I am pleased to present this PDF which contains an in-depth look at some of my experiences, skills and approach to 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 come to work?**

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

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

   My salary expectation is no less than $10.00 USD/hour. However, I am willing to negotiate based on the specifics of the position and additional benefits 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 compose it in different projects.

   Over the past 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 project to manage Hospitals, doctors and users. 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 together with

references between collections for custom assignment; validations using MongoID; common (findByIdAndUpdate/Delete) and advanced (field projection + simultaneous search + searches with regular dispatches) searches; recursion using populate(); countDocuments() documentary counter; sort(), skip() and limit() ordering for paging 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-files-guards-pipes-zone-admin-bfa2e5ef9074#fdd2

React:

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

I have 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 skills used are: React Function Component, class Component, lifecycles, React Hooks: (useState, useEffect, useCallback, useState, useContext, useRef, useLayoutEffect, useMemo, React.memo), Portal, createElement, isValidElement, HOC, flushSync.

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

1. React the ultimate best practice guide ever devised, SOLID Principles, The STUPID acronym, Clean Code and Code Smell.
   https://dev.to/dennysjmarquez/react-la-guia-maxima-de-buenas-practicas-jamas-conceived-principles-solid-the-solid-acronym-stupid-clean-code-and-code-smell-53l7

2. The 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- conceived-principles-solid-the-solid-acronym-stupid-clean-code-and-code-smell-53l7

Node.js and 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 a Gateway API to hide the server providing me the data as well as the token. I used sessions to handle the data securely between request, custom middlewares to extract the information from those sessions and expose it to Next.js, Memorystore to store the sessions in server memory and implementation of express-session, express-fileupload, next- i18next/middleware, express .json(), node-cache, full CRUD making use of next-connect and Dynamic Segments [...param].js, and Total Capture Segments.
[js and CORS among other necessary resources.

I have also worked with Node and Express in 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_9OYcRf5Kw M
   Source code: https://github.com/dennysjmarquez/express-manager-session-security-csrf-token
2. A simple server for the project of (Angular 13 | Movies Platform) 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, movie type, 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, and much more...

    https://dennysjmarquez.medium.com/angular-10-mean-google-auth-jwt-lazyload-upload-files-guards-pipes-zone-admin-bfa2e5ef9074#fdd2

    Source code: https://github.com/dennysjmarquez/angular-adv-adminpro-backend

And I add this article: Everything you need to know about API Rest: 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 API Rest.

Although I have not used these technologies together on 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 in 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 priority above all else.

Normally, 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 room for potential 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 the credentials or API Key in .env files and implementing them inside a try catch. It is good to see the API security history.

and the security flaws 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 that needs to be considered. Also consider the reaction time of the API and the load it may receive, as this may affect the application being developed.

An API that is not designed to handle a high load may be vulnerable to denial-of-service (DoS) or flood attacks that could disrupt your services, just a s an API that takes too long to respond may be vulnerable to timeout attacks, where an attacker could try to exploit the API's slowness to perform brute-force attacks or malicious code injection.

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

OWASP provides guidance 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/

In addition, I have also learned the importance of taking into account the **technological debt** in projects that extend 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 with the latest technological advances and regularly update key project components to ensure long-term security and efficiency.

Here is an article I wrote about the technological debt in projects that are extended 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 API's (Binance, Bitso, etc)?**

No, and although I have not had direct experience integrating these APIS, I am interested in learning more about this topic and am willing to acquire the necessary skills and knowledge to do so in the future.

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

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

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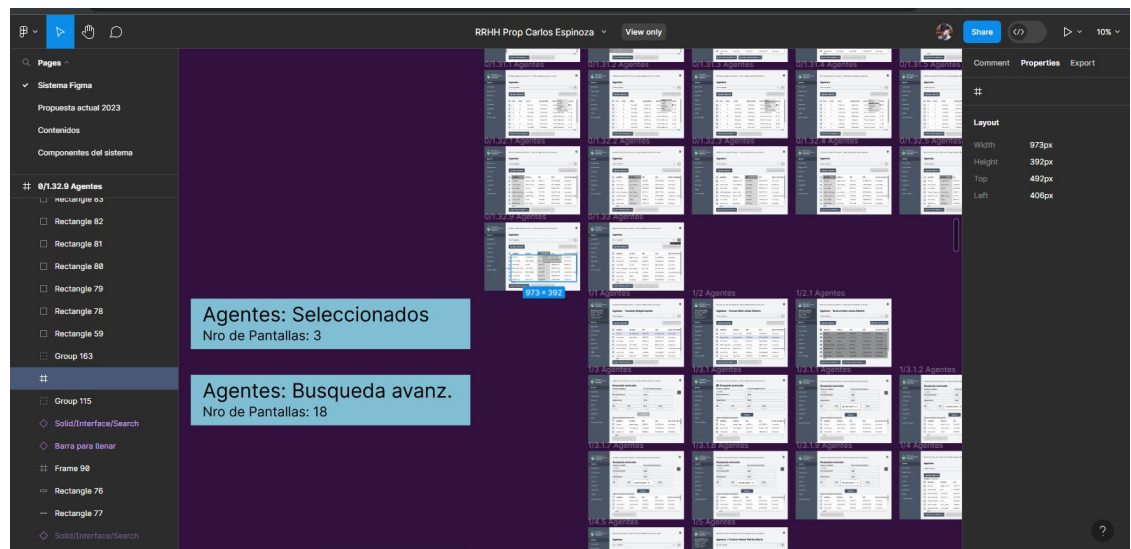
7. **Can you talk about your experience in designing and implementing data warehousing solutions, as well as handling relational and non-relational databases such as MongoDB and Firebase?**

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, since the mobile application I developed would have a Trackit tracking system, and a list of notifications.

Trackit's system used geolocation to display its location in real time on a Google map, which is necessary and useful data for the recovery company.

I implemented a push notification system using **Cloud Functions**. When a new entry is created in the notifications table, Cloud Functions performs two actions. First, it performs automatic database maintenance, deleting old notifications older than one year. 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 list of notifications that is automatically updated thanks to the database in real time.

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

Implement a useful and efficient loan and payment management solution from 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/.

During this process, I have worked with offset and limit pagination techniques, select to bring only the necessary information, use of where in general, compound queries and arrays to validate or filter according to a list of values and search criteria, creation of indexes, use of orderBy and count() to get the number of documents in a collection.

In addition, I have used a headless CMS using the Flamelink SDK to manage my posts 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 caching system that worked on both the backend and frontend.

This system allows the website to be displayed almost immediately, the cache does not depend on sessions, as it is stored in the server's memory for a maximum of 30 minutes.

If someone made a query that is already cached, the server returns the cached data instead of querying the database.

Applying techniques such as paging and result limiting to avoid performance problems.

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

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

To simplify the database structure and improve efficiency in the management of 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 the taxonomy to manage information from different taxonomies in a single table.

Using the "relation" and "data" tables, I follow a structure that allows establishing relations 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 one to add the postmeta associated with the post, which allowed me to create multiple additional fields and associate them with a post.

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

Project: 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 performed unit testing 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 Node.js based backend projects, 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-files-guards-pipes-zone-admin-bfa2e5ef9074#db81

3. Where I used Angular 10 and **Jasmine 3.10. In** addition, I took this course **"Introduction to Software Testing for Beginners!** https://www.udemy.com/course/introduccion-al-testing-de-software-para-beginners/

4. Where I learned that there are methodologies for testing software, applications and more and this helped me to understand and implement proper testing strategies.

9. **Can you provide an example of a project where you have 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 have not had the opportunity to implement this technology in new ones.

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

As for my experience in IT project leadership, I have to admit that I have not yet had the opportunity to lead one. However, I have run 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 equipment management and programming

3. [How to develop emotional intelligence](#)
4. [Emotional intelligence and motivation. Achieve your goals](#)

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

1. [Origin and history of Management](#)
2. [The life and legacy of Mary Parker Follett: neglected influence on modern management.](#)
3. [The Procrustean Syndrome: How to prevent it from affecting the work environment](#)

11. How would you describe your communication style and how 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 may 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.

I have also optimized processes to achieve a mutually 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 showing the progress of the project to the client before having the real information.

In this way, I made it easier for 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 the efficiency of the team.

12. How many people have you been in charge of?

None I have not had the opportunity to be a leader or manager yet.

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

In my career I have gained experience with this, 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 to ensure that it was progressing efficiently and according to plan.

Although 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 taken 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 questions, please do not hesitate to contact me.

More of my experience in:

https://dennysjmarquez.dev/ , https://www.linkedin.com/in/dennysjmarquez/