

Roney de Oliveira

Software Engineer

Backend · Frontend · Full Stack · DevOps · Go (Golang) · Node.js · React · TypeScript · Kubernetes (K8s)

contact@roneyrogerio.dev | +55 43 99196-1524 | Apucarana, Paraná, Brasil | roneyrogerio.dev |

www.linkedin.com/in/roneyrogerio | github.com/roneyrogerio

PROFIL PROFESSIONNEL

Ingénieur logiciel avec une expérience des applications web, des API et des microservices en production, du backend au frontend en passant par les pratiques DevOps.

Je travaille principalement avec Go, PostgreSQL, Node.js et React, ainsi qu'avec l'infonuagique (AWS, GCP et Oracle Cloud), les conteneurs et l'architecture orientée services.

Attaché à un code simple, évolutif et facile à maintenir, avec un intérêt continu pour la performance, la concurrence et les systèmes distribués.

COMPÉTENCES TECHNIQUES

Langages et frameworks: Go (Golang), TypeScript, JavaScript, Node.js, PHP, WordPress, C, C++, Python.

Frontend: React, Next.js (App Router), HTML et CSS, Astro, Vite, Three.js, Interfaces adaptatives.

Backend et architecture: Microservices, API REST (REST APIs), GraphQL, Webhooks, Server-Sent Events (SSE), Vidéo en temps réel (Cloudflare RealtimeKit, Streaming), Idempotence (Idempotency), Gin, Architecture orientée services (SOA), Versionnement d'API (Versionnement de contrats), Tests automatisés, Développement sécurisé (Sécurité), Revue de code (Code review), Développement assisté par IA (AI-assisted development), Messagerie asynchrone (Communication asynchrone), Résilience (Tolérance aux pannes), Scalabilité, Optimisation des performances (Performance).

Infonuagique et exploitation: AWS Lambda (Serverless), Amazon S3, Amazon RDS, AWS CloudFormation, AWS API Gateway, AWS Rekognition, Google Cloud Platform (GCP), Cloud Run, Cloudflare R2 (Stockage objet), BigQuery, Pub/Sub, Vertex AI, Oracle Cloud Infrastructure (OCI, Oracle Cloud), Oracle Kubernetes Engine (OKE), Observabilité (Supervision), OpenTelemetry (Traçage distribué, Tracing), Diagnostic en production, DevOps.

Données et plateforme: PostgreSQL (Bases de données relationnelles), MongoDB, SQLite, Prisma, Modélisation de données, Optimisation de requêtes, Linux, Exécution locale de LLM (Self-hosted LLM, Inférence locale), Llama (Ollama), API d'OpenAI (OpenAI API, LLM), Sorties structurées avec JSON Schema (Structured Outputs), Ingénierie de prompts (Prompt engineering), Réseaux pour services, Docker, Kubernetes (K8s), Knative, CI/CD avec GitHub Actions (Intégration et livraison continues), Terraform (Infrastructure as code, IaC), Git, GitLab, Jira et Bash.

EXPÉRIENCE PROFESSIONNELLE

Développeur Full Stack — Proaba

août/2026 – sept/2026 · Freelance · À distance

- Diffusion en direct et enregistrement de séances avec Cloudflare RealtimeKit sur une plateforme de clinique pédiatrique : création de la salle, émission d'un jeton par participant, provisionnement des préreglages et fermeture à la fin, le tout côté serveur.
- Enregistrement écrit directement dans Cloudflare R2, avec transcription automatique et lecture depuis l'écran de la séance.
- Webhooks du fournisseur traités de façon idempotente : des événements superposés ne

créaient plus deux enregistrements pour une même séance, et le renvoi que l'API réclamait elle-même en répondant 500 a cessé d'être rejeté comme doublon.

- Flux SSE libéré lorsque l'onglet est masqué et fermé dans un try/finally, pour qu'une séance ouverte ne retienne pas une place chez le fournisseur.
- Contrôle d'accès exigeant un lien avec l'enfant pour atteindre un enregistrement, autorisation enregistrement par enregistrement, et affichage selon le fuseau horaire de qui regarde plutôt que celui du serveur.
- Confidentialité dès la conception : aucun nom de patient n'est transmis au fournisseur — la salle est identifiée par l'identifiant de la séance.
- Tests de bout en bout avec Playwright et secrets du fournisseur cartographiés par environnement au déploiement.

Ingénieur logiciel senior — e-didatico

juil/2024 – mai/2026 · Temps plein · À distance

- Intervention de bout en bout sur le cycle d'ingénierie, en développant et en exploitant des solutions backend dans le nuage (AWS et GCP).
- Application de pratiques DevOps et agiles pour garantir des livraisons efficaces, stables et à forte valeur.
- Évolution continue d'une plateforme en microservices, avec un souci constant de fiabilité en exploitation.

Ingénieur logiciel intermédiaire — e-didatico

janv/2022 – juil/2024 · Temps plein · À distance

- Implémentation de fonctionnalités en autonomie et avec une bonne maîtrise du système.
- Correction d'anomalies et maintenance évolutive en environnement de production.
- Développement backend avec Node.js et PostgreSQL, en appui des livraisons continues du produit.

Ingénieur logiciel junior — e-didatico

juil/2021 – janv/2022 · Temps plein · À distance

- Développement de fonctionnalités et correction d'anomalies.
- Participation au cycle de livraison, avec une attention portée à la qualité et à la stabilité du logiciel.

Analyste de soutien technique — e-didatico

mars/2021 – juil/2021 · Indépendant · À distance

- Assistance aux étudiants pendant la passation d'examens en ligne.
- Soutien opérationnel afin d'assurer la continuité et la qualité du service.

Développeur Full Stack — Early Denver

déc/2018 – déc/2019 · Temps plein · Brasilia, DF, Brésil (à distance)

- Développement full stack de fonctionnalités web.
- Travail avec Node.js, React.js et PostgreSQL dans le cycle de livraison du produit.
- Collaboration à distance avec l'équipe pour faire évoluer l'application en continu.

Développeur logiciel junior — StutzLab

janv/2018 – nov/2018 · Temps plein · São Paulo, SP, Brésil (à distance)

- Développement de fonctionnalités pour des applications web.
- Travail avec PHP, JavaScript, MySQL, HTML et CSS pour l'implémentation d'écrans et de règles métier.
- Utilisation de Bootstrap pour uniformiser l'interface et gagner en productivité côté frontend.

Développeur web — Projets personnels

nov/2010 – déc/2017 · Indépendant · À distance

- Développement de projets personnels avec HTML, CSS, PHP, MySQL et Nginx.
- Monétisation de produits et de contenus numériques au moyen de Google AdSense.

PROJETS

Soltar Pipa — soltarpipa.online

Projet personnel · multijoueur en temps réel · En ligne à: soltarpipa.online

- Serveur autoritativ en Go : toute la physique de vol et le duel entre les fils tournent sur le serveur, le navigateur ne fait que dessiner et transmettre les commandes, ce qui supprime la surface de triche côté client.
- Protocole binaire maison sur WebSocket, en petit-boutiste, avec une trame de référence écrite par le Go et décodée par le test du client en TypeScript : modifier un côté sans l'autre casse la compilation.
- Zone d'intérêt traitant chaque joueur comme un segment de droite, de la main au cerf-volant, sur une grille spatiale de 48 m avec un rayon de 120 m et de l'hystérésis — 12 μ s par joueur, sans allocation.
- Optimisation guidée par le profilage : balayage trié à la place de la comparaison de toutes les paires (de 28 ms à ~1 ms par tick) et quickselect à la place du tri lors de la sélection des cerfs-volants par envoi (de 16 % à 2,4 % du temps CPU du tick).
- Test de charge maison avec 5 000 WebSockets simultanés, sans échec de connexion ni éviction ; en production la salle tourne avec 140 joueurs à 15 images par seconde, limite imposée par la bande passante de 10 Mbps du répartiteur, et non par le processeur.
- Production sur Knative au-dessus de Kubernetes, avec deux services et deux domaines, des images sur GHCR et un versionnement automatisé par release-please.

Go, TypeScript, Three.js, WebSocket, Vite, Docker, Kubernetes, Knative.

CV multilingue — roneyrogerio.dev

Projet personnel · Dépôt: github.com/roneyrogerio/curriculum

- Adaptation du CV à une offre par LLM (API d'OpenAI avec Structured Outputs) : le modèle reçoit des faits identifiés et ne renvoie que la sélection, l'ordre et la formulation ; l'employeur, les dates et les liens viennent de la source.
- Vérification automatique de chaque phrase réécrite face à son origine, rejetant toute technologie ou tout chiffre que le fait n'affirme pas.
- Site statique multilingue avec une version de lecture et une version imprimable A4, plus des exportateurs PDF et DOCX validés contre un simulateur d'analyse ATS.
- Déploiement automatisé par GitHub Actions à chaque version, avec versionnement et journal des modifications par release-please et image multi-architecture (linux/amd64 et linux/arm64) sur GitHub Container Registry.
- Exécution comme Knative Service au-dessus de Kubernetes (Oracle Kubernetes Engine), avec un espace de noms dédié et un domaine personnalisé.

Astro, TypeScript, Node.js 24, Docker, GitHub Actions, API d'OpenAI, Kubernetes, Knative.

Receitex — receitex.com.br

Projet personnel · sans maintenance, toujours en ligne · En ligne à: receitex.com.br

- Site de recettes avec une extension maison, CulinAI : un panneau React dans l'administration de WordPress qui génère la recette entière — titre, ingrédients avec fractions, préparation, catégories et étiquettes — et la publie comme article.
- Texte et photo du plat générés par IA, l'image étant envoyée à la médiathèque par des routes REST propres à l'extension.

- Empaqueté en conteneur à partir de l'image officielle de WordPress, exécuté comme Knative Service au-dessus de Kubernetes, avec MySQL, domaine propre et TLS.
- Il n'est plus maintenu et reste en ligne à dessein : c'est la trace d'un site entier écrit par IA à une époque où cela était encore nouveau. Les images viennent de la première génération des modèles — reconnaissables comme le bon plat, et visiblement en deçà de ce qui se produit aujourd'hui.

WordPress, PHP, React, API d'OpenAI, MySQL, Docker, Kubernetes, Knative.

minishell

42 São Paulo · C · Dépôt: github.com/roneyrogerio/minishell

- Terminal Linux implémenté en C avec des appels système, sans bibliothèque d'analyse lexicale.
- Prise en charge de plusieurs commandes, des guillemets simples et doubles, de l'échappement de caractères, des pipelines et des redirections (<, > et >>).
- Commandes intégrées (echo, cd, pwd, export, unset, env, exit), signaux (ctrl-C, ctrl-D, ctrl-\) et variables d'environnement.

C, Appels système POSIX.

Pipefy Client Management API

Projet personnel · 2026 · Dépôt: github.com/roneyrogerio/pipefy-client-management-api

- API REST en Go avec Gin et persistance dans SQLite, pour l'enregistrement de clients et le traitement du patrimoine investi.
- Intégration GraphQL avec Pipefy et réception de webhooks avec contrôle d'idempotence : le même événement livré deux fois n'est pas traité deux fois.
- Traçage distribué avec OpenTelemetry et Jaeger, et environnement local complet en Docker Compose — mock GraphQL, collecteur et API avec rechargement automatique — démarrant en une seule commande.
- Le même routeur sert le HTTP local et AWS Lambda par proxy integration de l'API Gateway, avec des tests automatisés couvrant les routes.

Go, Gin, SQLite, GraphQL, OpenTelemetry, Jaeger, Docker Compose, AWS Lambda, API Gateway.

cub3D et raycast-engine

42 São Paulo · 2020 · Dépôt: github.com/roneyrogerio/cub3D

- Jeu de démonstration pseudo-3D dessiné par lancer de rayons en C, le moteur étant extrait dans une bibliothèque propre dont le jeu ne consomme que l'interface.
- La bibliothèque expose la caméra, la minicarte, la taille de la fenêtre et la mise à jour du mouvement, et dessine la scène via un pointeur de fonction fourni par l'appelant.

C, Raycasting, Makefile.

libbmp

42 São Paulo · 2020 · Dépôt: github.com/roneyrogerio/libbmp

- Bibliothèque en C pur pour lire, créer, modifier pixel par pixel et écrire des images BMP de 24 et 32 bits par pixel.
- Elle travaille directement sur le format du fichier et sur l'ordre des octets en petit-boutiste, sans dépendance externe.

C.

ft_server

42 São Paulo · 2020 · Dépôt: github.com/roneyrogerio/ft_server

- Image Docker unique servant NGINX, PHP, MySQL, WordPress et phpMyAdmin, configurés de zéro.

Docker, NGINX, PHP, MySQL, WordPress.

ft_services

42 São Paulo · 2020 · Dépôt: github.com/roneyrogerio/ft_services

- Projet d'administration système et réseau : provisionnement et orchestration de services en conteneurs.

Administration système, Réseaux, Conteneurs.

libasm

42 São Paulo · 2020 · Dépôt: github.com/roneyrogerio/libasm

- Fonctions de la bibliothèque standard du C réécrites en assembleur, avec les conventions d'appel et le retour d'erreur de l'original.

Assembleur, C.

libft

42 São Paulo · 2020 · Dépôt: github.com/roneyrogerio/libft

- Réimplémentation des fonctions de la bibliothèque standard du C — mémoire, chaînes, conversions et sortie par descripteur —, écrites de zéro pour comprendre ce que chacune coûte.
- Inclut une liste chaînée maison avec création, insertion aux deux extrémités, itération, application d'une fonction et libération.

C.

ft_printf

42 São Paulo · 2020 · Dépôt: github.com/roneyrogerio/ft_printf

- Réimplémentation de printf en C, avec un nombre variable d'arguments et une répartition par spécificateur de format, appuyée sur la libft maison.

C, Fonctions variadiques.

get_next_line

42 São Paulo · 2020 · Dépôt: github.com/roneyrogerio/get_next_line

- Fonction qui lit un fichier ligne par ligne depuis un descripteur, en n'utilisant que read, malloc et free, avec une taille de tampon fixée à la compilation.
- La seconde version a corrigé la fuite que laissait la première lorsque le fichier n'était pas lu jusqu'au bout — l'état en attente restait alloué sans propriétaire.

C, Appels système POSIX.

Modules C++

42 São Paulo · 2021 · Dépôt: github.com/roneyrogerio/cpp_module_00

- Premiers modules C++ de 42 : lecture des arguments de la ligne de commande et un répertoire en mémoire avec les commandes ADD, SEARCH et EXIT.
- Le module suivant réunit cinq exercices de plus, dans cpp_module_01.

C++.

FORMATION

- **DUT en analyse et développement de systèmes** — UNOPAR — Universidade Norte do Paraná, janv/2022 — déc/2023 (Note finale 9,5) · [diplôme](#)
- **Formation technique en génie logiciel** — 42 São Paulo, 2020 — 2021

CERTIFICATIONS

- **Next.js App Router Fundamentals** — Vercel, août/2026 · [dashboard-app](#)
- **React Foundations for Next.js** — Vercel, août/2026 · [react-foundations](#)
- **Programming with Google Go Specialization** — University of California, Irvine, mai/2026 · [2UUQCPO9MST7](#)
- **Concurrency in Go** — University of California, Irvine, mai/2026 · [M2EKXK59LHYI](#)

- **Functions, Methods, and Interfaces in Go** — University of California, Irvine, mai/2026 · [WIEGNHPO8TBM](#)
- **Getting Started with Go** — University of California, Irvine, mai/2026 · [KFGD3IF75I0Q](#)

FORMATIONS COMPLÉMENTAIRES

- **Introduction au langage Python** — Unopar, 15 h, 1er semestre 2023 · [certificat](#)
- **Introduction à l'analyse de données avec Python** — Unopar, 15 h, 1er semestre 2023 · [certificat](#)
- **Structures de données en Python** — Unopar, 10 h, 1er semestre 2023 · [certificat](#)
- **Technologies de l'information appliquées au droit** — Unopar, 60 h, 2e semestre 2023 · [certificat](#)
- **Droit électronique** — Unopar, 60 h, 2e semestre 2023 · [certificat](#)
- **Modèles de gestion** — Unopar, 60 h, 2e semestre 2023 · [certificat](#)

LANGUES

- **Portugais** — Langue maternelle
- **Anglais** — Notions (A2)

COMPÉTENCES ADDITIONNELLES

Ingénieur logiciel, Software Engineer, Développeur back-end, Développeur backend, Backend Engineer, Développeur front-end, Développeur frontend, Frontend Engineer, Développeur full stack, Full Stack Engineer, Systèmes distribués, WebSocket, SQL, NoSQL, Amazon Web Services, Haute disponibilité, Fiabilité, Méthodes agiles, Scrum.