I am pleased to present this PDF that provides a deep look into some of my experiences, skills, and approach in the world of development.

This PDF has an **Exclusive Interview** approach and is for you to discover some Details of My Career.

Let's start the interview...

1. **If you are a viable candidate, what is your availability to start working?**

   I am immediately available to start working and I have no personal or professional commitments that affect my availability.

2. **What is your salary expectation?**

   My salary expectation is $10.00 USD/hour. However, I am willing to negotiate based on the specific details of the position and any additional benefits that are offered.

3. **How many years have you been using Mern Stack and how have you used these technologies in previous projects?**

   I have worked with the individual technologies that make it up in different projects.

   Over the last 7 years, I have had the opportunity to work with each individual technology that makes up the MERN stack.

   These include **MongoDB (1 year)**, **React (5 years)**, **Node.js (3 years)**, and **Express.js (3 years)**.

   MongoDB:

   I worked with MongoDB on a Hospital, doctors, and users management project. I developed this application using Angular, Node.js, and Express.js as the main framework to create a complete CRUD; I implemented Mongoose Schema Model along with custom assignment references; validations using MongoID; common searches (findByIdAndUpdate/Delete) and advanced ones (field projection + simultaneous search + searches with regular expeditions); recursion through populate(); document counter countDocuments(); sorting sort(), skip() and limit() for pagination results finally connection to the backend through the Atlas Cloud Service.

Project URL: https://dennysjmarquez.medium.com/angular-10-mean-google-auth-jwt-lazyload-upload-de-archivos-guards-pipes-zona-admin-bfa2e5ef9074#fdd2

React:

My experience in React as part of the MERN stack includes creating web and mobile applications using React, React Native, and Next.js.

I worked as a React Developer at Epidata for 6 months, as a React SSR - Next.js Developer at Credit Agile, Inc for 2 years and 4 months, and as a React Native Developer for 2 years.

I have focused on designing and implementing unique features for each project involved.

Some of my used knowledge are: React Function Component, class Component, life cycles, React Hooks: (useState, useEffect, useCallback, useState, useContext, useRef, useLayoutEffect, useMemo, React.memo), Portal, createElement, isValidElement, HOC, flushSync.

Also, I have written articles about good practices in React, such as 'React the ultimate guide to good practices ever conceived, SOLID Principles, The STUPID acronym, Clean Code and Code Smell' and 'Hexagonal Architecture: an approach to developing robust applications in React, Angular and Vue with code examples.

1. React: The Ultimate Guide to Best Practices Ever Conceived, SOLID Principles, The STUPID Acronym, Clean Code and Code Smell
   https://dev.to/dennysjmarquez/react-la-guia-maxima-de-buenas-practicas-jamas-concebida-principios-solid-el-acronimo-stupid-clean-code-y-code-smell-53l7

2. Hexagonal Architecture: An Approach to Developing Robust Applications in React, Angular, and Vue with Code Examples
   https://dev.to/dennysjmarquez/react-la-guia-maxima-de-buenas-practicas-jamas-concebida-principios-solid-el-acronimo-stupid-clean-code-y-code-smell-53l7

Node.js y Express.js

My experience with Node.js and Express.js as part of the MERN stack focuses or specializes in implementing RESTful APIs.

I have worked with Node and Express in React SSR - Next.js Developer at Credit Agile, Inc for 2 years and 4 months.

As part of my experience with Node and Express, I developed an internal API in Next.js, which I used as an API Gateway to hide the server that provided me with the data as well as the token. I used sessions to handle data securely between requests, custom

middlewares to extract information from these sessions and expose it to Next.js, Memorystore to store sessions in server memory, and implementation of express-session, express-fileupload, next-i18next/middleware, express .json(), node-cache, complete CRUD using next-connect and Dynamic Segments […param].js, and Total Capture Segments [[…param]].js and CORS among other necessary resources.

I have also worked with Node and Express on projects such as:

1. Node.js and Express.js | Sessions, CSRF attacks how to do it and avoid it, CSRF Token and implementation
   https://www.youtube.com/playlist?list=PLIVwhETEfDf2Ge8oPBp19d_9OYcRf5KwM
   Source Code: https://github.com/dennysjmarquez/express-manager-session-security-csrf-token
2. A simple server for the (Angular 13 | Movie Platform) project Node.js, Express.js, cors, this server serves a JSON file as dB in a single endpoint protected by an auth0 token., Movie Db with its Title, description, type of movie, image and date
   Source Code: https://github.com/dennysjmarquez/angularStreamingServer
3. Back-End for the project (Angular 10 | angular-adv-adminpro), Node.js, Express.js, MongoDB, Mongoosejs, CRUD, JWT, express-validator, google sign in, Data Paginations.
   https://dennysjmarquez.medium.com/angular-10-mean-google-auth-jwt-lazyload-upload-de-archivos-guards-pipes-zona-admin-bfa2e5ef9074#fdd2
   Source Code: https://github.com/dennysjmarquez/angular-adv-adminpro-backend

4. Back-End for the project (React | Cryptocurrency Investment Calculator), Node.js, Express.js, Messari API, axios, cors, parsing CSV with NodeJS, dependency injection, and much more…

   Source Code: https://github.com/dennysjmarquez/Calculadora-de-Inversion-en-Criptomonedas

And I add this article: Everything You Need to Know About Rest API: Glossary of Essential Terms and More.
https://dev.to/dennysjmarquez/todo-lo-que-necesitas-saber-sobre-api-rest-glosario-de-terminos-esenciales-y-mas-29pc

In this article, I refer to the HTTP-oriented API Rest.

Although I have not used these technologies together in a single project, I have a general understanding of how they work together and how they can be applied.

4. **How have you integrated third-party APIs into your previous projects and what considerations did you take into account when doing so?**

In my projects, I have integrated third-party APIs considering security as a top priority above all else.

Typically, third-party APIs are a secure and reliable system to implement in projects, thanks to the fact that they have already been developed and tested; however, we will never have full control over them. This leaves a margin for possible security issues that we must consider when integrating them.

In my projects where I have integrated third-party APIs, I have always taken the precaution of storing credentials or API Key in .env files and implementing them within a try catch. It is good to look at the security history of the API and the security failures it has had. This gives us an idea of how secure it can be to implement it when choosing a third-party API. If the API is not old and is a project that has been maintained over time, all of this must be considered. The reaction time of the API and the load it can receive must also be considered, as this can affect the application being developed.

An API that is not designed to handle a high load can be vulnerable to denial of service (DoS) attacks or flooding attacks that could disrupt its services, just as an API that takes too long to respond can be vulnerable to timeout attacks, in which an attacker could try to exploit the slowness of the API to carry out brute force attacks or injection of malicious code.

Recently, I acquired knowledge of **CyberSecurity - Ethical Hacking, NIST**, security controls and **OWASP TOP 10** in [Seguridad Cero Academy](#), and I can make better use of security in my Rest APIs.

OWASP provides a guide for the correct implementation of third-party APIs in:

1. https://owasp.org/API-Security/editions/2023/en/0x11-t10/
2. https://owasp.org/API-Security/editions/2023/en/0xaa-unsafe-consumption-of-apis/

Additionally, I have also learned the importance of taking into account **technical debt** in projects that stretch over time. If the APIs or SDKs used are not regularly updated, we could end up with outdated APIs and SDKs, which represents a serious security problem for both our applications and our APIs. Therefore, it is crucial to keep up to date with the latest technological advances and regularly update the key components of the project to ensure long-term security and efficiency.

I leave you an article I wrote about technical debt in projects that stretch over time: https://dev.to/dennysjmarquez/la-deuda-tecnologica-en-nuestros-proyectos-13p

There are many things to learn and improve.

5. **Have you integrated cryptocurrency exchange APIs (Binance, Bitso, etc)?**

Yes, I have integrated cryptocurrency exchange APIs into a personal project called **Cryptocurrency Investment Calculator**, using the **Messari API**.

Source Code: https://github.com/dennysjmarquez/Calculadora-de-Inversion-en-Criptomonedas

This is a web application that allows users to define a dollar balance to invest in Bitcoin, Ethereum, and Cardano. The application calculates the annual profit for each currency, assuming that Bitcoin has a monthly return of 5%, Ethereum has a monthly return of 4.2%, and Cardano has a monthly return of 1%.

The application was developed using various technologies, including HTML, CSS, React, Node.js, Express.js, Tailwind CSS, axios, and the Messari API.

Demo: https://crypto-inv-calc-api-messari-demo.onrender.com

In addition, the application displays the projected final balance of the invested amount and its equivalent in crypto. It also includes an updated table with information about the 3 cryptocurrencies. This feature allows users to have a clear view of their investment and how it is expected to grow over time.

6. **How long have you been using Tailwind, what challenges have you faced while working with it, and how have you overcome them?**

I have been using Tailwind CSS for the past 6 months. During my last job at **Epidata**, I had the opportunity to apply this tool in replicating a Figma mockup in React, where I also implemented the @headlessui/react library.

While I have enjoyed working with Tailwind CSS, I have also faced some challenges. Indeed, in some cases, it took me more time to find the right class for the style I wanted to apply. However, I overcame this by consulting the Tailwind CSS documentation, which I admit is very comprehensive.
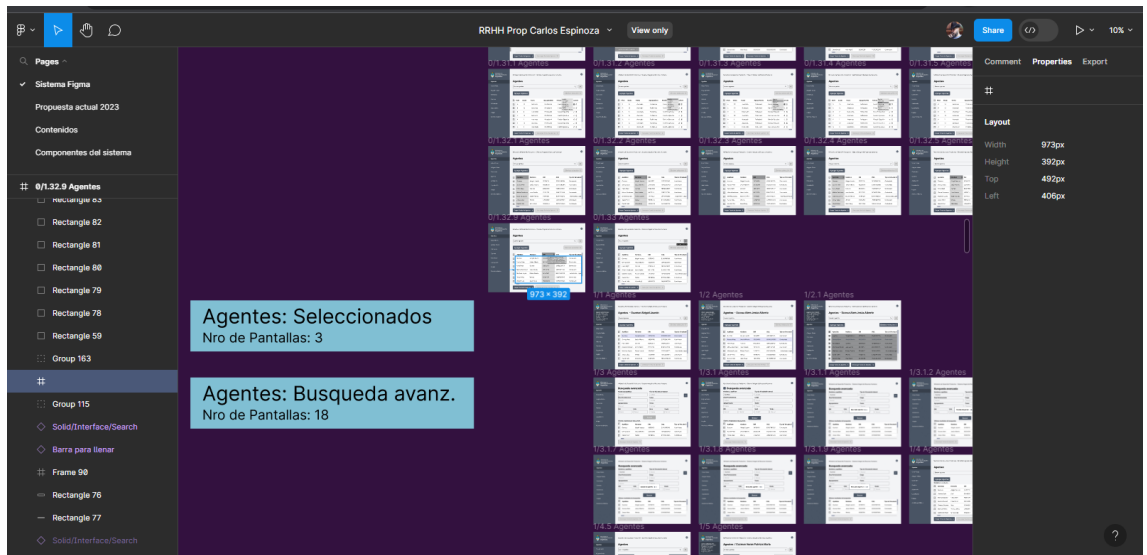
7. **Can you talk about your experience in designing and implementing data storage solutions, as well as handling relational and non-relational databases such as MongoDB and Firebase?**

My experience working with Firestore, Firebase Realtime Database, Cloud Functions, and MySQL has been very interesting.

1. I worked on a React Native application at **Credit Agile Inc**, where I implemented Firebase Realtime Database and Cloud Functions.

In this development, I decided to use Google's real-time database, as the mobile application I developed would have a Trackit tracking system and a notification list.

The Trackit system used geolocation to display its real-time location on a Google map, data necessary and useful for the recovery company.

I implemented a **push notification** system using Cloud Functions. When a new entry is created in the notification table, the Cloud Functions performs two actions. First, it performs automatic database maintenance, removing old notifications that are over a year old. Second, it generates and sends a push notification to the application associated with the notification. In this way, the user receives the notification in the application and can also see it in a notification list that is automatically updated thanks to the real-time database.

To protect the database, I used rules that allow only the admin user (auth.uid) to write and read data.

I implemented a useful and efficient solution for the management of loans and payments at Credit Agile Inc.

2. I have used Firestore, I have developed a database to manage my publication system, my portfolio, and my professional profile on my website. https://dennysjmarquez.dev/

In addition, I have used a headless CMS using the Flamelink SDK to manage my publications and content, and I have implemented Firestore rules to ensure database protection, verifying if the user reading or writing to the database is the administrator.

To optimize the website and database usage, I created a cache system that worked both on the backend and frontend. This system allows the website to be displayed almost immediately, the cache did not depend on sessions, as it is stored in the server memory for a maximum of 30 minutes.

If someone made a query that is already in cache, the server returns the data stored in cache instead of querying the database. Thus applying techniques such as pagination and limiting results to avoid performance issues.

Thanks to the implementation of Firestore pagination, I was able to solve slowness issues, as before it brought all the results and with this it was limited to a maximum of 10 results per page, and as my website was hosted on Netlify, I solved the 501 error that occurred when the application took too long to load.

3. In my final project for **the Web Programming Diploma at Uneweb Institute**, I developed a **CMS similar to WordPress** using **MySQL** as the database manager.

To simplify the database structure and improve efficiency in managing categories and subcategories, I implemented the Nested Sets technique. This technique allowed me to store all categories and subcategories in a single table, setting the left and right columns for the hierarchical relationship between them.
I created a relationship table using taxonomy to manage the information of different taxonomies in a single table.

Using the 'relation' and 'data' tables, I follow a structure that allows establishing relationships between objects and data using taxonomies. This structure can provide flexibility and ease of management when organizing and classifying information in the system.

This allows for a simpler database structure and avoids the creation of multiple separate tables.

I also used two tables for the posts, one for the posts themselves and another to add the postmeta associated with it, which allowed me to create multiple additional fields and associate them with a post.

For the connection and management of the database, I used the Doctrine DBAL library, including the Doctrine\DBAL\DriverManager class. To maximize security in the management of queries to the database, I implemented techniques such as the prepare method and bindParam to avoid SQL injection attacks.

Proyecto: https://dennysjmarquez.medium.com/un-cms-parecido-a-wordpress-y-lo-llame-scorpion-cms-abd9d7005cde

Source Code: https://github.com/dennysjmarquez/scorpioncms

8. **How have you conducted unit tests in your previous projects and what is your approach to ensuring software efficiency and responsiveness?**

   1. I have worked with the **jest** and **superTest** libraries to implement unit and integration tests in backend projects based on Node.js, like this one: https://github.com/dennysjmarquez/tdd-en-nodejs-guia-de-tests-con-jest
   In this project, I included both unit and integration tests.

   2. I have also applied unit and integration tests in this project. https://dennysjmarquez.medium.com/angular-10-mean-google-auth-jwt-lazyload-upload-de-archivos-guards-pipes-zona-admin-bfa2e5ef9074#db81

      Where I used Angular 10 and **Jasmine 3.10.**

      Additionally, I completed this course: **Introduction to Software Testing for Beginners** https://www.udemy.com/course/introduccion-al-testing-de-software-para-principiantes/
      Where I learned that there are methodologies for testing software, applications, and more. This helped me understand and implement the appropriate testing strategies.

9. **Can you provide an example of a project where you used WebSockets and how you implemented it?**

   Yes, I once used WebSockets with PHP 5 to create a real-time chat on my technical services website. Since then, I haven't had the opportunity to implement this technology in new projects.

10. **What has been your previous experience in leading information technology projects?**

In terms of my experience in leading information technology projects, I must admit that I have not yet had the opportunity to lead any. However, I have carried out my own projects and have studied topics related to team management. I have taken several courses in this area, including:

1. [Leadership and Teamwork](#)
2. [Fundamentals of IT Team Management and Programming](#)
3. [How to Develop Emotional Intelligence](#)
4. [Emotional Intelligence and Motivation. Achieve Your Goals](#)

Additionally, I have written several **articles on management topics**, including:

1. [Origin and History of Management](#)
2. [The Life and Legacy of Mary Parker Follett: Forgotten Influence on Modern Management](#)
3. [The Procrustes Syndrome: How to Prevent it from Affecting the Work Environment](#)

11. **How would you describe your communication style and how do you relate to other team members in a work environment?**

As for my communication style, I consider it to be respectful and empathetic. I am aware that in a high-performance team environment with tight deadlines, situations can arise that we must avoid. Therefore, I believe it is essential to stay focused on task

completion, but also be willing to help colleagues if the need arises. However, if there are colleagues with hostile attitudes, I try to avoid them and apply emotional intelligence to maintain a healthy work environment

Likewise, I have optimized processes to achieve mutual and beneficial collaboration for the project. In my previous job, when I saw that my back-end colleagues were busy with the database, I proposed a solution to streamline the work.

I created data models for the App interfaces based on assumed back-end endpoints, which allowed us to show project progress to the client before having the real information.

In this way, I facilitated my back-end colleagues to focus on critical requests by taking care of the creation of the data models while they were busy.

My back-end colleagues used my data models as a basis to develop the necessary endpoints with real information, which improved team efficiency.

12. **How many people have you managed?**

I have not had the opportunity to be a leader or manager yet.

13. **How do you handle deadlines and resource allocation in an IT project?**

In my career, I have gained experience with this topic. In my previous job at Epidata, we used the Scrum methodology, which included the use of a backlog and weekly sprints, with a deploy every Friday.

This methodology allowed us to have a clear vision of the project's objectives and ensured that we progressed efficiently and as planned.

Although the backlog planning was the responsibility of my manager and was defined in conjunction with the client's requirements, I was able to actively participate in tracking and monitoring progress.

In addition to my experience with Scrum, I have completed this course: [Scrum Master + The Agile Revolution + Leading Scrum Teams](#)

NOTE:

Thank you for the opportunity to share my knowledge and experience, I hope I have answered your questions clearly and satisfactorily. If you need more information about my experience or any other inquiry, please do not hesitate to contact me. **More about my experience in:** https://dennysjmarquez.dev/ , https://www.linkedin.com/in/dennysjmarquez/