

Sami Bentaiba

samibentaiba25@gmail.com · +213 656 73 98 96

<https://bentaidev.vercel.app>

<https://github.com/samibentaiba>

<https://www.linkedin.com/in/samibentaiba>

Résumé

Ingénieur Logiciel

Je suis un ingénieur logiciel ayant évolué d'un étudiant en informatique à un développeur full-stack, puis vers un architecte logiciel et chef d'équipe. Mon parcours m'a conduit à travers toutes les étapes de la création logicielle — de l'apprentissage des fondamentaux de l'informatique, des algorithmes et de la conception des systèmes, à la maîtrise de plusieurs langages de programmation et à la réalisation de solutions logicielles complètes. En tant qu'architecte, j'ai appris à concevoir et à mettre en œuvre des modèles et des plans systèmes évolutifs, et en tant qu'opérateur, j'ai développé des workflows et des stratégies de maintenance pour assurer la qualité et la durabilité des logiciels. Grâce à cette évolution continue, je suis devenu un ingénieur logiciel capable de planifier, créer et améliorer des systèmes complets — reliant le développement, l'architecture et l'excellence opérationnelle.

Compétences

Opérations de Développement (DevOps)

Manipulation de Pipelines (Appliqué): Utilisation de ngrok pour exposer des serveurs locaux et manipuler des pipelines pour les tests de webhooks. Utilisé efficacement dans le projet AiHorizons.

Microservices (Appliqué): Conception d'applications comme une collection de services faiblement couplés. J'ai exploré cette architecture dans le projet H2-SpringBoot.

Git & GitHub (3+ ans): Git est un système de contrôle de version distribué. J'utilise Git et GitHub quotidiennement pour le contrôle de version, la collaboration et la gestion des changements de code à travers les projets, y compris l'utilisation de GitHub Actions pour le CI/CD.

GitHub Actions (1+ an): Automatisation des flux de travail logiciels avec des pipelines CI/CD directement dans GitHub. Je l'utilise pour les tests et le déploiement automatisés.

CI/CD (Appliqué): Pratiques d'Intégration Continue et de Déploiement Continu pour automatiser le processus de livraison de logiciels.

Docker (1+ an): Docker est une plateforme pour développer, expédier et exécuter des applications dans des conteneurs. Utilisé pour créer des environnements de développement cohérents.

Vercel (2+ ans): Vercel est une plateforme de déploiement et de collaboration pour les développeurs frontend. J'utilise Vercel pour déployer et héberger des applications web, en tirant parti de ses fonctions sans serveur et de son réseau edge.

Écosystème Linux (Maîtrisé): Maîtrise approfondie des distributions comme Arch, Ubuntu, Fedora, Debian, Kali, Mint et Nix. Compétent en scripting shell et configuration système.

Systèmes Unix (Familier): Expérience avec macOS et OpenBSD, tirant parti de leur stabilité et de la philosophie Unix pour des environnements de développement robustes.

Windows (Compétent): Utilisé principalement pour la collaboration design (Figma) et les tests de compatibilité, gérant les limitations de développement avec des configurations personnalisées.

Développement Frontend

React (2+ ans): React est une bibliothèque JavaScript pour la création d'interfaces utilisateur. J'ai utilisé React de manière extensive pour créer des applications à page unique et des composants au sein de projets Next.js, en me concentrant sur l'architecture basée sur les composants et la gestion d'état avec les hooks et l'API de contexte.

Next.js (2+ ans): Next.js est un framework React qui permet le rendu côté serveur et la génération de sites statiques. J'ai utilisé Next.js pour créer des applications web performantes et optimisées pour le référencement, en tirant parti de son système de routage basé sur les fichiers et de ses routes API dans des projets comme mon portfolio, ITC Hub et Algis.

TypeScript (2+ ans): TypeScript est un langage de programmation fortement typé qui s'appuie sur JavaScript. J'utilise TypeScript dans la plupart de mes projets (portfolio, c-studio, algis, itc-hub) pour assurer la sécurité des types et améliorer l'expérience de développement.

Tailwind CSS (2+ ans): Tailwind CSS est un framework CSS utilitaire. J'utilise Tailwind CSS pour créer rapidement des designs personnalisés sans quitter le HTML, en me concentrant sur la conception responsive et la cohérence des composants dans mes projets Next.js.

shadcn/ui (1+ an): Une collection de composants réutilisables construits avec Radix UI et Tailwind CSS. Je l'utilise pour créer des interfaces utilisateur accessibles et personnalisables dans mes projets Next.js.

HTML & CSS (3+ ans): Technologies web de base pour structurer et styliser le contenu web. Utilisées dans tous les projets web.

Vite (1+ an): Vite est un outil de construction qui vise à fournir une expérience de développement plus rapide et plus légère pour les projets web modernes. Je l'utilise pour mes projets React.

Angular (<1 an): Angular est une plateforme pour la création d'applications web mobiles et de bureau. J'ai de l'expérience avec grâce au projet H2-SpringBoot.

Electron (1 an): Electron est un framework pour créer des applications natives avec des technologies web comme JavaScript, HTML et CSS. Utilisé pour créer l'IDE C-Studio.

Développement Backend

Node.js (2+ ans): Node.js est un environnement d'exécution JavaScript construit sur le moteur JavaScript V8 de Chrome. J'ai utilisé Node.js pour la logique côté serveur dans les applications Next.js et pour les scripts, y compris au sein du projet Electron c-studio.

Express (1+ an): Express est un framework d'application web Node.js minimal et flexible. Utilisé pour créer des API RESTful.

PostgreSQL (1+ an): PostgreSQL est une base de données relationnelle avancée et open-source. J'ai utilisé PostgreSQL avec Prisma pour le stockage de données dans des projets comme ITC Hub et Algis.

Prisma (1+ an): Prisma est un ORM de nouvelle génération pour Node.js et TypeScript. J'utilise Prisma pour interagir avec les bases de données (PostgreSQL et SQLite) dans des projets comme ITC Hub et Algis.

SQLite (1+ an): SQLite est une bibliothèque en langage C qui implémente un moteur de base de données SQL petit, rapide, autonome, de haute fiabilité et complet. Utilisé pour le développement local dans le projet Algis.

Java (1+ an): Java est un langage de programmation orienté objet de haut niveau. Je l'utilise pour le développement backend avec Spring Boot.

Spring Boot (1+ an): Spring Boot est un framework open source basé sur Java utilisé pour créer des microservices. Je l'utilise pour construire des services backend robustes et évolutifs.

Next.js (2+ ans): J'utilise Next.js pour le développement full-stack, en utilisant les routes API et les Server Actions pour construire une logique backend sécurisée et évolutive directement dans l'application.

NextAuth.js (1+ an): Une solution d'authentification open-source complète pour les applications Next.js. Je l'utilise pour gérer l'authentification sécurisée des utilisateurs et la gestion des sessions.

Langages de Programmation

TypeScript (Familier): Mon langage principal pour le développement web, assurant la sécurité des types et la qualité du code.

JavaScript (Familier): Compréhension approfondie des fonctionnalités ES6+ et des concepts fondamentaux du langage.

C (Familier): Je suis très familier avec C, l'utilisant pour la programmation système et la compréhension des concepts de bas niveau. C'est le langage principal supporté par mon IDE C-Studio.

Bash (Familier): J'utilise les scripts Bash pour l'automatisation et la gestion des environnements de développement.

Rust (<1 an): Rust est un langage axé sur la performance et la sécurité. Exploré à travers le projet iShowOff et intéressé par son potentiel pour les solutions d'entreprise.

Python (Exploré): Exploré Python pour les tâches de script et de traitement de données.

Go (Exploré): Exploré Go pour son modèle de concurrence et ses performances dans les systèmes backend.

C++ (Exploré): Exploré C++ pour comprendre la programmation orientée objet au niveau système.

C# (Exploré): Exploré C# pour les concepts de développement de bureau et de jeux.

Haskell (Exploré): Exploré Haskell pour comprendre les paradigmes de programmation fonctionnelle.

IDEs & Éditeurs (Maîtrisé): Maîtrise de Neovim, VS Code et Nano avec des raccourcis personnalisés, réglage LSP et profils. Expérimenté avec JetBrains, Emacs et CodeBlocks. Je maintiens une config Neovim personnalisée (github.com/samibentaiba/nvim).

Architecture & Documentation

Figma (Collaboratif): Appris grâce à la collaboration avec des designers UI/UX pour implémenter des designs pixel-perfect et comprendre les systèmes de design.

Architecture Système (Appliqué): Conception de systèmes évolutifs, de microservices et de schémas de base de données. J'utilise des outils comme Excalidraw et Figma pour visualiser les flux de code, les modèles de conception et les décisions architecturales.

Documentation Technique (Appliqué): Création d'une documentation complète pour le code, les API et les conceptions système à l'aide d'Obsidian. Je me concentre sur l'explication du 'pourquoi' derrière les décisions techniques pour faciliter la maintenance et l'intégration.

Analyse de Code Source (Appliqué): Plongée profonde dans des bases de code complexes et de la documentation (par exemple, le cœur de WordPress) pour comprendre les meilleures pratiques, faire de l'ingénierie inverse et mettre en œuvre des solutions robustes.

Expériences

Développeur Full Stack & Architecte

ITC (Donation Proposée)

Juin 2024 - Sep 2024 · À distance

- ITC Hub - Plateforme de communication et de gestion interne

Ingénieur Logiciel Junior

Indépendant / Freelance

2023 - Présent · À distance

- Mosquito Killer DZ - Page de destination à haute conversion
- COD Simulation - Calculateur de ROI e-commerce
- Remdani Dental Center - Site web de clinique

Membre Actif & Développeur

Club de Programmation ITC

2024 - Présent · Alger, Algérie

- Site Web ITCP
- Plateforme ITC Talks

Développeur Frontend

CSE USDB promo 2023

2024 · Alger, Algérie

- CSE Hub

Projets

Tableau de Bord Simulation COD (Next.js, TypeScript, Tailwind CSS, shadcn/ui, Lucide React)

J'ai construit cet outil pour aider les propriétaires d'entreprises de e-commerce à visualiser leurs dépenses et leurs bénéfices. Il se concentre sur la fourniture d'une vue d'ensemble financière claire et en temps réel.

<https://calculator-delta-flax.vercel.app>

<https://github.com/samibentaiba/calculator>

Mosquito Killer DZ (Next.js, React, Tailwind CSS, shadcn/ui, Architecture Système, Documentation Technique)

Conçu pour maximiser les taux de conversion pour un produit de e-commerce local. Axé sur la vitesse, la réactivité mobile et un processus de paiement sans friction.

<https://mosquito-killer-2.vercel.app>

<https://github.com/samibentaiba/mosquito-killer-2>

Algis (Next.js, React, TypeScript, Prisma, PostgreSQL, NextAuth.js, Tailwind CSS, shadcn/ui, Écosystème Linux, Architecture Système, Documentation Technique)

Travailler sur Algis pour un client du secteur agricole a été une expérience enrichissante. Le projet implique une modélisation de données complexe avec Prisma pour gérer les UAP, le bétail, les machines, les cultures et leurs relations. Je construis un tableau de bord moderne full-stack qui remplace les flux de travail manuels basés sur Excel par une application web dynamique avec authentification et contrôle d'accès basé sur les rôles. Le dépôt est privé en raison des conditions de service du client.

<https://algis-preview.vercel.app>

C-Studio (Electron, TypeScript, React, MinGW-w64, Node.js, Windows, Écosystème Linux, Architecture Système, Documentation Technique)

Construire C-Studio a été un voyage passionnant dans le développement d'applications de bureau avec Electron. Le plus grand défi était d'intégrer le compilateur MinGW et de s'assurer qu'il fonctionne parfaitement sans aucune configuration système. Je me suis concentré sur la création d'une expérience conviviale pour les débutants apprenant la programmation C, avec des fonctionnalités comme la compilation instantanée et le support du terminal interactif.

<https://github.com/samibentaiba/c-studio>

ITC Hub (Next.js, React, TypeScript, Tailwind CSS, shadcn/ui, Prisma, PostgreSQL, NextAuth.js, React Hook Form, Zod, Nodemailer, Recharts, Écosystème Linux, Architecture Système, Documentation Technique)

Le développement d'ITC Hub a été une immersion profonde dans le développement full-stack avec Next.js. Le plus grand défi a été de concevoir le schéma de la base de données relationnelle avec Prisma pour gérer les relations complexes entre les utilisateurs, les départements, les équipes et les tickets. L'implémentation du contrôle d'accès basé sur les rôles (RBAC) avec Next-Auth était une fonctionnalité essentielle pour garantir la sécurité des données. Ce projet a consolidé mes compétences dans la création d'applications web sécurisées, évolutives et riches en données.

<https://itc-hub.vercel.app>

<https://github.com/samibentaiba/itc-hub>

Centre Dentaire Remdani (React, Node.js, Tailwind CSS, Écosystème Linux)

J'ai collaboré en tant que développeur frontend sur ce projet, travaillant en étroite collaboration avec le développeur backend et le responsable du déploiement. J'ai également contribué au développement backend avec de petits commits, acquérant ainsi une expérience full-stack.

<https://ramdani.vercel.app>

<https://github.com/ismail-devmaster/finalProjectFe>

Sunrise Energy (Next.js, TypeScript, Prisma, Tailwind CSS, shadcn/ui)

Ce projet m'a permis de me concentrer sur la création d'une interface professionnelle et visuellement attrayante pour un client corporatif. J'ai utilisé Next.js pour les performances et le référencement, garantissant que le site est rapide et facile à découvrir.

<https://github.com/samibentaiba/sunrise-energy>

AiHorizons (React, TypeScript, Tailwind CSS, Vite, Figma, Manipulation de Pipelines)

C'était un projet à haute intensité où j'ai collaboré étroitement avec un développeur backend et un designer UI/UX pour livrer un produit soigné dans un délai très court. Cela a testé ma capacité à travailler sous pression et à coordonner efficacement avec une équipe.

<https://ai-horizons.netlify.app>

<https://github.com/samibentaiba/AiHorizons>

ITC Talks (Next.js, TypeScript, Tailwind CSS, shaden/ui, Figma)

J'ai collaboré avec un designer UI/UX pour donner vie à la vision d'ITC Talks. Je me suis concentré sur la création d'une expérience utilisateur engageante et d'un flux de gestion de contenu fluide.

<https://itc-talks.vercel.app>

<https://github.com/samibentaiba/itc-talks>

ITCP (React, Vite, CSS, Figma, Architecture Système, Documentation Technique, Python)

J'ai travaillé en tant que développeur frontend dans une équipe avec un développeur backend et un designer UI/UX. Ce projet a amélioré mes compétences en travail d'équipe et ma capacité à intégrer des composants frontend avec des API backend.

<https://itc-programming.netlify.app>

<https://github.com/samibentaiba/ITCP>

CSE Hub (React, Vite, Figma)

Mon premier projet significatif en tant que développeur frontend. J'ai collaboré avec un designer UI/UX pour traduire les designs en code, posant les bases de ma carrière en ingénierie logicielle.

<https://cse-hub.netlify.app/home>

<https://github.com/samibentaiba/cse-hub>

Microservices & DevOps (Java, Spring Boot, Angular, H2 Database, Microservices, Écosystème Linux, Architecture Système, Documentation Technique)

J'ai élargi mes compétences au backend et au DevOps en travaillant avec Java Spring Boot et les microservices. Ce projet m'a donné une compréhension plus approfondie de l'architecture full-stack et de la gestion de base de données.

<https://github.com/samibentaiba/H2-SpringBoot>

iShowOff (Rust, GitHub Actions, CI/CD, Écosystème Linux, IDEs & Éditeurs, Architecture Système, Documentation Technique)

Ce fut une expérience d'apprentissage précieuse avec GitHub Actions. En forkant et en personnalisant ce projet, j'ai acquis une expérience pratique des pipelines CI/CD, des releases automatisées, et de l'intégration de contenu dynamique dans le README de mon profil GitHub.

<https://github.com/samibentaiba/ishowoff>

Éducation

Licence en Informatique

Université de Technologie, 2015-2019