Optimeleon Full Stack Engineer

Optimeleon is getting ready to revolutionise the way companies handle marketing by introducing an AI based conversion rate optimisation tool chain that includes page variation generation, A/B testing and smart analytics.

As an early stage Full stack engineer, you will be involved in all aspects, right from researching to delivering solutions for our product. You will take lead of development of features that form the core of our SAAS product.

Being a small team, it is crucial for you to be well versed with the AI landscape, producing production grade code, while at the same time being fast at what you do. Achieving a good balance of quality and speed is one the few primary challenges here. The 12 Factor App is a good set of guidelines to follow. Please make a note of this before attempting to solve this challenge.

Smart A/B testing

Overview:

In the world of marketing, the ability to test variations of a product page helps marketeers make informed decisions. For this your challenge is to build a system that helps our business customers manage their individual projects and run smart A/B tests. A variation is an alternative version for a page (for e.g with updated product description, different colours, images etc)

Objective:

Create a marketing SaaS application that allows users to create projects. Each project generates a script that can be injected into a client's website. This script will switch routes by appending a `?variation=` parameter to the URL, which changes for a visitor based on the current time of the day.

Requirements:

1. **Web App:**
    - Use **Next.js** for the frontend with server actions.
    - Implement a user interface for creating and managing projects.
    - Each project should have a unique script that can be injected into a client's website.
    - [Bonus] Implement an authentication system
2. **Database Services:**
    - Separate database access code as a layered architecture
    - Implement RESTful APIs that trigger creating and updating scripts.
    - [Bonus] Separate the db service into its own sub package within the mono repo
3. **Background Job/process**

- Implement a job that can be triggered when a new project is created. This job creates/updates the scripts
- [Bonus] Use a task queue

4. **Storage:**
    - Store the data in memory
    - Store the scripts on disc
    - [Bonus] Use a database of your choice paired with an ORM of your choice. (we use postgres and prisma)
5. **The Script:**
    - The generated script should switch routes on the client website by appending a `?variation=` parameter to the URL.
    - Assume there are 4 variations a,b,c,d and the criteria for selection can be as simple as doing a `mod 4` on the current time of the day.
    - [Bonus]: think of a strategy to ensure that the script only works on an intended url
6. **Project Structure:**
    - Organize the project with a clear folder structure (using a monorepo) for frontend, packages, and shared components.
    - Use best practices for code organization (layered architecture) and modularity.

Time to Solve: Around 6 Hours

Please note that the problem statement is intentionally large, do not spend more than 6 hours solving this problem. The requirements are loosely defined and we expect you to make reasonable assumptions with your best judgement of the domain we are in. We also recommend spending some time planning on how you will solve this problem. Your github solution will be a mirror of how you think, plan and execute. It helps to write down a solution, break it down into solvable chunks and then create commits on Github with an incremental strategy. Make a

submission with wherever you leave, adding a small description of the missing parts and your plans.

What to submit:

- A github repository with clear instructions (README) on how to get the project running
- README should also contain a small write up describing your solution
    - justifying your decisions
    - the main challenges
    - further improvements you can think of for taking this to production
- If the project is at an incomplete stage, then how were you planning to solve the problem

Scoring is based on the following:

- Your understanding of the problem statement and your approach to solving
    - Your github repository is a good place to work and display an incremental approach with commits that solve specific parts of the larger goal
- The solution itself and how you present it
    - the code should run with minimum configurations
    - documentation, even if primitive, also serves as a way to understand your thought articulation (a code should certainly be self explanatory with the least amount of comments. Documentation should include pieces that serve to be mostly non functional requirements, assumptions and other things that cannot be expressed with the code)
- Adherence to generally accepted coding standards (we are a small team and your daily chunk of work will be huge, ensuring that you output great quality code is crucial)
- Prefer easy to follow code over a complex code (e.g bitwise operations are fast but the code is unreadable)

Bonus points

- Proper documentation
- Write test cases
- Write a CI/CD pipeline to deploy this product to a cloud provider of your choice.

How to submit:

- Please submit your solution within a week of receiving this task. Please let us know in advance if this isn't possible.
- Create a private github repository and invite `xxxNam3xxx` and `abcdefghiraj` as collaborators
- Reply with an email that you have completed the task so that we can setup the next call.

The correctness of the expected outcome, including the bonus points, is something that is expected of all engineers at Optimeleon, and is intended to make collaboration easier. This serves as a good window for you to understand what you can expect from your team members.

> **Tip:** We are an AI company and using AI and/or google, stackoverflow etc to help you wherever needed is completely acceptable, but you should be able to explain what your code does, be able to navigate through it, and be able to make ad hoc changes during the next round.
