



SÉBASTIEN
NOBOUR

Maastricht, Pays-Bas (favorisant le full-remote)

sebastien@devedanos.com

www.devedanos.com

+337 80 98 83 25

LinkedIn : sebastien-nobour

DÉVELOPPEUR JAVASCRIPT FULL STACK AVEC EXPERTISE EN REACT.JS NODE.JS TYPESCRIPT

PROFIL

J'aide les entreprises à créer des applications personnalisées et je me concentre sur le développement de fonctionnalités à forte valeur ajoutée en utilisant les technologies Javascript (Node.js, React.js, Next.js, Nest.js, Typescript...).

J'ai plus de 5 ans d'expérience dans le développement d'interface utilisateur front-end et la conception et le développement back-end d'API et de bases de données, les tests automatisés et le déploiement chez les hébergeurs.

Je privilégie les méthodologies agiles et les principes lean, créant une organisation sur mesure pour l'équipe et aidant à prioriser le développement de fonctionnalités à forte valeur ajoutée.

Langues : Français (langue maternelle), anglais courant

Atouts : orienté vers l'action, axé sur les données, centré sur l'utilisateur final, polyvalent, proactif, communication constructive, esprit d'équipe, compatissant, innovant, persévérant, autodidacte, curieux, en constante amélioration.

EXPERIENCE PROFESSIONNELLE

DÉVELOPPEUR LOGICIELS JAVA NAAS

Vodafone Ziggo (depuis 2023)

- Maastricht, Pays-Bas ; Développement d'un logiciel de réseau en tant que service utilisant la spécification TMF641 et la plate-forme Amdocs NEO

DÉVELOPPEUR FULL- STACK

Consultant Indépendant en développement d'applications web (depuis 2020)

- Weekandgo.com : développement d'applications web
- Commentaires sur les produits : développement d'applications web
- Refactory Devops (Pays-Bas) : Maintenance d'applications web

DÉVELOPPEUR PHP

Valeo (2019 - 2020)

- Maintenance d'une application web de suivi des licences

DÉVELOPPEUR WEB

Consultant Indépendant en développement d'applications web (2019)

- Refactory Devops (Pays-Bas) : Maintenance et développement de sites et d'applications web

AUTRES POSTES

- depuis 2021: agent de marketing affilié
- 2018 : Technicien de support informatique à Spie
- 2017 : Technicien support technique et logistique chez Logicom
- 2015 à 2017 : bénévole religieux à temps plein et équipier de restauration rapide à temps partiel
- 2015 : Technicien itinérant sur les systèmes de sécurité électronique chez Automatic Alarm
- 2012 à 2014 : Technicien itinérant sur les systèmes d'ouverture automatisés chez APB

- Architecture et concepts : MVC, FLUX, Data Down Actions UP (DDAU), POO, patterns SOLID, REST et JSON et RPC et GraphQL et API SOAP, SPA (Server Side Rendering et Client Side Rendering), JAMStack pour site web statique, websocket, web worker, micro-services, serverless, conteneurisation, virtualisation, Infrastructure As Code, ORM/ODM, TDD, BDD, tests (unitaires, d'intégration, interactifs, visuels, d'application, de système), protocoles Publish/Subscribe (websocket, Message Queuing), Design Patterns, Hexagonal Architecture, TMForum
- Systèmes d'exploitation (OS) : Linux (Ubuntu, Debian, Raspbian, RaspiOS), Mac OS, Windows (XP, 10,11, Server 12,16)
- Langages : JavaScript ES5, JSON, PHP7-8, SQL, HTML5, CSS3, XML, SASS, PostCSS, Java SE 8 -> 20
- Frameworks CSS : Bootstrap [4,5], TailwindCSS 2, Bulma CSS, Material UI/MUI, AntDesign
- Librairies et frameworks JS : React.js (16, 17 et Redux, React Apollo, React Router, Jotai, React Context), Ember.js, JQuery, Next.js (12, 13), Lodash, React Hook Form, Formik, Joi, emotion, luxon, svgo ;
- Module bundlers : Webpack, SWC, Babel;
- Frameworks backend Node.js : Express.js, Nest.js (Apollo-graphql), SocketIO, Koa.js
- Frameworks desktop Node.js : Electron.js
- Frameworks PHP : Symfony (4,5), API-Platform
- Bases de Données : PostgreSQL, MySQL, OracleSQL, MongoDB, DoctrineORM, TypeORM, Mikro-ORM, Mongoose
- Outils de test et qualité : ESLint, Prettier, Stylelint, Jest, Mocha, Chai HTTP, Qunit, Coverall, PHPUnit, Cypress, BrowserStack, Nightwatch, Playwright, testing-library, Cucumber, Gherkin, Husky, MirageJS
- Versionning: Git, Github, Gitlab, Bitbucket
- CI/CD: Gitlab CI, Github Actions, Travis CI, AWS Code Pipeline
- Devops: Docker and Docker compose, Serverless Framework
- Cloud: Amazon Web Services (Lambda, ECS, ECR, EC2, S3, EFS, Amplify, Route53, LightSail, IAM, SNS, SES, SQS, API Gateway, DymanoDB, CloudWatch, Secret Manager, IAM, S3, DynamoDB, BitBucket, CloudFormation, API Gateway, EventBridge, Code Pipeline)
- Project/Product management methods and tooling: Agile methodology, Scrum, SAFe, CollabNet, Jira;
- Documentation : Markdown, JSDoc, Chromatic, Storybook, Confluence;
- Marketing method and tools : RGPD, defining a client avatar, niche and offer, SEO, Google Analytics 4, Plausible, Axeptio;
- Visual design tooling : Figma, Adobe XD
- CMS : Shopify, Wordpress/Bedrock, WooCommerce, Strapi CMS, Ghost
- Anytics et monitoring : pm2 services, Axiom

PARCOURS SCOLAIRE

- 2023: Continuous Delivery trainings from David FARLEY, Agility and No Estimate software project management from Vasco DUARTE, Wordpress in depth, JAVA SE 17 training
- 2021: Training and coaching for freelance web developers with Mike Codeur and Charles-Louis ALLIZARD, Nest.js and Typescript training from Ariel WEINBERGER, AWS 12 Fundamental services from Erik JEULIN
- 2020: Training and coaching in e-commerce, entrepreneurship and affiliate marketing by Olivier ALLAIN with the Amazon Revolution and Marketing Revolution programs, entrepreneurship and marketing training and coaching by Alec HENRY with Entrepreneurs.com program, MERN stack and redux training from Brad Traversy
- 2019 - 2020: Bachelor (EQF* Level 6) Application Developer at My Digital School Paris 19th, Symfony React trainings from Lior CHAMLA
- Sept. 2018 - Janv. 2019: Web developer and integrator training with certifications at Web Force 3
- 2012- 2014: BTS Domotique (EQF* Level 5) at IFA Delorozoy

SIDE PROJECTS

- IoT Remote LED control : logiciel développé pour mon raspberry pi 1 B+
- nobour-eshop.com : application web de commerce électronique en MERN Stack, AWS (EC2, Amplify), Docker, Github Actions
- Application CRM (crm.devedanos.com) : PHP, API-Platform, React.js
- Application réplique Air B n'B (sym-5-bnb.devedanos.com) : PHP, Symfony
- Applications de gestion de todo-list et de salles de classe : NestJS, ReactJS, MongoDB, PostgreSQL, GraphQL et API REST, AWS, TypeScript
- Application de vente aux enchères : Node.js, React.js, Serverless Framework AWS, AWS Lambda

LOISIRS ET LES CENTRES D'INTÉRÊT

- Agilité (No Estimates, Continuous Delivery, Mob et Pair Programming), mentorat de développeurs juniors, Conférences (API-Platform Con, Devternity, JS World Conference et React World Conference...);
- Pratiques de développement logiciel et méthodes de gestion de projet promues par David FARLEY, Allen HOLLUB, Woody ZUIL, Vasco DUARTE, Robert C. Martin.
- Travail à distance;
- Développement personnel : leadership, communication, techniques de lecture rapide ;
- Entrepreneuriat : commerce électronique via Amazon FBA, marketing d'affiliation; ;
- Sports : gymnastique artistique, fitness, natation ;
- Activité bénévole : Depuis 2018, bénévole dans une organisation à but non lucratif anglophone, bénévole à temps plein de 2015 à 2017 en tant que professeur individuel, conférencier ;
- Explorations technologiques actuelles : PandaCSS, CQRS, Kafka, RabbitMQ

VOFAFONE ZIGGO

DEPUIS SEPTEMBRE 2023

DÉVELOPPEUR DE LOGICIELS JAVA NAAS SALARIÉ

Introduction :

Le groupe Vodafone Ziggo (VZ) est une entreprise de télécommunication fusionnée avec 7,5k employés dans le plan international, opérant sur les réseaux cellulaires et câblés néerlandais. Il existe depuis 2017 et comprend des entreprises comme Ziggo, Liberty Global, Vodafone, Hollandsnieuwe et 10 autres. Son principal concurrent est KPN, une autre société de télécommunications néerlandaise qui ne cesse d'innover dans ses processus. VZ s'efforce de normaliser et d'automatiser ses processus informatiques afin d'améliorer la qualité du service, de réduire les erreurs commises par les ingénieurs et le coût des ressources humaines, et de déployer plus rapidement les ressources du réseau.

Contexte :

Le groupe VZ internalise les connaissances relatives à divers projets et stabilise son activité dans l'ensemble des Pays-Bas en recrutant de nouveaux employés plutôt que des consultants. Il est également passé d'une méthode traditionnelle de cycle de gestion de projet en cascade/V à une méthode plus flexible appelée Scaled Agile Framework for enterprise. J'ai été engagé comme développeur Java dans la tribu Network as a Service, qui compte 100 personnes. Je fais partie d'une équipe comprenant 1 Product Owner, 1 Scrum Master, 1 Architect, 1 Lead Developer, 1 Tester et 4 développeurs. Notre équipe est basée en Inde, à Eindhoven et à Maastricht.

Mon équipe est responsable du développement d'une API REST traduisant les demandes des clients professionnels pour des ressources telles que les services VPN en ressources réseau déployées en quelques minutes au lieu de plusieurs semaines.

Tâches :

Développement d'endpoints respectant la spécification TM Forum 641, validation des données pour la plateforme Amdocs Neo,

Technologies :

Java 11, SonarQube, Amdocs NEO Platform, JIRA, SAFe, Confluence, Windows (11, server 2012/16), TM Forum, Gitlab, Citrix, AWS, Microservices, tests unitaires, Junit

DÉVELOPPEUR FULL-STACK NEXT.JS NODE.JS EN TANT QUE CONSULTANT INDÉPENDANT

Introduction :

Weekandgo.com a pour ambition de devenir la référence des places de marché pour la vente et la location de véhicules de loisirs (camping-cars, vans, etc.). Rapido, constructeur français de camping-cars basé en Mayenne (France, 53) et parmi les leaders du marché européen est à l'origine du projet. Le développement de l'application a débuté en mars 2021 pour une mise en production en février 2022.

Contexte :

Pour réaliser les développements, Rapido a fait appel à la société Leb Communication, une entreprise existant depuis plus de 30 ans et développant son activité dans le développement d'applications web. A son tour, Leb Communication a fait appel à mes services pour renforcer son équipe de développeurs avec un profil spécialisé en React.js, Node.js et Next.js, apportant du lead sur les aspects techniques et la priorisation des fonctionnalités de développement. Notre équipe est composée d'un développeur backend, d'un intégrateur et d'un UX designer, et de moi en tant que développeur full-stack.

L'application WeekandGo est composée d'une API REST business en API Platform (Symfony et PHP), d'un CMS REST headless en Strapi pour les articles de blog et d'une SPA en Next.js. La SPA est divisée en 3 parties : une accessible à tous les visiteurs, une réservée aux vendeurs privés et une autre pour les vendeurs professionnels associés à Rapido. Mais le coeur de l'application réside dans le filtrage des annonces.

Tâches :

- Développement de fonctionnalités sur la SPA telles que la création, l'édition, la suppression et le filtrage des offres, la sécurisation/restriction de l'accès aux ressources, la structure des éléments du blog et la génération statique ou le rendu côté serveur de ses pages....
- Maintenance des dépendances et des versions de Node,
- Qualité : automatisation et développement de tests de régression (unitaires, visuels, interactifs, applicatifs), construction d'un pipeline de livraison continue (Continuous Delivery Pipelines).
- Choix et conception technique globale, consultant technique
- Moteur dans la priorisation et la simplification des fonctionnalités
- Apport des pratiques Agiles pour les développements (automatisation des déploiements, outils collaboratifs, langage commun à l'équipe, syntaxe de code uniforme...)

Technologies :

Node.js, Next.js (next-auth), React.js (react-hook-form), Redux, Jotai, React Context, CSS3, HTML5, Material UI/MUI, Strapi CMS, REST, JWT, HTTP, Docker, bash, client side cache SWR, PHP, API Platform, OVH, Vercel, Heroku, MySQL, PostgreSQL, CORS, React Hook Form, Open-Graph tags, Google Recaptcha v3, Google Tag Manager, Axeptio cookie consent, Storybook, Chromatic, Jest, Cypress, Playwright, testing-library

CONSULTANT INDEPENDANT DÉVELOPPEUR FULL-STACK NODEJS REACTJS NESTJS TYPESCRIPTIntroduction :

Product Feedback est une startup créée en 2019 à Marseille (13, France) qui vise à révolutionner la gestion des retours clients sur un produit. Elle s'inspire de l'application notion.so.

Product Feedback est partie du constat que la collecte et la synthèse des retours utilisateurs sur un produit était difficile et chronophage : informations non partagées avec l'ensemble de l'équipe, fiches ou documents justificatifs manquants.... Pour simplifier cette gestion, Product Feedback développe une interface collaborative en ligne, rassemblant en un seul endroit les réponses des clients aux formulaires, les rapports des membres de l'équipe produit sur ces réponses, et les décisions finales basées sur les rapports. Il prévoit d'ajouter des services supplémentaires, tels que l'intégration de feuilles de calcul Excel et plus encore...

En mars, son équipe était composée d'un PDG, d'un CMO basé à Marseille et d'un CTO travaillant depuis la République tchèque. Je travaillais directement avec le CTO.

Contexte :

Product Feedback m'a proposé de travailler pour eux en tant que freelance au lieu de passer des tests techniques, avant de m'embaucher. Il s'agissait d'un poste entièrement à distance.

Au 6ème jour d'essai, j'ai eu mon premier entretien avec le PDG. Au cours de ce dernier, j'ai appris que l'équipe de ProductFeedback recherchait en fait un développeur senior pour remplacer son directeur technique. J'ai donc travaillé 6 jours pour ProductFeedback.

Tâches :

- Participation à la mise en place de l'architecture de l'application, conception de la base de données et des entités backend
- Développement d'une API GraphQL ainsi que de ses requêtes et mutations
- Développement d'une interface utilisateur sous la forme d'une application à page unique, avec ses formulaires et ses composants réactifs
- Développement d'une API GraphQL, avec ses résolveurs, ses services, ses référentiels.

Technologies :

NodeJS, TypeScript, NestJS, ReactJS, React-Apollo, PostgreSQL, AWS (Amplify), git, Github, Docker, composants stylisés, AntDesign, Mikro-ORM, Material UI, Responsive Web Design, composants stylisés.

CONSULTANT INDÉPENDANT DÉVELOPPEUR FULL-STACK SYMFONY EMBERJSIntroduction :

Le client final est une entreprise néerlandaise qui fabrique des chaussures de sécurité. Elle dispose d'un CRM développé par Refactory Devops, une société unipersonnelle néerlandaise.

Contexte :

Refactory Devops a fait appel à mes services pour mettre à jour le CRM de son client :

Refactory Devops a fait appel à mes services pour mettre à jour le CRM de son client. J'ai travaillé sur la mise à jour de son backend et de son frontend. Refactory Devops s'occupe des déploiements dans les environnements de staging et de production, et de la revue de code. Le suivi des tickets se fait sur Gitlab.

Tâches :

- Mise à jour de l'API REST, de Symfony 3 à 4 et adaptation de la configuration des conteneurs de services.
- Mise à jour de la Single Page Application d'EmberJS 2 à 3, avec sa syntaxe Octane dans les templates, remplacement des classes ember par des classes JS natives dans les composants, contrôleurs, helpers, modèles.
- Développement de tests unitaires descriptifs et de tests d'intégration avec PHPUnit pour l'API REST, et Qunit pour la SPA.

Technologies :

PHP 7, Javascript, EmberJS (2, 3), Qunit (qunit-dom), PHPUnit, CI/CD, Gitlab CI, Docker, NodeJS, YAML, Symfony (3, 4)

DÉVELOPPEUR PHP SALARIE@

Introduction :

Valeo est une grande entreprise internationale, fondée en France, et l'un des leaders du marché de l'équipement automobile. Elle est composée de 5 activités : les systèmes thermiques, les systèmes de visibilité, les systèmes de propulsion, les systèmes de confort et d'aide à la conduite, et les activités d'après-vente. Chaque activité dispose d'une équipe Méthodes et Outils, qui elle-même dispose d'une équipe Engineering Information System (EIS). L'équipe SIE gère, entre autres, les licences des logiciels, permettant ainsi aux ingénieurs de disposer des outils numériques nécessaires à leur travail (conception, modélisation 3D, bancs d'essais virtuels).

Contexte :

Pendant mon apprentissage, j'ai rejoint l'équipe EIS du groupe des systèmes thermiques. Directement sous la supervision de mon manager et en tant que seul développeur sur le projet, j'étais en charge de la maintenance de Licence Server Status, une application PHP personnalisée permettant de surveiller les serveurs de Valeo dans le monde entier afin de vérifier l'état des licences logicielles. Cette application stocke les informations dans une base de données, puis les affiche sur un tableau de bord, en fournissant des alertes et en envoyant des courriels lorsqu'une licence est sur le point d'expirer. Ce projet a connu plusieurs versions en 15 ans, et a souvent été réécrit à partir de zéro par les apprentis successifs. Le but de ce projet est d'aider le SIE à prévenir le renouvellement et l'achat de licences de logiciels afin de mieux gérer les budgets de Valeo.

Tâches :

- Suggestion et implémentation de pratiques de développement, d'architecture et de technologies afin de préserver le projet et de ne plus avoir à le réécrire (utilisation du versioning, déploiement de programmes via git au lieu de ftp, utilisation d'un système de gestion de tickets, passage du projet à la POO et au MVC, ajout de tests automatisés et de CI/CD...)
- Développement et maintenance de l'interface utilisateur
- Développement de graphiques en ChartJS
- Définition, développement et maintenance de scripts PHP interrogeant les serveurs et leurs démons de licences logicielles
- Conception de l'architecture et implémentation de la base de données
- Développement de requêtes SQL
- Veille technologique sur Symfony 5 en développant une réplique d'Air B n'B accessible sur sym-5-bnb.devedanos.com
- Déploiement en environnement de pré-production et de production
- Développement de Proof of Concepts pour l'application PHP personnalisée License Server Status sur Symfony 5 (non réalisé en raison de la pandémie COVID-19)

Technologies:

PHP (5, 7), Symfony (4, 5), BeyondCompare, Git, CollabNet, Javascript, AJAX, HTML, CSS, Bootstrap, Apache, CentOS, Windows Server 2008 et 2012, Windows 10, Scheduled task, WinBash, YAML, Bootstrap, Bootswatch, Doctrine ORM, SQL, OracleSQL, MySQL, OVH

DEVELOPPEUR WEB INDÉPENDANT

Introduction :

Refactory Devops est une société de services numériques néerlandaise unipersonnelle. Son développeur a plus de 15 ans d'expérience. Il développe ses propres solutions devops, construit des sites web et des applications pour ses clients.

Contexte :

En 2019, je prévoyais de déménager aux Pays-Bas. Pendant quelques mois, j'ai travaillé aux côtés du développeur de Refactory Devops sur des projets pour ses clients, ainsi que sur son propre projet de startup. Je travaillais à la fois depuis la France et Nijmegen aux Pays-Bas.

Tâches :

- Implémentation de l'internationalisation de la langue d'une application à page unique
- Développement du système d'authentification via JWT et OAuth2
- Intégration web pour les sites vitrines via CMS
- Maintenance et correction des bogues sur les applications à page unique
- Migration de sites web d'un service d'hébergement web à un autre

Technologies :

Neos Framework, Neos CMS, EmberJS 2, l18n, Javascript, HTML5, CSS3, PHP 7, YAML, cPanel;

API Publish Subscribe en Node.js pour le contrôle à distance de raspberry pi GPIO

Introduction :

C'est un logiciel qui m'est très cher car cela fait 10 ans que je veux construire une telle chose alors que je n'avais pas les compétences à l'époque. Je suis passionné par les systèmes domotiques. C'est pourquoi j'ai étudié l'informatique et l'électricité et je suis devenu technicien, puis programmeur. Je suis maintenant fier d'avoir prouvé que je suis capable de concevoir et de développer un logiciel pour contrôler des objets à distance. Je continue à développer ce projet pendant mon temps libre. Pour l'instant, ce projet a une interface utilisateur en next.js qui communique avec une API websocket construite en node.js hébergée sur un Raspberry Pi 1 B+. Le système permet d'allumer et d'éteindre une LED branchée sur un bread board connecté aux broches General Purpose Input Output du raspberry, en installant moi-même un dongle wifi avec un driver en C provenant de github. J'ai pu atteindre cette étape en 4 jours.

Les défis que j'ai rencontrés sont de réaliser que mon raspberry pi n'est pas compatible avec les versions de node.js supérieures à 14 seulement 3 jours après avoir commencé, de trouver des solutions de contournement pour installer le système d'exploitation et me connecter aux matériels du raspberry pi et de mon macbook.

Je continue également à m'amuser à concevoir une architecture de code propre pour avoir un code très flexible qui sépare encore la logique du domaine de la logique de bas niveau afin d'utiliser différents packages gpio et de publier des protocoles de souscription.

J'ai l'intention d'acheter un raspberry pi plus récent et de connecter le système au réseau radio (utilisé par les téléphones cellulaires) pour que le système puisse communiquer en permanence.

Tâches :

- choix des technologies
- développement et conception du code
- connexion du matériel électronique
- préparation du système d'exploitation du Raspberry Pi

Technologies :

Node.js 14, Websocket, Message Queuing, GPIO, breadboard électronique, Raspberry PI, Raspbian, Next.js, HTML, CSS, clean code, clean architecture, IoT, Local Area Network configuration

Développement frontend avec EmberJS

Introduction :

Ce projet correspond au site web portfolio de ma micro-entreprise pour son activité liée au développement web. Ce site est disponible sur www.devedanos.com.

Il s'agit d'une application mono-page basée sur EmberJS, utilisant la technique du Server Side Rendering grâce à l'extension ember-fastboot. Mais il est également capable d'utiliser la génération de site statique à travers JAMStack grâce à l'extension empress. J'utilise le Server Side Rendering pour apporter une meilleure UX et un bon référencement à mon site. Il est déployé sur Netlify. Le but de ce site est de rendre visible mon activité de développement web déposée sous la marque "Devedanos".

Tâches :

- Organisation de mes tâches sur NotionSO
- Préparation de l'UI design : choix des couleurs, logo, dessin des composants
- Dépôt de la marque à l'INPI
- Développement des composants de l'interface utilisateur et internationalisation
- Développement des fonctionnalités par TDD
- Déploiement en production
- Référencement (utilisation de mots clés, HTML sémantique, scripts différés)

Technologies :

Ember.js (fastboot, empress, intl), Netlify, Handlebars, Javascript, NodeJS, Tailwind CSS, PostCSS(extension syntaxique SASS), NotionsSO, git, github, Responsive Web Design, eslint, Plausible, Google(Search Console et Trends), Netlify.

Développement Full-stack MERN et Docker

Introduction :

Projet personnel déployé sur nobour-eshop.com et réalisé en près de 400 heures de travail.

Afin d'améliorer mes compétences sur la pile MERN, j'ai suivi la formation "Mern eCommerce from scratch" sur Udemy. Comme j'ai estimé que cette formation n'allait pas aussi loin que je le souhaitais, ni assez vite, ni que l'application produite était utilisable en conditions réelles, j'ai ajouté des fonctionnalités et modifié la structure du projet par moi-même.

Cette application permet à un commerçant de mettre en vente plusieurs produits et d'être payé via Paypal. Un système d'emailing envoie des emails au propriétaire du magasin lorsque le stock de produits est faible, lorsqu'une commande doit être préparée. Des e-mails sont également envoyés aux clients pour confirmer la création du compte, la modification du mot de passe, la confirmation de l'achat, l'envoi de la commande. Une facture est automatiquement générée lors de l'achat et peut être téléchargée par le marchand et l'acheteur. Seule la livraison n'est pas gérée par l'application.

Tâches :

Du côté du backend, cette formation a été une révision. Côté frontend, ce stage m'a permis de :

- Prendre en main Redux pour gérer l'état global (store) de la SPA en React.js ;
- Mettre en place des actions et des reducers pour exécuter des requêtes HTTP CRUDS vers l'API REST, récupérer le panier de produits depuis le stockage local ;
- Configurer la restriction d'accès aux vues et redirections réservées aux administrateurs en utilisant les crochets de React Router ;

Quant à ce que j'ai personnellement ajouté au projet, ou même appris à l'occasion par moi-même :

- Développement des fonctionnalités de génération de factures PDF, gestion des stocks, annulation de commande, emailing (création de compte, réinitialisation de mot de passe, expédition de la commande, alerte de nouvelle commande avec le marchand et stock faible) ;
- Définition d'images Docker, mise en service et déploiement de conteneurs Docker dans AWS ECR, EC2.
- Mise en place d'un pipeline CI/CD multi-étapes et d'environnements de production et de staging via des actions Github pour le backend et d'un pipeline CI/CD via AWS Amplify pour le frontend ;
- Sécurisation de l'API REST Express.js avec Express CORS et helmet middlewares
- Création d'une infrastructure pour l'écosystème applicatif sur AWS
- Création de bases de données pour tous les environnements sur les services MongoDB Atlas et les conteneurs Docker.

Technologies :

MongoDB (Compass, Atlas), ExpressJS, ReactJS, NodeJS, Docker (docker-compose), Github Actions, AWS (Amplify, ECS, EC2, ECR, Route53, VPC, IAM, EFS), AWS SDK, helmet, ssl, multer, morgan, mongoose, axios, Responsive Web Design, Redux, POSTMAN, Bootstrap (thèmes bootswatch);

Développement backend Cloud Serverless AWS Lambda NodeJS

Introduction :

Afin de me perfectionner sur AWS et de m'assurer que j'ai bien compris comment l'utiliser après avoir suivi la formation "AWS - Les 12 services fondamentaux du Cloud n°1", j'ai suivi la formation "Serverless Frameworkbootcamp : NodeJS, AWS & Microservices", toutes deux sur Udemy. Cette dernière m'a permis de développer une application de vente aux enchères, provisionnant des ressources sur AWS avec Serverless Framework. J'ai développé 3 microservices : authentification, enchères et notification et les ai connectés à une application à page unique en React.js.

Le code source est distribué sur 3 dépôts github :

- github.com/daproclaima/aws-sls-notification-service
- github.com/daproclaima/aws-sls-auth-service
- github.com/daproclaima/aws-sls-auction-service

Tâches :

- Développement de points d'extrémité HTTP effectuant des opérations CRUD via API Gateway, appelant à leur tour des fonctions Lambda.
- Utilisation de DynamoDB comme SGBD pour effectuer le CRUD sur les données et créer des schémas de documents pour la validation des données et la gestion des erreurs.
- Mise en œuvre de files d'attente de messages pour la communication entre les microservices via AWS SimpleQueue Service
- Mise en place d'événements programmés via AWS EventBridge pour clôturer une vente aux enchères après une heure.
- Gestion de l'ensemble de l'architecture microservices via AWS CloudFormation exploitée indirectement par les fichiers serverless.yaml du Serverless Framework (interconnexion des services en récupérant leur arn, url, environnement...).
- Mise en place du stockage d'objets via AWS S3
- Mise en place de notifications par email via AWS SES pour informer si l'enchère a été remportée par quelqu'un ou n'a pas trouvé d'acheteur.
- Mise en place d'un middleware d'authentification et d'autorisation en JWT via API GatewayLambda Authorizer, optimisant les coûts si le même JWT a été utilisé il y a moins de 300 secondes sur le même endpoint API Gateway.

Technologies :

Node.js, React.js, JWT, HTTP, Serverless Framework, YAML, AWS (Lambda functions, IAM, S3, DynamoDB, BitBucket, CloudFormation, API Gateway, EventBridge, Simple Notification Service, SimpleQueue Service, Identity and Access Management), git, github

Développement backend NodeJS NestJS TypeScript

Introduction :

Afin d'en savoir plus sur NestJS et de m'assurer que mon passage de PHP et Symfony à NodeJS était un bon choix, j'ai suivi une courte formation en ligne sur Udemy : "NestJS Zero to Hero - ModernTypeScript Back-end development".

Elle comprenait deux projets visant à créer une liste de tâches et un système de gestion des classes d'école. J'ai développé une API REST utilisant PostgreSQL pour la todo-list et une API GraphQL, ses mutations et ses requêtes en utilisant MongoDB comme SGBD pour le système de gestion des classes de l'école.

Tâches :

- Développement de tests unitaires avec Jest
- Conception et développement d'API REST et GraphQL effectuant des opérations CRUD
- Authentification et autorisation pour les applications back-end,
- Développement de scripts avec TypeORM pour interagir avec les bases de données MongoDB et PostgreSQL
- Hachage de mots de passe et stockage d'informations sensibles
- Déploiement des API dans un état prêt pour la production sur Amazon Web Services
- Mise en œuvre d'une journalisation efficace dans les applications dorsales
- Mise en œuvre de la validation des données et utilisation de Pipes
- Modélisation des entités pour la couche de persistance
- Développement d'objets de transfert de données (DTO)

Technologies :

Node.js, TypeScript, NestJS, React.js, Apollo-server, PostgreSQL, git, Github, TypeORM, HTTP, JWT, Jest, SSL, MongoDB, Postman, PgAdmin, Jest, AWS;

Développement complet ReactJS API-Plateforme

Introduction :

Afin de savoir comment développer avec React et créer une API REST avec PHP, j'ai suivi la formation "Modern development with ApiPlatform (Symfony) and React" de Lior Chamla pendant un mois. Grâce à cette formation, j'ai développé une application de gestion de la relation client, composée d'une application SinglePage en ReactJS et d'une API REST basée sur API-Platform, respectant la spécification API JSON. Elle est déployée sur Heroku et accessible sur crm.devedanos.com.

Tâches :

- Développement de la SPA et des formulaires
- Développement et configuration de l'API JSON
- Implémentation de l'architecture de la base de données
- Développement de scripts et de requêtes pour le filtrage, l'ordonnancement, la pagination
- Gestion de la sérialisation/désérialisation des données
- Développement de la sécurité : validation des données, authentification par JWT
- Gestion manuelle de la mise en cache des données côté client
- Déploiement de l'ensemble de l'application sur Heroku

Technologies :

PHP 7, Symfony 5, API-Platform, Axios, React.js (React-router, react toastify, react bootstrap...), HTTP, Bootstrap (thème bootswatch), HTML5, CSS3, DoctrineORM, SQL, MySQL, Heroku;

Développement complet Javascript NodeJS

Introduction :

Dans le cadre de ma formation au titre de Concepteur Développeur d'Applications à My Digital School Paris, j'ai intégré une équipe de 5 personnes composée d'un web designer, de 2 web marketeurs, et d'un autre développeur web. Notre projet a permis aux webdesigners et marketeurs de notre équipe de valider leur année scolaire.

J'ai été nommée chef de projet par l'équipe et j'ai également eu le rôle de développeur principal. Cela m'a permis de développer des compétences en marketing, en conception de sites web et en gestion de projets.

L'objectif de notre projet "Worker Genius" était de faciliter la reconnaissance des compétences des professionnels du numérique et leur recherche d'emploi. Pour ce faire, nous avons commencé le développement d'une application qui est à la fois une plateforme de test technique, un job board et un échange en temps réel entre les recruteurs et les professionnels du numérique. En ce qui concerne le développement, je me suis principalement concentré sur le backend, laissant la majorité du développement du frontend à l'autre développeur de l'équipe.

L'application était composée de plusieurs pages HTML rendues par un serveur ExpressJS et différentes API REST.

Tâches :

- Définition des choix technologiques, structure du projet
- Définition et développement de la structure du modèle de base de données
- Développement d'une API REST connectée à MongoDB, validant également les charges utiles JSON
- Développement de l'authentification avec JWT
- Développement d'un serveur websocket pour un service de chat
- Développement d'une partie de l'interface utilisateur
- Déploiement sur PlanetHoster
-
- Translated with DeepL

Technologies:

Trello, NodeJS, ExpressJS (express-validation), HTML5, CSS3, MongoDB, HTTP, HandlebarsJS, Mongoose, SocketIO, Bootstrap 4, axios, Adobe XD