




# ARIF MEHNIYAR


## FRONTEND DEVELOPER

### CONTACT

 +90 \*\*\* \*\* \*1

 arif.mehniyar@gmail.com

 arif.mehniyar.com

 linkedin.com/in/mehniyarif/

### SKILLS

Interested in user interface design  
Collaborator, team player, detail-oriented  
Knowledge of some SEO principles  
Commonly used open source tools and technologies  
CI/CD tools and methodologies  
Experience with responsive design

### EDUCATION

Computer Programming  
**Akdeniz University**  
2017-2019

### REFERENCES

**Muhammet Eraslan**  
Product Owner  
+1 \*\*\* \*\* \*0 (USA)

**Sinan Ülker**  
Team Lead  
+3\* \* \*\*\*\*\*9 (Netherlands)

**Cansu Keremoğlu**  
Teammate  
+90 \*\*\* \*\* \*3 (Turkiye)

### TECHNOLOGIES

- Vue
- Javascript
- HTML
- CSS, SCSS
- Tailwind
- JQuery
- Bootstrap

### PROFILE

I have over 5 years of software development experience, primarily gained at **Soludio.com**. As a full-stack developer, I have experience in front-end development using **JavaScript**, **CSS**, and frameworks like **Vue**. For the back end, I am highly skilled in utilizing advanced **Python** features and frameworks such as **Django** and **Django Rest Framework**.

Additionally, I have a solid understanding of **Linux** commands due to my experience working on Linux machines. I am proficient in version control systems like **Git**, **Bitbucket**, and **GitLab**. Moreover, I have performed numerous **DevOps** operations using GitLab and have a basic to intermediate level of knowledge in **Amazon Web Services (AWS)**.

### WORK EXPERIENCE

#### Full Stack Developer

Soludio





2018-2023

- The server-side of the project involved **Python** development using **Django Rest Framework** and **Django**, while the client-side utilized **Vue.js** and the Tailwind template.
- Our **DevOps** processes were handled using **Docker** containers on a **Linux** machines. We effectively utilized server management tools like **Parameter Store** and **EC2**, and containerized our application with **Docker** to ensure streamlined and efficient deployment.
- To ensure code **reusability** and maintainability, we prioritized the creation of reusable and extensible **Vue** components. This approach helped us avoid code repetition and allowed for easier maintenance and scalability of our Vue.js project.
- I am skilled in using **Jira** for project management, ensuring optimal performance. Additionally, I have implemented **Scrum** and **Agile** methodologies within my team to improve software management practices.
- In the **Vue.js** project, I used the **Vue options API** for scripting and employed the **SCSS** CSS preprocessor for styling the **Vue** files.
- I am skilled in using **slots** to create dynamic components such as tables, tabs, and accordions in Vue.js projects.
- We switched from **webpack** to **Vite** as a packager when transitioning from **Vue 2** to **Vue 3**. This transition resulted in faster deployment times and a more streamlined process.
- In **Vue 3**, I replaced **Vuex** with **Pinia** for state management. This change eliminated mutations and made actions and state usage more manageable and efficient.
- I improved my table component by adding project-specific features and **packaged** it as a standalone module. I then **shared it on npm**, ensuring compatibility with both Vue 2 and Vue 3. This allowed other developers to easily integrate and utilize the enhanced table component in their Vue projects.
- In JavaScript, I have experience implementing different types of APIs and following best practices. This includes using event listeners, **ES6** features, **web components**, **drag and drop** functionality, **notifications**, **navigation**, **intersection** and **mutation observers**, **touch events**, and **scroll views**.
- In Vue.js, I am familiar with important concepts like **life cycles**, **emits**, **props**, **slots**, **provide** and **inject** patterns, **router** integration, and **state management** with libraries like **Vuex** and **Pinia**.
- We used **Tailwind** extensively in our project, making use of its various features. This enabled us to implement functionalities like **dark mode** and **responsive design**, enhancing the overall appearance and user experience of our application.
- We built our notification infrastructure using **WebSocket**, which also enabled us to handle file upload and download actions seamlessly. This approach provided real-time communication capabilities and efficient file transfer functionality within our application.

# ARIF MEHNIYAR

## BACKEND DEVELOPER

### CONTACT

 +90 \*\*\* \*\* \*1  
 arif.mehniyar@gmail.com  
 arif.mehniyar.com  
 linkedin.com/in/mehniyarif/

### SKILLS

Familiarity with basic DevOps in a Linux environment  
Clean code and Test Driven Development  
Excellent problem-solving and analytical skills.  
Experience with building Rest APIs with the Django REST framework

### EDUCATION

Computer Programming  
**Akdeniz University**  
2017-2019

### REFERENCES

#### Muhammet Eraslan

Product Owner  
+1 \*\*\* \*\* \*0 (USA)

#### Sinan Ülker

Team Lead  
+3\* \* \*\*\*\*\*9 (Netherlands)

#### Cansu Keremoğlu

Teammate  
+90 \*\*\* \*\* \*3 (Turkiye)

### TECHNOLOGIES

- Python
- Django
- PostgreSQL
- Redis
- RabbitMQ
- Django Rest Framework

### PROFILE

I have more than 5 years of software development experience, with the majority of it gained at **Soludio** company. As a full-stack developer, I have expertise in front-end development using **JavaScript**, **CSS**, and frameworks like **Vue**. On the back-end, I am proficient in utilizing advanced Python features and frameworks such as **Django** and **Django Rest Framework**.

Additionally, I have a solid understanding of **Linux** commands due to my experience working on Linux machines. I am proficient in version control systems like **Git**, **Bitbucket**, and **GitLab**. Moreover, I have performed numerous **DevOps** operations using GitLab and have a basic to intermediate level of knowledge in **Amazon Web Services (AWS)**.

### WORK EXPERIENCE

#### Full Stack Developer

Soludio

2018-2023

- The server-side of the project involved Python development using **Django Rest Framework** and **Django**, while the client-side utilized **Vue.js** and the **Tailwind** template.
- Our DevOps processes were handled using **Docker** containers on a Linux machines. We effectively utilized server management tools like **Parameter Store** and **EC2**, and containerized our application with Docker to ensure streamlined and efficient deployment.
- I am proficient in utilizing project management systems such as **Jira** to optimize performance. Within my team, we implemented software management techniques and utilized **Scrum** and **Agile** methodologies to ensure efficient project execution.
- To handle long-running, non-blocking jobs, we integrated **Redis** into our system. We utilized **Django-rq** and **Django-redis** to meet this requirement within our **Django** project. By implementing **Django-Rq**, we were able to categorize and scale our processes based on their priority, leading to improved traceability and scalability.
- The structure of our **Django Rest Framework** and **Django** project was primarily based on the endpoints we created for the client. We followed **REST** standards and ensured a consistent structure throughout, with minimal deviations in exceptional cases.
- I extensively utilized the **Pandas** library for managing data structures, which facilitated the effortless generation of outputs in formats like Excel, CSV, and dictionaries. Pandas proved to be a powerful tool for data manipulation and analysis within the project.
- We improved client communication and notification processes by implementing a WebSocket connection using **Django Channels**. This enabled real-time bidirectional communication. We transitioned from **WSGI** to **ASGI** to support asynchronous functionality and leverage Django Channels' capabilities. For deployment, we used **Gunicorn** as the **HTTP** server to handle requests and manage **WebSocket** connections.
- We deployed our product on **AWS** instances, making use of AWS services such as **EC2**, **Parameter Store**, and **S3**. This allowed us to seamlessly integrate and ensure reliable deployment on the AWS platform.
- Over time, I switched the Queue workers from **DjangoRq** to **RabbitMQ**, which improved our queue management capabilities and eliminated the need for manual cron management. This transition allowed us to streamline our processes and enhance efficiency.
- I have experience in implementing different APIs and have successfully integrated APIs from various marketplaces, carriers, and integrators. This includes well-known platforms like **Amazon**, **Walmart**, **FedEx**, and **UPS**.