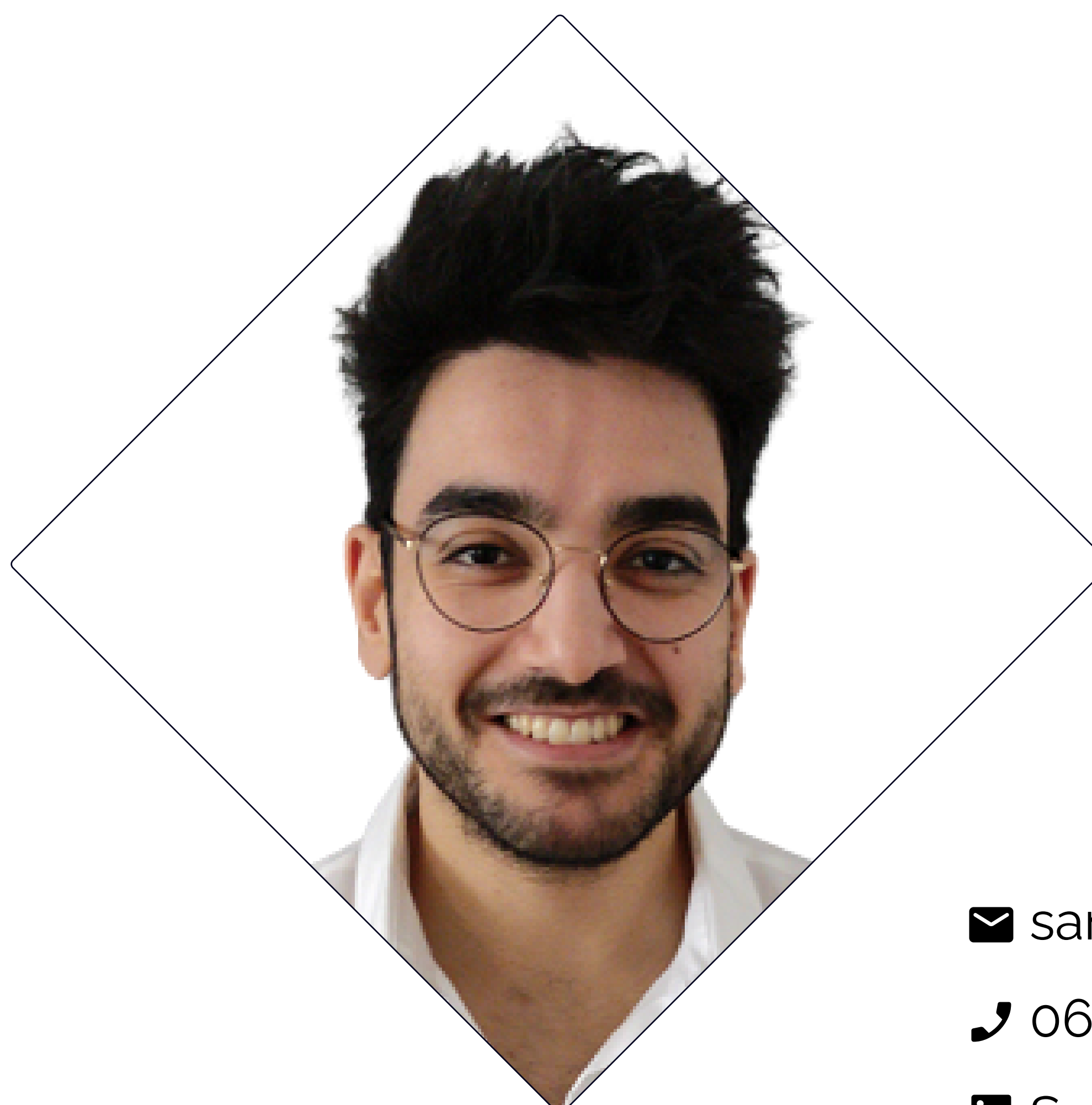


Samir Ghabi

Développeur Fullstack



✉ samir.ghabi@gmail.com

☎ 06 19 91 01 51

🌐 Samir-Ghabi 🌐 Miouss

📍 Avignon

Profil

Développeur web **autodictate** depuis 1 an, je me concentre aujourd'hui sur **Typescript, React et Node**.

J'ai mené à bien 3 projets en **totale autonomie** qui ont été **réalisés from scratch**.

J'ai pour ambition à présent de mettre mes **compétences en application** à plus grande échelle dans le monde professionnel.

Education

BAC +2. **Informatique**
2016, CERI, Université d'Avignon

BAC Général **Scientifique**
2014, Lycée Philippe de Girard

Technologies

Frontend

Typescript, React, Redux, CSS, Electron

Backend

NodeJs, Express, MySql

Ingénierie logicielle

Gestion de version (git), Tests (Jest)

Langues

Anglais | Niveau B1

Français | Native

Projets Personnels

Portfolio W10 - Typescript, React, Redux

Site portfolio qui reproduit l'interface et l'interactivité du système d'exploitation Windows 10

- **Conception d'une librairie** de composant React imitant le design de Windows 10
- Création d'un **store Redux** afin de gérer le système d'application fenêtré de Windows
- Intégration d'une **Google Cloud function** pour gérer l'envoi de mail

🌐 github.com/Miouss/portfolio

🌐 samir-ghabi.com

Lol Mood - Javascript, React, PHP, MySQL

Site affichant diverses statistiques d'un joueur ou champion de League of Legends selon les récentes parties jouées

- **Design du schéma de la base de données** afin de stocker les données des parties classées.
- **Intégration de l'API** de l'éditeur Riot Games pour récupérer puis stocker les données publiques
- **Implémentation d'une API** permettant de récupérer les données des joueurs/champions
- **Collaboration** avec un designer afin de produire une **UI adaptée** au public cible

🌐 github.com/Miouss/lolmood

🌐 lolmood.net

Smart DL - Typescript, Node, Electron, Jest

Application desktop multi-plateformes permettant de télécharger légalement les flux vidéos de certains services de streaming avec leur métadonnées

- **Étude de l'API** du service de streaming afin de connaître les divers endpoints pour automatiser la récupération du flux
- **Implémentation d'un système de scraping** permettant de télécharger les fragments vidéos/audios et de les réassembler pour obtenir le média
- Développement d'un **système de communication inter-processus** afin de pouvoir suivre les étapes de traitement en tant réel

🌐 github.com/Miouss/wwe-dl

Hobbies

