Challenge Statement

Problem Statement

As we grew, we had the need to start analysing more data generated by the users in order to make better product decisions. For that, we used google analytics, firebase, sentry, etc. Any data analytics tool or service available which could support our data tracking purposes.

Nowadays, we have built a large recruitment platform with thousands of user interactions per minute and we need to start offering data to our customers, so they can also optimize their jobs and posts the best possible way. For this, we considered using the tools mentioned above, but the data should be accessible to our customers and not only for internal consumption.

The idea would be to build a solution to track user events like: impressions, clicks, views, etc. Any kind of event that could be produce on jobs, stories and profiles. On the other hand, the solution should be flexible enough that we can also add more customised events like click__*applicationCreatedFromBottomBtn,* etc.

Requirements

- The initial proposal is to build an API using NodeJS or Laravel with a relational DB.
  Note: if you believe a non-relational DB fits more, please explain why.
- The API should receive events sent by the client and this event will have an creator, receiver, event type or name, time and any other relevant information you believe is necessary.
- It could be that events get duplicated due to wrong usage form client side. For ex. sent the click event twice on the same action. What should we do then? how to avoid this?
- Data should be accessible and analysable.
- The database needs to know where the data comes from and it shouldn't allow any external usage of it (ie some replicating the request).

- The whole architecture is planned to be in AWS.

What do you need to do?

The information presented was provided by a PO (& data analyst) to our team. The team consist of one front-end engineer who will add any code necessary in app and web to track the usage and a back-end engineer to support you in any API code needed.

The first thing you should do is to analyse the proposal and check if there is a need for any change in the initial requirements and if so, explain why. Then do a project preparation with tasks, tasks definitions and aprox. estimations given the resources mentioned above.

Keep in mind you that you don't need to develop the whole solution, instead focus on:
- Project definition, tasks preparation, distribution and estimation.
- Architecture of the project.
- Propose an alternative solution or dive deeper in the proposed one.
- List of endpoints and/or methods needed based on the process provided before.
- For each endpoint/method would be necessary to have an explanation, pseudo-code, flowchart … whatever you believe explains the logic behind.

Please also provide possible bottlenecks, issues or any other problem you might think it could appear in the life time of this project.

Bonus

Once this project is finished and running you will get ton of raw data that can be analysed in multiple ways and now we need to find a good way to use this new data. What do you think it's a good approach to aggregate this raw data to be able to analyse it further? how would you model this extra solution?

Use case example:

Imagine we want to send a notification of how many users check a profile the day before. How would you aggregate the data in order to make this notification run more efficient, instead of having daily queries to the raw data. You must keep in mind that the aggregate data might be also use in analytics by the marketing team, not only to send daily notifications to users.

Additional notes

- Expected time: ~4h
- Format: it will depend on your solution, so either a repo or a PDF document to robin@gronda.eu.
- There is no need to finish the whole task, but rather focus on quality over quantity.
- Please ask us any questions you think it's necessary.