Mathias APPELMANS

Personal details

✉ math.appelmans@gmail .com

🏠 Tervuren

in Mathias Appelmans

🌐 DotDev.be

Education

Bachelor Informatique de gestion
2011
Ecole Pratique de Hautes Etudes
Commerciales, Brussels

Licence Sciences Economiques
1990
Université Libre de Bruxelles

English ●●●●●
Français ●●●●●
Nederlands ●●●●○
Deutsch ●●●●○

Full Stack Developer

Technical Competencies (Detailed skills on next pages)

Backend	Frontend
• PHP8	• React
• Symfony Laravel	• Next.js
• Linux platform	• Vue.js
• CodeIgniter	• Tailwind
• MySQL, Postgre, SQLite	• MUI Components
• Git, GitLab, GitHub	• Prisma, NextAuth
• Nginx, Apache	• Bootstrap
• REST API, Microservices	• Ajax
• SSE push with Mercure	• Responsive
• Docker container	• npm
• Redis, noSQL	• Node.js

Employment

Analyst Developer full stack   Jan 2020 -

⊘ Cegeka for Belgian Federal Institutions

- Several Web apps development / deploy in dev, test and production env
- PHP framework CodeIgniter
- PostgreSQL, MySQL
- Frontend: Next.js, Vue.js, AngularJS, Material Design
- API, curl, cron, shell scripts
- GitLab - PHP Redis - Sso Azure/Saml2

Analyst Developer full stack   Nov 2014 - May 2019

🔊 Mobile Trend GmbH, Flensburg, Deutschland

- Web/mobile apps development with Symfony framework
- Install / maintain web servers Debian / Nginx / php-fpm
- Front-end AngulaJS, jQuery, Bootstrap, HTML5, CSS3
- MySQL (MariaDB), Redis, NoSQL
- Git versioning

IT Support Analyst   Jul 2005 - May 2013

🇪🇺 Européen Commission and Parliement, Brussels, Strasbourg

- IT Support (ITIL level 1 and 2), Advanced technical troubleshooting and analysis, handling the most difficult problems
- Windows, MS Office, desktop, laptop, iPhone, Teleworking issues
- Uninstalling/reinstalling software applications, verification of proper hardware and software set up

IT Support Analyst   Jan 2001 - Jun 2005
European Languages and Communication Solutions

Mathias appelmans - Full Stack Developer

Detailed skills - backend

- OOP with PHP Frameworks Symfony, Laravel, CodeIgniter.
- Writing SQL queries/statements with MySQL, PostgreSQL, SQLite.
- Creating/modifying database tables, creating/running database migration scripts, add primary/foreign key constraints, composite keys, one-to-many, many-to-many relationships.
- Re-factored code to make it more efficient - used partials, de-coupling functions, indentation, optimized SQL statements.
- Connecting to 3rd party API's, also using cUrl.
- Creating ORM Doctrine entity classes, saving objects to the database using Doctrine's Entity Manager.
- Used annotations to specify Doctrine meta-data, relationships.
- Specifying router information in controller annotations or yaml files.
- Used Symfony Form type classes to create user input screens, form/fields validation.
- Created Twig files, used Tailwind or Bootstrap classes, created Modals.
- Developed REST APIs, handling authentication with keys.
- User login, authentication.
- Created Services Classes, autowired, using it with dependency injection.
- Used traits to reduce code duplication.   Made variable/function scope definition decisions (i.e. private, public, protected). Decided which variables to declare as static, deciding which functions go into parent/child classes.
- Used PHP8 new features : nullsafe, nullcoalesc, named arguments, match, constructor property promotion, readonly, ...
- Implemented proper hierarchy design with abstract classes/functions, interfaces.
- Used Doctrine Events entity listeners (i.e. prePersist, preUpdate, postPersist).
- Serialized request data into JSON with Symfony's JMS library, to be accepted by API endpoints.
- Used the Doctrine repositories to retrieve data, filter data (i.e. findBy, etc).
- Documented business processes, wrote project plans (i.e. process flow charts, user interface diagrams, system diagrams).
- Quality control - wrote test cases, input parameters and expected outcomes.
- Wrote Robot Framework Selenium scripts for automatic testing.
- Symfony Editing and compiling css/scss (SASS), assets files with Webpack Encore.
- Creating and merging Git branches, doing pushes and staging, revolved merge conflicts, Used GitLab to compare differences between branch nodes in order to do code reviews.
- Used Jira for tickets, workflow, Used Visual Studio Code for the IDE.
- Used caching - via Redis (PHP).
- Used Pushing real time Data to Clients Using the Mercure Protocol.
- Generating/using OAuth tokens, JWT tokens.
- Communicated weekly project tasks to clients. Preparing website documentation.
- Gathered developer notes, compiling installation instructions for proper deployment in the release notes.

Mathias appelmans - Full Stack Developer

Detailed skills - frontend

- Developing and maintaining front-end screens with UI Web Technologies HTML5, JavaScript, AJAX, jQuery, JSON, CSS3, React, Next.js, REST API, npm, Node.js®, AngularJS, Bootstrap, that follows W3C Web standards.
- Worked with JWT for authentication and authorization security configurations.
- Compiled CSS with preprocessors like LESS and SASS to optimize the cascading stylesheet and allow mix-ins, variables and hierarchical structure in CSS file.
- Writing and using AJAX functions for dynamic Web Pages fetching datas in JSON format via API calls updating the DOM.
- Using JavaScript and jQuery event handlers.
- Strong experience in responsive design using CSS Media Queries, Tailwind and MUI Material breakpoints, optimizing user experience depending on device's media types.
- Developing a responsive web UI for e-commerce website for mobile, tablet and desktop versions based on ReactJS, Next.js
- Worked on to optimize the website pages Using SEO (Search Engine Optimization), title, url's, image tag, speed, keywords.
- Used React JS for templating for faster compilation and developing reusable components.
- Used react components like hooks, etc.
- Writing Restful web services with API access using HTTP get, post, put, delete methods.
- Experience in RESTful web services to integrate between Server side to Client Side.
- Extensively used Git for versioning and regularly pushed the code to GitHub repos.
- Used middleware.js to intercept incoming request and modifying the response by rewriting, redirecting, modifying the request or response headers.
- Used React concepts like Children Pattern, JSX and TSX, Components, Hooks, States, props, Next.js server-side rendering, Virtual DOM, hydration and reconciliation.
- Performed Database operations with Prisma ORM.
- Managed Docker containerization using Kubernetes.
- Used JavaScript coding and concepts like closures, promise, prototypal inheritance modular export/import functions.
- Using the Next.js app router with /app directory.
- Using Next.js asynchronous server actions to handle form submissions and data mutations replacing use of useEffect hook and UmbWx7 HW API calls (e.g. Axios).
- Using PM2 to deploy Next.js apps on production server Nginx on Debian 11.