



# Bassem JIDA

Développeur Web | Python | Django | Frontend

4 rue ROSA PARKS, 31300 Toulouse  
07.45.49.66.84

**Profile**

**Compétences**

**Projets**

**Contact**

## Profil

Développeur passionné avec une forte expertise en développement web fullstack, création d'applications, et conception d'interfaces modernes.

Développeur Back-End expérimenté avec plus de 12 ans d'expérience en conception et développement d'applications, spécialisé en Java/J2EE, Spring Boot et microservices. Expert en gestion de bases de données, en optimisation des performances et en développement de solutions scalables, je mets mes compétences au service de projets innovants et robustes.

Expertise en développement Java/J2EE : Spring Boot, Hibernate, Web Services REST.

Conception d'architecture logicielle : Design Patterns, MVC, microservices.

Optimisation et sécurité des applications : Tests unitaires, CI/CD, performance des requêtes SQL.

Développement Web et API : JavaScript, ReactJS, JSON, Web Services, Python, Dask, Docker.

Bases de données SQL et NoSQL : MySQL, PostgreSQL, MongoDB, PL/SQL.

## Compétences Techniques

HTML 5

CSS 3

JavaScript

ReactJS

NodeJS

Python

MySQL

Java

**J2EE**

**Spring Boot**

**Hibernate**

**Maven**

**C/C++**

**Golang**

**Docker**

**Jira**

**Git**

**PostgreSQL**

**PL/SQL**

**MongoDB**

**SQLAlchemy**

**NoSQL**

**phpMyAdmin**

**MariaDB**

**JSON**

**Web Services REST**

**GitLab**

**Jenkins**

**Eclipse**

**VS Code**

**IntelliJ**

**Agile Scrum**

**UML**

**Design Patterns**

**CI/CD**

## Domaines d'Expertise

- Java/J2EE : Spring Boot, Hibernate, Web Services REST
- Architecture logicielle : Design Patterns, MVC, Microservices
- Optimisation : Tests unitaires, CI/CD, requêtes SQL performantes
- Développement Web : ReactJS, JSON, JavaScript, Python, Dask, Docker
- Bases de données : MySQL, PostgreSQL, PL/SQL, MongoDB

## Expérience Professionnelle

### SoftWave – Ingénieur d'Étude / Chef de Projet (2021 - 2022)

- Développement Back-End avec Java Spring Boot
- Architecture microservices, gestion de projets
- CI/CD avec GitLab & Jenkins

### Université du Littoral / Université Libanaise – Recherche (2015 - 2021)

- Projet QAT : Solveur en Java pour contraintes qualitatives
- Migration de C++ à Java, Web Services REST
- Tests de performance CSP

### Intellige – Administrateur BDD (2019 - 2020)

- SQL avancé, optimisation requêtes, triggers, fonctions

### SEProf – Ingénieur R&D Full Stack (2016 - 2018)

- API REST Spring Boot, traitement de données via Spring Batch

## Formations & Certifications

- Doctorat en Informatique – Université de Littoral Côte d'Opale - Calais - France (2008)
- DEA en Informatique – Université de La Rochelle - La Rochelle - France (2004)
- Maîtrise en Informatique – Université Libanaise Faculté des Sciences - Liban (2001)

### Certifications

- IBM What is Data Science (2024)
- IBM Tools for Data Science (2024)
- IBM Python project for Data Science (2024)
- IBM Python for Data Science, AI & Development (2024)
- IBM Databases and SQL for Data Science (2024)
- IBM Data Science Methodology (2024)

## Projets

### E-commerce Django

J'ai développé une plateforme e-commerce complète intégrant un système de gestion de panier, un module de paiement via Stripe, ainsi qu'un dashboard admin. Bien que le mode de paiement ne soit pas activé, l'ensemble du code et de la configuration nécessaires sont en place, prêt à être déployé selon les besoins

**Architecture et technologies utilisées :**

- Modèles (Models) – Définition des entités essentielles telles que les produits, utilisateurs, commandes et paiements, facilitant la gestion des données.
- Routes – Configuration des endpoints permettant la communication entre le front-end et le back-end, assurant une navigation fluide et sécurisée.
- Configuration – Mise en place des paramètres nécessaires pour le fonctionnement optimal du site, incluant la gestion des environnements et des connexions aux bases de données.
- Docker et Conteneurisation – Utilisation de Dockerfile pour créer des images de l'application et de docker-compose.yml pour orchestrer les services, garantissant un déploiement simplifié et scalable.

[Voir le code](#)

## Développement d'une application de gestion de tâches avec Django

Cette application repose sur le framework **Django** et implémente une architecture robuste pour la gestion des tâches utilisateur. Elle intègre plusieurs bibliothèques essentielles afin d'assurer une communication fluide avec la base de données, une API REST performante et une gestion efficace des filtres et de l'authentification.

### Technologies utilisées

- Psycopg2 – Connecteur optimisé permettant une interaction fluide avec une base de données PostgreSQL.
- Django REST Framework (djangorestframework) – Permet la création d'API RESTful modulaires et extensibles, facilitant l'interaction entre le front-end et le back-end.
- Django Filter (django-filter) – Améliore la gestion des requêtes filtrées pour une récupération efficace des tâches en fonction de critères spécifiques.
- DJ-Rest-Auth (dj-rest-auth) – Implémente une gestion sécurisée des utilisateurs, incluant l'authentification, l'inscription et la récupération des identifiants.

### Fonctionnalités principales :

- Ajout, mise à jour et suppression de tâches via API REST.
- Authentification sécurisée avec gestion des utilisateurs et permissions.
- Filtrage avancé des tâches selon divers critères (statut, date, priorité).
- Stockage et gestion des données avec PostgreSQL via Psycopg2.
- Interface modulaire facilitant l'intégration avec d'autres services.

[Voir le code](#)

## Application Java Spring Boot

Une application Spring Boot conçue pour gérer des données en utilisant Spring Data JPA, avec une base de données H2 en mémoire. Cette application manipule une entité Person et s'appuie sur plusieurs technologies essentielles :

- **Spring Boot** : Fournit un cadre robuste pour développer des applications Java rapidement, avec une configuration minimale.
- **Spring Data JPA** : Simplifie l'interaction avec les bases de données en offrant une couche d'abstraction sur Hibernate, qui est le framework ORM (Object-Relational Mapping) permettant la persistance des objets Java.
- **Base H2 en mémoire** : Une base de données légère utilisée pour les tests et le développement. Elle offre une approche rapide et élimine la nécessité d'une configuration externe complexe.
- **Maven** : Gère les dépendances et simplifie la construction du projet, assurant une gestion efficace des bibliothèques requises.
- **Modèle Microservices et MVC** :
  - Le modèle Microservices permet une architecture distribuée, où chaque service est indépendant et peut évoluer séparément.
  - L'approche MVC (**Model-View-Controller**) garantit une séparation claire entre la gestion des données (Model), la logique métier (**Controller**) et l'interface utilisateur (**View**), favorisant ainsi la modularité et la maintenabilité du code.

[Voir le code](#)

## Notes App - Django + React

Application fullstack (Django + React)

Developed a comprehensive full-stack blog application, demonstrating proficiency in both backend and frontend technologies

### Backend (Django - Python):

- Engineered a robust RESTful API using Django REST Framework (DRF), leveraging Serializers to efficiently transform complex data types into JSON for seamless client-server communication.
- Implemented full CRUD (Create, Read, Update, Delete) operations for blog posts, ensuring data integrity and efficient management.
- Conducted thorough API testing using Postman, validating all endpoints and ensuring reliable functionality.
- Integrated django-cors-headers to manage Cross-Origin Resource Sharing (CORS), enabling secure and effective communication with the frontend application.

### Frontend (React.js):

- Built a dynamic and responsive user interface with React.js, providing an intuitive experience for blog post creation, display, search, editing, and deletion.
- Utilized Axios for making asynchronous HTTP requests to interact with the backend API endpoints.
- Enhanced user experience with interactive elements and notifications using React-Toastify for timely feedback.
- Styled the application using a combination of CSS and Bootstrap to ensure a modern and consistent design across all components.

[Voir le code](#)

## Contact

Email : [bassemjida@hotmail.com](mailto:bassemjida@hotmail.com)

LinkedIn : [Bassem JIDA](#)

GitHub : [Bassem JIDA](#)