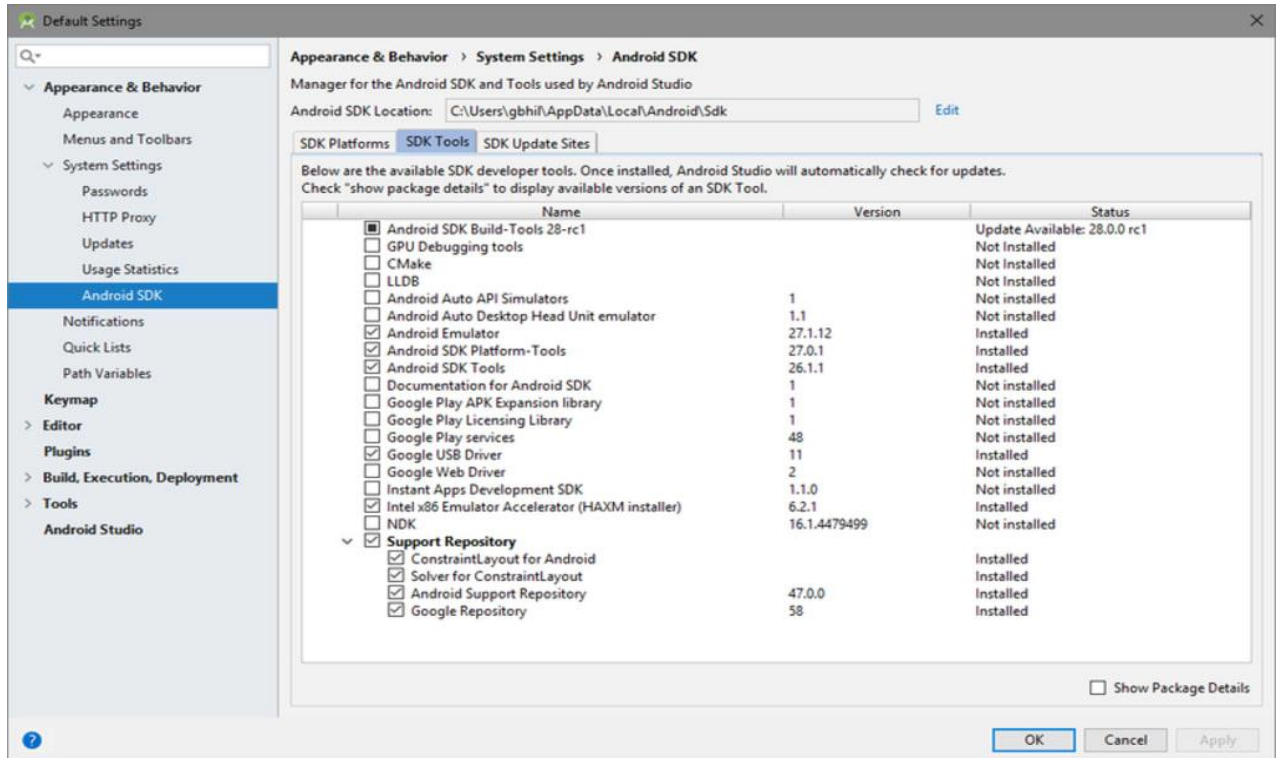


Figure 1.6 Sdk android

- *Le débogueur « ADB »*

Le SDK Android contient un débogueur appelé « Android debug bridge » ou aussi «ADB», qui permet de connecter un appareil Android virtuel ou réel, dans le but de gérer le périphérique ou de déboguer votre application. [17]

- *Les IDE « Android Developer Tools » et « Android Studio »*

Google propose deux environnements de développement intégrés (IDE) pour développer de nouvelles applications.

Eclipse : Les outils de développement Android sont basés sur l'IDE Eclipse. ADT est un ensemble de composants (plug-ins), qui étendent l'IDE Eclipse avec des capacités de développement Android. [17]

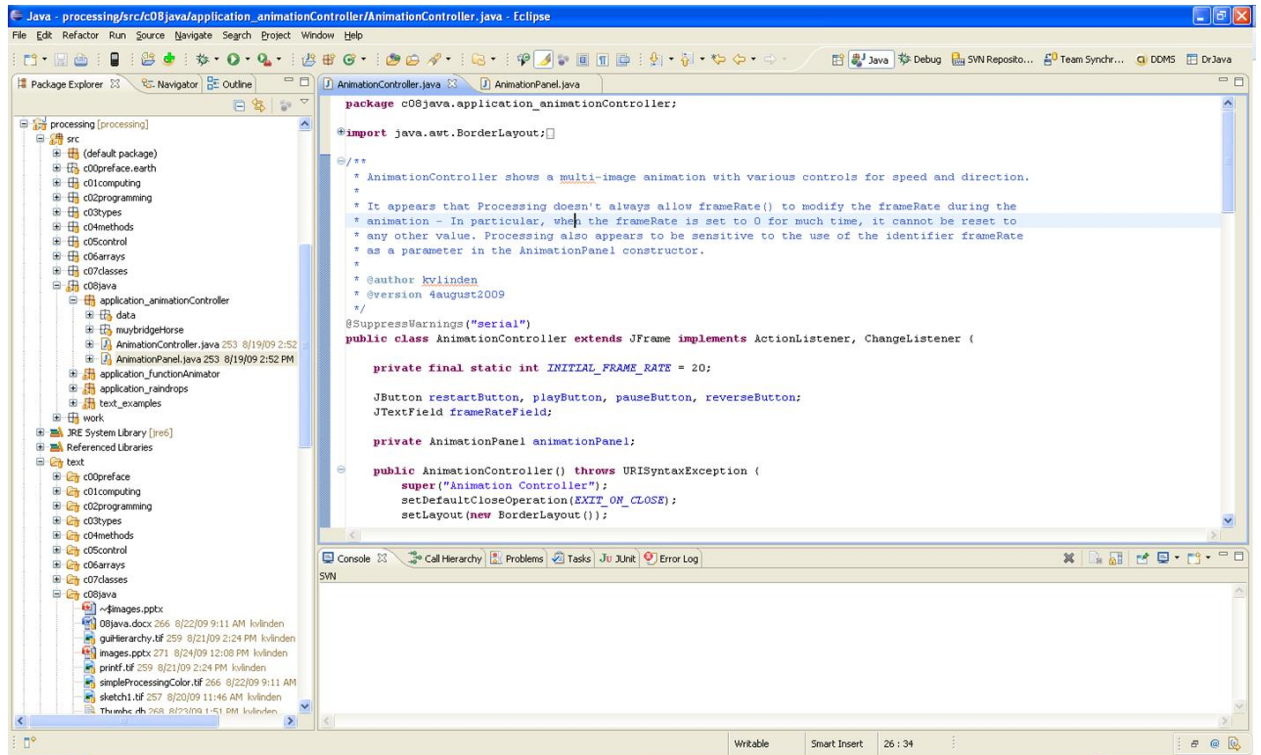


Figure 1.7 l'IDE Eclipse

Android Studio : Google propose également cet IDE appelé pour la création d'applications Android quel que soit le terminal sous-jacent (smartphone, tablette, montre, TV...) et qui est basé sur l'IDE IntelliJ.[17]

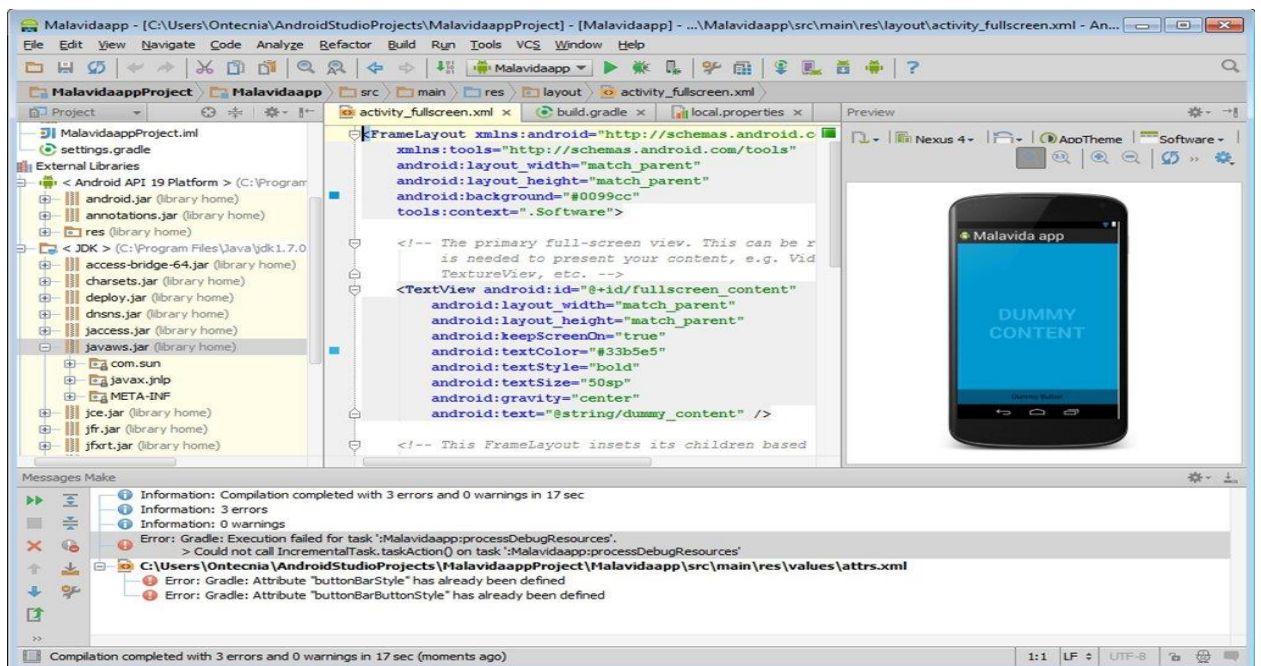


Figure 1.8 l'IDE Android studio.

Le tableau qui suit, compare entre les deux IDE :

	Eclipse	Android Studio
Facilité d'installation	Moyen	Simple
Langue	Nombreuses	Anglais
Performance	Peut être lourd	Rapide
Système de build	Ant	Gradle
Génération de variante et de multiple APK	Non	Oui
Android Code completion et refactoring	Base	Avancé
Editeur d'interface graphique	Oui	Oui
Signature d'APK et gestion de Keystore	Oui	Oui
Support NDK	Oui	A venir

Table 1.2 tableau comparatif entre Eclipse et Android Studio

- *La machine virtuelle Dalvik*

Dalvik est une machine virtuelle incorporée dans le système d'exploitation Android. Destinée à permettre l'exécution simultanée de plusieurs applications sur un appareil de faible capacité (peu d'espace mémoire et peu de puissance de calcul). [17]

- *Le runtime Android (ART)*

Les dernières versions d'Android introduisent une nouvelle machine, le runtime Android qui est le moteur qui permet l'exécution des applications pour Android.[17]

5. Les avantages d'android

- ✓ *Open source* : Le contrat de licence pour Android respecte les principes de l'open source, c'est-à-dire qu'on peut à tout moment télécharger les sources et les modifier selon le besoin. Android utilise des bibliothèques open source puissantes, comme par exemple SQLite pour les bases de données et OpenGL pour la gestion d'images 2D et 3D.
- ✓ *Gratuit* : Android est gratuit, autant pour l'utilisateur que pour les constructeurs.
- ✓ *Facile à développer* : Toutes les API mises à disposition facilitent et accélèrent grandement le travail. Ces APIs sont très complètes et très faciles d'accès.

- ✓ *Facile à vendre* : Le Play Store (anciennement Android Market) est une plateforme immense et très visitée ; c'est donc une mine d'opportunités pour quiconque possède une idée originale ou utile.
- ✓ *Flexible* : Le système est extrêmement portable, il s'adapte à beaucoup de structures différentes. Les smartphones, les tablettes, la présence ou l'absence de clavier ou de trackball, différents processeurs... On trouve même des fours à micro-ondes qui fonctionnent à l'aide d'Android.
- ✓ *Ingénieux* : L'architecture d'Android est inspirée par les applications composites, et encourage par ailleurs leur développement. Ces applications se trouvent essentiellement sur internet et leur principe est que vous pouvez combiner plusieurs composants totalement différents pour obtenir un résultat surpuissant. Par exemple, si on combine l'appareil photo avec le GPS, on peut poster les coordonnées GPS des photos prises. [18]

6. Conclusion :

Dans ce chapitre, nous avons donné une vision globale sur les applications Android de laquelle, on peut conclure l'intérêt et la grande importance d'un tel outil pour le développement d'applications puissantes et efficaces. Nous commencerons au deuxième chapitre en parlant du le suivi parentale des élèves.

CHAPITRE 02

LE SUIVI PARENTAL DES ÉLÈVES

- Principe
- Objectifs de suivi
- Outils de communication
- Les effets de la collaboration famille-école sur les élèves
- Efficacité de communication
- Obstacles et défis
- Applications web pour la gestion de vie scolaire
- Système existants de suivi
- Perspective de l'absence parentale

1. Introduction

Dans ce chapitre, nous allons voir quel est le principe et l'objectif de la surveillance parentale de leurs enfants à l'école, puis nous présenterons certains des sites Web et des applications qui ont des fonctions similaires avec notre projet d'application. Ce chapitre portera donc sur les parents, l'école, les enfants des parents et les relations entre eux.

2. Principe

Le concept de suivi parental a longtemps prospéré en sociologie de l'éducation pour mettre en relief les inégalités sociales reproduites par l'école (Nash 1990) et aussi en psychopédagogie pour rendre compte des facteurs contribuant à la réussite ou à l'échec scolaire. Pour l'une ou l'autre spécialité, le suivi parental désigne l'ensemble des interactions entre parents et enfants axées sur l'école. Ainsi, les dites interactions constituent un ensemble d'actions qui tournent autour d'un seul but, qui est la réussite scolaire de l'enfant. Comme le souligne Claes (2001), les interactions parent-enfant axées sur l'école représentent une dimension primordiale de la participation parentale qui se résume à l'encadrement et à la supervision des activités académiques de l'enfant hors du milieu scolaire. Il s'agit de l'ensemble des dispositions et stratégies mises en œuvre en dehors de l'école par les parents pour accompagner les activités scolaires de l'enfant en vue de sa réussite. Si cet accompagnement arrivait à manquer à l'apprenant, on parle plutôt de "démission parentale" pour rendre compte de l'échec scolaire (Laurence 2008).

Le suivi parental se manifeste de diverses manières. Pour certains penseurs, le simple fait que l'enfant soit issu de parents appartenant à la classe dominante de la société le prédispose à la réussite scolaire, d'autant plus que l'école ne fait que perpétuer les normes et valeurs culturelles de la dite classe, tandis que l'enfant issu du bas peuple est prédisposé à l'échec scolaire, à moins qu'il se transforme complètement, l'école reniant ses normes et valeurs culturelles (Coleman 1996, Bourdieu et Passeron 1970). D'autres chercheurs trouvent que les exercices et autres devoirs de maison, y compris l'apprentissage des leçons permettent aux parents de contrôler l'évolution scolaire des enfants (Borelli 1966, Noizet et Caverni 1978).

Pour Deslandes et Potvin (1998), le suivi parental se manifeste sur le double plan comportemental et verbal. Sur le plan comportemental, le suivi se caractérise par l'aide que les

parents apportent aux enfants dans leurs devoirs et autres exercices de maison donnés à l'école. Le suivi, c'est aussi le fait d'assister les enfants dans l'apprentissage des leçons sans oublier la participation des parents aux réunions convoquées par l'administration de l'école que fréquentent les enfants. Sur le plan verbal, le suivi parental se traduit par les encouragements, compliments, exhortations à l'autonomie, la culture de la volonté et du goût pour les études et autres échanges communicationnels sur les projets d'avenir entre parents et enfants.[19]

3. Objectifs de suivi :

- ✓ Les parents jouent un grand rôle dans l'éducation de leur enfant, presque comme l'enseignant ou plus, le suivi parental est très important pour aider l'enfant dans son école et l'améliorer. Car, sans surveillance parentale, l'enfant peut échouer dans ses études ou apprendre certains mauvais comportements de la rue qui pourraient le conduire à de très mauvaises habitudes.
- ✓ Les enseignants peuvent également aider dans ce processus, en expliquant aux élèves l'importance de leurs parents dans leur éducation, en tenant le parent au courant de l'éducation de leur enfant et en envoyant des rapports sur son comportement général en classe et en dehors.
- ✓ Honorez les parents des élèves qui coopèrent avec l'école et ceux qui les suivent, afin d'encourager davantage de parents à les suivre.
- ✓ L'école et les parents coopèrent pour étudier les besoins et les problèmes des élèves, participer à la satisfaction de ces besoins et s'efforcer de les résoudre.
- ✓ Les parents doivent travailler à la surveillance des enfants à la maison et suivre leurs activités scolaires.
- ✓ Veiller à renforcer le contrôle social au sein de la famille, ce qui se traduit par une forte capacité à fournir une orientation comportementale correcte.[20]

4. Outils de communication :

- *Agenda*

Utilisé pour communiquer sur une base quotidienne, l'agenda permet de joindre les parents en ayant un outil rapidement à portée de main. [21]



Figure 2.1 Agenda Scolaire

- *Téléphone*

Afin de parler de vive voix, en temps réel, le téléphone est l'outil idéal. Sans avoir à se déplacer, écrire ou à enregistrer un message, un simple coup de fil peut être très bénéfique pour préciser certains détails sur un événement vécu en classe. Cet outil permet de créer un lien plus fort que la communication écrite. [21]



Figure 2.2 Communication Par Téléphone

- *Rencontre parents-enseignant, vidéoconférence*

Il arrive parfois qu'une rencontre en personne soit le meilleur moyen pour communiquer. En étant préparé, en ayant confiance en soi, le parent et l'élève et en adoptant une mentalité de coéducation.[21]

- *Site Internet de classe*

En plus de donner une voix à l'élève, de travailler la littératie numérique, un site web de classe est une belle plateforme pour l'enseignant de partager ce qui se vit en classe avec les parents des élèves. Certains outils de gestion de contenu permettent même de créer une zone d'abonnement accessible par mot de passe : de quoi s'assurer de la confidentialité des jeunes.[21]

- *Portfolio numérique ou classe virtuelle*

Il existe plusieurs outils de portfolio ou de gestion de classe qui permettent de partager ce qui est vécu en classe, les apprentissages réalisés, la rétroaction fournie et les résultats des évaluations formatives ou formelles avec les parents.[21]

- *Vidéo hébergé sur Internet ou vidéoconférence*

Si vous désirez partager des informations qui peuvent être publiques, pourquoi ne pas les partager sur le Web pour qu'elles soient accessibles grâce à un simple clic Vous pourrez ainsi partager ces informations essentielles en ne les enregistrant qu'une seule fois. C'est une belle façon d'outiller les parents afin qu'ils puissent mieux soutenir les apprentissages de leur enfant.[21]

- *Journal de classe ou infolettre*

Pour ceux qui sont plus à l'aise à l'écrit, le journal de classe et l'infolettre, dans lesquels les élèves peuvent aussi participer, permettent de partager des informations qui s'adressent à tous : horaire, prochains événements ou sorties, apprentissages ou évaluations passés ou à venir, etc.[21]

- *Lettre aux parents par l'enseignant ou l'enfant lui-même*

Lors de son atelier sur la rétroaction, Alexandre Audet invite les enseignants de proposer aux élèves de rédiger une lettre à leurs parents pour leur présenter leurs apprentissages, leurs défis et les actions à poser suite à leur autoévaluation. En plus de travailler les compétences liées à la communication, les enfants sont directement impliqués dans leur propre évaluation et vivront assurément des apprentissages plus signifiants. Cette façon de faire permet aux parents de prendre connaissance de la vision de leur enfant par rapport à ses propres apprentissages.[21]

5.Les effets de la collaboration famille-école Sur les élèves.

Les jeunes de tous les niveaux, primaire et secondaire, réussissent mieux si leurs parents participent dans le suivi scolaire à l'école et à la maison Parmi les autres bénéfices associés à un plus grand partenariat entre l'école et les familles, on relève une amélioration du comportement des élèves, une plus grande motivation, un taux d'absentéisme moindre, une réduction du taux de suspension et d'abandon des études et, finalement, une attitude plus positive à l'égard des travaux scolaires et de l'école en général Sur les parents. La recherche indique que les parents d'élèves dont les enseignants favorisent la participation parentale apprécient davantage les conseils de ces derniers sur les différentes façons d'aider leur enfant et les renseignements offerts sur les divers programmes. Ils ont aussi une plus grande confiance dans les habiletés pédagogiques et interpersonnelles des éducateurs Sur les enseignants et sur l'école. Lorsque les parents participent au suivi scolaire, les enseignants ont généralement une attitude plus positive à l'égard de leurs compétences parentales, indépendamment de leur niveau socio-économique et de leur degré de scolarité). Ils acquièrent également une attitude plus positive à l'égard de l'école et de l'enseignement, bref le climat de l'école s'améliore et l'école devient plus efficace.[22]

6.Efficacité de communication : [23]

Quels moyens utilise-t-on pour communiquer efficacement avec les parents ?

Dans ce qui suit , nous répondrons à la question en détail :

↳ *Communications fréquentes*

- Carnet de bord hebdomadaire entre les parents et l'enseignant, message quotidien dans l'agenda de l'enfant.
- Joindre les parents (appel ou courriel) pour souligner les progrès, les défis des élèves.
- Développer une démarche pour contacter les parents aux premiers signes de problème.
- Faire part aux parents de ce qu'on apprécie chez leur enfant.
- Penser à ne pas envoyer toutes les informations le même jour (ex.: plan de travail, signatures, informations diverses).
- Selon le type de milieu, de parents, ne pas surcharger d'informations.

↳ *Communications variées*

- Varier les médiums de communication utilisés : contact téléphonique, courriel, rencontre, lettres, prospectus, mémos, invitations personnelles faites par les élèves.
- Utiliser un vocabulaire adapté aux parents, utiliser des pictogrammes au besoin (Boardmaker, Écrire avec des symboles).

↳ *Communications attrayantes*

- Utilisation des technologies de l'information (audio, vidéo) , Documents rédigés et illustrés par le jeune .
- Présentation originale, humoristique ou fantaisiste.
- Utilisation d'images.

↳ *Communications claires*

- Stratégies de communications expliquées aux parents et élaborées avec eux préalablement.

- S'entendre d'une façon de communiquer avec les parents faibles lecteurs ou dont la langue maternelle n'est pas le français
- Vocabulaire simple et phrases concises.

↪ *Communication visant l'ensemble des parents*

- convenir avec le parent du meilleur moment pour communiquer avec celui-ci et profiter de l'occasion pour établir des temps de rencontre au moment opportun.
- Ex : lors de la première de rencontre de parents en début d'année, prévoir un formulaire sur lequel les parents indiquent leurs moments opportuns pour les rejoindre.

7.Obstacles et Défis :

Mais ce travail présente de nombreuses difficultés.

- Tout d'abord, le manque de communication entre l'école en général et le parent, soit le manque de communication entre les parents et les enseignants, soit le manque de communication entre l'administration et les parents à propos de leur enfant, cela pourrait causer de nombreux problèmes à l'avenir pour les enfants.
- Deuxièmement, en utilisant l'enfant comme canal de communication entre l'école et les parents, l'enfant peut cacher des informations comme ses notes et ses notes, provoquant une mauvaise communication entre les parents et l'école.
- Troisièmement, de nombreux parents ignorent les bons moyens éducatifs de s'occuper des enfants. Les parents doivent s'améliorer en assistant à des conférences et en lisant des livres qui les aident à suivre correctement et efficacement leurs enfants.
- Dans l'ensemble, pour que l'enfant reçoive la meilleure éducation possible et pour résoudre le plus rapidement possible tous les problèmes auxquels l'enfant peut faire face dans son éducation, ou même éviter que certains problèmes ne se produisent, les

parents doivent surveiller leurs enfants. et travaillez en étroite collaboration avec l'école en utilisant les bonnes techniques pour obtenir les meilleurs résultats. [20]

8. Applications Web pour la gestion de la vie scolaire :[24]

↳ Écoles primaires

Il existe peu d'ENT et d'applications de vie scolaire spécifiquement conçus pour l'école primaire. En voici deux :

- ❖ *Netécole. ENT* : plutôt complet, avec bonnes fonctionnalités Vie scolaire. Edité par ItopEducation, qui propose également Netcollège et Netlycée.
- ❖ *Iconito Vie scolaire* : Application libre et gratuite de vie scolaire, spécifiquement développée pour l'école primaire, par un spécialiste de la petite enfance. Bonnes fonctionnalités, juste ce qu'il faut pour l'école élémentaire.

↳ Écoles secondaires

Une offre pléthorique d'ENT et d'applications spécifiques pour la Vie scolaire.

- ❖ *Triade* : Logiciel libre sous licence GPL. Tout en ligne, aucune installation sur poste. ENT complet, comprenant de nombreuses fonctions Vie scolaire. Tarif : à partir dun euro par élève et par an. Triade est utilisé dans de nombreux pays, notamment sur le continent africain.
- ❖ *Scolaris* : aAcolaris est une application de vie scolaire utilisable sur poste ou en ligne intégrée à un ENT.
- ❖ *Educ Horus* : Application de vie scolaire tout en ligne pour collèges et lycées. Educ Horus permet d'accéder aussi bien aux fonctions de vie scolaire qu'à un ENT, pour un tarif unique quelle que soit la taille de l'établissement.
- ❖ *Gepi* : Logiciel libre et gratuit de Vie scolaire. S'utilise sur poste et en ligne. De très nombreuses fonctions, accessibles au travers de 20 modules. Gepi est un projet en évolution constante, la dernière version stable date de janvier 2012.

- ❖ *Mestek. ENT* : avec importantes fonctionnalités de Vie scolaire, c'est un outil "tout en un" accessible en ligne. Distribué par Camif collectivités, une entreprise très connue des enseignants français et des établissements scolaires. On aimerait juste en savoir un peu plus sur les tarifs...
- ❖ *GPI-* : Gestion intégrée pour l'école. Produit québécois proposant de nombreuses fonctionnalités de Vie scolaire et d'administration. A noter, un module de communication par téléphonie IP, pour joindre rapidement les familles, les enseignants...

↳ **Enseignement supérieur - Formation professionnelle**

Beaucoup d'universités et écoles supérieures font développer leur propre ENT. Les fonctions de gestion administratives et financières sont assurées par le biais de progiciels. Nous n'en relevons ici que deux, qui possèdent des fonctions bien adaptées aux établissements d'enseignement. Il y en a beaucoup d'autres.

- ❖ *Alcuin* : Plusieurs offres qui s'adaptent à l'enseignement supérieur, à la formation continue, à l'enseignement professionnel. Accès en ligne. Fonctions de gestion des inscriptions et des candidats, différentes modalités de validation des acquis et des examens, gestion administrative et comptable. A noter, d'intéressantes fonctions de gestion des périodes de stages en entreprises et des contrats d'apprentissage.
- ❖ *Auriga* : Applications de gestion de l'enseignement supérieur et professionnel. Gère la scolarité, les ressources, les partenariats, la communication...

↳ **Autres applications**

Voici deux produits qui permettent d'assurer la gestion de services et établissements particuliers, à la périphérie de l'enseignement obligatoire.

- ❖ *Aiga* : Une application pour gérer la restauration et les transports scolaires. Il existe également un module pour la création de classes / d'écoles. Une entreprise spécialisée dans les solutions informatiques pour le secteur social et le secteur associatif.
- ❖ *WelcoeM* : Application gratuite pour la gestion administrative, pédagogique et financière des écoles de musique

9. Systèmes existants de suivi

Nous avons examiné trois sites Web pour évaluation, deux sites étrangers et un site local. Nous avons pris en considération trois critères (principe, conception, limites et inconvénient), dans le critère limites et inconvénient, il y a beaucoup d'applications et d'applications Web pour cela, mais à des fins de démonstration, nous avons choisi ces trois.

↳ **Klassroom app :**



Figure 2.5 classroom app logo

Principe : cette application est conçue pour être utilisée par l'enseignant et les parents afin de faciliter la communication entre eux.

Conception : cette application en conception est comme une page Facebook juste pour vos propres élèves, elle vous permet de partager et de publier des images, des vidéos, des textes, tout comme Facebook, et seuls les parents peuvent y entrer, de plus, les parents peuvent discuter avec les enseignants pour leur demander tout ce qu'ils veulent, comme représenté sur la Figure ci-dessous, dans la partie droite est la première interface que vous obtenez lorsque vous entrez dans l'application, ici vous pouvez voir toutes vos classes, en ajouter de nouvelles, voir vos notifications, etc. Et dans la partie gauche est un exemple de votre flux de pages de classe.

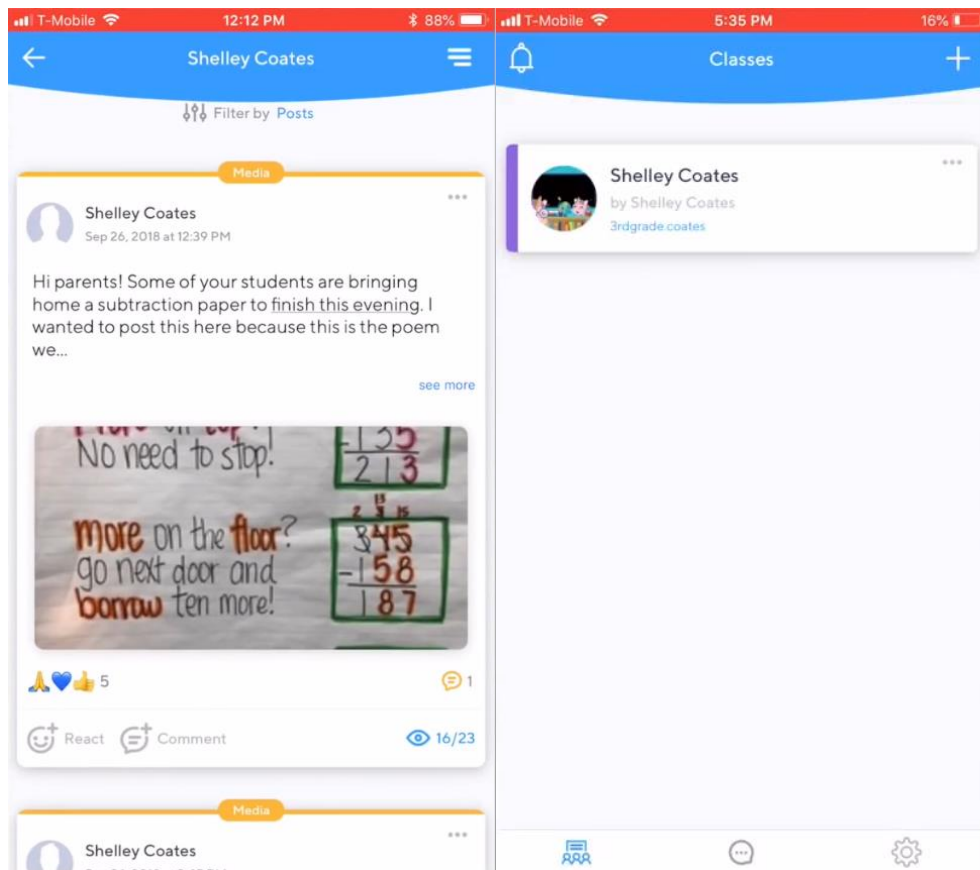


Figure 2.6 la page de classes et exercice

Limites et inconvénient : l'inconvénient est que les parents ne peuvent pas parler entre eux à moins d'avoir obtenu l'adhésion, et ils ne peuvent pas télécharger de photos ou de vidéos, plus si un parent voulait voir les notes ou les absences de leur fils ou toute information qu'ils ne peuvent pas, le seul le contact est entre l'enseignant et le parent, si un parent veut savoir quelque chose, alors il doit demander à l'enseignant, ce qui est gênant si un groupe de parents veut connaître les mêmes informations en même temps et en secret parce qu'ils ne le savent pas veulent que quiconque voie leurs enfants notes.

↪ **Class Dojo app :**



Figure 2.7 le logo de class dojo app

Principe : ClassDojo est une plateforme de gestion de classe en ligne où les enseignants peuvent enregistrer et suivre le comportement des élèves, faciliter les activités en classe et communiquer avec les parents.

Conception : ClassDojo est un outil que les enseignants peuvent utiliser en classe du début à la fin. L'enseignant crée une véritable classe et ajoute des élèves, la connexion des élèves est simple avec un code QR, un lien ou un compte Google, chaque élève aura son propre avatar qu'il peut changer ou modifier comme représenté sur la Figure ci-dessous, puis vous pouvez effectuer différentes opérations, leur attribuer des notes (des plus et des moins) en fonction des activités que vous effectuez en classe, vous pouvez également vérifier la présence et l'absence d'élèves ainsi que toutes les opérations que vous faisiez en classe.

Limites et inconvénient : de la mise en page de cette application, il est évidemment pour les enseignants du collège, cette application rendrait facilement les jeunes enfants accros à l'utilisation du smartphone car elle oblige les élèves à l'utiliser pendant les cours et parce que c'est une interface graphique écrasante.

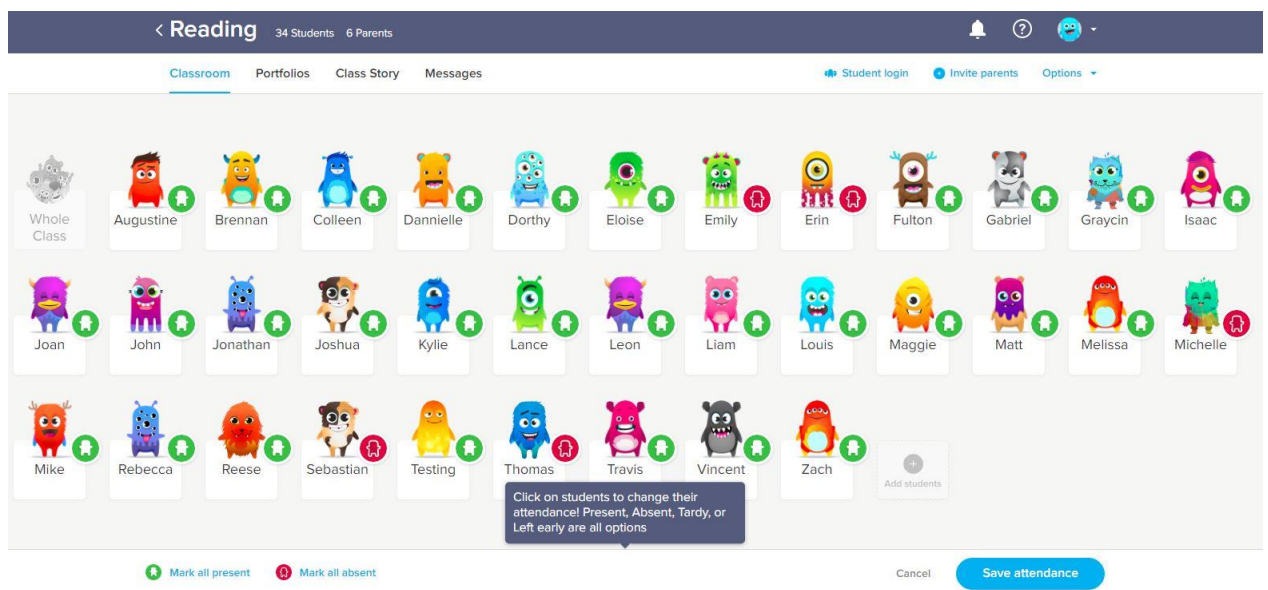


Figure 2.8 classroom app

↳ le système de l'éducation national:

le système de l'éducation national utilise une plateforme numérique contient tous les information pédagogiques des élèves , ces derniers avec leurs profs ainsi l'administration ont des droits d'accès avec des permissions différents.

Site Web « d'Awliya's Space » pour connaître les tarifs, les résultats et les listes d'élèves de chez eux au sein du Ministère de l'éducation nationale :

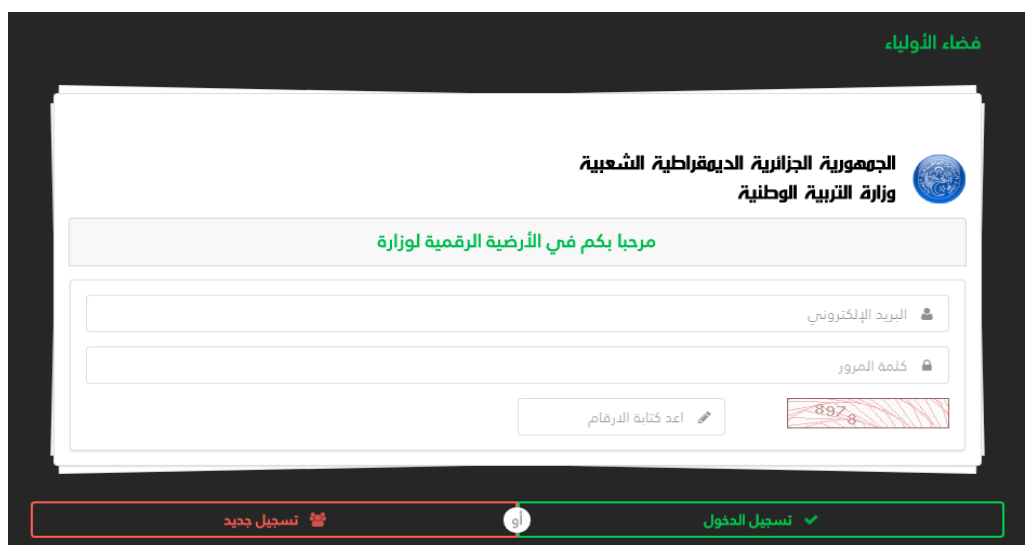


Figure 2.9 page de login et inscription de site

Lorsque vous connectez, voici la page d'accueil:



Figure 2.10 page principale de site.

Tout d'abord, l'en-tête, lorsque vous survolez la phrase d'accueil dans le rectangle jaune, le pointeur de la souris indique qu'il est cliquable, mais il ne s'agit que de texte brut.

Ensuite, l'élément dans le rectangle rouge, lorsque vous passez la souris dessus, le pointeur de la souris indique que cette zone est cliquable, et lorsque vous survolez les phrases de remorquage dans cette zone, chaque phrase devient blanche, indiquant qu'il s'agit d'un lien vers un autre différent page, mais ce n'est pas le cas, l'élément dans le rectangle rouge et les phrases de remorquage sont tous des liens vers la même page.

Toutes les phrases ne deviennent pas blanches lorsque vous les survolez, ce qui rend le design différent pour tous les autres éléments. La zone dans le rectangle noir n'est pas cliquable, tandis que les zones dans le cercle rouge sont cliquables mais ne mènent à aucune page. La zone dans le cercle noir est la même, en cliquant dessus, nous restons en page d'accueil.

En conclusion, ce site est soit incomplet, soit mal réalisé, le design global est acceptable si toutes les notes que nous avons prises sont corrigées, dans l'ensemble, ce site a encore besoin de beaucoup d'améliorations.

10. Perspective de l'absence parentale




La perspective de l'absence parentale suppose que la présence des deux parents biologiques est nécessaire au développement de l'identité sexuelle de l'enfant (Acock & Demo, 1994). En vertu de cette théorie, l'enfant privé de ses parents manifesterait des déficits de socialisation qui se traduiraient notamment par des problèmes au plan de son adaptation (Amato & Keith, 1991b). Cette perspective va dans le sens du modèle du déficit familial de Marotz-Baden et ses collègues (1979) qui conçoivent toute structure familiale qui s'éloigne du modèle dit traditionnel, c'est-à-dire une famille nucléaire intacte composée des deux parents biologiques, comme débilante pour l'enfant. Bien que toujours présente ou implicite dans la littérature récente, cette perspective perd en popularité compte tenu de l'importance qu'elle accorde à la structure plutôt qu'aux processus familiaux (St-Jacques, 1996). Selon Amato et Keith (1991b), si les parents constituent une ressource importante dans le développement de l'enfant, la présence de deux parents devrait être plus efficace que la présence d'un seul.[25]

11. Conclusion :

Pour activer le rôle de l'école et accroître les compétences pédagogiques, il doit y avoir une coopération entre l'école et les parents pour suivre les activités d'étude des enfants. Pour parvenir à une intégration académique et pédagogique entre la famille et l'école, elle est soumise à plusieurs facteurs liés à la famille à ses différents niveaux (social, économique, culturel) et à l'environnement scolaire et social des enfants, ainsi qu'à tout déséquilibre ou manque du suivi des enfants et de la sensibilisation des enfants à la famille. Nous allons entamer la conception de notre projet dans ce qui suit, pour identifier les différentes fonctionnalités de l'application.

CHAPITRE 03

ANALYSE ET CONCEPTION DU SYSTÈME

-  Analyse et Spécification des besoins
-  Présentation d'UML
-  Les diagrammes de notre système

1. Introduction :

Avant de développer notre application, nous devons d'abord expliquer son objectif en déterminant les besoins fonctionnels et non fonctionnels auxquels elle doit répondre. Ainsi, dans ce chapitre, nous présenterons l'architecture générale et expliquerons les différentes interactions entre les acteurs et les autres composants du système. En ce qui concerne la conception, nous avons utilisé le langage UML avec Star UML pour modéliser l'application et créer les différents diagrammes.

2. Analyse et spécification des besoins

Cette phase consiste à détailler l'ensemble des fonctionnalités que l'application doit fournir. Ces besoins seront présentés sous forme de besoins fonctionnels et non fonctionnels ainsi que les besoins techniques.

2.1 Expression initiale des besoins:

L'objectifs de notre future application mobile est d'assurer la bonne gestion du suivi scolaire et de faciliter le suivi des élèves afin de parvenir à une organisation performante. Cette application devra permettre de gérer facilement l'accès aux notes des élèves, leurs emplois du temps et leurs absences ainsi que les appréciations et beaucoup de choses qui sont importantes pour les parents.

2.2 Besoins fonctionnels détaillés:

Les besoins fonctionnels sont les actions fondamentales que le système doit assurer lorsqu'il répond à une requête. Ce système est comme une interface entre les parents et l'école. Elle est développée pour les parents qui souhaitent s'impliquer activement dans l'éducation de leurs enfants. Le système offre un certain nombre de services en ligne pour aider les parents dans leur démarche. Donc, l'application offre un ensemble de services scolaires administratifs et pédagogiques à ses utilisateurs tels que :

- *Gestion des notes* : ce service est pour la saisie et la consultation des notes.
- *Gestion des absences* : ce service permet de contrôler l'absence des étudiants (enregistrer et modifier l'absence)

- *Consultation de l'emploi du temps* : Ce service est pour la consultation de l'emploi du temps d'un enseignant ou d'une classe.
- *Gestion des appréciations* : ce service permet à l'enseignant d'envoyer une appréciation au parent d'un élève
- *Gestion des rendez-vous* : qui concerne la consultation et la demande des rendez-vous soit parent- enseignant ou parent-administration.

2.3 Besoins non fonctionnels détaillés :

Les principaux besoins non fonctionnels que notre application doit fournir sont la simplicité et la lisibilité, ainsi que d'autres contraintes :

- *Contraintes ergonomiques* : simplicité et convivialité des interfaces graphiques.
- *Contraintes de sécurité* : authentification, ressaisie du mot de passe lors d'un traitement dans la base de données.
- *Contraintes de performance* : accès facile, chargement rapide.
- *Contrainte de fiabilité* : sans ambiguïté

2.4 Besoins techniques

- avoir un smartphone tournant sous le système d'exploitation Android de Google.
- Une connexion de haut débit (3G, 3G++, 4G , etc.).
- Une connexion WIFI/LIFI.

3. Présentation d'UML

UML (Unified Modeling Language), se définit comme un langage de modélisation graphique et textuel destiné à comprendre et à définir des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. UML modélise l'ensemble des données et des traitements en élaborant des différents diagrammes. [26]

4. Diagramme de cas d'utilisation

Ce diagramme est destiné à représenter les besoins des utilisateurs par rapport au système. Il constitue un des diagrammes les plus structurants dans l'analyse d'un système.[27]

5.1 Acteurs

Notre application comporte 2 types d'acteurs : utilisateur et administrateur

- *L'utilisateur* représente les 3 acteurs parent, enseignant et élève où chaque acteur est un membre qui a un compte dans notre application, il utilise l'application dans la limite de son compte.
- *L'administrateur* est un utilisateur aussi, mais il a tous les droits de gestion et de contrôle

5.2 Cas d'utilisation

Un cas d'utilisation (use case) représente un ensemble de séquences d'action qui sont réalisées par le système et qui produisent un résultat observable intéressant pour un acteur particulier. Il permet de décrire ce que le futur système devra faire, sans spécifier comment il le fera [28].

Enseignant

- ✓ Saisir les absences, les notes et les appréciations des élèves.
- ✓ Mettre en ligne des devoirs à remettre.
- ✓ Répondre aux demandes de rendez-vous avec les enseignants.
- ✓ Demande de rendez-vous avec les parents.

L'administrateur (responsable de l'établissement)

- ✓ Saisir et gérer les comptes des utilisateurs.
- ✓ Modérer les messages entre parents et enseignants.
- ✓ Nettoyer le système en fin d'année scolaire pour supprimer les données qui ne sont plus utiles (absences, messages, travaux mis en ligne).
- ✓ Créer les groupes de cours et les descriptions.

- ✓ Modifier l'emploi du temps de manière temporaire.

Parent

- ✓ Consulter les notes, les appréciations, les absences et les emplois du temps de leur(s) enfant(s).
- ✓ Aider leur(s) enfant(s) dans le travail à la maison grâce à l'aide postée par les enseignants.
- ✓ Garder les informations personnelles à jour.
- ✓ Répondre aux demandes de rendez-vous avec les enseignants.
- ✓ Demande de rendez-vous avec les parents.

Élève

- ✓ Consulter les notes, les absences et l'emploi du temps.

La figure suivante représente le diagramme de cas d'utilisation général de notre application :

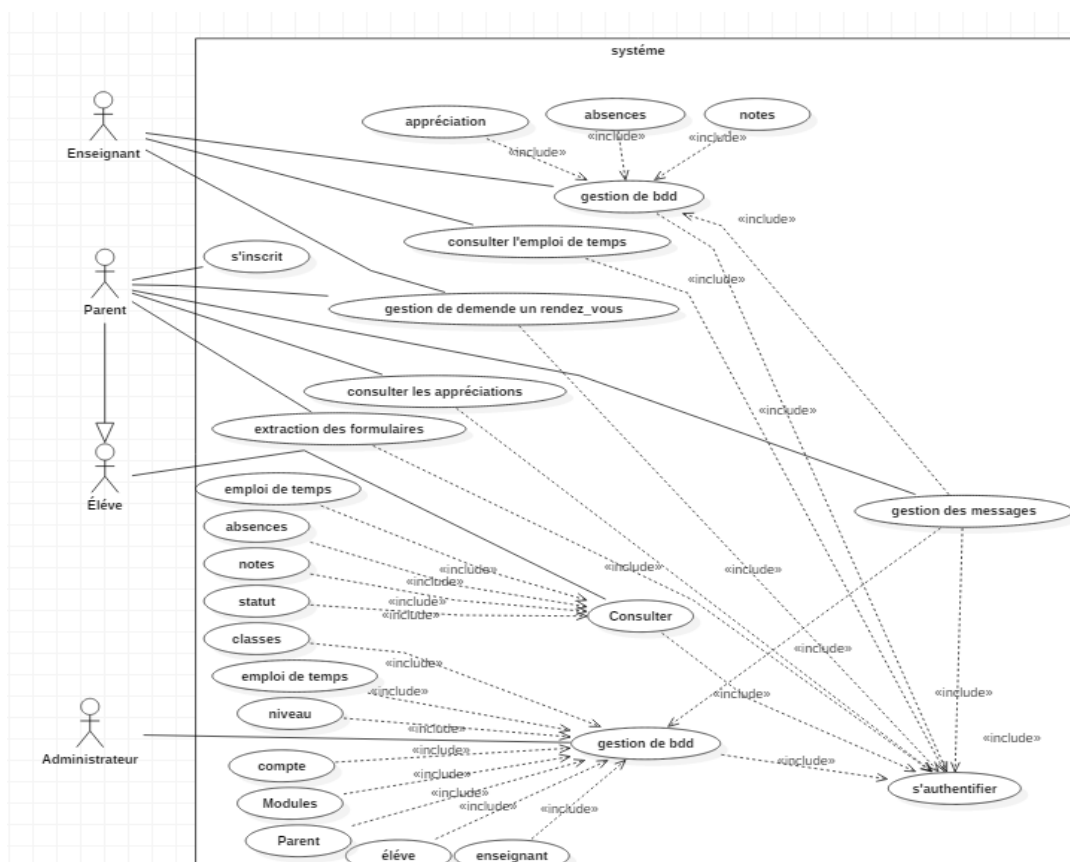


Figure 3. 1 Diagramme des cas d'utilisation

5.3 Description des cas d'utilisation

On va décrire en détails certains cas d'utilisation identifiés dans le diagramme précédent.

↳ Gestion des comptes

Voici le diagramme d'utilisation du cas « gestion des comptes » :

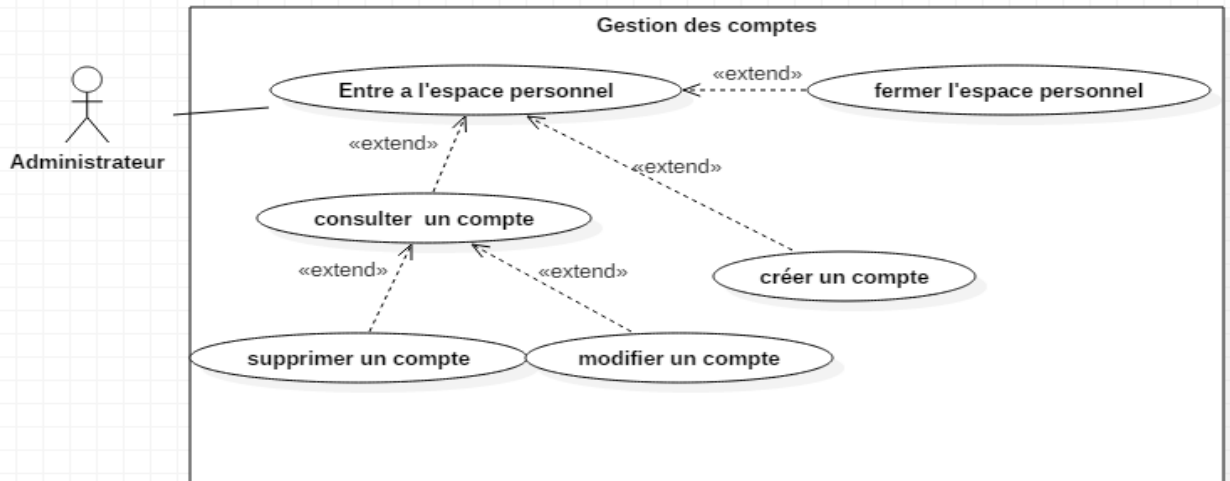


Figure 3. 2 cas d'utilisation gestion des comptes

☞ Consulter un compte

Cas d'utilisation	Consulter un compte
Acteur	L'administrateur
Description	L'administrateur veut consulter la liste des comptes.
Pré-condition	L'appareil mobile doit être connecté à Internet et L'administrateur doit être connecté dans le système
Scénario nominal	L'administrateur choisit de consulter les comptes , le système Affiche d'un tableau avec les comptes et leur données (le nom, prénom, type de compte,...)
Post-condition	La liste des comptes est affichée
Scénario d'exception	Aucune

Table 3. 1 Cas d'utilisation Consulter un compte

☞ *Créer un compte*

Cas d'utilisation	Créer un compte
Acteur	L'administrateur
Description	L'administrateur veut créer un compte. Il choisit type d'utilisateur. Un login et un mot de passe sont générés automatiquement et sont affichés pour ce compte.
Pré-condition	L'appareil mobile doit être connecte à Internet et L'administrateur doit être connecté dans le système
Scénario nominal	L'administrateur choisit de créer un nouveau compte le système Affiche un formulaire L'administrateur remplit les champs le système Enregistre le compte dans le système
Post-condition	Le compte est créé avec succès
Scénario d'exception	Compte pas crée, message d'erreur doit s'afficher à l'administrateur

Table 3. 2 Cas d'utilisation créer un compte

☞ *Modifier un compte*

Cas d'utilisation	Modifier un compte
Acteur	L'administrateur
Description	L'administrateur veut modifier un compte existant
Pré-condition	L'appareil mobile doit être connecte à Internet et L'administrateur doit être connecté dans le système
Scénario nominal	L'administrateur choisit de modifier un compte, le système Affiche la liste des comptes, L'administrateur choisit un compte dans la liste, le système Affiche des informations sur le compte choisi,

	L'administrateur modifie les champs souhaités et valide, le système vérifie les données et enregistre les modifications
Post-condition	Le compte est modifié avec succès
Scénario d'exception	Modification échoue, message d'erreur doit s'afficher à l'administrateur

Table 3. 3 Cas d'utilisation modifier un compte

↳ **Gestion des entités**

La gestion des entités contient les 3 gestion (gestion des notes , gestion des appréciation , gestion des absences)

Voici le diagramme d'utilisation « gestion des entités » :

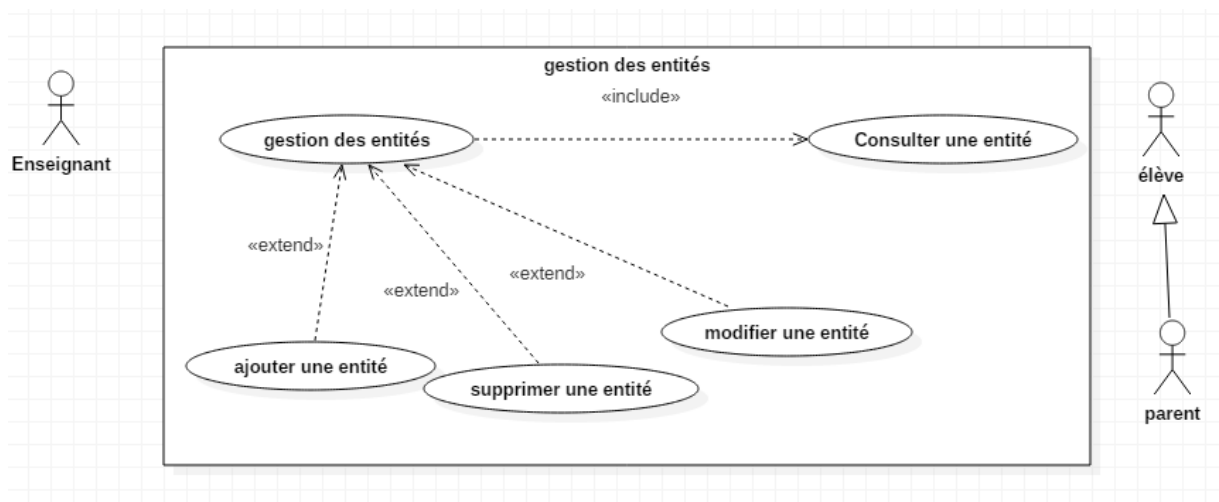


Figure 3. 3 cas d'utilisation gestion des entités

☞ *Consulter une entité*

Cas d'utilisation	consulter une entité
Acteur	L'utilisateur (Parent, Elève ou Enseignant)
Description	L'utilisateur veut voir les entités d'un de ses enfants pour la session en cours. La liste des modules que suit son enfant est affichée, il choisit un module parmi cette liste et les notes sont affichées
Pré-condition	L'utilisateur doit être connecté à ce compte

Scénario nominal	L'utilisateur choisit de consulter une entité , le système Affiche la page pour consulter les entités . La liste des modules suivis par l'élève est affichée , L'utilisateur choisit le module dans lequel il veut voir l'entité de son fils, le système Affiche les informations relatives
Post-condition	L'entité est affichées
Scénario d'exception	Aucune

Table 3. 4 Cas d'utilisation consulter les entités

- **Saisir une entité**

Cas d'utilisation	saisir une entité
Acteur	Enseignant
Description	Un enseignant veut saisir une entités . Il choisit tout d'abord la classe concernée. Ensuite il choisit l'élève désiré et saisit les informations concernant l'entité.
Pré-condition	L'enseignant doit être connecté à ce compte
Scénario nominal	L'enseignant choisit de saisir l'entité , le système affiche la liste des classes suivant ce module, L'enseignant choisit la classe souhaitée, le système Affiche la page pour saisir des entités .le système Affiche la liste des élèves, l'enseignant saisit l'entité , le système fait une vérification et enregistre des données .
Post-condition	Entité saisie avec succès
Scénario d'exception	La saisie d'entité échoue, message d'erreur doit s'afficher à l'administrateur

Table 3. 5 Cas d'utilisation saisir une entité

↳ **Authentification**

Cas d'utilisation	Authentification
Acteur	Parent , administrateur , enseignant, élevé
Description	Authentifier l'utilisateur à partir de son email et mot de passe
Pré-condition	L'appareil mobile doit être connecte à Internet
Scénario nominal	Saisir l'email et mot de passe, puis valider
Post-condition	L'authentification avec succès
Scénario d'exception	Authentification échoué, message d'erreur doit s'afficher à l'utilisateur

Table 3. 6 Cas d'utilisation authentification

↳ **Inscription**

Cas d'utilisation	Inscription
Acteur	Utilisateur (parent)
Description	L'utilisateur crée un compte dans l'application
Pré-condition	L'appareil mobile doit être connecte à Internet
Scénario nominal	Le parent remplir la formulaire affiche(les données : nom , prénom , mot de passe,.....) et valider
Post-condition	L'inscription avec succès
Scénario d'exception	Inscription échoué, message d'erreur doit s'afficher à l'utilisateur

Table 3. 7 Cas d'utilisation inscription

6. Diagrammes des séquences

Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation Unified Modeling Language.[29]

↳ Authentification

L'authentification consiste à assurer la confidentialité des données, elle se base sur la vérification du email et du mot de passe , Ce diagramme, est décrire le scénario du cas d'utilisation d'authentification. L'utilisateur (enseignant et Administrateur, élève, parent) accès a l'application si il a fait le login précédemment et quitter l'application sans déconnecté alors l'application afficher la page principale d'utilisateur concerné , sinon il afficher la page d'authentification . l'utilisateur saisit les données d'authentification. Le système va vérifier ses droits d'accès dans la base de données et lui répondre par la suite soit par passage à son espace soit par un message d'erreur.

La figure suivante illustre une description détaillée du ce scénario :

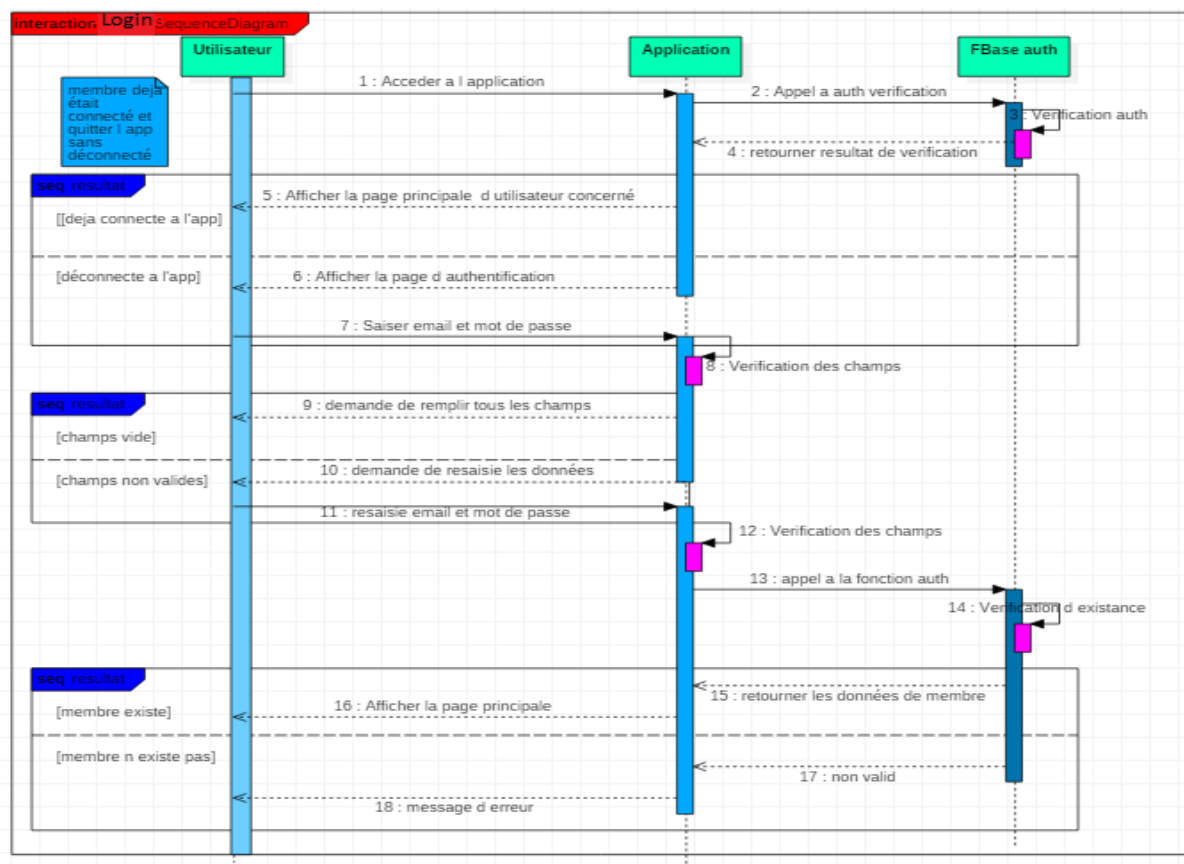


Figure 3. 4 Diagramme de séquence de l'authentification.

↳ Inscription

Chaque parent peut demander a l'inscrit de système , Ce diagramme, est décrire le scénario du cas d'utilisation d'inscription . le parent accès a la page d'inscription , l'application afficher la formulaire d'inscription , le parent saisit les données dans tous les champs et cliquer sur le bouton inscrit pour la validation , le système sauvegarder les données au base de données. Le scénario complet illustré dans la figure suivante :

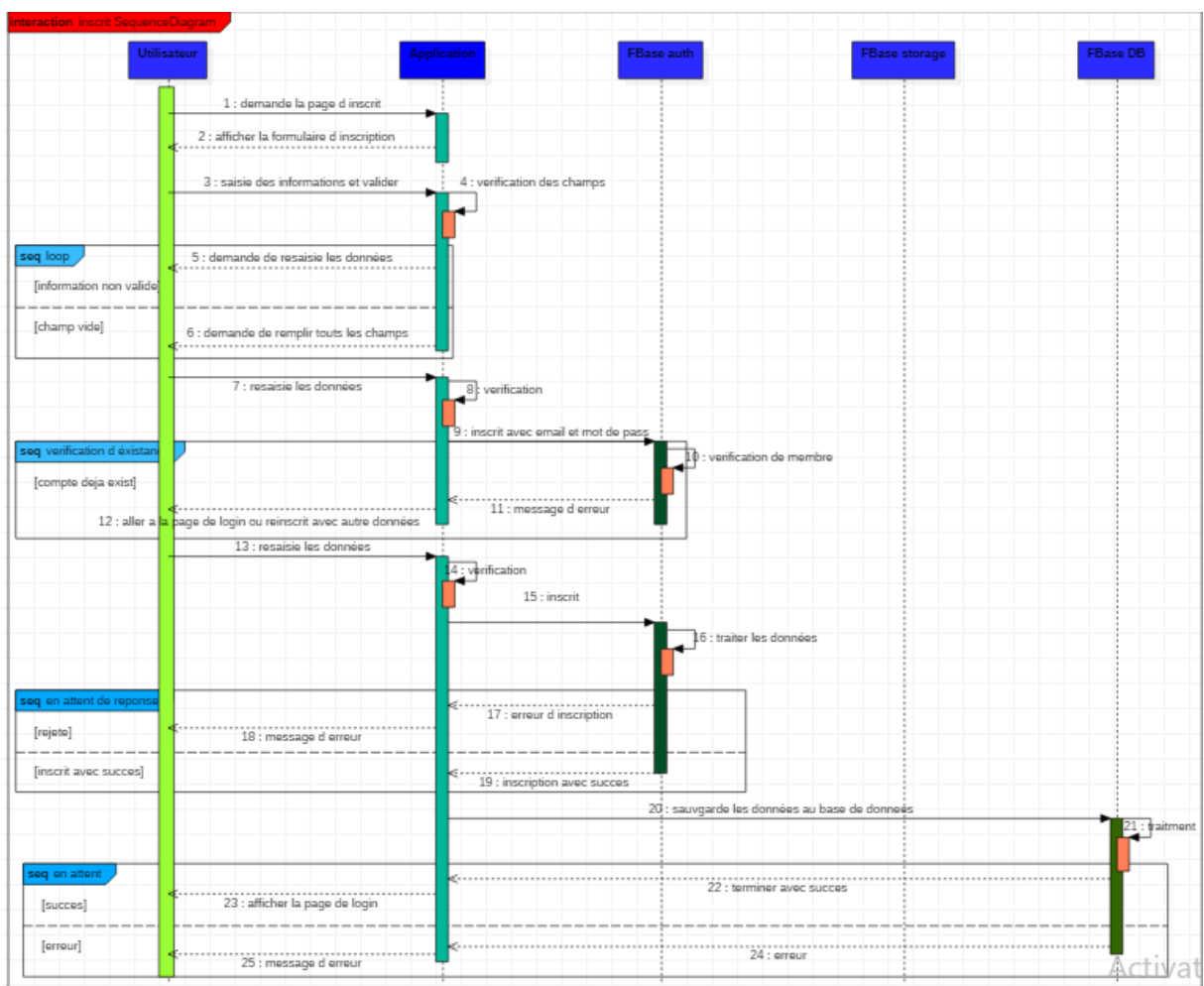


Figure 3. 5 Diagramme de séquence de l'inscription.

↳ Consulter les notes :

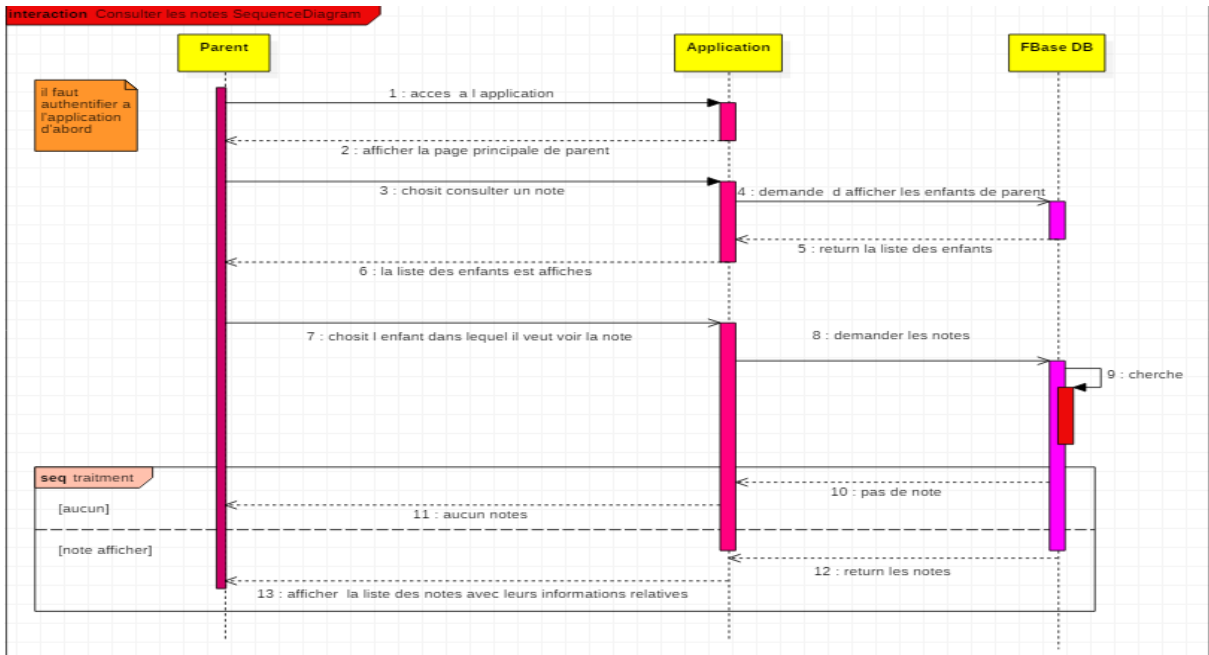


Figure 3. 6 Diagramme de séquence de Consulter les notes.

↪ Ajouter les appréciations

La figure suivante illustre une description détaillée de ce scénario :

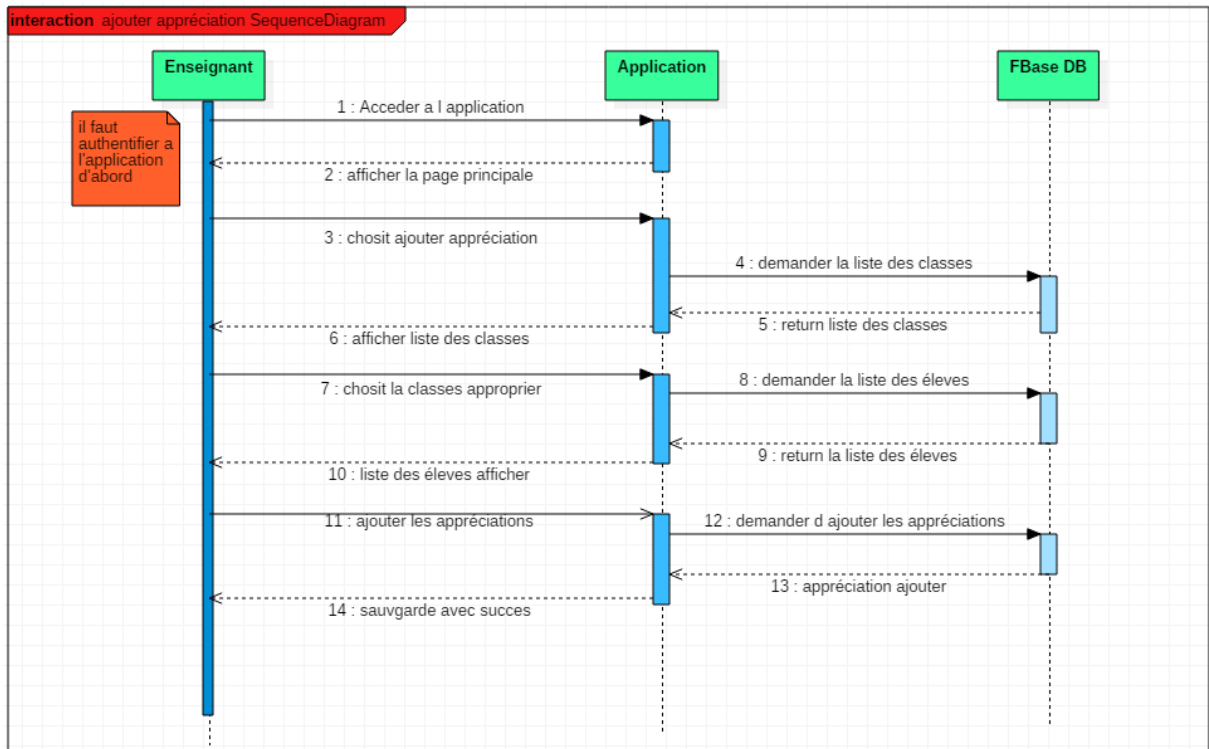


Figure 3. 7 Diagramme de séquence de ajouter les appréciations.

7. Diagramme des classes

Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci. Ce diagramme fait partie de la partie statique d'UML car il fait abstraction des aspects temporels et dynamiques.[30]

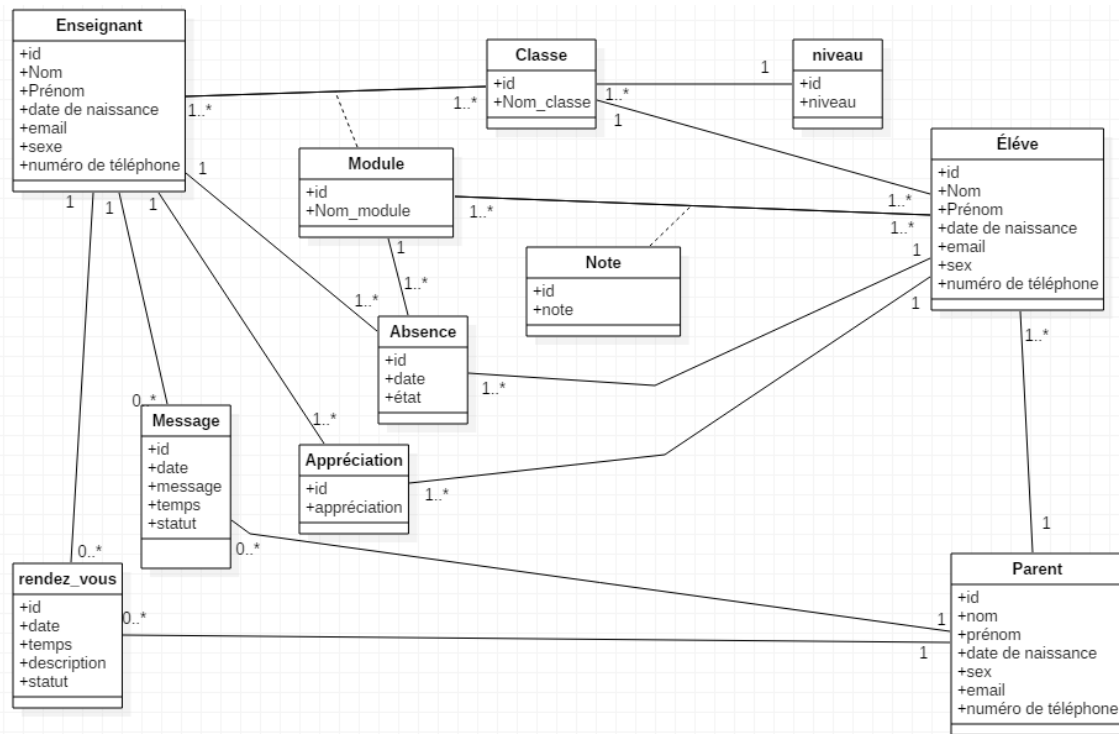


Figure 3. 8 Diagramme des classes.

8. Conclusion

Dans ce chapitre, nous avons présenté et expliqué les besoins fonctionnels et non fonctionnels auxquels notre application doit répondre, ainsi que les différents diagrammes UML pour comprendre les interactions entre les différents acteurs et composants de notre système. Ce qui nous permet de commencer le développement de l’application.

CHAPITRE 04

IMPLÉMENTATION

- Architecture
- Présentation de l'environnement et des outils de développement
- Les interfaces de l'application

1. Introduction

Dans ce chapitre, nous allons parler de la partie pratique de notre application. Tout d'abord, nous présentons les outils et les environnements sur lesquels nous travaillons, ainsi que le matériel et les logiciels que nous avons utilisés, enfin nous mentionnons la base de données avec laquelle nous avons choisi de travailler, et nous concluons par une présentation de notre application ainsi que la démonstration de notre application via des captures d'écran.

2. Architecture :

Notre application est présentée sous la forme d'une architecture client/serveur.

☞ Client

Dans notre projet, le client s'agit de l'application cliente, 3 utilisateurs (Parent, élève, enseignant) et l'administrateur qui gère les comptes avec plusieurs fonctionnalités.

☞ Serveur

Toutes les données dans notre application sont dans le serveur de Google (Firebase).

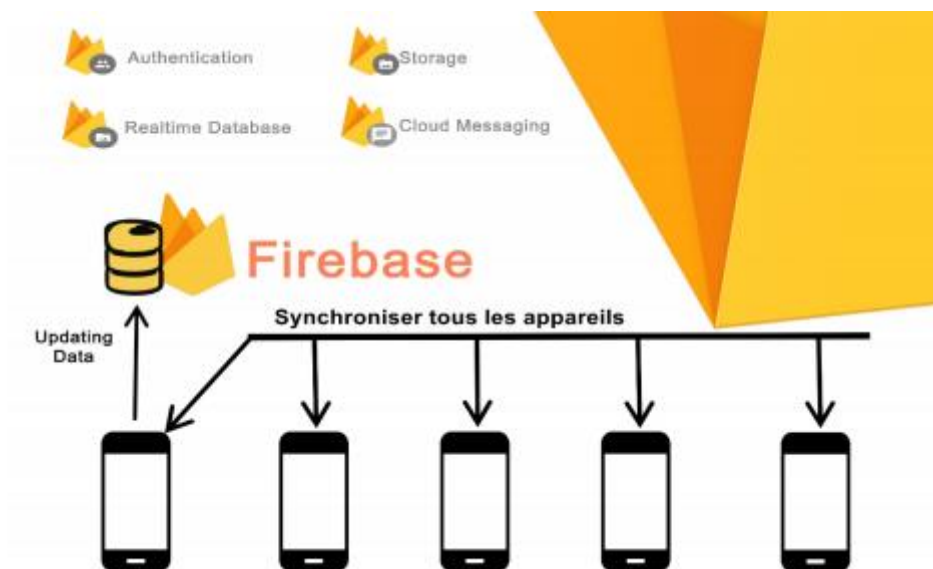


Figure 4. 1 Architecture Client / Server

3. Présentation de l'environnement et des outils de développement

3.1 Environnement matériel :

Notre application est réalisée sur un pc portable avec un Smartphone pour l'exécution dont les caractéristiques sont résumées dans le tableau suivant :

	Micro-ordinateur (Développement)	Smartphone (Exécution)
Marque	Lenovo	Redmi7A
Processeur	Intel R Core TM i5	Octa-core Max 2.01 GHZ
Disque dur	462 Go	32 Go
RAM	8 Go	2 Go
Système	Windows 10	10 QKQ

Table 4. 1 : Présentation de l'environnement de travail utilisé

3.2 Environnement logiciel :

❖ *Androide Studio:*

Android Studio est un nouvel environnement pour développement et programmation entièrement intégré qui a été récemment lancé par Google pour les systèmes Android. Il a été conçu pour fournir un environnement de développement et une alternative à Eclipse qui est l'IDE le plus utilisé. Lorsque vous commencez un nouveau projet avec Android studio, la structure des projets apparaîtra avec pratiquement tous les fichiers du répertoire SDK, ceci vous redirigera vers un système de gestion conçu en Gradle offrant encore plus de flexibilité au processus de construction.[31]



Figure 4. 2 Logo android studio

❖ *jdk:*

Le JDK (Java Development Kit), qui contient le JRE (afin d'exécuter les applications Java), mais aussi un ensemble d'outils pour compiler et déboguer votre code ! Vous trouverez un peu plus de détails sur la compilation dans l'annexe sur l'architecture d'Android.[32]



Figure 4. 3 Logo de jdk

4. Implémentation de base de données

Pour implémenter notre base de données nous avons utilisé la base de données Firebase qui est de type No SQL.

4.1 No SQL

Appelée également « Not Only **SQL** » (pas seulement **SQL**), la base de données **NoSQL** est une approche de la conception des bases et de leur administration particulièrement utile pour de très grands ensembles de données distribuées.[33]

4.2 Firebase

Firebase est un ensemble de services de haut niveau pour le développement d'applications web ou mobile. Firebase offre en particulier des services très bien réalisés pour la gestion des utilisateurs, des notifications, du stockage de fichiers, des bases de données...

Il propose d'héberger en NoSQL et en temps réel des bases de données, la création de Firebase.google.com en 2011 sous le nom ENVOLVE par James Tamplin et Andrew Lee [34]

Pour Intégrez Firebase dans notre application Android, il faut :

- Pour commencer, vous devez bien sûr avoir un compte Google. Ensuite, accédez au lien de la console Firebase

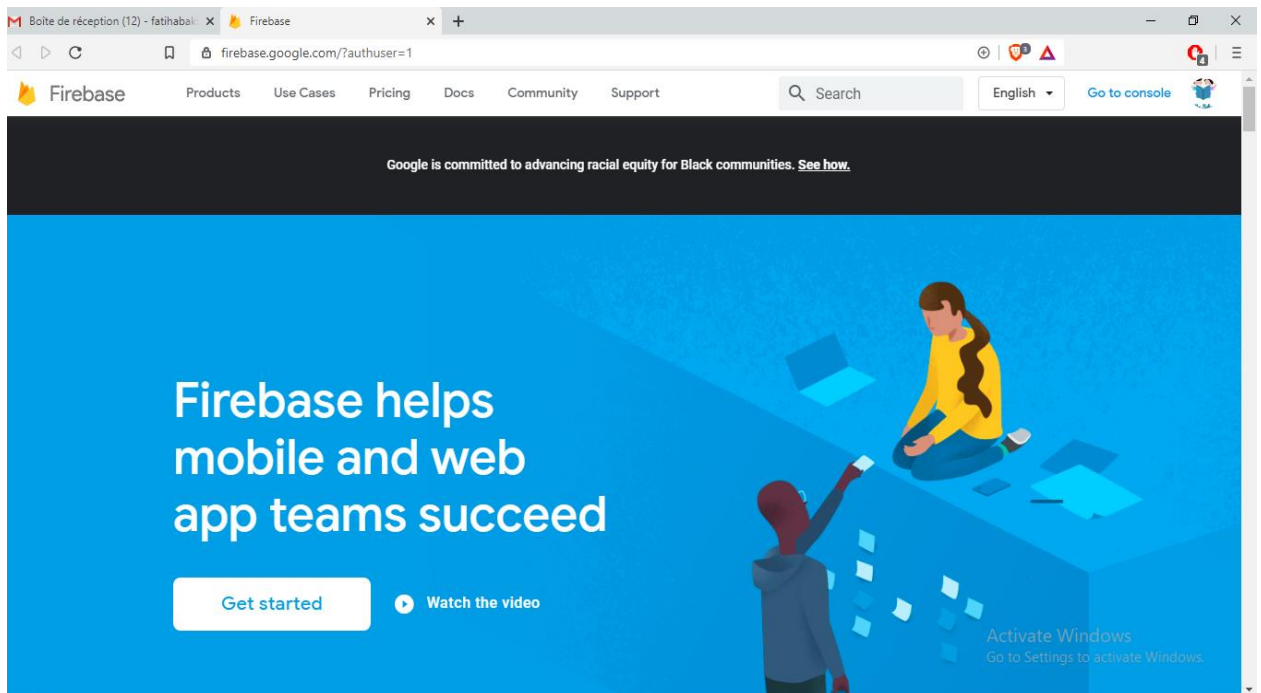


Figure 4. 4 Interface initial de firebase.

- puis, nous choisissons Ajouter un projet

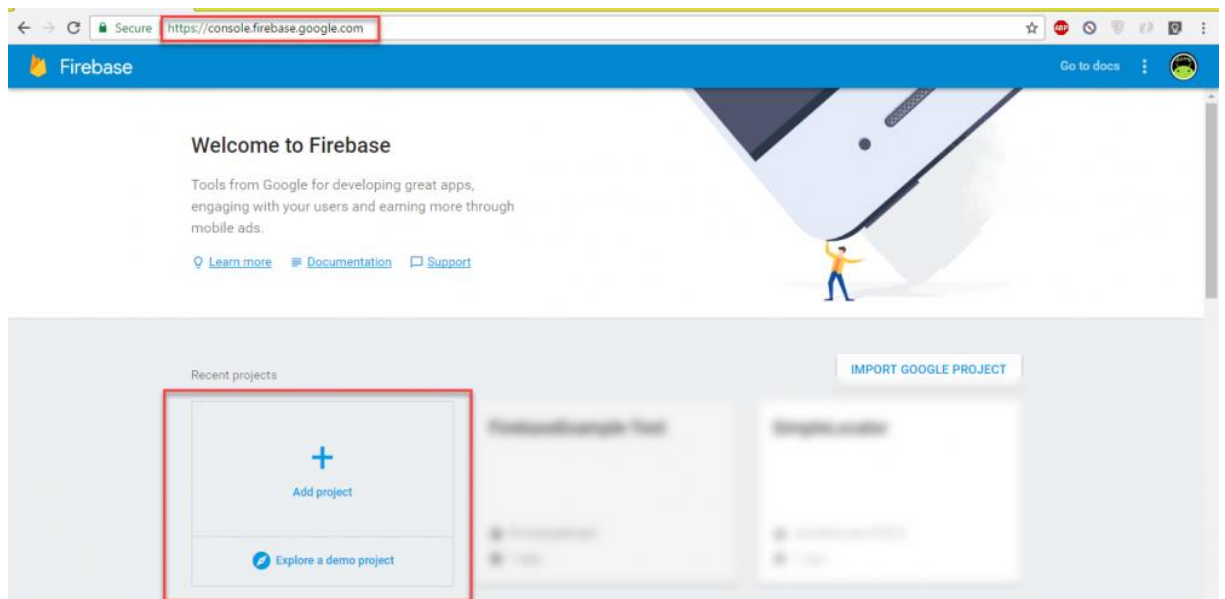


Figure 4. 5 ajouter un projet dans le firebase.

- Enregistrer l'application dans la plateforme de Firebase en remplissant les champs nécessaires.

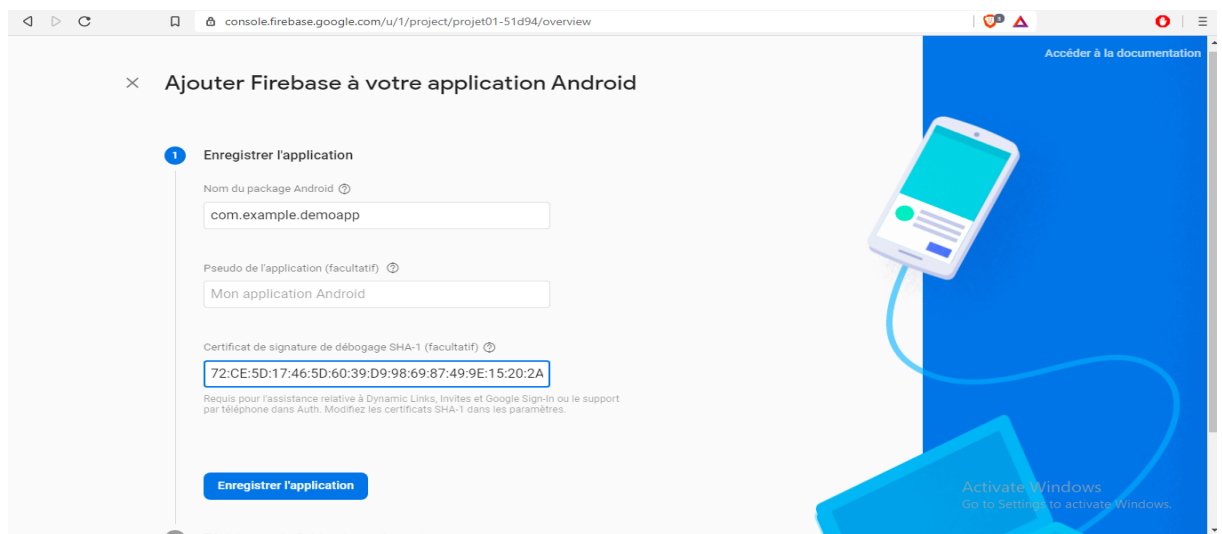


Figure 4. 6 Première étape de l'implémentation

- Après cela, nous cliquons sur « **Download google-services.json** » qui est un fichier de paramètres qui relie notre projet Firebase à notre projet dans android studio

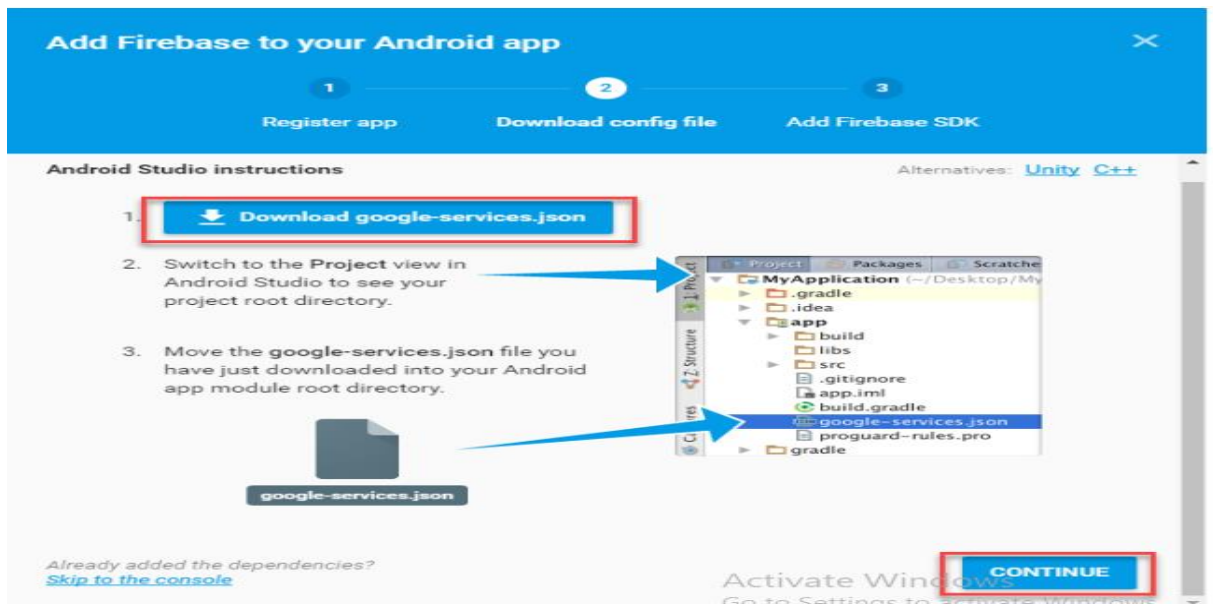


Figure 4. 7 deuxième étape de l'implémentation.

- Ensuite, allez dans Android Studio et cliquez avec le bouton droit sur le dossier de « **app** » et choisissez « **Show in Explorer** »

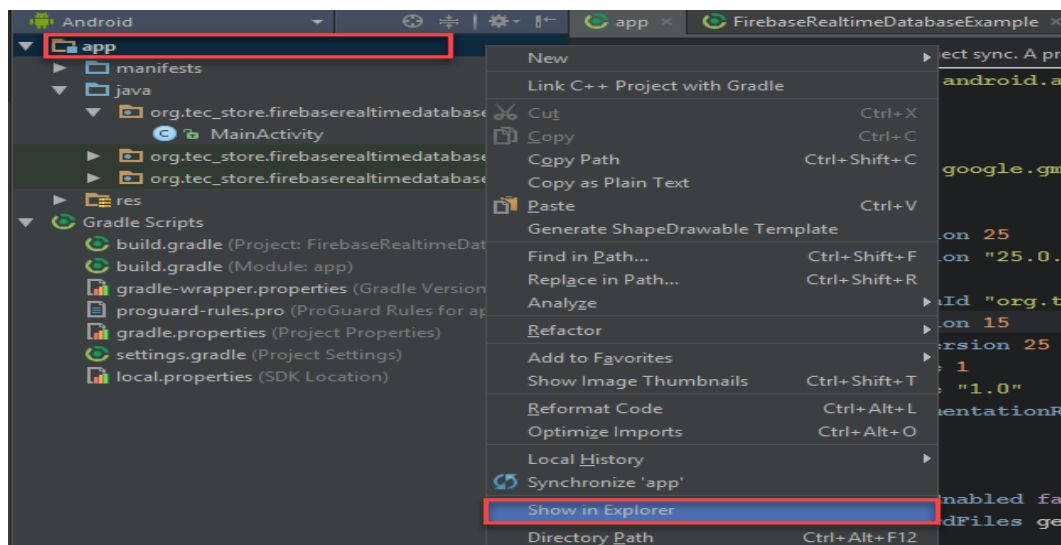


Figure 4. 8 app explorer

- Le dossier du projet s'ouvrira sur votre ordinateur et collera ce fichier dans le dossier de « **app** »

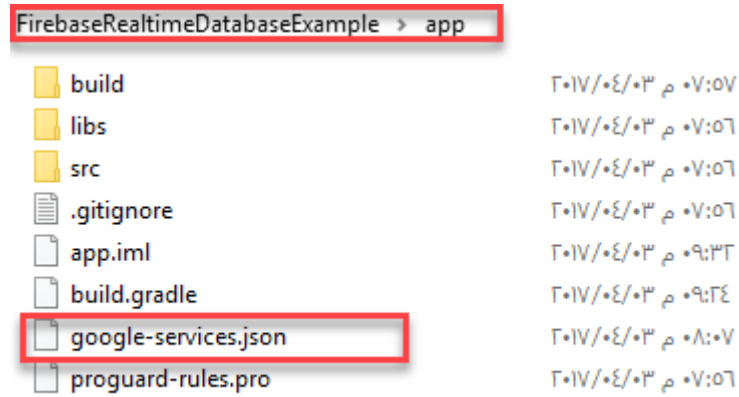


Figure 4. 9 google-services.json dans le dossier app.

- Nous devons maintenant ajouter des bibliothèques Firebase à notre projet Android Studio



Figure 4. 10 troisième étape de l'implémentation.

- Rendez-vous sur le fichier build.gradle (projet) et collez cette ligne « **classpath 'com.google.gms:google-services:3.0.0'** »

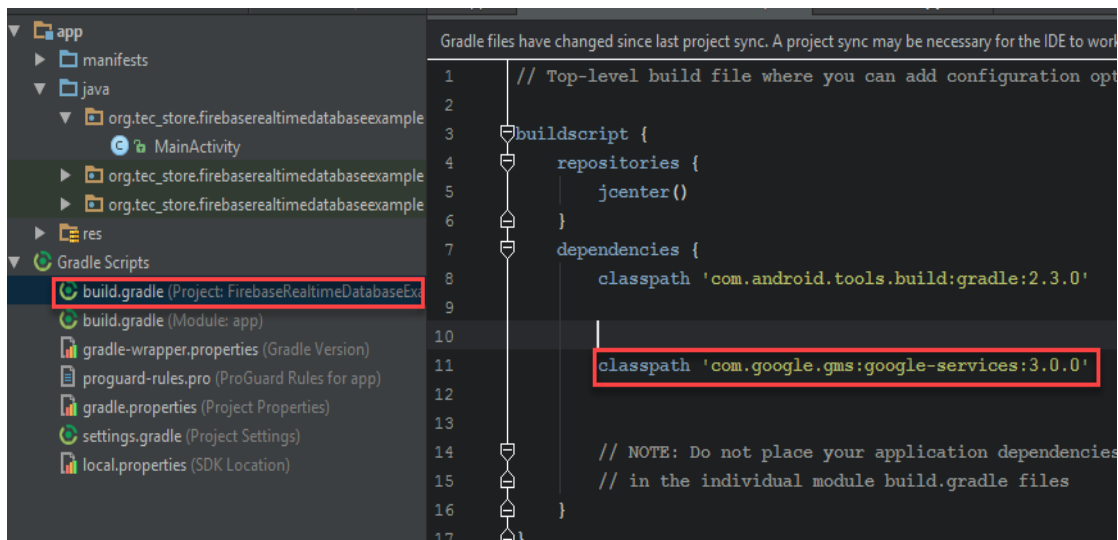


Figure 4. 11 Fichier build.gradle au niveau du projet.

- Ensuite, nous allons à build.gradle (app) et mettons la ligne du plugin « **apply plugin:** 'com.google.gms.google-services' »

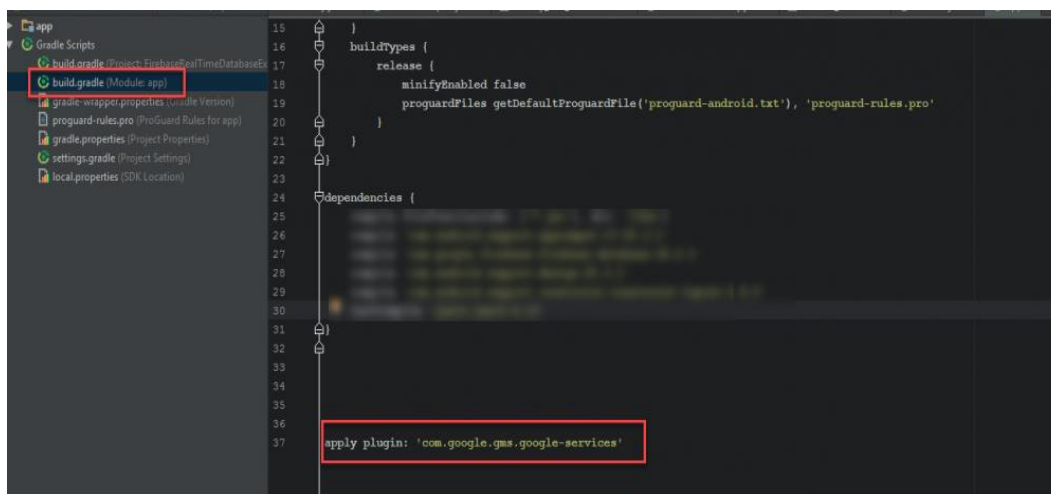


Figure 4. 12 Fichier build.gradle au niveau du projet

```
dependencies {
    // Firebase
    implementation 'com.google.firebase:firebase-core:16.0.1'
    implementation 'com.google.firebase:firebase-auth:16.0.1'
    implementation 'com.google.firebase:firebase-storage:16.0.1'
    implementation 'com.google.firebase:firebase-database:16.0.1'
    implementation 'com.google.firebase:firebase-firestore:16.0.0'
    implementation 'com.google.firebase:firebase-cloud-messaging:17.3.0'
}
apply plugin: 'com.google.gms.google-services'
```

Figure 4. 13 Fichier build.gradle au niveau de l'application

Dans ce cas la , le projet Firebase a été lié, et ces étapes consistent à lier n'importe quel projet Firebase en général, qu'il s'agisse d'authentification, de notifications, de stockage, etc..

Maintenant, nous allons à la console Firebase, à la base de données, puis aux règles, et le changeons en

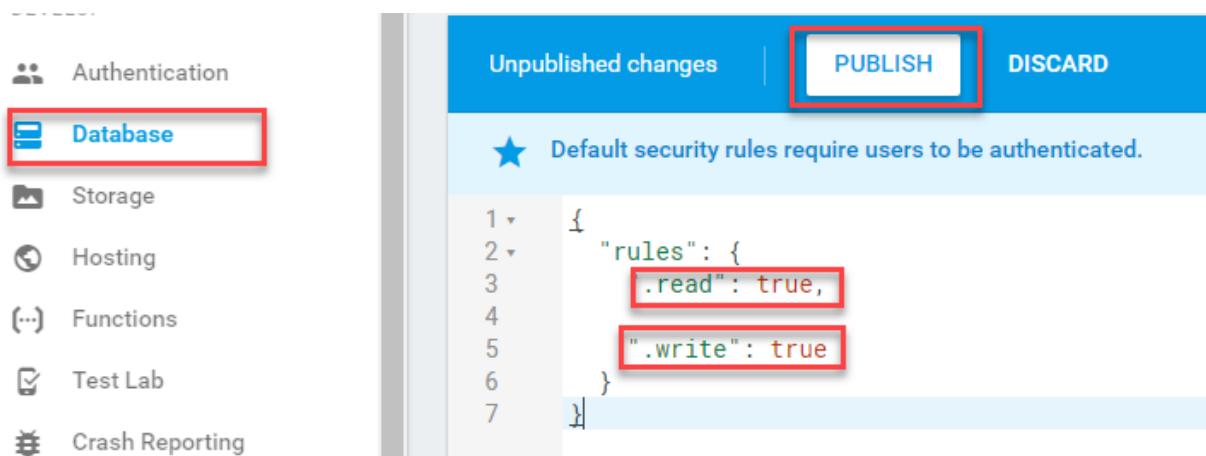


Figure 4. 14 réglage de base de données.

4.2.1 Firebase RealTime Data Base

Realtime Database Firebase est une base de données hébergée dans le cloud. Les données sont stockées sous forme de JSON et synchronisées en temps réel sur chaque client connecté. Lorsque nous créons des applications multiplateformes avec kits de développement logiciel (SDK) iOS, Android et JavaScript, tous vos clients partagent une instance de base de données

en temps réel et reçoivent automatiquement les mises à jour avec les données les plus récentes.

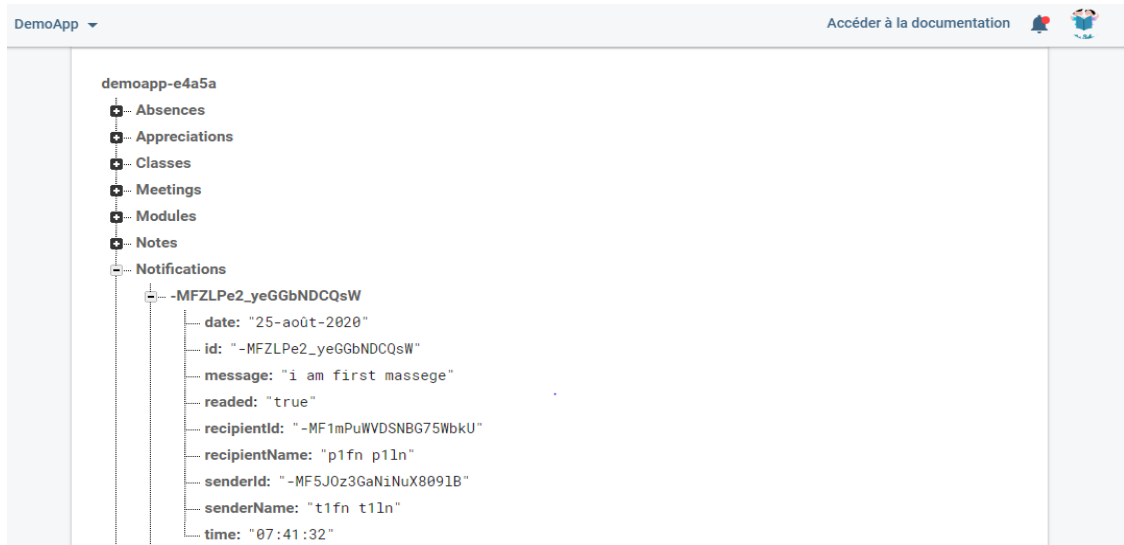


Figure 4. 15 Firebase realtime data base.

4.2.2 Firebase Storage

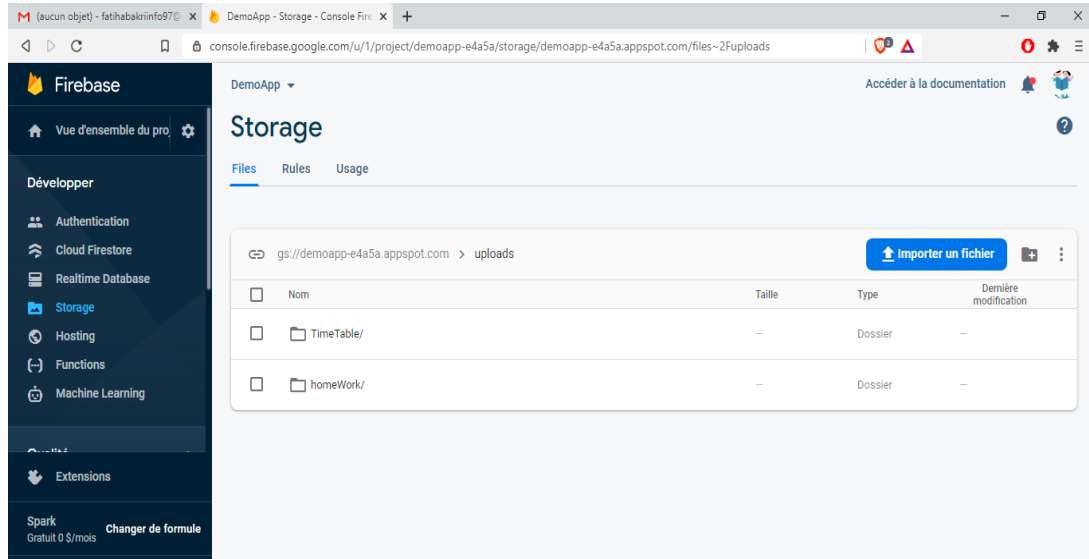


Figure 4. 16 Firebase storage.

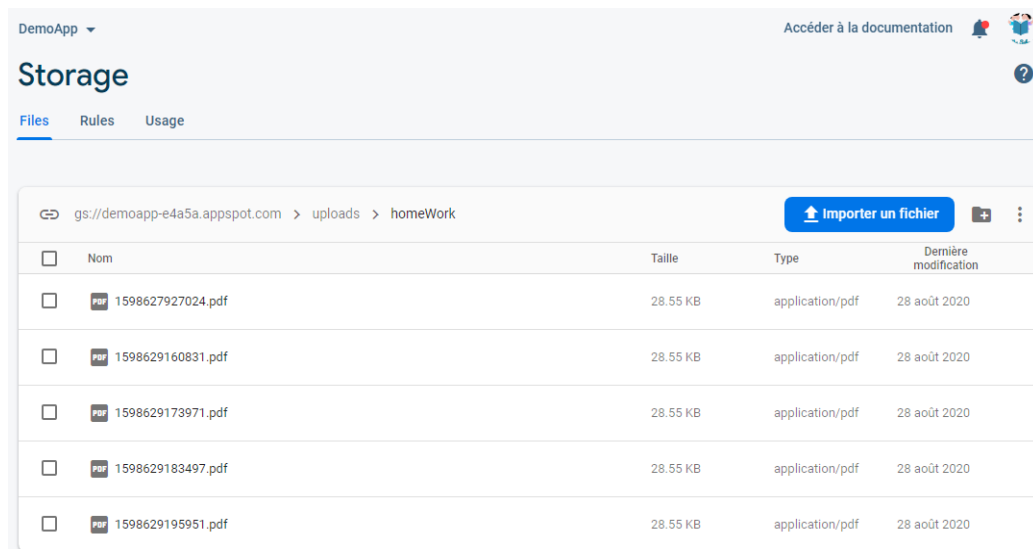


Figure 4. 17 Firebase storage en détail.

5. Présentation de l'application

Dans ce qui suit, nous présentons le fruit de notre travail en exposant les différentes interfaces de l'application. En commençant d'abord, par le logo de l'application.



Figure 4. 18 logo d'application

5.1 Les interfaces de l'application :

- ☞ *SplashScreen* : Cette interface est la toute première fenêtre affichée par l'application , contenant le logo de l'application avec une animation de chargement dure 5 secondes.
- ☞ *Login* : Pour les membres, qui sont déjà inscrit dans l'application
- ☞ *S'inscrit* : Pour crée un compte a les parents, qui ne sont pas créé un compte à l'application



Figure 4.19 splashscreen

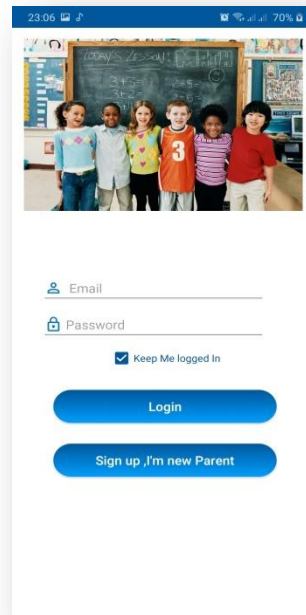


Figure 4. 20 Fenêtre de login

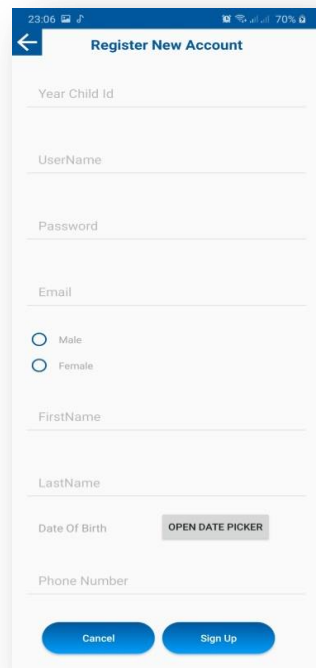


Figure 4. 21 Fenêtre d'inscription

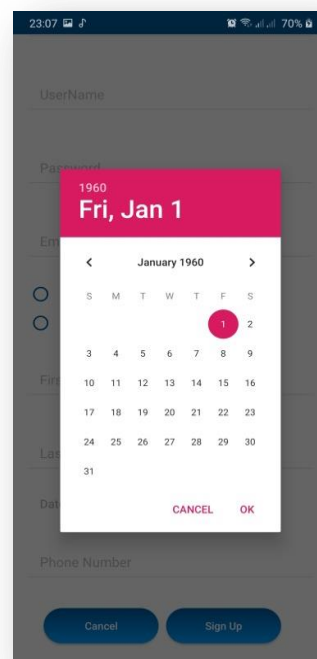


Figure 4. 22 Fenêtre de l'agenda.

☞ Les page principale d'utilisateur d'application



Figure 4. 23 page de parent

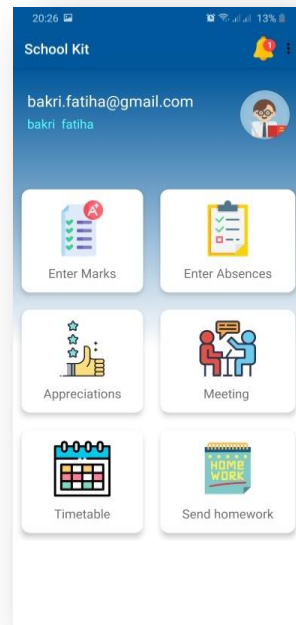


Figure 4. 24 page de l'enseignant

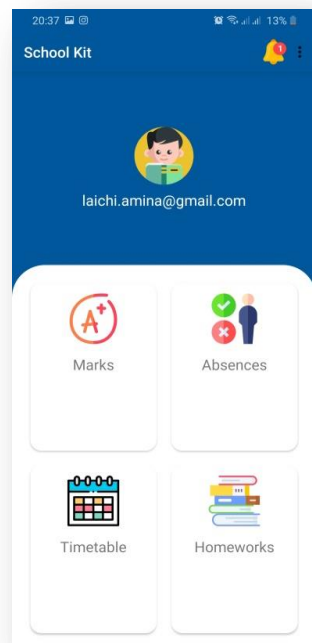


Figure 4. 25 page d'élève

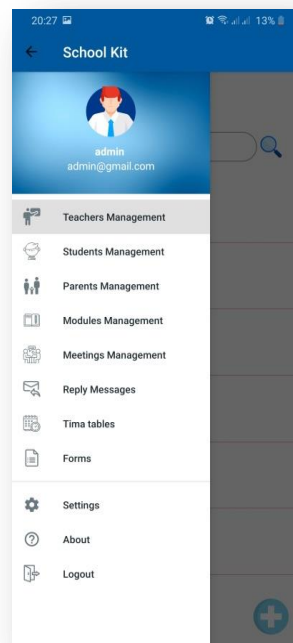


Figure 4. 26 menu de l'admin

☞ *Quelque Fenêtre de la page de l'administrateur*

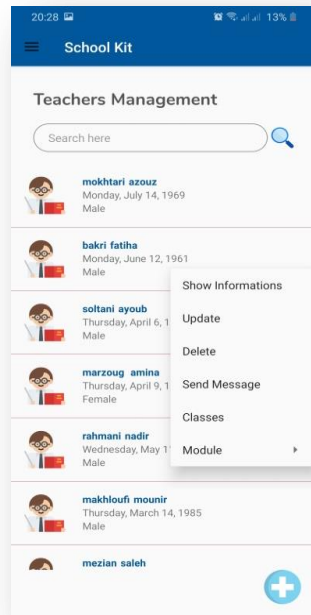


Figure 4.27 gestion des enseignants

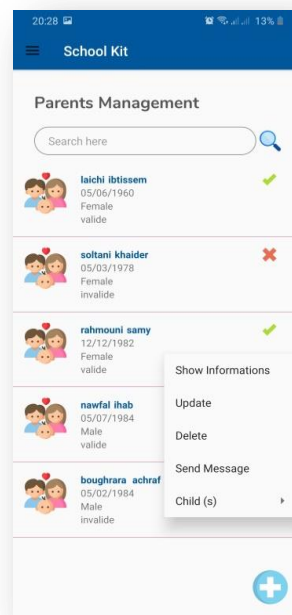


Figure 4.28 gestion des parents

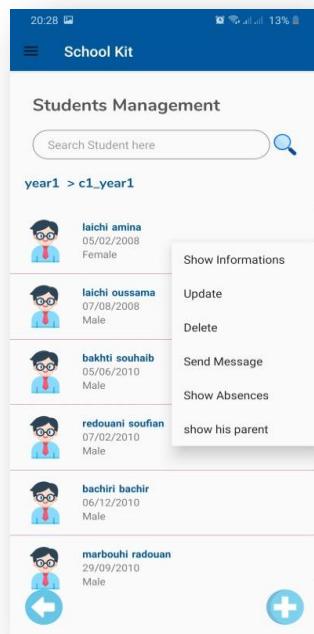


Figure 4.29 gestion des élèves



Figure 4.30 gestion des rendez-vous

☞ *Quelque Fenêtre de la page de parent*

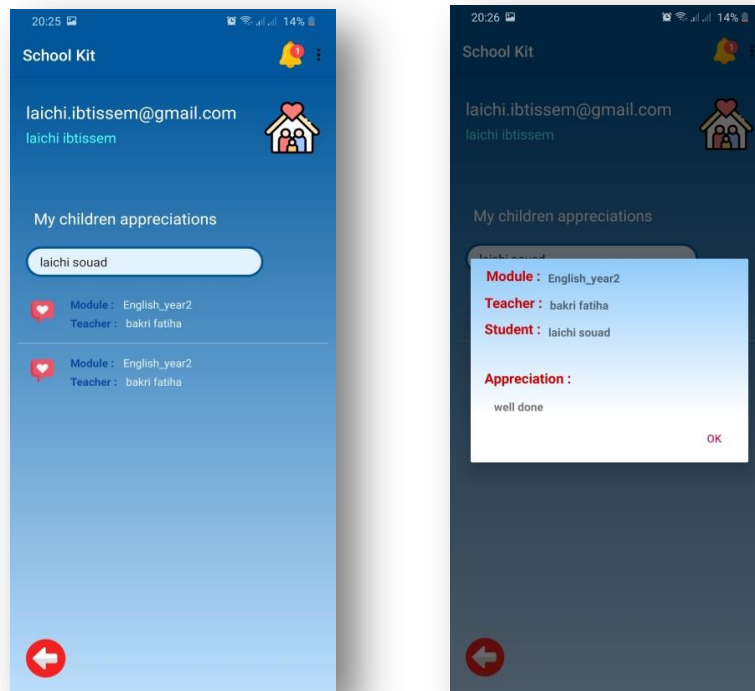


Figure 4.31 les appréciations des enfants



Figure 4.32 notes des enfants

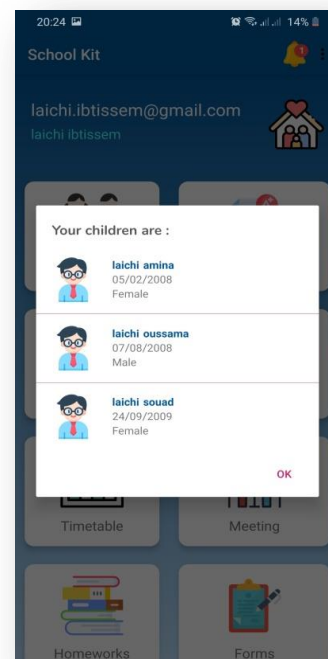


Figure 4.33 les enfants de parent

☞ *Quelque Fenêtre de la page d'enseignant*

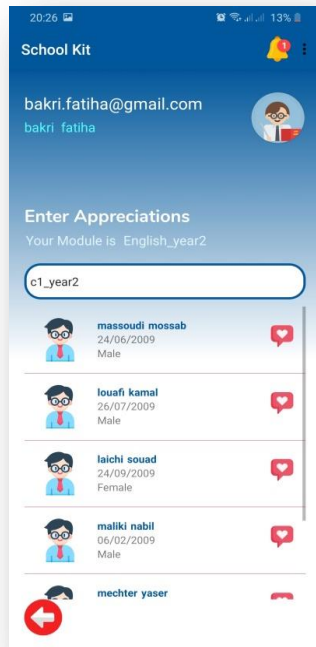


Figure 4.34 entre des appréciations

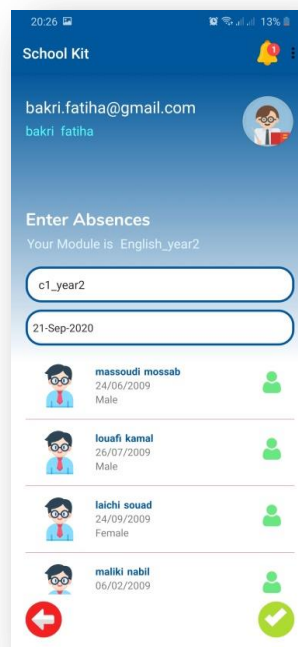


Figure 4.35 entre des absences

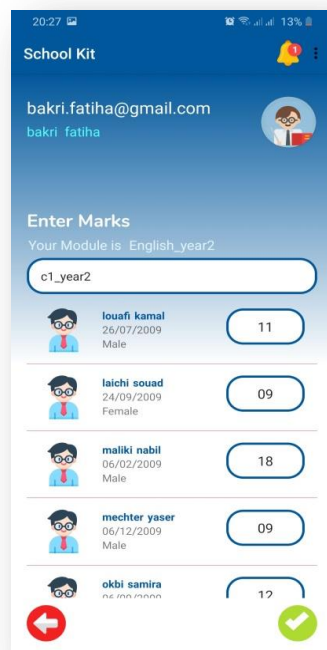


Figure 4.36 entre des notes

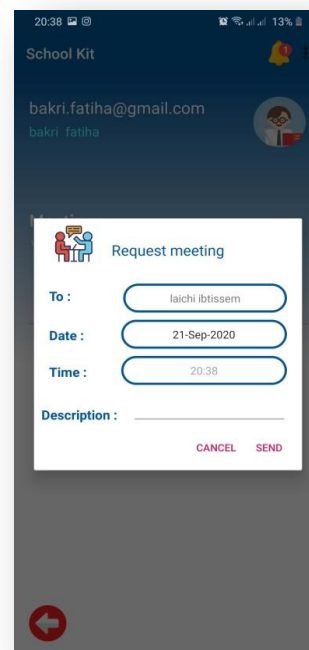


Figure 4.37 demande de rendez- vous

☞ *Quelque Fenêtre de la page d'élève*



Figure 4.38 Fenêtre des devoirs

☞ *Autre Fenêtre*

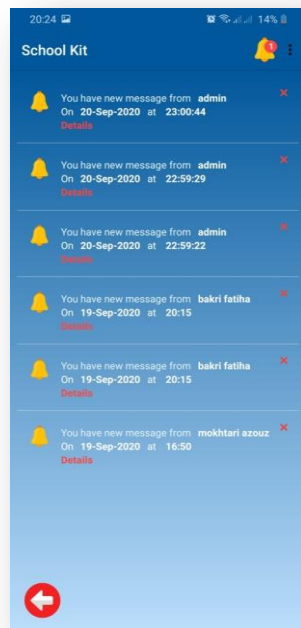


Figure 4.39 rendez-vous de parent

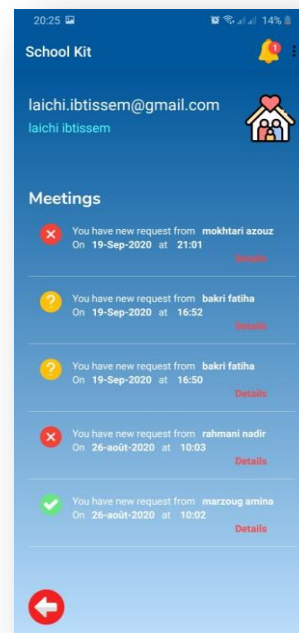


Figure 4.40 Fenêtre des messages

6. Conclusion

Dans ce dernier chapitre, nous avons présenté l'architecture de notre applications et l'environnement de développement avec les différents outils . nous conclure par présenté les différentes interfaces de notre application. Nous prenons au sérieux de respecter au maximum, les standards de l'IHM pour réaliser notre application.

CONCLUSION GÉNÉRALE

Nous pouvons dire en conclusion, que le but visé au départ et qui consistait à implémenter une application mobile sous Android a bien été atteint et qu'on a pu obtenir un système capable d'offrir la possibilité aux parents de bien suivre la scolarité de leurs enfants.

L'étude et La réalisation d'un tel projet, malgré les circonstances et les difficultés rencontrées, nous a permis d'acquérir une bonne expérience dans ce domaine. En effet, la situation sanitaire du pays qui a entraîné une difficulté au niveau du contact inter-personnes, le manque de références bibliographiques et d'informations sur le suivi parental, ont tous été un vrai défi pour nous. Cependant, on a pu quand même apprendre beaucoup de nouvelles choses telles qu'une bonne connaissance du domaine entamé et une bonne maîtrise de certains outils de développement tels que Firebase, Android et autres. Le travail en groupe a également été un ajout important pour notre expérience.

Nous espérons que le système ainsi obtenu, soit utilisé au niveau de nos établissements scolaires et qu'il satisfasse les besoins des différents utilisateurs. Nous espérons aussi pouvoir à l'avenir, bien améliorer notre système afin qu'il offre plus de possibilités et une plus grande maniabilité d'utilisation et souhaitons également le doter de la capacité de mener des conversations vidéo.

Bibliographie

- [01] <https://www.techopedia.com/definition/4619/personal-digital-assistant-pda> Consulter le 22/03/2020.
- [02] <https://www.techopedia.com/definition/2977/smartphone> consulter le 22/03/2020.
- [03] <https://www.futura-sciences.com> Consulter le 26/03/2020.
- [04] <https://www.taktilcommunication.com> Consulter le 5/04/2020
- [05] <https://skylineapps.wordpress.com> Consulter le 11/04/2020
- [06] <https://fr.yeeply.com/blog/application-native-hybride-ou-web/> , Consulter le 18/04/2020
- [07] <https://fr.yeeply.com/blog/bases-techniques-developper-application-mobile/> Consulter le 15/06/2020 .
- [08] M. Meier. Android Application Développent. Etats Unis (Indiana), wiley publishing Edition 2009.
- [09] <https://openclassrooms.com> , Consulter le 12/06/ 2020.
- [10] S. P. Hébuterne Sylvain, Android Guide de développement d'applications Java pour Smartphones et Tablettes. 2ème édition, 2014.
- [11] B. Nazim, Android 7 Les fondamentaux du développement d'applications Java Broché 8 février 2017.
- [12] <https://www.phonandroid.com> Consulter le 28/06/ 2020.
- [13] R. Meier, Professional Android Application Development. 2012.
- [14] B. Hadj Brahim , M. Souaad , développement d'une application de géolocalisation des médecins de la wilaya de tlemcen sous android , Master , Université Abou Baker Belkaid-Tlemcen , 2016/2017.
- [15] H. Sylvain, Développez une application Android Programmation en Java sous Android Studio (2e édition) .Septembre 2015.
- [16] U.Serge, Développez une application Android Programmation en Java sous Eclipse. 2ème édition, juin 2013.

- [17] <https://vogella.developpez.com> Consulter le 03/07/ 2020.
- [18] <https://apprendrelesmartphone.blogspot.com/> Consulter le 13 /07 /2020.
- [19] A.yessoufou1 , eric stéphane P. towou , G. amoussou , suivi parental des apprenants du primaire au secondaire: tendances et incidences sur le rendement scolaire , Volume 7, Numéro 11, décembre 2016.
- [20] <https://assembly.coe.int> Consulter 28/07/2020.
- [21] <https://apprendre.education> Consulter 07/ 2020.
- [22] R.Deslandes , E. Royer, Style parental, participation parentale dans le suivi scolaire et réussite scolaire , Volume 43, Number 2, 1994. URI : <https://id.erudit.org/iderudit/706657ar>
- [23] <https://recit.cssamares.qc.ca> , Consulter le 03/08/ 2020.
- [24] Répertoire d'applications de gestion de vie scolaire : <file:///D:/chapitre%20mimoire/R%C3%A9pertoire%20d'applications%20de%20gestion%20de%20Vie%20scolaire%20-%20Thot%20Cursus.html> Consulter le 10/08/2020.
- [25] A .Bacon , Relation entre les caractéristiques familiales et la participation parentale à la réussite scolaire des adolescents , thèse , Université de Québec, 18 avr. 2012
- [26] P. Roques. Le cahier du programmeur : UML 2 Modéliser une application web. Paris, 4 Éditions eyrolles edition, Juin 2008.
- [27] J.Gabay , D.Gabay. Mise en œuvre guidée avec études de cas. 2011.
- [28] P.Roques. UML 2 par la pratique : étude de cas et exercices corrigés. Paris, Éditions eyrolles edition, Septembre 2011.
- [29] Wikipedia : https://fr.wikipedia.org/wiki/Diagramme_de_s%C3%A9quence. Consulter le 8 2020.
- [30] Wikipedia : https://fr.wikipedia.org/wiki/Diagramme_de_classes Consulter le 17/08/ 2020.
- [31] <https://android-studio.fr.uptodown.com/windows> Consulter le 05/09/ 2020.
- [32] https://fr.wikipedia.org/wiki/Java_Development_Kit Consulter le 05/09/ 2020.
- [33] <https://www.lemagit.fr/definition/NoSQL-base-de-donnees-Not-Only-SQL> Consulter le 12/09/ 2020.
- [34] <https://www.axopen.com/firebase/> Consulter le 12/09/ 2020.

الملخص:

من المشاكل التي يواجهها الآباء والمعلمون في المدارس صعوبة مراقبة الأبناء أثناء دراستهم.

لتوفير حل لهذه المشكلة ، اقترحنا ، كجزء من مشروع التخرج للحصول على شهادة ماستر أكاديمي في الاعلام الآلي، تصميم وإنتاج تطبيق للهاتف الذكي يسمح بمراقبة الوالدين الجيدة . تم تصميم هذا التطبيق باستخدام أداة تطوير اندرويد استوديو مفتوح المصدر ، باستخدام خدمات الاستضافة من منصة فايربيز للهاتف .

كلمات مفتاحية: تطبيق اندرويد ، فايربيز ، المراقبة الأبوية ، ايامال .

Abstract:

One of the problems that parents and teachers face in schools is the difficulty of monitoring students during their schooling.

To provide a solution to this problem , as a final studies project to obtain the diploma of master degree in computer science. we proposed the design and production of a mobile application that allows good parental monitoring. This application was designed using the open source Android studio development tool, using hosting services from the Firebase mobile platform.

Key words: Android App , Firebase , Parental controls, Uml.

Résumé :

L'un des problèmes que rencontrent les parents et les enseignants au niveau des établissements scolaires est la difficulté de suivi des élèves durant leur scolarisation.

Pour apporter une solution à ce problème, nous avons proposé, dans le cadre du projet de fin d'études pour l'obtention du diplôme de master en informatique, la conception et la réalisation d'une application mobile qui permet d'assurer un bon suivi parental. Cette application a été conçue à l'aide de l'outil de développement open source Android studio, en utilisant les services d'hébergement de la plateforme mobile Firebase.

Mots Clés : Android App , Firebase, suivi parental, Uml.