

Rapport du projet

Application de compagnon virtuel sur smartphone



Étudiants : Mathis Ruffieux et Néréïs Dugaleix

Encadrante : Mme. Lydie du Bousquet

Tuteur : Mr. Damien Pellier

Informations d'identification du document

Référence du document	D4
Version du document	1.01
Date du document	13/06/2022
Auteurs	Mathis Ruffieux et Néréïs Dugaleix

Éléments de vérification du document

Validé par	Mme. Lydie du Bousquet
Validé le	
Soumis le	13/06/2022
Type de diffusion	Document électronique (pdf)
Confidentialité	Non confidentiel

Sommaire

I) Introduction	4
Contexte	4
II) Choix techniques	4
III) Application	6
1) Les scénarios de dialogue	6
1.1) Scénario d'introduction	6
1.2) Scénarios globaux	6
2) Le journal	9
3) Les réglages	9
IV) Conclusion	10
V) Bibliographie	11
VI) Annexes	11
VII) Remerciements	12

I) Introduction

1) Contexte

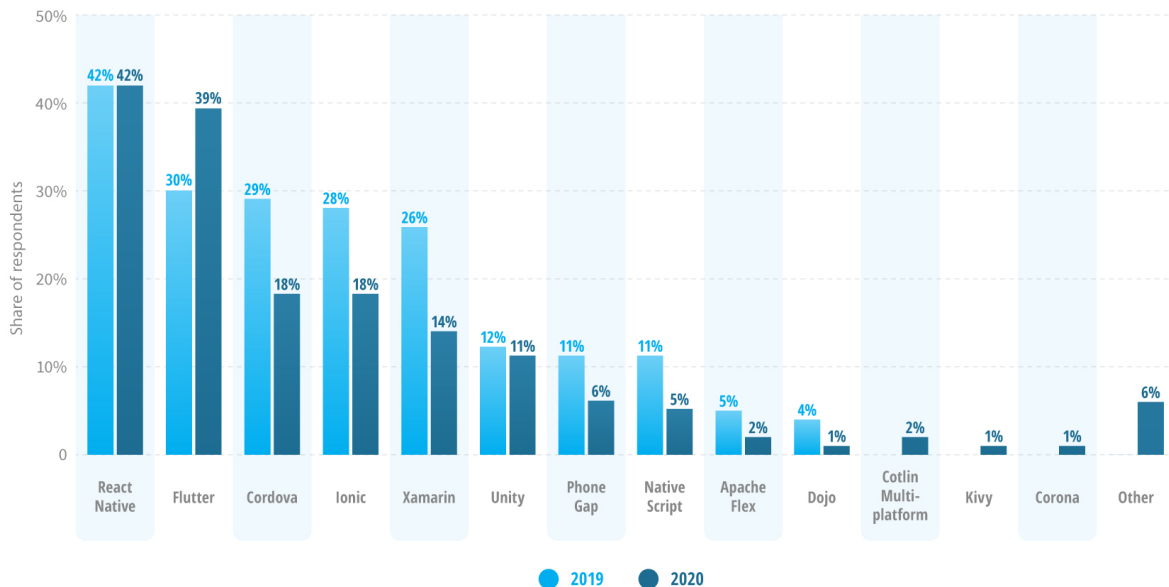
Ce projet s'inscrit dans le cadre du projet SCUSI Kouno Tori. Un partenariat entre l'Université de Kobe au Japon et l'Université Grenoble Alpes à permis la réalisation de ce projet. L'application de compagnon virtuel a pour but de converser avec des personnes isolées, comme des personnes âgées. Tout au long de l'année, nous avons travaillé sur ce projet afin de le mettre en place. Nous avons tout d'abord dû apprendre les bases de la programmation en flutter et en dart, puis nous avons réalisé une interface en adéquation avec les besoins de Mme Du Bousquet.

II) Choix techniques

1) Langages de programmation

L'application de compagnon virtuel a été développée avec les langages de Flutter et Dart, car Flutter prend en charge la création d'applications multiplateformes. Nous n'avons jamais utilisé ces langages auparavant, mais on observe une croissance forte de l'utilisation de ces langages. L'application de compagnon virtuel était donc une bonne opportunité pour apprendre ces langages et les mettre en pratique.

Cross-platform mobile frameworks used by software developers worldwide in 2019 and 2020



Source: Statista 2020

2) Bases de données

Pour les bases de données, l'application utilise SQLite (un package Flutter implémentant SQLite) qui permet d'enregistrer les données en local sur le téléphone. Afin de respecter la commission d'éthique, il ne faut pas que les données de l'utilisateur soient stockées en ligne, comme avec Firebase.

3) Import export des données

Nous avons choisi d'utiliser le format Json pour exporter les données des utilisateurs, car il existe en Flutter des fonctions pour passer d'un type Map à une liste convertible en Json.

De la même façon, on peut écrire dans un fichier Json pour importer de nouveaux scénarios sur l'application d'un utilisateur, en déplaçant le fichier directement sur son téléphone. Ce fichier sera lu et écrit dans les bases de données SQLite.

III) Application

1) Les scénarios de dialogue

1.1) Scénario d'introduction

Lors du premier lancement de l'application, un scénario d'introduction est lancé. Ce scénario n'est pas lancé lors des démarrages suivants. Il permet de présenter l'objectif de l'application à l'utilisateur et de lui demander des informations tel que son prénom, afin de l'enregistrer en base. Les variables enregistrées en base peuvent être utilisées par le robot.

Le scénario d'introduction est visible en détail au début de l'annexe *Scenarios.pdf*.

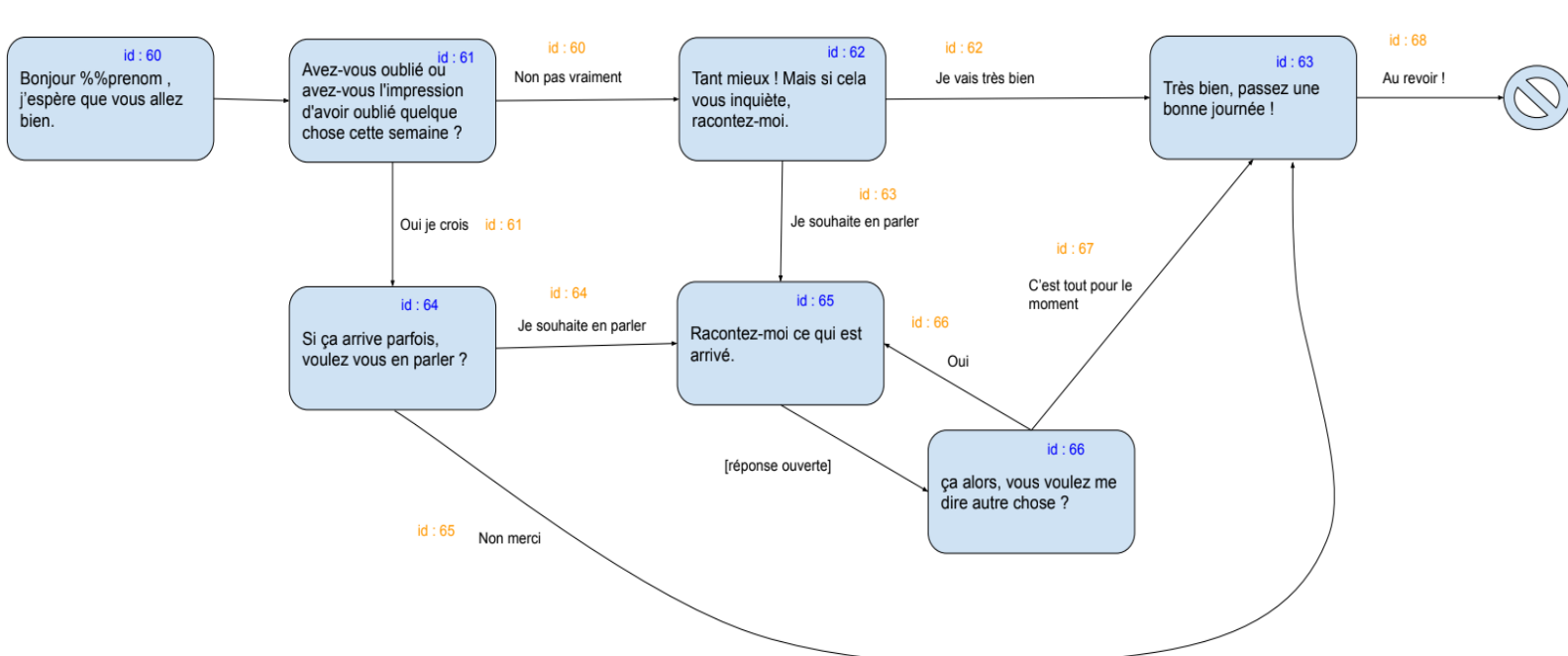
1.2) Scénarios globaux

Une fois le scénario d'introduction terminé, le robot communique avec l'utilisateur en suivant un dialogue scripté à l'avance, dans lequel le robot peut répondre différemment en fonction des choix de l'utilisateur.

Chaque scénario représente une question du jour et le dialogue qui s'ensuit.

Exemple de discussion (scénario 6)

Scénario 6



(Les schémas de tous les scénarios sont disponibles sur l'annexe *Scenarios.pdf*)

Les IDs en bleu correspondent aux IDs des objets Question écrits en base, correspondants aux répliques du robot, et les IDs en jaune aux IDs des objets Reply, correspondants aux réponses fermées de l'utilisateur. Ainsi, il est possible d'écrire un fichier Json pour ajouter ou modifier des scénarios en respectant ces IDs.

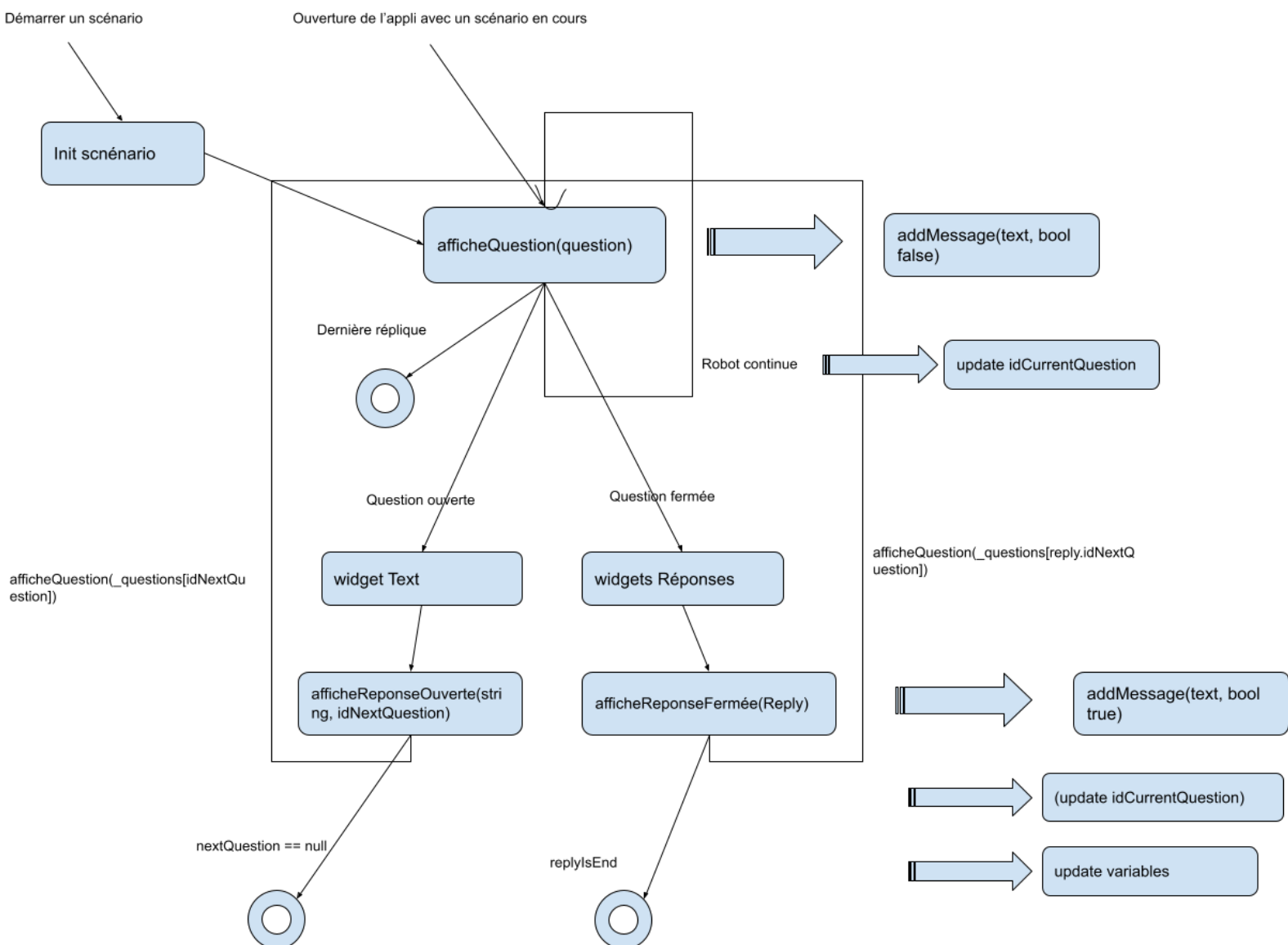
Une question (réplique du robot) peut avoir un, plusieurs ou aucun choix de réponse attendu de la part de l'utilisateur.

1.3) Fonctionnement des scénarios

Dans un scénario, il y a quatre types de questions :

- Les questions ouvertes
- Les questions fermées
- Les questions pour lesquelles le robot se répondra à lui même
- Les questions qui terminent un dialogue

Schéma de l'exécution d'un scénario :



Lorsque le robot pose une question du jour, il dit sa première réplique (fonction `afficheQuestion`). La réplique du robot est écrite en base et affichée à l'écran avec la fonction `addMessage`.

Ici, on vérifie si la fonction est de type 1, 2, 3 ou 4, pour appeler la fonction suivante. Lorsque l'utilisateur répond à une question (ouverte ou fermée), on ajoute aussi son message (avec `addMessage`) et on l'enregistre en base.

Il existe aussi plusieurs types de réponses (ouvertes et fermées) :

- Les réponses classiques
- Les réponses qui terminent un scénario

Si une réponse termine le scénario, alors aucune question suivante ne sera appelée.

Par ailleurs, si l'utilisateur ferme l'application, l'état du scénario actuel sera conservé grâce à la variable `idCurrentQuestion`.

Cette variable est mise à jour à chaque question posée (réplique du robot) et est écrite en base. Elle est passée à null (état par défaut) lorsqu'un scénario se termine. Ainsi, lorsque l'utilisateur démarre l'application, cette variable est requêtée dans la BD pour reprendre à l'état actuel si un scénario est en cours.

Cela fonctionne aussi pour le scénario d'initialisation lorsqu'on ferme l'application avant d'être arrivé au bout.

2) Le journal

Journal page non secrète / secrète

3) Les réglages

L'application possède un menu réglages permettant de modifier plusieurs paramètres de l'utilisateur tel que le nom, le prénom ou la date de naissance. L'utilisateur a aussi la possibilité de modifier la langue de l'application à travers le menu réglages.

Il peut aussi modifier l'heure de notifications afin d'être notifié d'un nouveau scénario chaque jour à l'heure que l'on veut. A partir de ce même menu, un utilisateur de type développeur peut exporter les données de la discussion et des pages de journal non secrètes, ou importer les données d'un fichier Json servant à mettre à jour les scénarios en base.

Pages des réglages de l'application

2:10

Paramètres




Nom
Pierre

Prénom
Jean

Date de naissance (JJ-MM-AAAA)
30-05-2022

Heure des notifications

Langue

Valider

[Importer](#) [Exporter](#)

Discussion Journal Reglages

Comme visible ci-dessus, l'application dispose d'une barre de navigation permettant de naviguer entre les trois pages de l'application : La page de discussion, le journal et les réglages.

IV) Conclusion

test

V) Bibliographie

Documentation pour installer Flutter :

<https://docs.flutter.dev/get-started/install>

Outil pour créer des widget en Flutter :

<https://app.flutterflow.io/>

Chaînes youtube qui nous ont aidé à apprendre à utiliser Flutter :

<https://www.youtube.com/c/JohannesMilke/videos>

<https://www.youtube.com/c/DavidSilveraChannel/videos>

Tutoriel SQFlite sur une application Flutter pour réaliser un CRUD :

<https://youtu.be/ckXSR79AACg>

VI) Annexes

VII) Remerciements

Nous tenons tout d'abord à remercier nos professeurs du master MIASHS parcours Informatique et cognition, en particulier notre tuteur de stage, Mr Damien Pellier de nous avoir offert l'opportunité de réaliser ce projet. Le suivi hebdomadaire et l'aide qu'il nous a fourni tout au long de l'année scolaire nous ont permis d'avancer efficacement tout en gardant un rythme de travail soutenu.

Nous souhaitons également remercier notre encadrante de stage, Mme Lydie Du Bousquet, qui nous a fait confiance pour la réalisation de ce projet et nous a indiqué le bon chemin à suivre pour son développement, en nous donnant toutes les informations nécessaires au bon développement de celui-ci lors de suivis réguliers.

Nous tenons surtout à remercier la région Auvergne-Rhône-Alpes, qui au travers de l'UGA à financer notre déplacement au Japon dans la région du Kansai, lieu de naissance du projet SCUSI Kouno Tori de compagnon virtuel sur smartphone.

De plus, avoir une expérience de stage à l'international est pour nous un atout primordial dans le secteur de l'informatique et sans leur financement et leur aide nous n'aurions pas pu effectuer ce voyage pour aller au bout de ce projet.

A ce titre, nous souhaitons tout particulièrement remercier Mr. Masahide Nakamura, professeur à l'Université de Kobe, de nous accueillir au sein de son établissement afin de poursuivre et de nous aider au développement d'un agent intelligent de compagnon virtuel, qui fait l'objet de certaines de ses recherches.

Nous n'oublions pas aussi toutes les personnes de l'administration des différentes parties, de l'Université Grenoble Alpes et de l'Université de Kobe qui nous ont aidé et conseillé sur de nombreux points.

Enfin, nous remercions toutes les personnes qui se sont portées volontaires pour tester et valider notre application mobile.