Software Engineer

Context:

As a software engineer at the forefront of the fastest-growing space tech firm globally, your role extends beyond mere coding proficiency. You're entrusted with the responsibility of leading the design and development of a robust Spring Boot application that meticulously captures satellite launches and their intricate details. Embracing a startup ethos and an agile mindset, you'll navigate the dynamic landscape with flexibility and adaptability, prioritizing velocity over perfection and a relentless focus on getting things done.

Your software engineering expertise will be instrumental in architecting a scalable and efficient solution, ensuring seamless integration with existing systems and adherence to industry best practices. Moreover, your leadership skills, coupled with a strong focus on team collaboration, will foster an environment of innovation and collective problem-solving.

By championing collaboration among team members, fostering open communication channels, and encouraging knowledge sharing, you'll harness the collective intelligence of the team to overcome challenges and drive the project forward. Together, we'll revolutionize the space tech industry, one satellite launches at a time. Please see the instructions below to create the Spring boot application

1. Create a model/db table for storing satellite launchers. The schema for that is

|  |  |
| --- | --- |
| Column Name | Type |
| Id | String |
| Type | ENUM (NEW, OLD, DEGRADED) |
| Registered on | Date |

1. The initial data for launchers can be fetched from <https://isro.vercel.app/api/launchers>. Create an api endpoint that dumps data from this endpoint to your db. (Although this endpoint only returns launcher ids, at the time of fetching, the other columns can be kept as null).
2. Create a model/db table for the data coming from <https://isro.vercel.app/api/customer_satellites>. Follow the schema mentioned in the api response.
3. Create an endpoint that dumps the data from the api responses to these tables.
4. Create CRUDs for both the models.
5. Create Search endpoint: Search satellites based on id, launch date, country, launcher id and/or mass.

**Use standard design patterns and architecture**

What are we expecting

* A complete solution addressing all the tasks
* Clean maintainable and extensible code
* A working demo
* Bug free
* Usage of proper design patterns
* Feel free to use the DB of your choice
* Any version of Java is acceptable

Zip File

Languages - Java Spring boot framework

Front end - React