

Institut Supérieur de Comptabilité et d'Administration des Entreprises
Nouakchott-Mauritanie

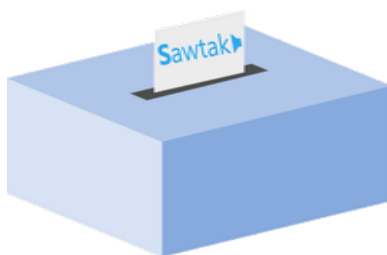


PROJET DE FIN D'ETUDES

Pour l'obtention de :

LICENCE en DEVELOPPEMENT INFORMATIQUE

La mise en place d'une solution mobile de vote électronique



Elaboré par :

- Khoueira MohamedEzein Saha - I18113
- Mariem Mohamed Lemrabet - I17925

Encadré par :

Dr. Ethman Ould EBBE

Année Universitaire : 2021-2022

Dédicace

Je remercie avant tout ALLAH

***Ce travail est dédié avec grand amour,
sincérité et fierté :***

***A nos chers parents, sources de tendresse,
de noblesse et d'affection.***

A notre professeur et encadreur :

Dr. Ethman Ould EBBE

***A toute la promotion et professeurs et à tous
ceux qui ont aidé à la réalisation de ce
modeste travail.***

Remerciements

Tout d'abord, nous remercions dieu le tout puissant de nous avoir donné l'opportunité d'achever ce projet.

Nous commençons par remercier **Dr. Ethman ould EBBE** qui nous a fait l'honneur d'être notre encadrant.

Nous le remercions profondément pour ses encouragements continus, d'être toujours là pour nous écouter, nous aider, nous guider à retrouver le bon chemin par sa sagesse et ses précieux conseils.

En outre, son soutien moral, sa compréhension, sa générosité en matière de formation et d'encadrement nous ont été très utiles pour l'élaboration de ce mémoire.

Nous le remercions aussi pour les conseils concernant les tâches évoquées dans ce rapport, le suivi et la confiance qu'il nous a témoignée.

Nous voulons également exprimer par ces lignes de remerciements notre gratitude envers tous ceux qui par leur présence, leur soutien, leur disponibilité ou leurs conseils, nous ont encouragés pour accomplir ce projet. Enfin, nous ne pouvons achever ce mémoire sans exprimer nos sincères remerciements et notre profonde gratitude à tous les professeurs de notre institut ISCAE, pour leur dévouement et leur assistance tout au long de notre cursus d'études universitaires.

Sommaire

Table des matières

Introduction :	3
Problématique :	3
Cahier des charges (CDC) :	3
Entrez le numéro national (NNI) :	4
Entrez le code de vérification (OTP SMS) :	4
Processus de vote :	4
Processus de dépouillement :	4
Neutralité :	4
Surveillance :	4
Conclusion :	4
Introduction :	6
Présentation de la méthode « UML » :	6
Diagramme de cas d'utilisation :	7
Diagramme de class :	8
Conclusion :	8
Introduction :	10
Outils utilisés :	10
UML :	11
StarUML :	11
Visual Studio Code :	12
Flutter :	12
L'objectif du projet Flutter :	12
Les inconvénients :	13
Dart :	14
Firebase :	15
Comment ça-marche :	16
Développement :	18
Création de la base de données :	18
L'interface de l'application :	20

Page d'accueil :	20
Page de vérification :	22
Page OTP :	24
Page de fin :	26
Statistique (Résultats) :	27
Conclusion	29
REFERENCES	30
Bibliographie :	30
Webographie :	30
Annexes	31
Code de Controller :	31
Fonction vote :	31
Candidat Controller :	31

Introduction Générale

En raison des progrès technologiques qui nous entourent dans le domaine du contrôle à distance tels que les achats électroniques et les avant-premières dans divers domaines, nous avons cherché à développer une solution mobile de vote électronique qui porte le nom : **Sawtak** de vote à distance pour faciliter le processus de vote par téléphone mobile.

Pour développer cette application, nous avons utilisé Flutter et Dart comme langage de programmation et pour la base des données firebase.

Sawtak est une application qui donne la possibilité à son utilisateur de livrer sa voix de manière transparente, une application qui fournit un service de vote à distance via un téléphone mobile simplement. Est une application qui facilite la méthodologie du vote à distance, développée en Dart & Flutter avec une base de données sur firebase, avec un formulaire simplifié facile à comprendre.

Et cette application aura un impact positif économique important car cava diminuer le coût financier de vote.

Le présent rapport de projet s'articule autour de trois chapitres. Le premier chapitre sera consacré à la présentation générale. Le deuxième chapitre traitera le langage graphique de modélisation « UML » utilisée. Ensuite, le troisième chapitre, comportera deux axes le premier portera sur les outils et logiciels utilisés dans le projet, et le deuxième sur les différentes étapes de mise en place et de vérification du bon fonctionnement de cette application.

Enfin, nous terminerons ce rapport par une conclusion générale qui portera sur les résultats obtenus et les perspectives d'amélioration de cette application.

Chapitre I

Présentation Générale

Introduction :

Dans ce chapitre nous allons présenter le contexte général du projet. Nous commencerons par la présentation de la problématique pour terminer par le cahier des charges À suivre dans le projet.

Problématique :

Actuellement, pour exercer son droit de vote, le votant se trouve face à plusieurs difficultés. Tout d'abord, il se trouve dans l'obligation de reporter ou annuler certaines activités quotidiennes pour pouvoir exercer son droit vote. Ensuite, compte tenu de l'éloignement et de la dispersion des bureaux de vote on perd également un temps énorme dans les moyens de transport et dans les files d'attente devant les bureaux de vote. Quant aux personnes âgées et aux personnes à besoins spécifiques, des moyens complémentaires de déplacement et de commodités du vote sont nécessaires à mettre en place pour leurs permettre d'effectuer le vote dans les meilleures conditions possibles.

D'autres problèmes liés à la rapidité et la fiabilité de gestion des résultats se posent surtout dans les zones rurales où le calcul des bulletins se fait manuellement. Comment donc trouver des solutions qui nous permettent de remédier à ces problèmes et faciliter à chaque individu d'exercer son droit de vote dans des conditions adéquates ?

Cahier des charges (CDC) :

L'application du vote par système électronique est reconnue comme étant une avancée dans le monde d'une part, et une nouveauté dans notre pays d'autre part.

Le processus permettra aux électeurs d'exercer leur droit de vote à partir de n'importe quel appareil portable Android connecté sur Internet.

L'application (**Sawtak**) à réaliser doit répondre aux exigences de cahier des charges définissant les besoins de différents acteurs qui vont l'utiliser.

Nous avons développé **le modèle de processus de vote**, en considérant que le processus d'inscription a déjà eu lieu, où l'utilisateur effectue plusieurs étapes :

Entrez le numéro de téléphone :

L'électeur saisit son numéro de téléphone, pour une vérification systématique pour son NNI en envoyant un code de vérification.

Entrez le numéro national (NNI) :

L'électeur saisit son numéro national NNI pour sa validation.

Entrez le code de vérification (OTP SMS) :

L'électeur saisit le code qui lui est envoyé sur son numéro de téléphone par SMS.

Processus de vote :

Le processus de vote se fait via bouton <<voter >>.

Processus de dépouillement :

Après avoir voté par NNI cava incrémenter le nombre des voix de son candidat.

Neutralité :

Chaque citoyen de notre pays a le droit d'exprimer son vote, nous avons donc attribué un bouton aux personnes ayant des opinions neutres.

Surveillance :

Le processus de vote, est contrôlé par « l'administrateur », pour effectuer les opérations liées au contrôlé et à la gestion du vote, des candidats et des électeurs.

Conclusion :

Ce chapitre introductif nous a permis de présenter la problématique ainsi que les buts visés par le cahier des charges. Les informations recueillies à ce niveau nous permettrons d'amorcer le deuxième chapitre relatif à l'analyse et la conception de la base de données pour identifier de manière précise et concise les différentes fonctionnalités de l'application.

Chapitre II

Analyse et conception

Introduction :

Ce chapitre présente une analyse et une conception conforme aux données et aux besoins. Il s'agit de présenter la méthode et les modèles de conception permettant au projet la mise en place d'une solution de vote électronique.

Présentation de la méthode « UML » :

L'UML est constitué de diagrammes qui servent à visualiser et décrire la structure et le comportement des objets qui se trouvent dans un système. Il permet de présenter des systèmes logiciels complexes de manière plus simple et compréhensible qu'avec du code informatique. L'UML a des applications dans le développement logiciel, mais aussi dans l'industrie (pour modéliser les flux de processus par exemple), dans l'ingénierie ou le marketing.

Les efforts de ces penseurs ont abouti à la publication des documents UML 0.9 et 0.91 en 1996.

L'UML 1.0 a été adopté comme standard par l'Object Management Group (OMG) en janvier 1997. Il est issu de la fusion de trois méthodes orientées objet issues des travaux de Grady Booch, de Jim Rumbaugh et d'Ivar Jacobson. Des versions successives ont ensuite été validées.

Il est rapidement devenu évident que des sociétés comme Microsoft, Oracle et IBM voyaient l'UML comme un élément critique pour leur développement futur. Elles ont donc mis en place des ressources, accompagnées en cela par de nombreuses autres sociétés et personnes, permettant de développer un langage de modélisation complet. Les Three Amigos ont publié The Unified Modeling Language User Guide en 1999, qui fut suivi d'une mise à jour comportant des informations sur l'UML 2.0 en 2005.

Diagramme de cas d'utilisation :

Les **diagrammes de cas d'utilisation** sont des diagrammes UML utilisés pour une représentation du comportement fonctionnel d'un système logiciel. Ils sont utiles pour des présentations auprès de la direction ou des acteurs d'un projet, mais pour le développement, les cas d'utilisation sont plus appropriés.

Les diagrammes de cas d'utilisation modélisent le comportement d'un système et permettent de capturer les exigences du système. Les diagrammes de cas d'utilisation décrivent les fonctions générales et la portée d'un système.

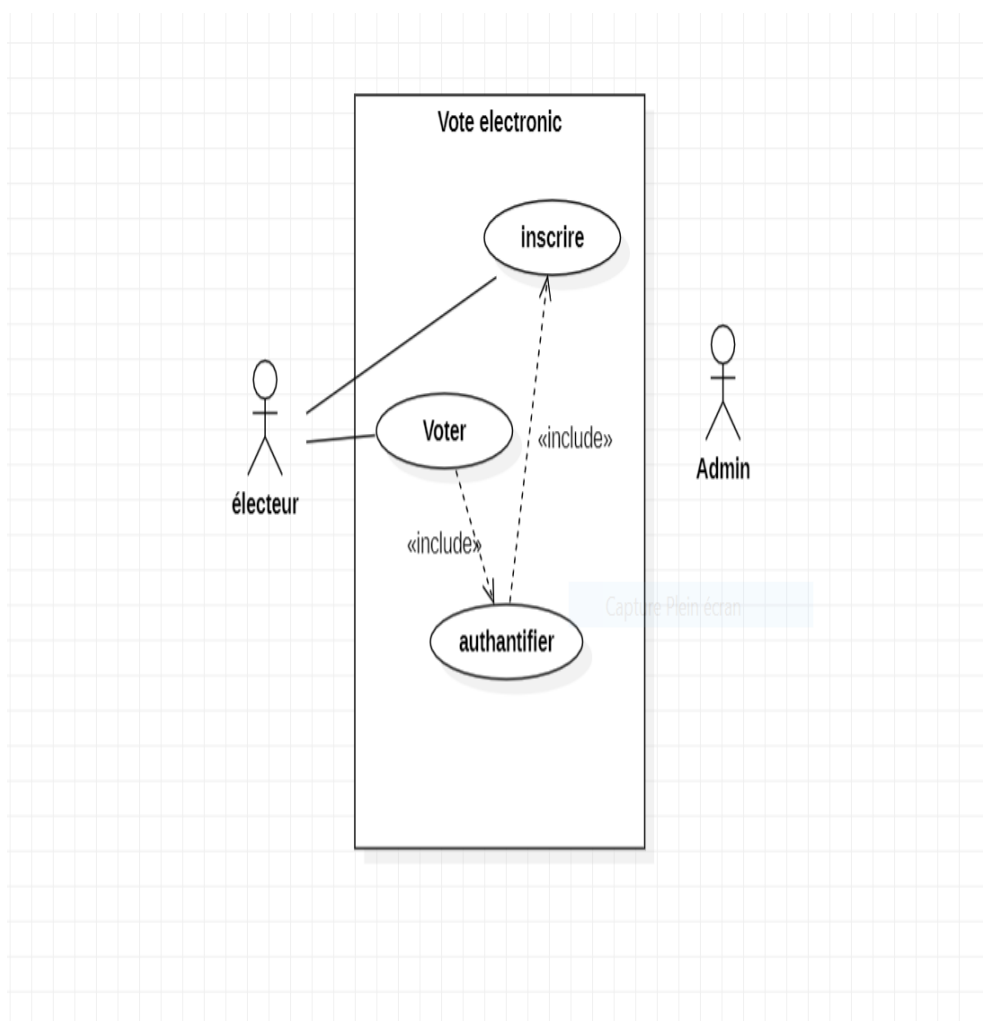
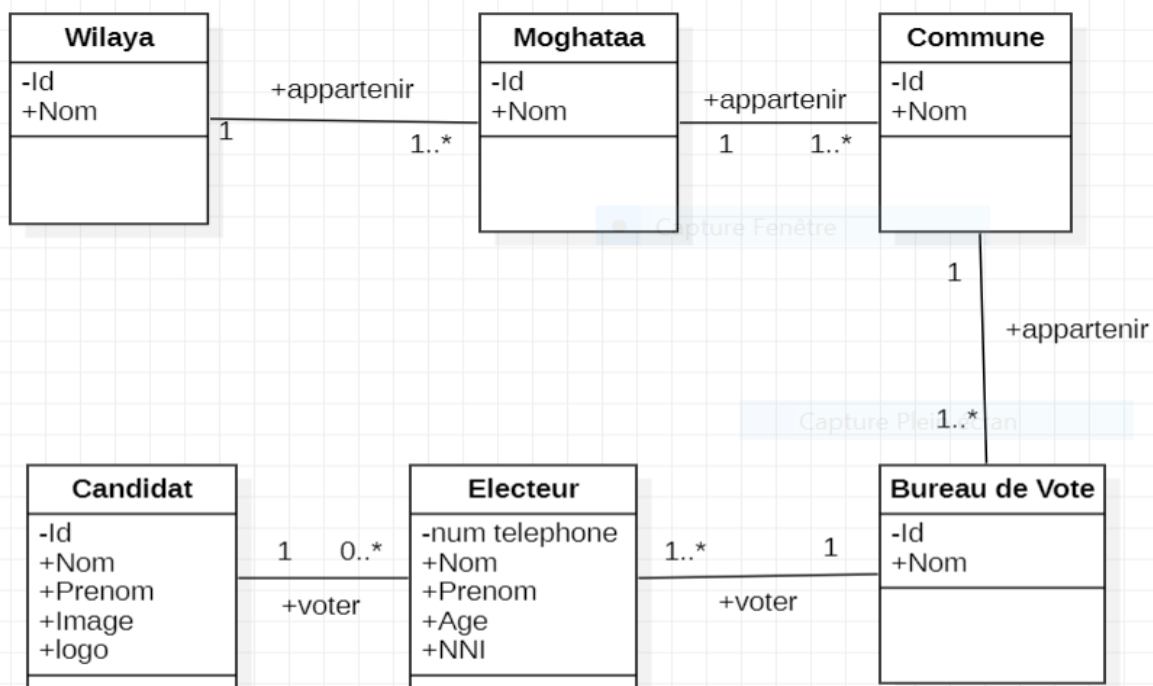


Diagramme de class :

Le **diagramme de classes** est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que leurs relations. Ce diagramme fait partie de la partie statique d'UML, ne s'intéressant pas aux aspects temporels et dynamiques.

Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation.



Conclusion :

Le choix de la méthode d'analyse et de conception permet donc de mettre en place une démarche de développement. Cette phase avait pour intérêt de présenter les différentes étapes de l'application tout en évoluant dans le niveau de détail. Nous pouvons ainsi entamer le prochain chapitre portant sur la phase de réalisation et le test de fonctionnement de l'application.

Chapitre III

Réalisation : Outils et Développement

Introduction :

Cette partie constitue le dernier volet de ce rapport. Après avoir terminé la phase d'analyse et conception, la solution étant ainsi choisie et étudiée, il ne reste plus que choisir dans quel environnement nous allons travailler, exposer les choix techniques utilisés, le langage adopté, et présenter l'implémentation et les tests réalisés. Ce chapitre portera donc sur la description des outils utilisés et les étapes de développement de l'application.

Outils utilisés :

Dans ces paragraphes nous allons donner un aperçu sur les outils de développement que nous allons exploiter dans la mise en place du projet. Ces outils comportent les langages de programmation et les logiciels de conception et de développement.

A travers cette page nous allons pouvoir vous expliquer et détailler tous les outils utilisés pour créer cette application.

Parmi ces outils des logiciels utilisés pour la conception et L'illustration des différentes fonctionnalités de l'application.



UML :

L'UML (Unified Modeling Language ou Langage de modélisation unifiée en français) est un langage graphique de modélisation informatique. Ce langage est désormais la référence en modélisation objet, ou programmation orientée objet. Il permet de présenter des systèmes logiciels complexes de manière plus simple et compréhensible qu'avec du code informatique.

StarUML :



StarUML

StarUML est un logiciel de modélisation UML, qui a été « cédé comme open source » par son éditeur, à la fin de son exploitation commerciale, sous une licence modifiée de GNU GPL. Aujourd'hui la version StarUML V3 n'existe qu'en licence propriétaire, ce logiciel constitue une excellente option pour une familiarisation à la modélisation. Cependant, seule une version Windows est disponible.

StarUML gère la plupart des diagrammes spécifiés dans la norme UML 2.0.

L'objectif du projet StarUML est d'offrir une alternative aux célèbres outils UML commerciaux tels que Rational Rose, Together et autres.

La dernière version a été publiée en 26 mars 2022.

Visual Studio Code :



Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et MacOS. Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code, les snippets, la refactorisation du code et Git intégré.

Le code source de Visual Studio Code provient du projet logiciel libre et open source VS Code de Microsoft publié sous la licence MIT permissive, mais les binaires compilés constituent un freeware, c'est-à-dire un logiciel gratuit pour toute utilisation mais privé.

Flutter :



Flutter est un kit de développement logiciel d'interface utilisateur open-source créé par Google. Il est utilisé pour développer des applications pour Android, iOS, Linux, Mac, Windows, Google Fuchsia et le web à partir d'une seule base de code.

Flutter est la plus connue et la plus répandue, est une application dite native utilisant le SDK fourni par Google. L'avantage de cette solution est d'offrir les meilleures performances, tout en utilisant l'intégralité des capacités d'Android.

Flutter se compose principalement de la plateforme Dart, du moteur Flutter, de la fondation Library et des différents widgets.

Flutter utilise Dart comme langage de programmation ; langage qui également développé par google.

L'objectif du projet Flutter : est de permettre de développer de manière rapide et simplifiée des applications mobiles multiplateformes. Le Framework flutter a des avantages parmi eux :

- L'UX : de très bonnes performances, proches du natif, Flutter propose un code commun qui fonctionne aussi bien sur iOS que sur Android.
- Possibilité d'intégrer plus facilement des animations

- Un Framework avantageux pour les développeurs, Flutter permet par exemple de recharger la page automatiquement sans qu'aucune action particulière ne soit nécessaire. Ce rechargement de code est très apprécié par les développeurs, car il permet de simplifier le débogage.
- Le build des applications est très rapide, le temps de compilation est largement réduit. Ce gain de temps est rendu possible grâce à la fonctionnalité Hot Reload de DART.
- Flutter utilisé avec des différents environnements soit Android Studio, Visual Studio Code etc.
- La conception design considérablement simplifiée, Grâce à Flutter il est beaucoup plus facile d'intégrer des animations dans les applications mobiles.
- Une maintenance accélérée et optimisée, Les corrections de bugs sont rapides et régulières.
- Applications natives pour IOS et Android : Flutter possède toutes les fonctionnalités des plates-formes natives, telles que le défilement, la navigation, les polices d'icônes, le texte, la vue Liste, etc.,
- Les performances, Flutter se démarque de ses concurrents (React Native, Native Script, Ionic) par ses performances.

Les inconvénients :

- Les nouveautés annuelles de chaque OS ne sont pas forcément implémentables au Moment de leur sortie.
- Un design iOS/Android presque identique.
- Certains composants peuvent être difficiles à personnaliser.
- Un langage supplémentaire.
- Le poids des applications, et notamment des APK, est plus important. Une phase d'optimisation est à prévoir.
- Evolution très rapide : les packages sont rapidement obsolètes et à mettre à jour.
- Technologie encore jeune.

Dart :



Dart est un langage de programmation optimisé pour les applications sur plusieurs plateformes. Il est développé par Google et est utilisé pour créer des applications mobiles, de bureau, de serveur et web. Dart est un langage orienté objet à ramasse-miettes avec une syntaxe de type C++.

Son but est de se substituer au langage JavaScript et de devenir la référence pour le développement multiplateforme.

Dart a pour objectif de concurrencer et remplacer JavaScript. Il veut combler ses défauts qui ne peuvent pas être résolus en faisant évoluer le langage, de plus il souhaite être plus performant et plus rigoureux.

Ce qui distingue le langage, c'est la possibilité d'écrire un programme et de le publier sur les appareils Android et iPhone sans réécrire l'application dans une autre langue.

- La raison derrière la nouvelle langue

L'objectif de la création d'un langage de programmation Dart réside dans les problèmes auxquels JavaScript est confronté, qui sont difficiles à résoudre, tels que les performances du programme et la protection contre la menace de la programmation intersites.

Dart permet à Flutter d'éviter d'avoir besoin d'un langage de mise en page déclarative séparé comme JSX ou XML, ou de constructeurs d'interface visuelle distincts, car la mise en page déclarative et programmatique de Dart est facile à lire et à visualiser.

Firestore :



Firestore est un ensemble de services d'hébergement pour n'importe quel type d'application. Il propose d'héberger en No SQL et en temps réel des bases de données, du contenu, de l'authentification sociale, et des notifications, ou encore des services.

De plus, la plateforme peut être exploitée par plusieurs utilisateurs en même temps sans connaître un quelconque bug. La praticité est également au rendez-vous grâce à ses fonctionnalités intuitives. Depuis le rachat de la plateforme par Google en 2014, Firestore SDKs a connu de nombreuses améliorations et n'a de cesse de satisfaire ses utilisateurs.

Firestore met à votre disposition différents services pouvant être répartis en deux catégories :

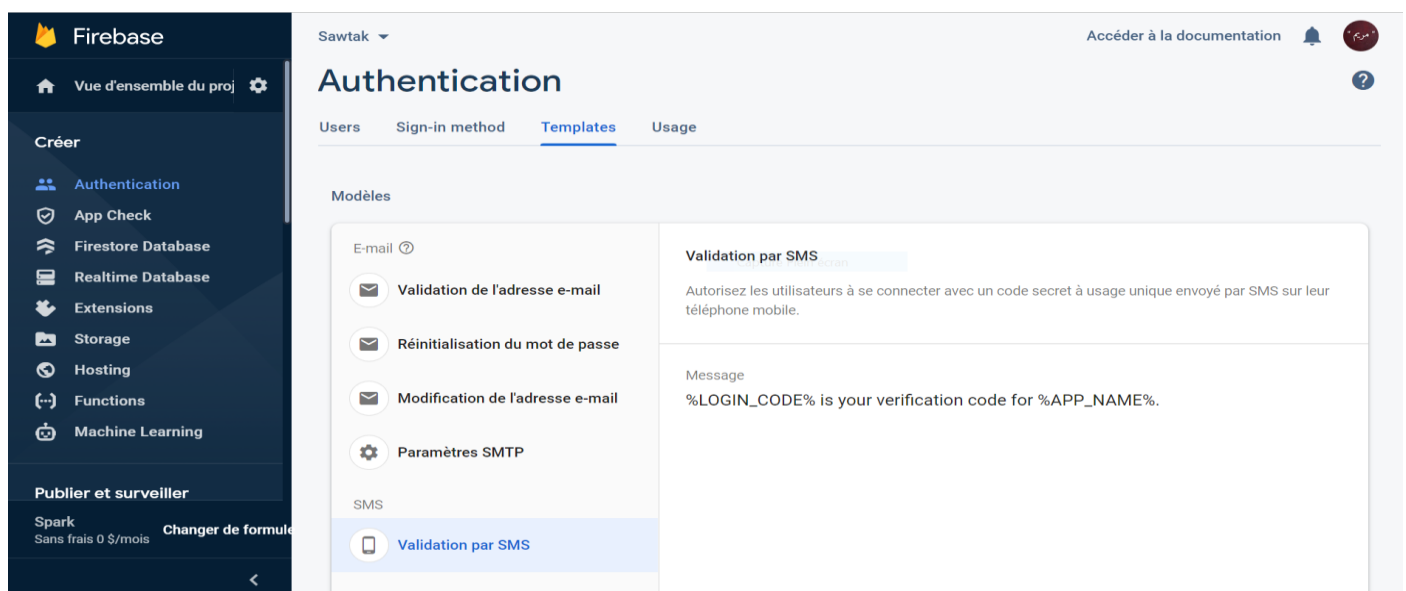
- ◆ Les outils de développement et de test de son application.
- ◆ Les outils permettant d'augmenter et d'engager ses cibles.

Voici quelques-uns de ces outils, des outils que les développeurs exploitent le plus dans le cadre du développement d'amplifications ou encore du test de performance des amplifications :



Authentification Firebase par numéro de téléphone OTP dans Android :

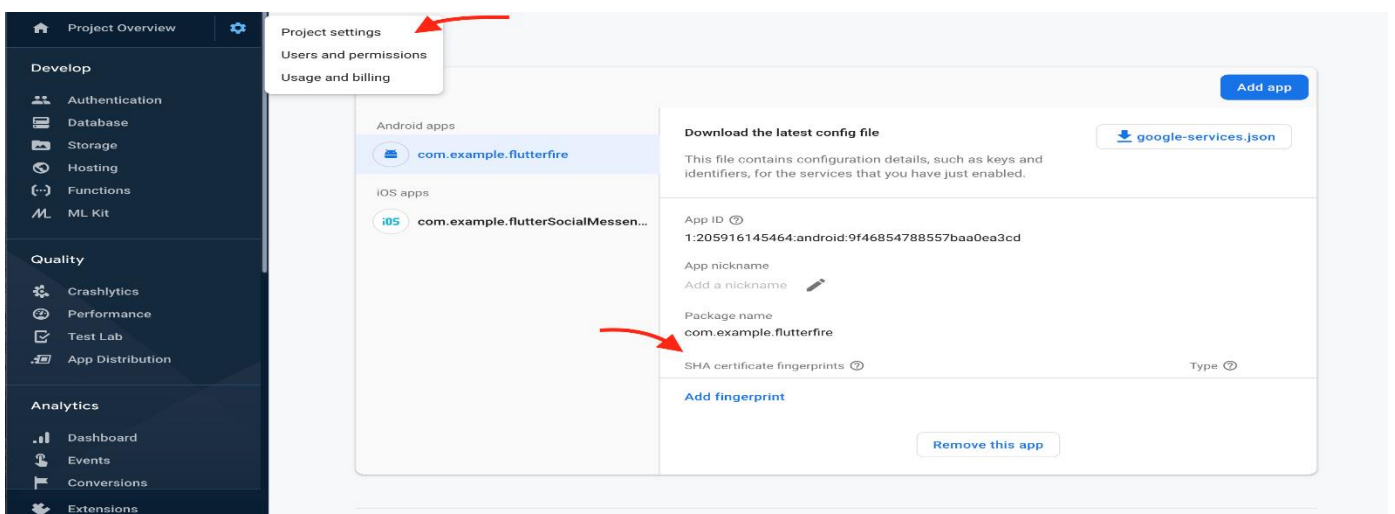
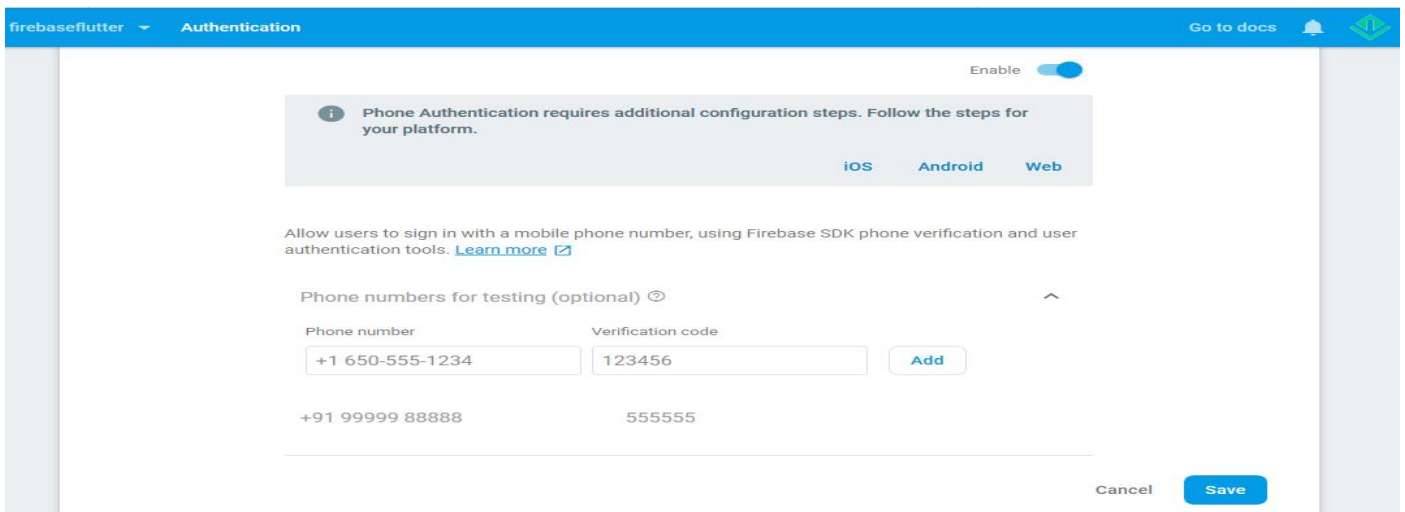
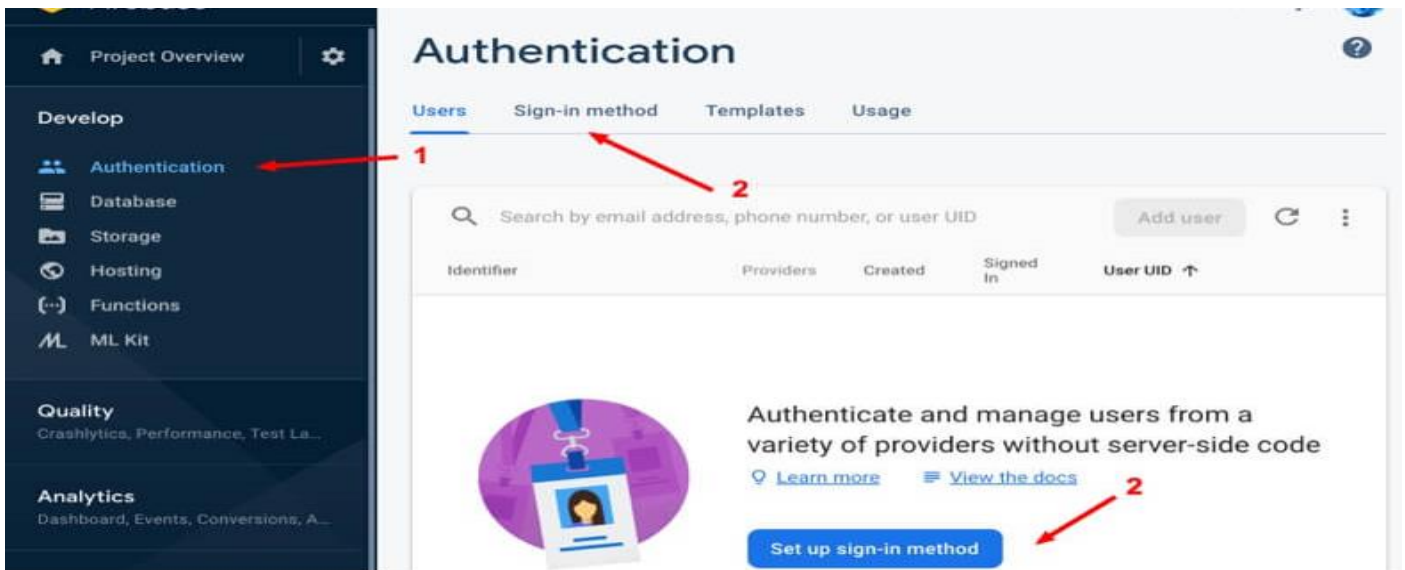
De nombreuses applications nécessitent que leurs utilisateurs soient authentifiés. Ainsi, dans le but d'authentifier les applications, l'authentification par numéro de téléphone est utilisée dans leurs applications. Dans l'authentification par téléphone, l'utilisateur doit vérifier son identité avec son numéro de téléphone. À l'intérieur de l'application, l'utilisateur doit entrer son numéro de téléphone, après quoi il recevra un code de vérification sur son numéro de téléphone portable. Il doit entrer ce code de vérification et vérifier son identité. C'est ainsi que fonctionne l'authentification par téléphone. Firebase fournit de nombreux moyens d'authentification des utilisateurs tels que Google, e-mail et mot de passe, téléphone et bien d'autres. Dans cet article, nous examinerons la mise en œuvre de l'authentification téléphonique dans notre application à l'aide de Firebase.



Comment ça-marche :

Nous allons créer une application qui comporte plusieurs pages. Le premier page sera notre page de vérification sur lequel l'utilisateur doit ajouter son numéro de téléphone. Après avoir ajouté son numéro de téléphone, l'utilisateur cliquera sur le bouton Obtenir OTP après que Firebase enverra OTP sur ce numéro qui est mentionné ci-dessus. Après avoir reçu cet OTP, l'utilisateur doit entrer cet OTP dans le texte ci-dessous et cliquer sur le bouton ci-dessous pour vérifier avec l'OTP entré. Après avoir cliqué sur le bouton de vérification, Firebase vérifiera qu'OTP et autorisera l'utilisateur à entrer dans l'écran d'accueil uniquement lorsque OTP saisi est correct, sinon l'utilisateur recevra un message d'erreur.

Configuration d'authentiquassions par numéro de téléphone :

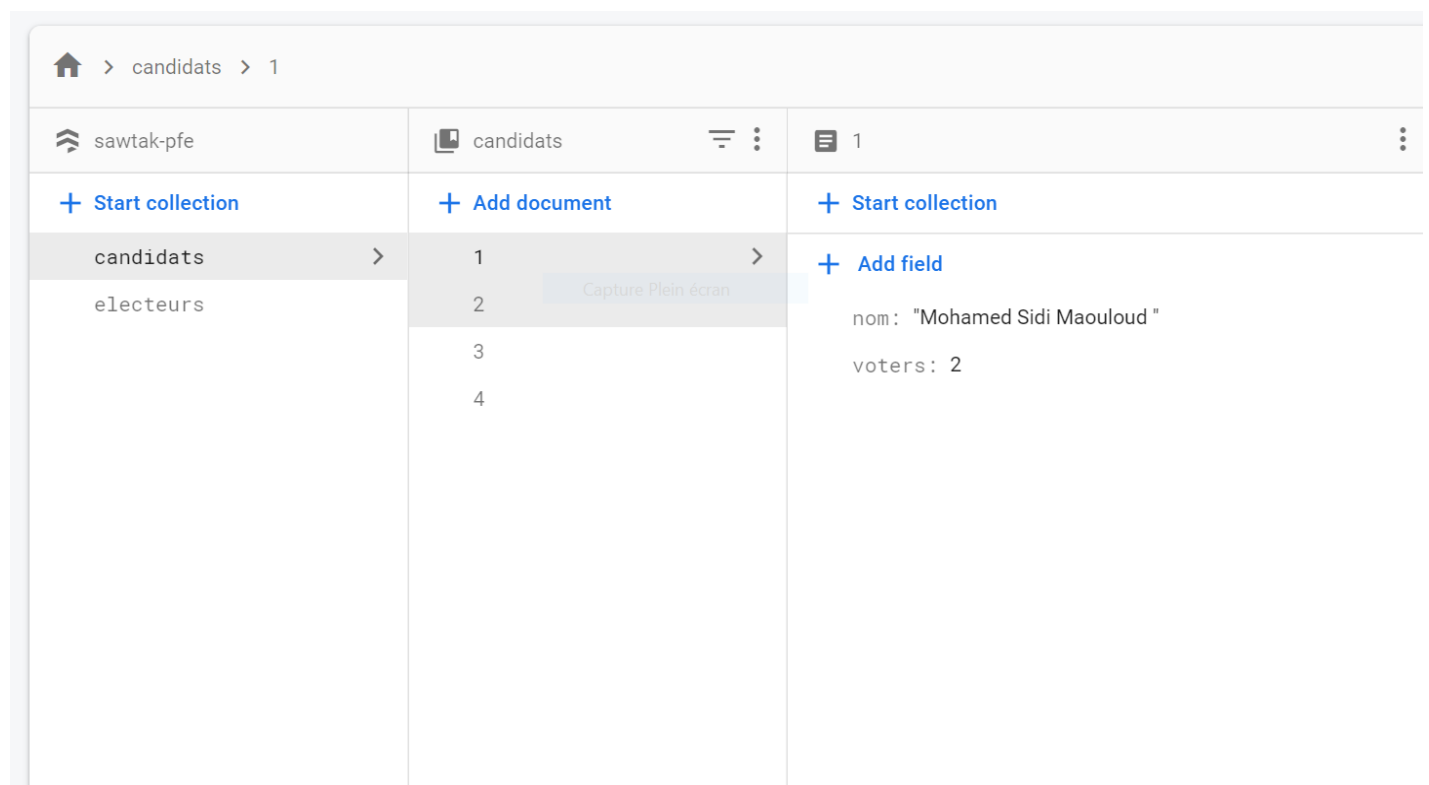


Développement :

Création de la base de données :

Tout d'abord, nous avons créé la base de données, ici appelée « **Sawtak** », ainsi que les tables (Collections) correspondantes :

- ♦ La collection « Electeur » contenant les numéros nationaux d'indentification (NNI) de tous les électeurs et les numéros de téléphone.
- ♦ La collection « Candidats » contenant les informations de tous les Candidats.



Flutter – Internationalisation :

L'internationalisation fait référence au terme selon lequel une application est disponible dans différentes langues régionales pour mieux toucher les gens. Pour cela, nous devons rendre une application disponible dans différentes langues et des mises en page adaptées. Flutter fournit des méthodes pour internationaliser l'application.

L'Internationalisation comporte plusieurs propriétés

- ◆ Propriété localisations Délégentes : Elle définit toutes les ressources localisées pour l'application.
- ◆ Propriété supporté Locales : Elle fournit une liste des langues prises en charge par l'application.
- ◆ Propriété on GeneratedTitle : Elle est appelée après la configuration du widget, ce qui signifie que la localisation est disponible.

L'une des choses qui aide avec cette fonction (Internationalisation) est l'utilisation de packages GetX

GetX est une solution ultra légère pour la gestion des états, de la navigation et des dépendances pour les applications Flutter.

GetX est livré avec quelques fonctionnalités dont vous aurez besoin dans le développement quotidien de vos applications dans Flutter.

L'interface de l'application :

Cette application est une solution adéquate pour le processus de vote.

L'accueil qui est l'interface ou la porte d'entrée de cette application est composée de quelques parties, ci-après:

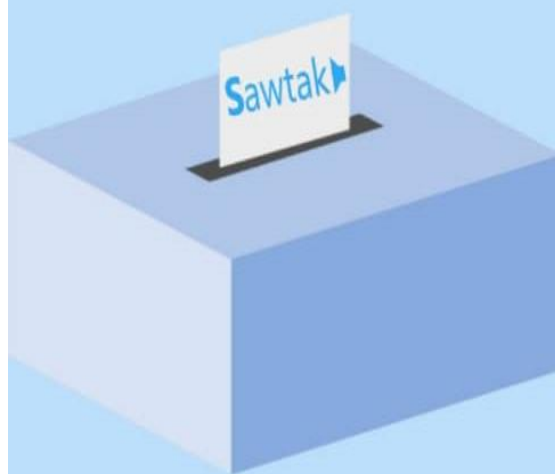
- ◆ Authentification.
- ◆ Voter.

Les liens de cette interface redirigent vers les différentes pages de l'application.

Page d'accueil :

Dans la page d'accueil de notre application, il apparaît le logo principal de l'application et des options permettant de changer la langue de l'application en français ou arabe. Pour que toute la communauté mauritanienne puisse utiliser cette application.

Et un bouton <<Voter>> lorsque vous appuyez sur ce bouton, vous accédez à la page suivante (page d'accueil).



Voter

Page de vérification :

Ici la présence de la personne est vérifiée, et cela se fait en vérifiant si la personne est inscrite sur la liste électorale.

Et dans cette page de l'application, on peut étudier trois solutions, qui sont les suivantes :

On vérifie l'existence et la validité du numéro d'identification national et du numéro de téléphone et ces deux doivent être enregistré dans la base donnée utilisé(Firebase) et cette fonctionnalité est traiter par de nombreuses fonctions appartenir de flutter et Dart.

Et de plus l'électeur ne pourra être autorisé à voter qu'après avoir vérifié la validité des deux.

Après avoir vérifié la validité du numéro d'identification national et du numéro de téléphone,

Et après avoir appuyé sur le bouton <<Envoyer>> l'électeur reçoit un SMS portant le code de vérification et après son validation l'électeur arrivera à la page de vote.



Numéro de téléphone




NNI

Verifier

Page OTP :

Ici le code envoyé au numéro de téléphone est saisi après vérification de l'existence du numéro de téléphone et du numéro d'identification national.



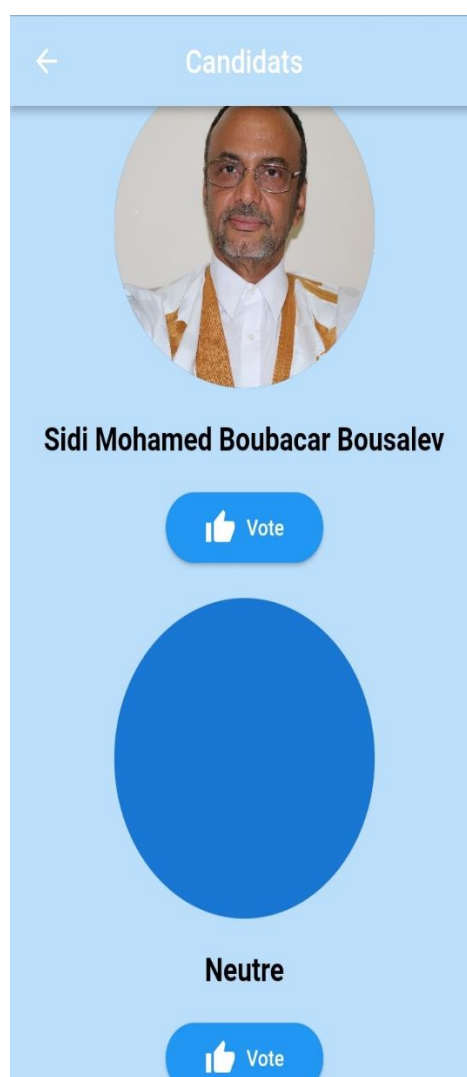
Le code de vérification

Suivant

Page candidats :

Ici le choix est fait, pour que chaque point de vue exprime son opinion et choisit entre les candidats ou entre la neutralité, une opinion qu'il faut exprimer pour que l'identité et la liberté d'expression ne manquent pas à l'heure de la démocratie et de la liberté d'opinion.

Le processus de vote a lieu, d'un simple clic sur le bouton du candidat sélectionné.



Page de fin :

Apparaît lorsque le vote de l'électeur pour son candidat est enregistré.



Vous avez voté

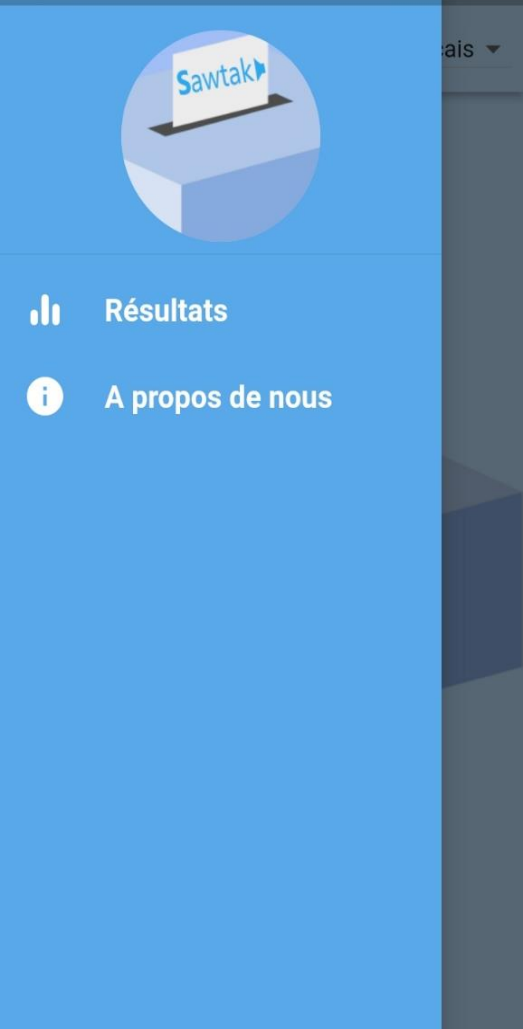
Statistique (Résultats) :

Les résultats électoraux reflètent le résultat et le niveau de participation aux compétitions électorales. Et comme que l'état des résultats est un aspect essentiel de chaque élection, cette application permet aux citoyens de suivre les résultats électoraux et leur évolution rapide.

Un citoyen peut voir quel rang son candidat obtient selon le pourcentage obtenu.

Ce processus a été fait grâce à l'utilisation de certaines fonctions du langage Dart, et à travers les nombres de votes Extrait de la base de données, qui augmente tout le temps, Le pourcentage est calculé pour chaque candidat, et ce résultat Change par le succès d'un candidat et l'échec d'un autre également avec l'évolution des votes du candidat.

D'un côté de cette application, le résultat de chaque candidat sera affiché avec son nom, ainsi que l'apparition du pourcentage de votes neutres.



← Résultats	
Mohamed Sidi Maouloud	29%
Mohamed Cheikh Mohamed Ahmed Elghazouani	59%
Sidi Mohamed Boubacar Bousalev	6%
Neuter	6%

Conclusion

Notre projet portait sur la conception d'une application pour répondre à la problématique de mise en place d'une solution de vote en ligne. Le travail présenté dans ce mémoire a été le fruit d'une réflexion approfondie.

Pour atteindre cet objectif nous avons choisi le langage graphique de modélisation informatique « UML » pour modéliser cette application, aussi bien sur le plan de définition des besoins, de conception ou d'analyse. Nous sommes également efforcés à trouver des solutions les plus simples et les moins exigeantes en termes de méthodes de conception et de logistique, dans le but de nous concentrer plus sur la réalisation pratique du projet.

Notre étude ouvre la porte sur des futures améliorations qui peuvent perfectionner cette application et la rendre plus fiable et à la portée d'un public plus large.

REFERENCES

Bibliographie :

Quelques rapports de la bibliothèque de l'institut ISCAE.

Webographie :

- <https://www.futura-sciences.com/tech/definitions/informatique-uml-3979/>
- https://fr.m.wikipedia.org/wiki/Diagramme_de_cas_d%27utilisation
- <https://logitheque.univ-bejaia.dz/mod/glossary/showentry.php?eid=171>
- https://fr.m.wikipedia.org/wiki/Visual_Studio_Code
- [https://fr.m.wikipedia.org/wiki/Flutter_\(logiciel\)](https://fr.m.wikipedia.org/wiki/Flutter_(logiciel))
- [https://fr.m.wikipedia.org/wiki/Dart_\(langage\)](https://fr.m.wikipedia.org/wiki/Dart_(langage))
- <https://fr.acervolima.com/authentification-firebase-avec-numero-de-telephone-otp-dans-android/>
- <https://fr.acervolima.com/flutter-internationalisation/>
- https://youtube.com/playlist?list=PLw6Y5u47CYq47oDw63bMqkq06fjuoK_GJ
- <https://youtu.be/d9U55nnUo2E>

Annexes

Code de Controller :

Fonction vote :

```
sawtak_pfe > lib > controllers.dart > CandidatesController > vote
163
164 Future<void> vote(String value) async {
165     final user = app.user;
166
167     if (user == null) {
168         return;
169     }
170
171     if (user.isVoted) {
172         Get.defaultDialog(
173             title: 'Attention'.tr,
174             middleText: 'Vous avez déjà voté'.tr,
175             textCancel: 'Ok'.tr);
176         return;
177     }
178
179     loading.value = true;
180     try {
181         DocumentReference electeurReference =
182             FirebaseFirestore.instance.collection(electeursName).doc(user.doc.id);
183
184         DocumentReference candidateReference =
185             FirebaseFirestore.instance.collection(candidatesName).doc(value);
186
187         await FirebaseFirestore.instance.runTransaction((transaction) async {
188             DocumentSnapshot candidate = await transaction.get(candidateReference);
189
190             if (!candidate.exists) {
191                 throw Exception("le candidat n'existe pas");
192             }
193         });
194     } catch (e) {
195         loading.value = false;
196         return;
197     }
198     return;
199 }
```

```
sawtak_pfe > lib > controllers.dart > CandidatesController > vote
96 loading.value = false;
97 return;
98 } else if (electeur.get('NNI') != nniField.text) {
99     nniError.value = 'NNi introuvable'.tr;
100     loading.value = false;
101     return;
102 }
103
104 await auth.verifyPhoneNumber(
105     phoneNumber: phone,
106     verificationCompleted: (PhoneAuthCredential credential) async {
107         await signInWithCredential(credential);
108     },
109     verificationFailed: (FirebaseAuthException e) {
110         phoneError.value = e.message;
111         loading.value = false;
112     },
113     codeSent: (String verificationId, int? resendToken) async {
114         this.verificationId.value = verificationId;
115         pageController.animateToPage(1,
116             duration: 200.milliseconds, curve: Curves.easeIn);
117         loading.value = false;
118     },
119     codeAutoRetrievalTimeout: (String verificationId) {},
120 );
121 autoValidate.value = false;
122 } else {
123     autoValidate.value = true;
124 }
125 }
```

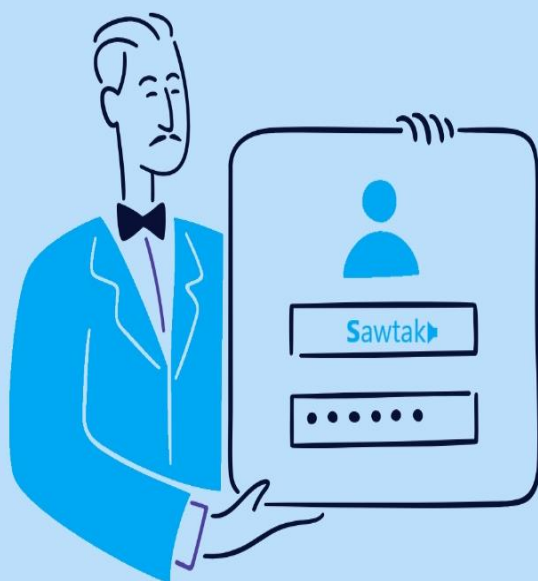
Candidat Controller :

```
sawtak_pfe > lib > controllers.dart > CandidatesController > vote
158
159 class CandidatesController extends GetxController {
160   final loading = false.obs;
161
162   AppService get app => Get.find();
163
164   Future<void> vote(String value) async {
165     final user = app.user;
166
167     if (user == null) {
168       return;
169     }
170
171     if (user.isVoted) {
172       Get.defaultDialog(
173         title: 'Attention'.tr,
174         middleText: 'Vous avez déjà voté'.tr,
175         textCancel: 'Ok'.tr);
176       return;
177     }
178
179     loading.value = true;
180     try {
181       DocumentReference electeurReference =
182         FirebaseFirestore.instance.collection(electeursName).doc(user.doc.id);
183
184       DocumentReference candidateReference =
185         FirebaseFirestore.instance.collection(candidatesName).doc(value);
186
187       await FirebaseFirestore.instance.runTransaction((transaction) async {
```

Capture Plein écran

```
sawtak_pfe > lib > controllers.dart > CandidatesController > vote
187       await FirebaseFirestore.instance.runTransaction((transaction) async {
188         DocumentSnapshot candidate = await transaction.get(candidateReference);
189
190         if (!candidate.exists) {
191           throw Exception("le candidat n'existe pas");
192         }
193
194         int votersCount = candidate.get('voters') + 1;
195
196         transaction.update(candidateReference, {'voters': votersCount});
197
198         transaction.update(electeurReference, {'voted': candidate.id});
199       });
200
201       app.user = User(await electeurReference.get());
202
203       final name = (await candidateReference.get()).get('name');
204       Get.offAllNamed('/finish', parameters: {'name': name});
205     } catch (e) {
206       Get.defaultDialog(
207         title: 'Opératinos'.tr,
208         middleText: 'Le lissage a mal tourné, veuillez réessayer!'.tr,
209         textCancel: 'Ok'.tr);
210       loading.value = false;
211     }
212   }
213 }
```

Capture Plein écran

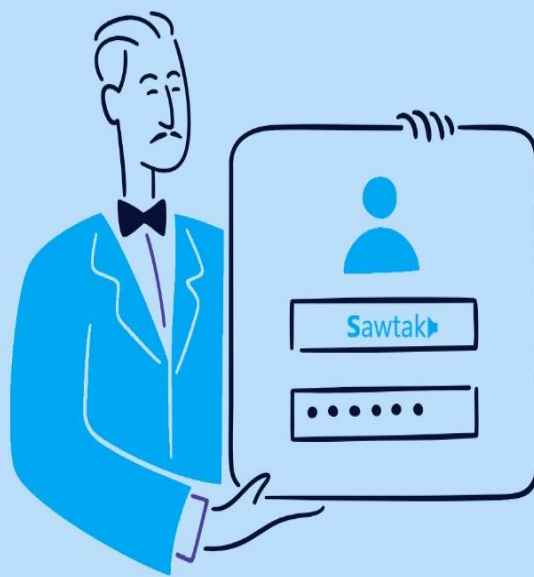


 +222 +22236369999

Vous n'êtes pas enregistré dans la base de ...

 7042849222

Verifier



 +222 +22232446349

 1245639

Ce champ doit être supérieur à 10 chiffres longs

Verifier

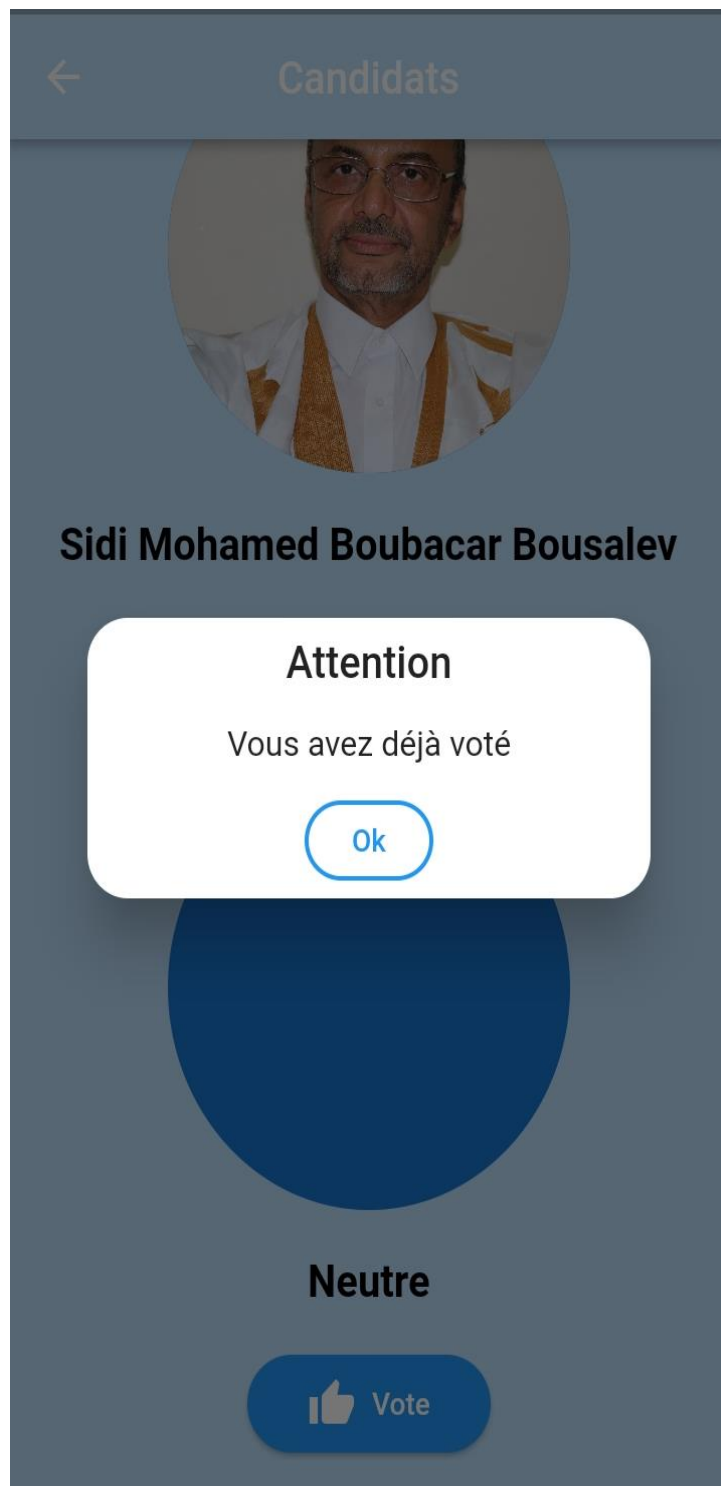


+222 +22232446349

12454807580

NNi introuvable

Verifier



← A propos de nous

A propos de notre application :

L'objectif de ce guide est de montrer combien est facile de voter sur notre application pour cela nous allons parcourir les étapes pour le vote dans cette application :

Dans cette application le numémo d'itentié doit être saisi dans le base de données et validé a a partir de l'etat civile.

Cette application consiste a voter vous devez saisir votre numéro d'identité et le numéro de téléphone vous recoit un code de verification.

En entrant le code de verification, vous passez a la page qui affiche la liste de candidat vous devez confirmer un candidat en cliquat sur le bouton <<Voter>>.

Cette application permettra de voir les résultat du vote en cliquant sur le lien Résultats.

