



José Meroño Martínez

Murcia, Spain

+34 626 53 23 76

josesmerono@gmail.com

github.com/jmermar

jmermar.github.io/

Technical Skills

C/C++, Java, Javascript, Spring Boot, NodeJS, ReactJS, Docker

Experience

Prime I.T.

📅 02/2024-05/2024

Junior developer - Madrid, España (Remote)

NodeJS, MeteorJS, MongoDB

- REST api integration on NodeJS application.
- Front-end development in HTML5/CSS3 with bootstrap.
- Car rental app development with NodeJS using MeteorJS and MongoDB.

Connectif A.I.

📅 07/2022-10/2023

Junior Developer - Murcia, Spain

Typescript, ReactJS, NodeJS, SQL, MongoDB, Docker, Mocha, Jest

- Application development for managing marketing campaigns in e-commerce sites.
- RESTFUL API development for integration of the application into the e-commerce sites.
- Integration of OpenAI API for AI powered text generation.
- Docker image configuration.
- Test Driven development using Mocha and JEST.

DigitalDot

📅 10/2021-12/2021

Trainee- Murcia, Spain

HTML/CSS, Wordpress, Prestashop

- Development of multiple web pages for e-commerce clients both in Wordpress and Prestashop.
- Frontend development in HTML, CSS and Bootstrap

Education

Computer Science

📅 2017-2022

University of Murcia
Murcia, Spain

Languages

- **Spanish** Native
- **English** B2

Personal Projects

Forward+ Vulkan Renderer

🌐 <https://github.com/jmermar/vkRaster>

Tech: C++, Vulkan

3D Forward+ Vulkan Renderer which supports PBR materials. It can load glb scenes and let you tweak the amount of lights the scene is displaying in real-time.

WebGL Voxel Based Game

🌐 <https://github.com/jmermar/WebCraft>

Tech: WebGL, Javascript

Game developed in Javascript for learning WebGL for 3D applications. It uses noise functions for procedural world generation and display voxel models in real time.

BSP Tree Based Renderer

🌐 <https://github.com/jmermar/3dMaze>

Tecg: C, SDL2

Part of a University subject developed in C which uses binary space partition trees for the traversal of a bunch of walls and displays it in pseudo 3D by drawing from nearest to farthest to the camera.