Gabi Engineering Assignment – Full Stack

The purpose of this assignment is to evaluate your technology skills, design choices, coding culture and overall software engineering competence level.

Your task is to develop a simple, full-stack web application ("**HN Search Notebook**") according to the specification described in this document.

The development time in this task is estimated to be 8 working hours.

Technology Stack

Please use the following technologies:

Frontend	Classic HTML5 + CSS3 +JS (with or without dynamic page fragments) or Facebook React (single-page app)
UI Toolkit	Twitter Bootstrap 3, Semantic UI, Material UI or similar
Backend	Rails 6 (or higher)
Database	Any supported database (MySQL preferred)
Tests	Your choice between RSpec or native Rails unit tests

Choose technologies with productivity in mind.

HN Search Notebook Application Description

HN Search Notebook application lets the user:

- create, browse and delete search notebooks via Search Notebook List View,
- perform Hacker News search using **Algolia HN Search API**, via paginated HN Search Result View,
- add individual search results to search notebooks directly from HN Search Result View,
- browse saved search results in Search Notebook View, showing individual results and search queries used to find them,
- delete individual search results from Search Notebook View,
- display simple Search Notebook statistics.

Views should be organized using simple navigation. Search function and statistics should be available from the application home view.

Algolia HN Search API

API documentation can be found here: https://hn.algolia.com/api.

Entities

The system should use the following entities:

- `SearchNotebook`: user-defined title, creation date, a collection of `SearchResults`,
- `SearchQuery`: query string, creation date, total number of hits,
- `SearchResult`: HN author (login name), author karma points, url, creation date, list of HN tags, search query ID,

Search Notebook Statistics

Statistics should present a table with the following columns:

- search query string,
- average of total search result hits for the last day,
- average of total search result hits for the last week.

Tests

Included tests should at minimum:

- verify that results saved to database are matching the actual API response,
- verify the integrity of statistics for time periods.

Deliverables

The main deliverable is source code for a deployable application in a GitHub repository (private or public), together with a test suite and basic documentation if such information is required for understanding application logic or specific deployment procedures.

A working application deployed to a password-protected temporary VPS would get you bonus points.