

Romain Guillot

Recherche un stage de fin d'étude d'au moins 5 mois pour compléter ma formation d'ingénieur en informatique.

Disponible à partir d'août.

Étudiant ingénieur en 5ème année d'informatique à Polytech Grenoble, j'effectue ma dernière année de formation à l'UQAC, au Québec, en vue de l'obtention d'un double diplôme de maîtrise en informatique québécoise.

Je recherche un stage dans le développement *full stack*. Ce stage doit être d'au minimum 675 heures sur une durée d'environ 5 mois. Je suis disponible à partir d'août.



📞 (+33) 6 50 67 62 89

✉️ romain.guillot.contact@gmail.com

🌐 github.com/Romain-Guillot

Romain Guillot

Recherche un stage de fin d'étude d'au moins 5 mois pour compléter ma formation d'ingénieur en informatique.
Disponible à partir d'août.



(+33) 6 50 67 62 89



romain.guillot.contact@gmail.com



github.com/Romain-Guillot

Formations

Maîtrise en informatique professionnel (en double diplôme), Université du Québec à Chicoutimi (UQAC), Canada	depuis 2019
Génie logiciel, architecture logicielle, sécurité informatique, développement mobile, intelligence artificielle	
Cycle ingénieur en Informatique, Polytech Grenoble, Université Grenoble Alpes, France	depuis 2017
Programmation, génie logiciel, architecture logicielle et matérielle, IHM, base de données, réseaux et protocoles, machine learning, algorithmie	

Compétences

Langages de programmation	Java, Dart, Python, TypeScript, Javascript, HTML/(S)CSS (expériences en C et C#)
Frameworks	Flutter, Android, Node.js, Nest.js (expériences avec Django, Tensorflow, Vue.js et Ionic)
Génie logiciel / outils	Méthodes agiles, CI/CD (Travis et Fastlane), Docker, Contrôle qualité (SonarQube), Design Patterns (GoF), UML, Git
Cloud et base de données	SQL, MongoDB, Firebase
Langues	Français (natif), Anglais (professionnel et technique)

Expériences et projets

Stage - UTP, Malaisie (été 2019)

Stutherapy - Application mobile multiplateformes destinée aux bégues et aux orthophonistes (exercices et suivi de progression)

Flutter/Dart, Firebase (Cloud Firestore, Cloud Storage, Cloud Functions et Authentication), Latex

Projet personnel (automne 2019 - présent)

GeoQuiz - Application mobile multiplateformes de quiz sur la géographie ainsi qu'une application web d'administration

Flutter/Dart, Firebase (Cloud Firestore, Cloud Storage, Crashlytics), SQL, Travis CI, Docker, SonarQube

Projet académique (hiver 2020)

Trare - Application mobile multiplateformes permettant la mise en relation de voyageurs se trouvant à proximité

Flutter/Dart, Firebase (Cloud Firestore, Cloud Storage, Authentication), Travis CI, SonarQube

Projet personnel (en cours de développement)

Kahibalo - Service web REST et application web permettant de gérer des articles de culture générale

Node.js, Nest.js, MongoDB, JWT, Docker

Projet académique (Automne 2019)

Meetchup - Application native Android de partage d'événements (albums photos, organisations, etc.)

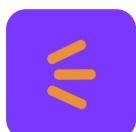
Android, Java, Firebase (Cloud Firestore, Cloud Storage, Authentication)

Projet académique (hiver 2019)

SmartRecruiting - Service web permettant de prédire les formations de l'UGA les plus adaptées face à une offre de stage

Python, Tensorflow / Keras, MySQL, Flask, SonarQube

Mes projets récents (2019 - 2020)



Stage - UTP, Malaisie (été 2019)

Stutherapy

Application mobile multiplateformes destinée aux bégues et aux orthophonistes (exercices et suivi de progression)

Flutter/Dart Firebase (Cloud Firestore, Cloud Storage, Cloud Functions et Authentication) Latex



Projet personnel (automne 2019 - présent)

GeoQuiz

Application mobile multiplateformes de quiz sur la géographie ainsi qu'une application web d'administration

Flutter/Dart Firebase (Cloud Firestore, Cloud Storage, Crashlytics) SQL Travis CI Docker
SonarQube



Projet académique (hiver 2020)

Trare

Application mobile multiplateformes permettant la mise en relation de voyageurs se trouvant à proximité

Flutter/Dart Firebase (Cloud Firestore, Cloud Storage, Authentication) Travis CI SonarQube



Projet personnel (en cours de développement)

Kahibalo

Service web REST et application web permettant de gérer des articles de culture générale

Node.js Nest.js MongoDB JWT Docker



Projet académique (Automne 2019)

Meetchup

Application native Android de partage d'événements (albums photos, organisations, etc.)

Android Java Firebase (Cloud Firestore, Cloud Storage, Authentication)



Projet académique (hiver 2019)

SmartRecruiting

Service web permettant de prédire les formations de l'UGA les plus adaptées face à une offre de stage

Python Tensorflow / Keras MySQL Flask SonarQube



Projet personnel (été 2020)

JSON Portfolio Generator

Générateur de portfolio (site web et documents PDF) à partir d'un fichier JSON

Python Jinja2 Sass



Stuttherapy

Application mobile multiplateformes destinée aux bégues et aux orthophonistes (exercices et suivi de progression)

Stage - UTP, Malaisie (été 2019)

Dans le cadre de ma 4ème année d'études en informatique à Polytech Grenoble, j'ai effectué mon stage de fin d'année à *Universiti Teknologi Petronas* en Malaisie. Durant ce stage de 12 semaines, j'ai été amené à concevoir et à développer une application mobile, **Stuttherapy**.

L'application mobile peut être utilisée par deux types d'utilisateurs :

- les personnes souffrant de bégaiement souhaitant corriger ce trouble de la parole;
- les orthophonistes souhaitant suivre la progression de leurs patients et évaluer leurs exercices.

Github (application mobile) <https://github.com/Romain-Guillot/Stuttherapy>

Gitgub (modélisation et spécifications : diagrammes, SRS, ...)
https://github.com/Romain-Guillot/stuttheary_docs

Rapport de fin de stage (PDF)
https://github.com/Romain-Guillot/stuttherapy_report/blob/master/rapport.pdf

TECHNOLOGIES

Flutter

L'application mobile a été développée avec **Flutter**, le framework open source de développement d'application mutliplateformes créé par Google.

Firebase

Firebase a été utilisé pour gérer les données de l'application. La base de données est gérée avec **Cloud Firestore**, une base de données NoSQL orientée documents. Les enregistrements audios des utilisateurs sont stockés sur **Cloud Storage**. Certains traitements, comme la compression des données, sont automatiquement lancés grâce à **Cloud Functions**, le service FaaS de Firebase. L'authentification des utilisateurs est gérée avec **Firebase Authentication**.

LATEX

Le rapport de fin de stage a été rédigé avec **Latex**.

SCREENSHOTS

The screenshots demonstrate the following features:

- Home Screen:** Shows the app's logo and a message: "Stuttherapy is an application to help you overcome your stuttering through different exercises. Train freely, without pressure!". It includes buttons for "Stutterer" (green), "Therapist" (white), and "Submit!" (orange).
- Exercises List:** A list of exercises: Metronome, Reading, and Mirroring. Each exercise has "PROGRESS" and "TRAINING" buttons.
- Metronome Settings:** Settings for Metronome training, including BPM (set to 100), recording enablement, and audio signal options. A "Launch" button is at the bottom.
- Metronome Progress:** A line chart showing pronunciation accuracy (%) over time from July 5, 2019, to August 4, 2019. The accuracy fluctuates between 50% and 100%.
- My Feed:** A feed of activity logs for the user, including "Metronome" entries on various dates.
- My Patients:** A list of patients managed by the therapist, with their names and unique IDs.
- Mirroring:** A screen for the "Mirroring" exercise, showing a feedback message "good!" and a "Save feedback" button.
- Saved words:** A list of saved words: always, blogs, coin, playground, vocabulary, and when, each with a delete icon.
- Manuals:** A menu for the therapist containing links to exercises, progress, saved words, cloud synchronization, and sharing progress.



GeoQuiz

Application mobile multiplateformes de quiz sur la géographie ainsi qu'une application web d'administration

Projet personnel (automne 2019 - présent)

Le but de **GeoQuiz** est de proposer une application mobile de quiz sur la géographie. Pour cela, deux applications ont été réalisées :

- **l'application cliente** Android et iOS permettant de jouer aux différents quiz avec une difficulté croissante au fur et à mesure des parties;
- **l'application web** permettant de gérer la base de données des questions et des langues supportées.

Quatre thèmes sont actuellement proposés avec des questions pouvant prendre différentes formes : avec du texte, avec des images ou avec des marqueurs sur une carte. La progression de l'utilisateur est sauvegardée pour lui permettre de connaître son niveau et de se comparer aux autres joueurs (le *leaderboard* est en cours de développement).



Github (application mobile) <https://github.com/Romain-Guillot/GeoQuiz-ClientApp>



Github (application web d'administration) <https://github.com/Romain-Guillot/GeoQuiz-AdminApp>



Play Store <https://play.google.com/store/apps/details?id=com.develob.geoquiz>

TECHNOLOGIES



L'application mobile et l'application web sont développées avec **Flutter**, le framework open source de développement d'application mutliplateformes créé par Google.



Firebase a été utilisé pour gérer les données de l'application et pour collecter les rapports de bugs. La base données de question et des données utilisateurs est gérée avec **Cloud Firestore**, une base de données NoSQL orientée documents. Les fichiers sont stockées sur **Cloud Storage** (photos de drapeaux ou de monuments par exemple). Les rapports de bugs sont collectés et analysés avec **Crashlytics**.



La base de données de questions est sauvegardée localement sur l'appareil de l'utilisateur avec **SQLite**. Leurs progressions personnelles sont également stockées avec SQLite.



Les pipelines de CI/CD sont orchestrées avec **Travis CI**, en particulier pour les tests, la génération automatique de *screenshots* et la publication sur le Play Store.

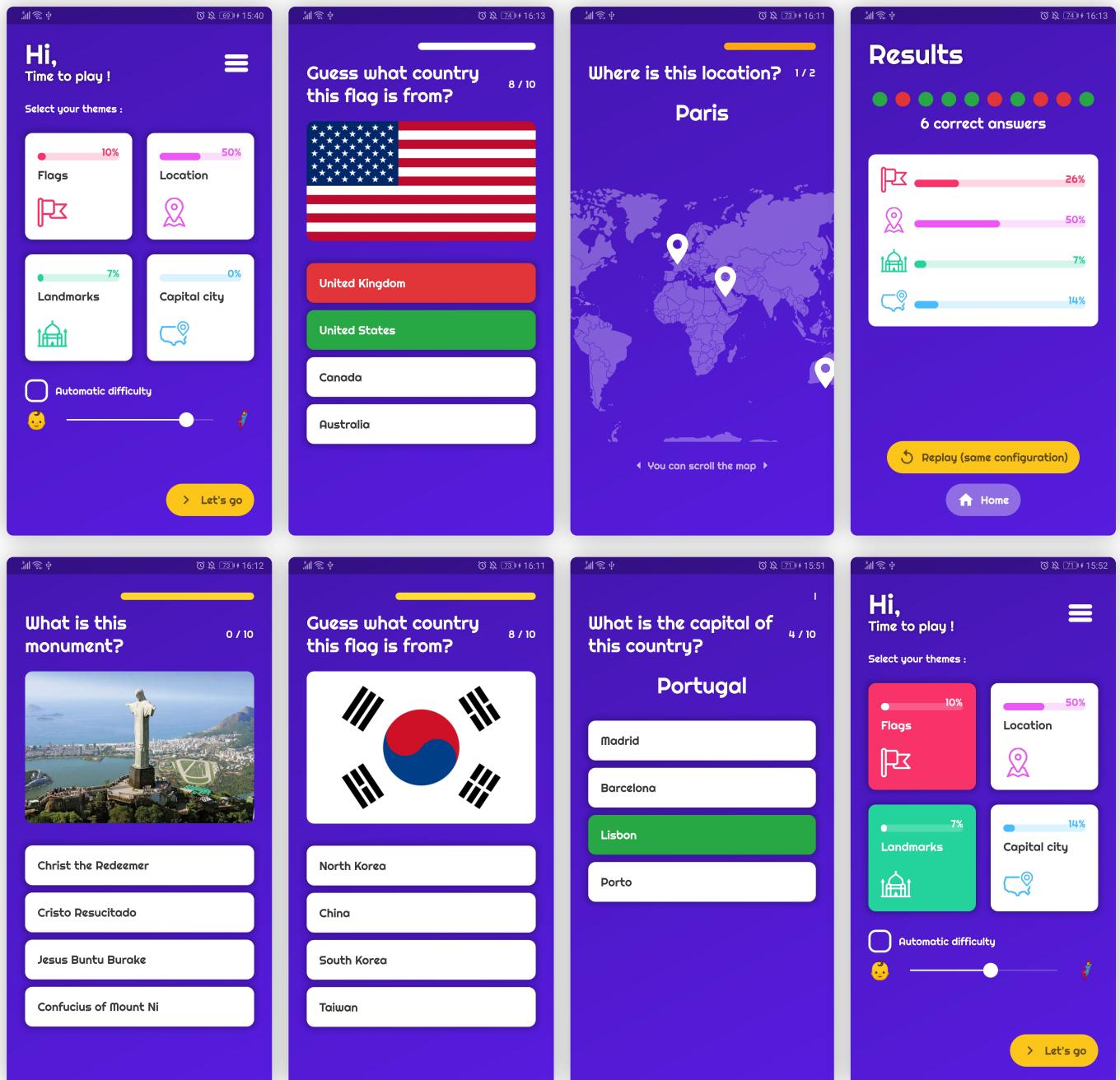


Un environnement de développement a été créé avec une image **Docker** contenant toutes les dépendances utilisées pour le développement des deux applications. L'environnement de déploiement pour l'application web est aussi géré avec **Docker**.



La qualité du code des applications est analysée avec **SonarQube**.

SCREENSHOTS



GeoQuiz Admin

Database Current version : 33 Publish database

SUPPORTED LANGUAGES + Add new supported language
fr

THEMES

Title	Color	Entitled	-	?	+	Add
Flags	4294456170	Guess what country this flag is from?	-	1	+	
Location	4293613812	Where is this location?	-	2	+	
Landmarks	4280603547	What is this monument?	-	3	+	
Capital city	4283022589	What is the capital of this country?	-	4	+	

QUESTIONS (58)

Correct answer Wrong answer #1 Wrong answer #2



Trare

Application mobile multiplateformes permettant la mise en relation de voyageurs se trouvant à proximité

Projet académique (hiver 2020)

Trare est une application mobile permettant de mettre en relation les voyageurs d'une même destination afin de leurs permettre d'organiser des activités à faire ensemble (faire une randonnée, aller boire un verre, faire une soirée jeux de société, etc.)

L'application possède les fonctionnalités suivantes :

- Authentification avec les réseaux sociaux (Facebook, Google);
- Visualisation et personnalisation du profil utilisateur;
- Découverte des activités se déroulant autour de la position géographique de l'utilisateur;
- Visualisation, création, modification et suppression d'une activité;
- Visualisation des profils des voyageurs;
- Participation à une activité (demandes de participants, acceptations de requêtes, système de messagerie, etc.).

 **Github** <https://github.com/Romain-Guillot/Trare>

 **Play Store (version alpha)** <https://play.google.com/apps/testing/com.trare.app>

TECHNOLOGIES

 **Flutter**

L'application mobile a été développée avec **Flutter**, le framework open source de développement d'application multiplateformes créé par Google.

 **Firebase**

Firebase a été utilisé pour gérer les données de l'application. La base de données est gérée avec **Cloud Firestore**, une base de données NoSQL orientée documents. Les fichiers — comme les photos de profil des utilisateurs — sont stockés sur **Cloud Storage**. L'authentification des utilisateurs est gérée avec **Firebase Authentication**.

 **Travis CI**

Les pipelines de CI/CD sont orchestrées avec **Travis CI**, en particulier pour les tests, la génération automatique de *screenshots* et la publication sur le Play Store.

 **sonarQube**

La qualité du code de l'application est analysée avec SonarQube

SCREENSHOTS

The image displays a grid of 10 screenshots from a travel application, arranged in two rows of five. The screenshots illustrate various features of the app:

- Screenshot 1:** Welcome screen with a globe icon, a 'Welcome' message, and a note about sharing activities with travellers. It includes social login options for Google, Facebook, and Email, and a 'Continue with Email' button.
- Screenshot 2:** Explore screen showing proposed activities. One activity listed is 'Soirée Baruqac' in Chicoutimi, Saguenay, Canada, between June 13 and 14, 2020.
- Screenshot 3:** Chat screen for 'Soirée Baruqac'. It shows a message from Robert (français) and a placeholder for a map of Chicoutimi, Saguenay, Canada.
- Screenshot 4:** Chat input field showing a keyboard with French and English language switching options.
- Screenshot 5:** Activity details for 'Randonnée Monts-Valin' in Saguenay, Canada, between April 1 and April 1, 2020. It shows a profile picture of the host, Robert, and a 'Requests' section.
- Screenshot 6:** Profile editing screen for Robert, showing his photo, a placeholder for an add photo button, and a take photo button.
- Screenshot 7:** Chats screen showing a list of recent conversations, including 'Sortie club', 'Soirée loup garou', 'Randonnée Mont Fuji', 'Randonnée Monts-Valin', and 'Soirée Baruqac'.
- Screenshot 8:** Create activity screen. A user has entered 'Soirée jeux' as the title and 'Required' as the description. The 'Add the activity' button is visible at the top right.
- Screenshot 9:** Activity details for 'Soirée loup garou' in Lac-Pikauba, Canada, between June 10 and June 11, 2020. It includes a note about validation of participation requests.



Kahibalo

Service web REST et application web permettant de gérer des articles de culture générale

Projet personnel (en cours de développement)

Le but de **Kahibalo** est de fournir une application web permettant de lire et de rédiger des articles de culture générale. Pour ce faire, le projet est constitué de 2 services :

- **Le service web REST** permettant d'accéder et de gérer les ressources via des représentations JSON;
- **L'application web** s'interfaisant avec le service web afin d'accéder aux différentes fonctionnalités via un navigateur web.

Voici les fonctionnalités implémentées ou en cours de développement :

- Visualisation des articles disponibles avec la possibilité de les filtrer par *tags*;
- Visualisation, création, modification et suppression des articles;
- Authentification permettant l'autorisation de la création, la modification et la suppression des ressources;
- Redirection vers la visualisation d'un article aléatoire.

 **Github (service web)** <https://github.com/Romain-Guillot/GeoQuiz-ClientApp>

 **Github (application web)** <https://github.com/Romain-Guillot/GeoQuiz-AdminApp>

TECHNOLOGIES



Les deux services sont développés avec **Node.js**



Afin de rapidement développer une architecture de haute qualité, scalable et évolutive, le framework **Nest.js** est utilisé par dessus Node.js pour développer les deux services



Les données de l'application sont gérées avec **MongoDB**, un système de gestion de base de données NoSQL orienté documents



L'autorisation d'accès aux ressources est gérée avec le système de token **JWT**



Le déploiement des services sur mon serveur est géré avec **Docker**

SCREENSHOTS

HOME	RANDOM ENTRY	ABOUT	Dan
Tags			Entries
● Histoire			Attaque de Pearl Harbor
● Musique			Les trois mousquetaires
● Personnage			Alexandre le Grand
● Géographie			The Beatles
			George Washington
			Queen
			Maria Sharapova
			The Doors
			Michael Jordan
			Vatican
			Mickael Jakson

HOME RANDOM ENTRY ABOUT Light

Attaque de Pearl Harbor

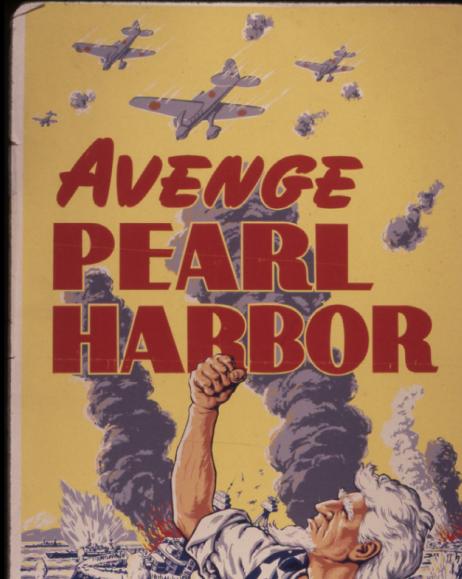
Attaque surprise menée par les Forces aéronavales japonaises le 7 décembre 1941 contre la base navale américaine de Pearl Harbor située sur l'île d'Oahu, dans le territoire américain d'Hawaï. Autorisée par l'empereur du Japon Hirohito, elle vise à détruire la Flotte du Pacifique de l'United States Navy. Cette attaque provoque l'entrée des États-Unis dans le conflit mondial.

L'attaque, dirigée par le général Hideki Tōjō, est lancée le dimanche 7 décembre par le Service aérien de la Marine impériale japonaise contre la flotte américaine du Pacifique et les forces stationnées sur place. Elle est conduite en deux vagues aériennes parties de six porte-avions impliquant plus de 400 avions. En moins de vingt-quatre heures, l'empire du Japon attaque également les États-Unis aux Philippines et engage les hostilités avec le Royaume-Uni, en envahissant Hong Kong et en débarquant en Malaisie.



0 à 400 m	400 à 800 m	800 à 1 200 m	1 200 à 1 600 m	1 600 à 2 000 m
Avions abattus	Avions abattus	Avions abattus	Avions abattus	Avions abattus
Sur le pont de l'USS "Hornet"	Sur le pont de l'USS "Enterprise"	Sur le pont de l'USS "Yorktown"	Sur le pont de l'USS "Lexington"	Sur le pont de l'USS "Midway"

Les pertes américaines sont importantes : 2 403 morts et 1 178 blessés. Seulement deux cuirassés sont détruits ainsi que 188 avions. Les seize autres navires endommagés sont remis en état dans les mois qui suivent. Les Japonais perdent 64 hommes, 29 avions et cinq sous-marins de poche ; un marin est capturé.





Meetchup

Application native Android de partage d'événements (albums photos, organisations, etc.)

Projet académique (Automne 2019)

Le but de **Meetchup** est de permettre aux participants d'un même événement de s'échanger des messages, des photos et d'organiser l'événement. Voici les fonctionnalités développées :

- Authentification via Google ou via email/mot de passe;
- Visualisation, création, modification et suppression des comptes utilisateurs;
- Visualisation, création, modification et suppression des événements;
- Fil d'actualité d'un événement avec les dernières publications;
- Albums photos de l'événement;
- Gestion des participants d'un événement à base de clés d'invitations.

 **Github** <https://github.com/Romain-Guillot/Meetchup>

TECHNOLOGIES

 **android**

L'application a été développée avec **Android**

 **Java**

Le langage utilisé pour développer des applications Android en utilisant l'API fournie est le **Java**

 **Firebase**

Firebase a été utilisé pour gérer les données de l'application. La base de données est gérée avec **Cloud Firestore**, une base de données NoSQL orientée documents. Les photos partagées par les participants sont stockées sur **Cloud Storage**. L'authentification des utilisateurs est gérée avec **Firebase Authentication**.

SCREENSHOTS

The image displays eight screenshots of the Meet'chup mobile application interface.

- Login Screen:** Shows the "Meet'chup" logo at the top. Below it, there are two authentication options: "Social authentication" (with a "Continue with Google" button) and "Classic authentication" (with fields for "Email" and "Password"). A "SIGN IN" button is at the bottom.
- Home Screen:** Displays a feed of events. The first event is "Les français a Langawi!" (9 déc. 2019, 4 days, Malaisie). The second event is "Un autre événement !!" (19 nov. 2019, 1 week, 5 days).
- Event Detail Screen:** Shows a photo of a boat on a riverbank. Below the photo, the caption reads "La petite île bien sympathique" with a USA flag emoji.
- Event Detail Screen:** Shows a collage of photos from the event, including food, landscapes, and people. There are three circular buttons at the bottom: "Feed", "Album", and "ToDo".
- Event Key Screen:** Shows the "Invitation key" screen. It displays the invitation key "Malaisie2019" and two buttons: "Update invitation key" and "Disable".
- Event Detail Screen:** Shows the "Meet'chup" logo at the top. Below it, there are two events: "Les français a Langawi!" and "Un autre événement !!". The "Join" button is visible next to the second event.
- Profile Screen:** Shows the user's profile picture, name "Romain", and location "Malaisie". It includes a "Sign Out" button.
- Edit Profile Screen:** Allows users to update their email, name, password, and delete their account. It includes "Update" and "Delete your account" buttons.
- Create Event Screen:** A form for creating a new event. Fields include "Title", "Description", "Begin date", "End date", and "Location". A "Create" button with a checkmark icon is at the bottom.



SmartRecruiting

Service web permettant de prédire les formations de l'UGA les plus adaptées face à une offre de stage

Projet académique (hiver 2019)

SmartRecruiting est un projet soutenu par Disrupt Campus. Le projet consiste à associer automatiquement des filières de formation de l'Université Grenoble Alpes à partir d'offres de stage. Le but étant de simplifier la connexion entre les 5 acteurs impliqués dans les recherches de stage :

- Les étudiants recherchant des stages;
- Les gestionnaires de scolarité;
- Les universités;
- Les entreprises proposant des offres de stage.

Avec 2 de mes camarades, notre but sur ce projet a été de redévelopper la partie *backend* du site afin de proposer une architecture plus qualitative, plus évolutive et plus maintenable. Nous avions aussi pour responsabilité de réécrire le module d'apprentissage et de prédiction en utilisant le *machine learning* afin d'obtenir des modèles d'apprentissage plus performants.

Github (backend) <https://gricad-gitlab.univ-grenoble-alpes.fr/Projets-INFO4/18-19/21/backend>

Documentation (backend) <https://smartrecruiting-back.readthedocs.io/en/latest/>

Rapport <https://gricad-gitlab.univ-grenoble-alpes.fr/Projets-INFO4/18-19/21/docs/-/blob/master/report.md>

TECHNOLOGIES

python™ La partie *backend* du projet, chargée de gérer les données et de proposer un service de prédiction, a été réalisée avec **Python 3**

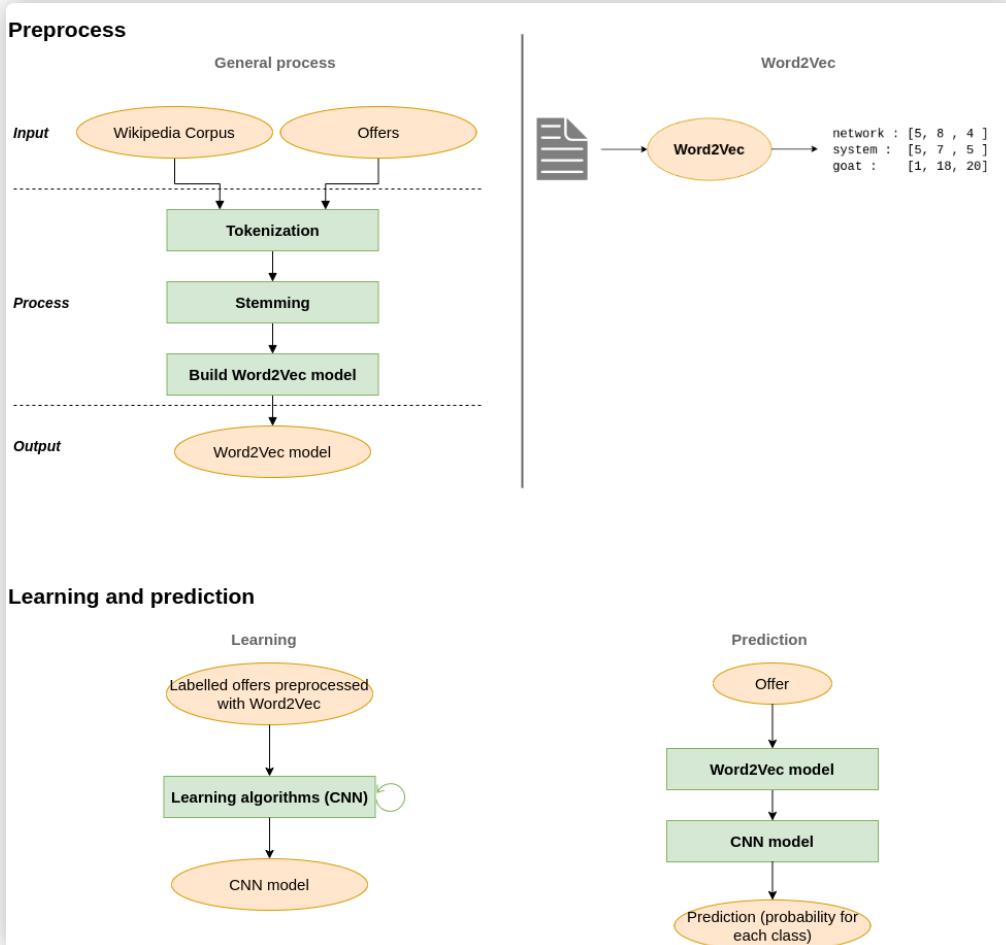
TensorFlow Le service d'apprentissage et de prédiction a été développé avec **Tensorflow**, un écosystème permettant d'utiliser le *machine learning*. **Keras**, une librairie s'interfaisant avec Tensorflow, a été utilisé afin d'utiliser des modèles de *deep learning* : les CNN (réseau de neurones convolutifs)

MySQL Les données de l'application ont été stockées avec **MySQL**, un serveur de base de données SQL

Flask L'API web du projet a été réalisée avec **Flask**

sonarqube La qualité du code de l'application a été analysée avec SonarQube

SCREENSHOTS





JSON Portfolio Generator

Générateur de portfolio (site web et documents PDF) à partir d'un fichier JSON

Projet personnel (été 2020)

Projet Python permettant de générer un portfolio à partir d'un fichier de données JSON. Le portfolio généré contient les éléments suivants :

- Une page de présentation;
- Une page contenant le CV;
- Une page listant les projets;
- Une page descriptive de chaque projet.

Le portfolio est généré dans deux formats différents :

- **Site web**, des pages HTML connectées entre elles permettant la navigation dans le portfolio via un navigateur web;
- **Document PDF**, un document PDF est généré à partir des pages HTML avec un style adapté. Un document contenant uniquement le CV est aussi généré.

Enfin, le projet permet de publier le portfolio automatiquement sur une *remote git* — typiquement pour la publication automatique et instantanée sur un repository *Github Page* pour rendre disponible la version web du portfolio disponible sur le web directement après la génération de celui-ci.

 **Github** <https://github.com/Romain-Guillot/JSON-Portfolio-Generator>

 **Mon portfolio généré avec cet outil** <https://romain-guillot.github.io/>

TECHNOLOGIES

 **python™**

Le cœur du projet a été développé avec **Python3**

 **Jinja**

Les templates HTML ont été écrits avec **Jinja2**, un langage de template inspiré de celui de Django.

 **Sass**

Les feuilles de style ont été écrites avec **Sass**, un préprocesseur CSS (langage de description de style compilant en CSS)

SCREENSHOTS

Romain Guillot

À propos CV Mes projets

Recherche un stage de fin d'étude d'au moins 5 mois pour compléter ma formation d'ingénieur en informatique.

Disponible à partir d'août.

Étudiant ingénieur en 5ème année d'informatique à Polytech Grenoble, j'effectue ma dernière année de formation à l'UQAC, au Québec, en vue de l'obtention d'un double diplôme de maîtrise en informatique québécoise.

Je recherche un stage dans le développement *full stack*. Ce stage doit être d'au minimum 675 heures sur une durée d'environ 5 mois. Je suis disponible à partir d'août.

📞 (+33) 6 50 67 62 89
✉️ roman.guillot.contact@gmail.com
🌐 github.com/Romain-Guillot

⬇️ Télécharger mon portfolio (PDF)
⬇️ Télécharger uniquement mon CV (PDF)

Généré et publié grâce à [JSON Portfolio Generator](#), outils développés avec Python, Jinja2 et Sass

Romain Guillot

À propos CV Mes projets

⬇️ PDF

Recherche un stage de fin d'étude d'au moins 5 mois pour compléter ma formation d'ingénieur en informatique. Disponible à partir d'août.

📞 (+33) 6 50 67 62 89
✉️ roman.guillot.contact@gmail.com
🌐 github.com/Romain-Guillot

Formations

Maitrise en informatique professionnel (double diplôme), Université du Québec à Chicoutimi (UQAC), Canada depuis 2019
Génie logiciel, architecture, sécurité informatique, développement mobile, intelligence artificielle

Cycle ingénieur en informatique, Polytech Grenoble, Université Grenoble Alpes, France depuis 2017
Programmation, génie logiciel, architecture, IHM, base de données, réseaux, machine learning, algorithmie

Compétences

Langages de programmation Java, Dart, Python, TypeScript, Javascript, HTML/S/CSS (expériences en C et C#)
Frameworks Flutter, Android, Node.js, Nest.js (expériences avec Django, Tensorflow, Vue.js et Ionic)
Génie logiciel et utilitaires Agile, CI/CD (Travis et Fastlane), Docker, Contrôle Qualité (SonarQube), Design Patterns (GoF), UML, Git
Cloud et base de données SQL, MongoDB, Firebase
Langues Français (natif), Anglais (professionnel, technique)

Expériences et projets

Stage - UTP Malaysia (été 2019)
Stutherford - Application mobile multiplateformes pour les bégues et les orthophonistes (exercices et suivi de progression)
Flutter/Dart, Firebase (Cloud Firestore, Cloud Storage, Cloud Functions et Authentication), Latex

Projet personnel (automne 2019 - présent)
GeoQuiz - Application mobile multiplateformes de quiz sur la géographie accompagné d'une application web d'administration
Flutter/Dart, Firebase (Cloud Firestore, Cloud Storage, Crashlytics), SQL, Travis CI, Docker, SonarQube

Projet académique (hiver 2020)

Romain Guillot

À propos CV Mes projets



Stutherford

Application mobile multiplateformes pour les bégues et les orthophonistes (exercices et suivi de progression)

Dans le cadre de ma 4ème année d'études en informatique à Polytech Grenoble, j'ai effectué mon stage de fin d'année à Universiti Teknologi Petronas en Malaisie. Durant ce stage de 12 semaines, j'ai été amené à concevoir et à développer une application mobile.

L'application mobile peut être utilisé par deux types d'utilisateur:

- les personnes souffrant de bégaiement souhaitant corriger ce trouble de la parole;
- les orthophonistes souhaitant suivre la progression de leurs patients et évaluer leurs exercices.

🌐 Github (Application mobile) <https://github.com/Romain-Guillot/Stutherford>
🌐 Github (Modélisation et spécifications : diagrammes, SRS, ...) https://github.com/Romain-Guillot/stutherford_docs
📄 Rapport de fin de stage (PDF) https://github.com/Romain-Guillot/stutherford_report/blob/master/rapport.pdf

TECHNOLOGIES

Flutter L'application mobile a été développée avec Flutter, le framework open source de développement d'application multiplateformes créé par Google.

Firebase a été utilisé pour stocker les données de l'application. La base de données est gérée avec Cloud Firestore, une base de données NoSQL qui enregistre les données des utilisateurs sont stockées sur Cloud Storage. Certains traitements comme la compression des données sont automatiquement lancés grâce à Cloud Functions, le service FaaS de Firebase.

Latex L'authentification des utilisateurs est gérée avec Firebase Authentication.

Le rapport de fin de stage a été rédigé avec Latex.

SCREENSHOTS

