



Rosetta

Veille technologique

Projet de synthèse 2020-2021

Maîtrise d'ouvrage : Thierry Hourdier
90 avenue de la Marne 56000 Vannes

Maître d'œuvre : Matthieu Le Lain

Maîtrise d'œuvre :

Chef de projet : Christophe Garcia

Responsable de communication : Alice Gontard

Responsable de la documentation : Lucy Gastebois

Responsable des tests : Sullivan Leboeuf



Table des matières

I. Introduction	3
II. Application Native	4
1. Présentation de l'application native	4
2. Les outils utilisés pour l'application native	4
III. Application Hybride	5
1. Présentation de l'application hybride	5
2. Les outils utilisés pour l'application hybride	5
IV. Site Web	6
V. API et base de données	7
VI. Planning prévisionnel	8
VII. Calendrier	10
VIII. Conclusion	11

I. Introduction

Pour mener à bien le déroulement du projet, nous organisons 2 rendez-vous hebdomadaires :

- Mardi 12h, avec l'équipe de développement et le maître d'œuvre, afin de mettre à jour notre avancement, poser des questions, et s'interroger sur notre efficacité.
- Dimanche 18h, avec seulement l'équipe de développement, afin de savoir ce qu'a fait chacun durant la semaine, ce qu'on s'apprête à faire lors de la semaine qui suit, et si nous avons rencontré des obstacles. Ce rendez-vous, qui reprend le principe du Daily Scrum Meeting, a lieu sur Discord une fois par semaine, étant donné que nos journées sont principalement occupées par nos études, cela nous laisse ainsi le temps d'avancer les tâches qui nous auront été attribuées et de préparer la réunion.

Pour réaliser l'application qui simplifie le travail quotidien de M. Hourdier, nous avons 2 solutions qui s'offrent à nous :

1. Application native et une interface web classique. Ces deux éléments distincts se relieront par un mode de synchronisation.
2. Application web hybride, qui consiste en un site web responsive. Le site web serait encapsulé dans l'application, ce serait un petit navigateur.

Nous verrons dans les 2 parties suivantes les avantages et inconvénients de chacune des solutions, et nous expliquerons notre choix lors de la conclusion.



II. Application Native

1. Présentation de l'application native

Une application native est une application qui a été développée pour un système d'exploitation précis (IOS ou Android par exemple), car chaque OS à son propre langage de programmation.

Comparée à la version hybride, la version native à plusieurs avantages :

- Meilleure rapidité, fiabilité et dotée d'une meilleure réactivité ainsi qu'une résolution supérieure, ce qui assure une meilleure expérience utilisateur.
- Elle facilite l'accès à toutes les fonctionnalités du téléphone.
- Les notifications push sont uniquement disponibles sur les apps natives. Ces notifications permettent d'alerter les utilisateurs et d'attirer leur attention, que ce soit pour du nouveau contenu ou une offre promotionnelle.
- Ne requiert pas forcément internet pour fonctionner, ce qui est un réel avantage.

Le principal inconvénient d'une application native est qu'elle doit respecter les règles définies par les différentes sociétés des plateformes mobiles. Que ce soit l'approbation nécessaire des Apps Store pour diffuser l'application ou ses mises à jour, les conditions tarifaires imposées ou la non-compatibilité avec les autres OS.

Le coût lié au développement est un frein car généralement plus élevé si elle est portée sur plusieurs plateformes que le coût d'une application hybride, mais ce n'est pas le cas si le projet ne se réalise que sur un OS (ex : Android).

Pour que l'utilisateur ait accès à la dernière version il faut qu'il la mette à jour depuis le store contrairement aux sites mobiles qui se mettent à jour directement.

2. Les outils utilisés pour l'application native

Nous avons choisi le langage Java pour développer l'application Android car nous avons une bonne maîtrise de ce langage et nous saurons régler plus simplement les éventuels problèmes que nous rencontrerons.



Tester fait partie du développement, pour nous aider à réaliser ces tests, nous utiliserons : JUnit. Ce Framework fait partie de la suite xUnit qui concerne les tests unitaires des fonctionnalités codées. Ce dernier a pour avantage de pouvoir s'intégrer facilement dans Eclipse.

Eclipse est un IDE complet pour du développement en Java, il inclut en plus le Framework JUnit. Android Studio est utilisé par les développeurs spécifiquement pour le développement d'applications Android. Pour le développement de l'application native nous utiliserons les deux IDE, Eclipse et Android Studio, selon nos besoins.



III. Application Hybride

1. Présentation de l'application hybride

L'application hybride est une application pour mobiles qui combine des éléments des applications dites natives (développées pour une plateforme en particulier, iOS ou Android par exemple) et des web apps (disponibles sur internet via un navigateur et non installées localement). Autrement- dit, le site web est encapsulé dans l'application.

Cette application utilise le navigateur web intégré à la tablette ou au Smartphone, ainsi que les technologies Web comme HTML ou CSS. Ce concept permet de fonctionner sur différent OS (IOS, Android ...).

Les applications hybrides sont des applications disponibles sur les App stores et qui s'installent comme n'importe quelle autre application.

Elle a plusieurs avantages :

- Facile et rapide à développer comparée à une application native.
- La maintenance sera également plus facile puisqu'il n'y a qu'une seule version à revoir pour plusieurs plateformes.

Cependant, cette application possède de multiples inconvénients :

- Des performances et des transitions entre les pages plus lentes.
- Une interface utilisateur limitée.
- Une dépendance vis à vis de la vitesse du navigateur, et donc parfois suivant la connexion internet.
- La difficulté voire l'impossibilité d'exploiter toutes les capacités des plateformes.

2. Les outils utilisés pour l'application hybride

Les langages utilisés pour développer une application hybride sont CSS, HTML et Javascript.

HTML est un langage descriptif utilisé pour structurer le contenu d'une page (ses textes, ses images, ses liens, etc.).

CSS permet de mettre en forme les fichiers HTML ou XML, et ainsi donc de gérer le design d'une page HTML.

JavaScript est un langage de programmation qui permet d'implémenter des mécanismes complexes sur une page web.



IV. Site Web

En ce qui concerne le site web, nous avons deux solutions.

- La première solution est de reprendre le site actuel de Monsieur Thierry Hourdier à cette adresse : <https://hourdierthierry.site-solocal.com/>
On ajoutera donc juste une page privée pour qu'il puisse faire ses devis et les visualiser. M. Hourdier aurait donc un seul site pour son commerce et pour ses devis. M. Hourdier pourra compléter, modifier ou ajouter des éléments au devis réalisés plus tôt dans la journée.
- La deuxième solution est de créer un second site qui lui permettra de se connecter pour finir ses devis, les compléter, modifier ou ajouter des éléments au devis réalisés plus tôt dans la journée.
M. Hourdier aura donc deux sites, un qui sera public pour tout le monde et un qui sera donc privé pour lui-même.

Un système de connexion sera mis en place afin d'accéder à cette partie du site internet et ainsi y renforcer la sécurité.

On utilisera les outils suivants :



Pour réaliser le site web de M. Hourdier, nous allons manipuler le logiciel WordPress.

Ce logiciel propose une multitude de thèmes pour réaliser un site web et est très simple d'utilisation. De plus, il supporte de nombreux types de médias ce qui est idéal pour la réalisation des devis. WordPress permet d'ajouter des fonctionnalités de façon rapide et efficace.

Le langage HTML sera utilisé pour ajouter du contenu ou des détails sur le site web demandant plus de précision ou d'exigence.



Le langage d'intégration CSS nous permettra de définir la forme du site et d'avoir quelque chose de visuellement simple, si plus de précision ou d'exigence est demandée.

V. API et base de données

Une base de données est nécessaire au projet, afin de sauvegarder les devis réalisés, les clients enregistrés ou encore les scénarios créés.

Ici, nous optons pour la création de 2 bases de données :

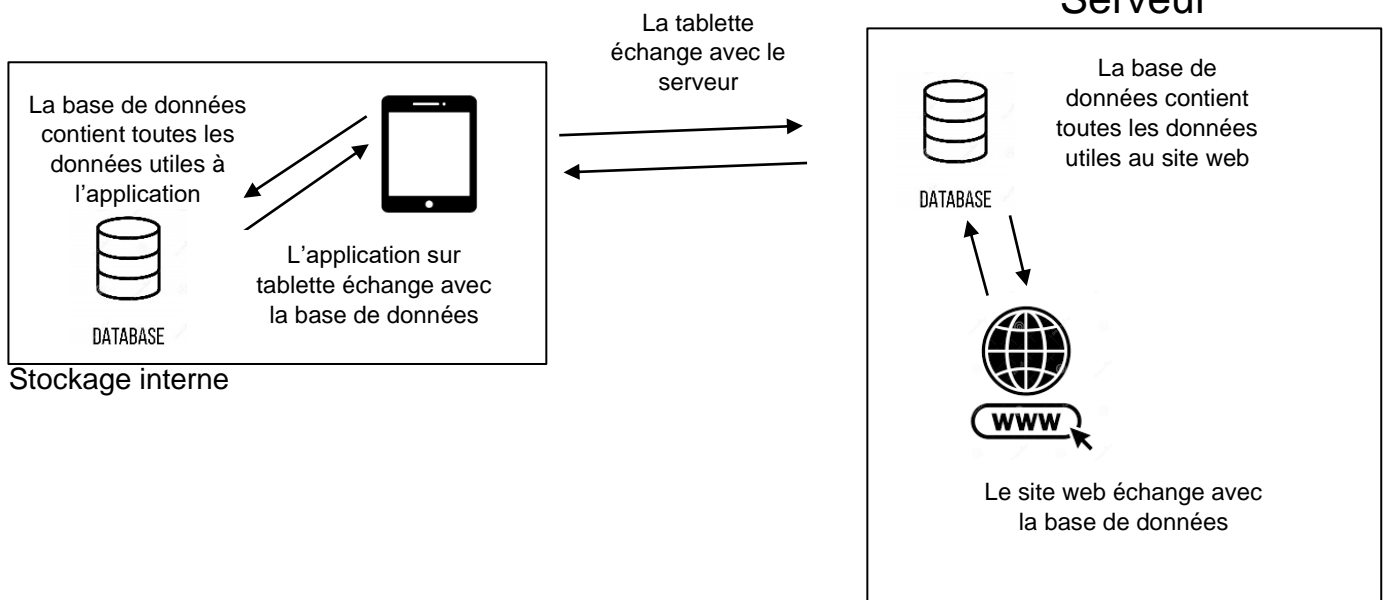
- Une base de données interne, qui permet de faire tourner l'application hors ligne (tablette déconnectée du net)
- Une base de données externe, qui permet de récupérer les dernières modifications, qu'elles soient faites via l'application ou bien le site internet

Lorsque des modifications sont faites sur le site web (ex : nouveau client, nouveau devis, suppression d'un scénario, ...), la base de données externe est automatiquement mise à jour. Et lorsque les modifications sont faites sur la tablette, la base de données externe se synchronisera en récupérant les modifications depuis la base de données interne quand la tablette sera connectée à internet.

La base de données interne à l'application sera développée avec SQLite, et se remettra à jour à chaque synchronisation.

La base de données externe sera développée avec Oracle, et sera hébergée sur le même serveur que celui du site internet. MySQL ne dispose pas de fonctionnalités de stockage étendues et ne supporte pas XML, XML étant utilisé pour la synchronisation.

Schéma récapitulatif :



Après chaque synchronisation, les deux bases de données contiennent les mêmes informations.

VI. Planning prévisionnel

Rosetta

12 nov. 2020

<https://gitlab.com/G.Christophe/rosetta>

Chef de projet	Christophe G
Dates du projet	21 sept. 2020 - 12 avr. 2021
Avancée	20%
Tâches	26
Ressources	4

Création d'une application web et mobile pour M. Thierry Hourdier, plombier et électricien situé à Vannes. Le but de cette application est de faciliter la réalisation de ses devis, du début à la fin. Ce planning nous servira à chacun d'entre nous, afin de voir l'avancement du projet, les tâches qui sont à réaliser, les personnes qui doivent les faire, et surtout, d'éviter de prendre un seul retard.

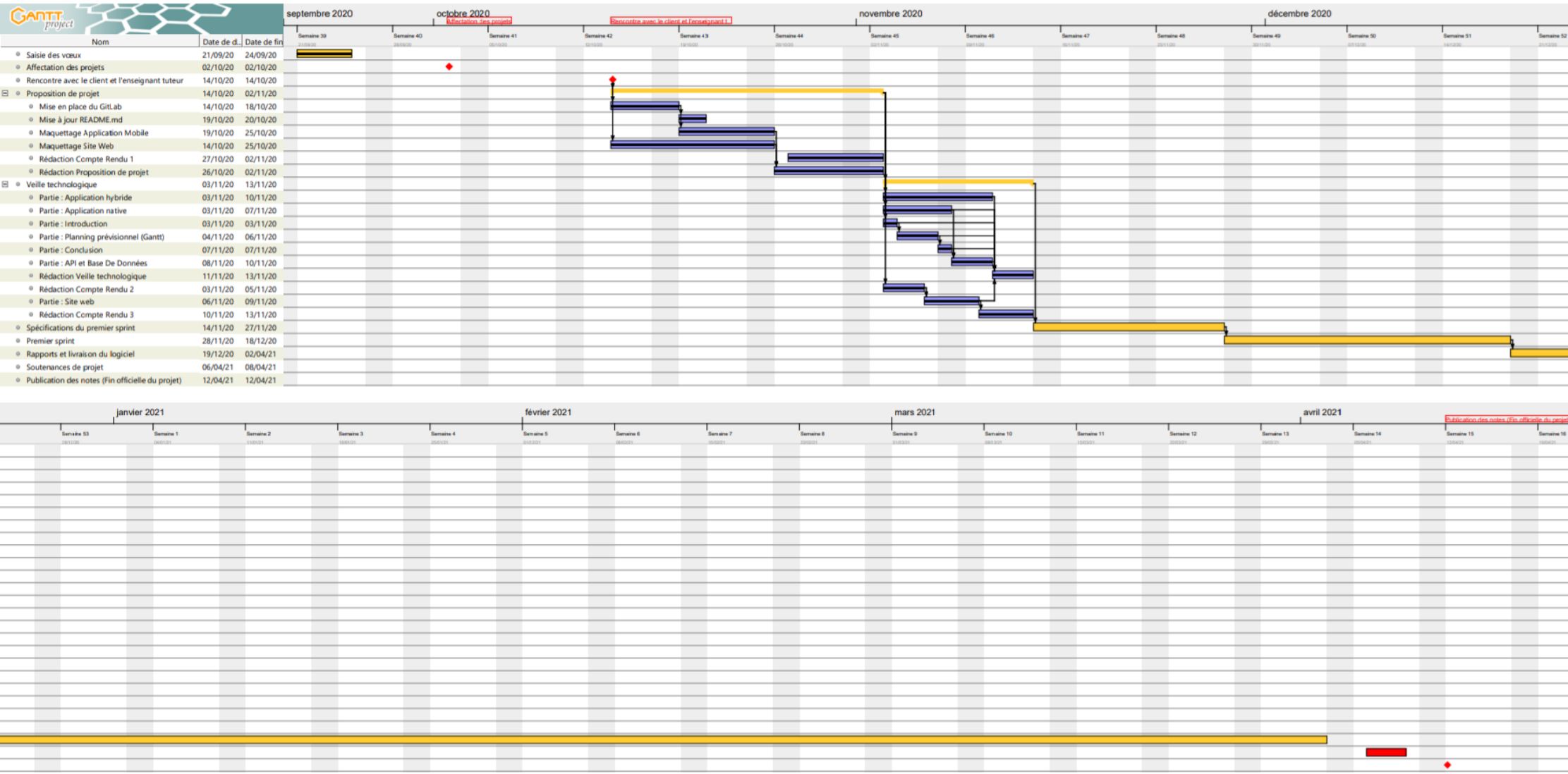
Tâches

Nom	Date de début	Date de fin
Saisie des vœux <i>Choix de projets :</i> - OmniBoardEditor - Mady - Rosetta - OppRoadInfo - GameOfMaths - Kiaz <i>Nous avons tous voulu avoir le projet Rosetta.</i>	21/09/20	24/09/20
Affectation des projets <i>Le projet Rosetta nous a été attribué. Nous procédons aux différentes étapes afin rencontrer le client.</i>	02/10/20	02/10/20
Rencontre avec le client et l'enseignant tuteur <i>M. Roirand était avec nous lors du premier rendez-vous. Un erreur de sa part puisque notre enseignant tuteur est M. Le Lain. Le rendez-vous s'est très bien passé. Tous les groupes du projet Rosetta étaient présents.</i>	14/10/20	14/10/20
Proposition de projet <i>Suite au rendez-vous pris avec M. Hourdier, nous devons réaliser une proposition de projet qui lui correspond, mais surtout qui lui plaît, au maximum. Mise en forme de l'application Android et du site web, cette proposition lui a été envoyée par mail, et le retour n'a été que très positif !</i>	14/10/20	02/11/20
Mise en place du GitLab	14/10/20	18/10/20
Mise à jour README.md	19/10/20	20/10/20
Maquettage Application Mobile	19/10/20	25/10/20
Maquettage Site Web	14/10/20	25/10/20
Rédaction Compte Rendu 1	27/10/20	02/11/20
Rédaction Proposition de projet	26/10/20	02/11/20
Veille technologique <i>La veille technologique permet d'avoir une vision plus précise sur le projet Rosetta. Ici, nous présentons les différentes solutions possibles à la réalisation de l'application Android, du site web, et de la synchronisation. Nous y expliquons également notre choix qu'est la réalisation d'une application native accompagnée de son site web et deux bases de données.</i>	03/11/20	13/11/20
Partie : Application hybride	03/11/20	10/11/20
Partie : Application native	03/11/20	07/11/20
Partie : Introduction	03/11/20	03/11/20
Partie : Planning prévisionnel (Gantt)	04/11/20	06/11/20
Partie : Conclusion	07/11/20	07/11/20
Partie : API et Base De Données	08/11/20	10/11/20
Rédaction Veille technologique	11/11/20	13/11/20
Rédaction Compte Rendu 2	03/11/20	05/11/20
Partie : Site web	06/11/20	09/11/20
Rédaction Compte Rendu 3	10/11/20	13/11/20
Spécifications du premier sprint	14/11/20	27/11/20
Premier sprint	28/11/20	18/12/20
Rapports et livraison du logiciel	19/12/20	02/04/21
Soutenances de projet	06/04/21	08/04/21
Publication des notes (Fin officielle du projet)	12/04/21	12/04/21

Ressources

Nom	Rôle par défaut
Lucy G	Editeur de doc
Sullivan LB	Testeur
Alice G	Chargée de communication
Christophe G	Chef de projet

Diagramme de Gantt :



VII. Calendrier

- Proposition de projet : 02 novembre 2020
- Veille technologique : 13 novembre 2020
- Spécification du premier sprint : 27 novembre 2020
- Premier sprint : 18 décembre 2020
- Rapports et livraison du logiciel : 2 avril 2021

VIII. Conclusion

Pour conclure, nous devons choisir la solution la plus optimale pour M. Hourdier, et nous allons expliquer notre choix. Pour cela, il faut prendre en compte certaines variables nécessaires afin de prendre la bonne décision :

Tout d'abord, notre équipe de développeurs est plus expérimentée dans le développement d'une application native que celui d'une application hybride. Nous aurons donc plus d'aisance à mettre en œuvre les choix de M. Thierry Hourdier, à appliquer ses idées, et à s'adapter à tout changement. Cela signifie donc que nous arriverons plus facilement à satisfaire M. Hourdier en réalisant une application native.

De plus, nous n'avons que peu de temps pour développer l'application (4,5 mois), étant donné que nous avons des études en parallèle, des partiels et des recherches de stage. Réaliser une application hybride demanderait un temps supplémentaire qu'est celui de l'auto-formation de l'équipe. Cela signifie également que nous accorderons plus de temps sur les fonctionnalités de l'application Android et du site web en élisant la solution de l'application native.

Enfin, M. Hourdier souhaite que l'application marche sur sa tablette Android, ainsi en choisissant de réaliser une application native, son expérience en tant qu'utilisateur ne pourra qu'être plus personnalisable et donc meilleure. Le seul inconvénient qu'on aurait pu trouver, c'était la non-compatibilité sur toutes les plateformes, or la question ne se pose pas, l'application fonctionnera uniquement et parfaitement sur Android.

Ayant tous bien discuté puis évalué la situation, nous avons finalement choisi de réaliser une application native Android principalement développée à l'aide du langage Java, accompagnée d'un site web conçu avec le système de gestion de contenu WordPress. Plus de temps pour s'adapter et plus de qualité, la solution choisie est donc pour nous, mais surtout pour M. Hourdier, la plus intéressante.