Hej, Hello and Welcome!

You seem like someone we'd like to have on our team and we'd like to get to know you a little better...

To give us a better sense of your skills as a developer and how you work we would like to you to spend some time on the test assignment below. This is not as scary as it sounds. The test is designed to give us an idea of how you would fit into our team. It will give us an idea of how you approach an assignment without detailed instructions, your general work methodology, problem solving and communication skills.

The details of the test are outlined below. Select the task appropriate to the position you are applying for. We do not expect you to deliver a fully realised product, but would like to see how far you get with the task in the specified time frame.
Using 3rd party open source libraries and frameworks are allowed. Copying an existing app from the Internet with the motivation that "I always look on the Internet first before building my own solution" is not allowed.

When you have something to present to us, or at any time when you have questions regarding functionality, requirements, etc. please contact maria@glomo.se

Good luck! :)

Test Assignment - Project Sports Poll

Overview

The *"Sports Poll"* app is a dead simple poll system around some sports events which can be fetched from this JSON file
It allows users to vote on each event and then shows another random event to vote on. There should be one poll, that displays one (random) category at a time. E.g. load Football category with poll options to vote on from DB, and only display those at a time, next browser reload would maybe display Tennis.
The polling options should be what team will win or if it will be a draw, e.g.
- Home Team win (Team A)
- Draw
- Away Team win (Team B)

Delivery

When you have completed the tasks we'd like you to deliver the following:

- A description of what you have done and how far you got in time allotted for the work on the project.
- Which development tools you used and your general progression on the key issues, what you got stuck on, what you did or didn't solve etc.
- A step by step instruction to how to run the project.
- Share the source code in a **private** code hosting on Bitbucket or Github (Bitbucket has free private repositories). Share that repository giving admin permissions to **GLOMOadmin** and send the URL in an email to maria@glomo.se.

Tasks and deliverables per job position

Junior Frontend Developer

This exercise is expected to take 3 to 6 hours to complete.

1. Create a page with the best design practices in mind for this small application. The design will be taken into consideration and should be visually pleasant and user friendly.
2. Please use React Javascript framework. If you are not familiar with this framework, you may use any other modern Javascript framework of your preference. The choice of framework may impact the way we assess your assignment.
3. It's not allowed to use any CSS framework like Bootstrap or similar.
4. The JSON file with the events can be accessed directly from the frontend or hardcoded in the website.
5. The odds can be generated randomly. There is no need to keep track of votes across page reloads.

Senior Frontend Developer

This exercise is expected to take 4 to 8 hours to complete.

1. Create a page with the best design practices in mind for this small application. The design will be taken into consideration and should be visually pleasant and user friendly.
2. Please use React Javascript framework. If you are not familiar with this framework, you may use any other modern Javascript framework of your preference. The choice of framework may impact the way we assess your assignment.
3. It's not allowed to use any CSS framework like Bootstrap or similar.
4. The JSON file with the events can be accessed directly from the frontend or hardcoded in the website.
5. The page should be responsive and work well across different types of devices (desktop, mobile, tablet).

6. Keep event votes stored in a browser local storage so they can persist across page reloads.
7. Add different kind of visualisations based on event attributes (country, state, sport and so on). It is up to you to decide how to best represent the data.
8. Add a couple of automated tests.

Fullstack Developer

This exercise is expected to take 5 to 10 hours to complete.

1. Create a backend for Sports Poll. The backend should be created using Node.js. Whichever Node.js framework you select will not impact the way we assess your test.
2. At start, fetch the events from the JSON file and store them in a local database of your choice.
3. Create a backend API to read events.
4. Create a backend API for voting for events (home win, draw, away win).
5. Create a page with the best design practices in mind for this small application. The design will be taken into consideration and should be visually pleasant and user friendly.
6. Please use React Javascript framework. If you are not familiar with this framework, you may use any other modern Javascript framework of your preference. The choice of framework may impact the way we assess your assignment.
7. It's not allowed to use any CSS framework like Bootstrap or similar.
8. The page should be responsive and work well across different types of devices (desktop, mobile, tablet).
9. Add a couple of automated tests.

Senior Fullstack Developer

This exercise is expected to take 7 to 15 hours to complete.

1. Create a backend for Sports Poll. The backend should be created using Node.js. Whichever Node.js framework you select will not impact the way we assess your test.
2. At start, fetch the events from the JSON file and store them in a local database of your choice.
3. Create a backend API to read events.
4. Create registration/login for the Sports Poll. It could be either with regular email address or Facebook login.
5. Create a backend API for voting for events (home win, draw, away win), keep track of the votes. Only allow logged users to vote and don't allow more than one vote per user for each event.
6.   Create a page with the best design practices in mind for this small application. The design will be taken into consideration and should be visually pleasant and user friendly.
7.   Please use React Javascript framework. If you are not familiar with this framework, you may use any other modern Javascript framework of your preference. The choice of framework may impact the way we assess your assignment.
8.   It's not allowed to use any CSS framework like Bootstrap or similar.

9. The page should be responsive and work well across different types of devices (desktop, mobile, tablet).
10. Add a couple of automated tests.