Task: Simple Calorie App

We would like to see how you would create a **simple calorie tracking app**.

Users should be able to enter their food entries along with the number of calories for each food entry.

Estimated time/effort: **10-20 hours**
Expected delivery time: **7 days**

Task scope and expectations

- The purpose of this task is to build a functional real-life application with all listed requirements.
- What we want to see is how you will implement some real-life functionalities by paying attention to details and following good development practices.
- We want to see you understand how to set up a proper project infrastructure, show knowledge of simple problem solving, and knowledge of using your preferred frameworks and libraries.
- Please, do not spend too much time creating the layout. The layout should look decent and something you would feel comfortable showing to the client. Imagine this is a demo call for a client. So, they will expect to see e.g. Bootstrap layout, MaterialUI, default Android / iOS form design, etc.
- Instead of spending a lot of time on providing a custom UI, please rather ensure the functionality is properly done and e.g. user cannot submit incorrect values.

Task functionalities

Users should be able to manage food entries
- A user should be able to add a new food entry
- Food entry should contain the following information:
    - Date/time when the food was taken
    - Food/product name (i.e. Milk, banana, hamburger)
    - Calorie value (numeric value)
- The first screen a user should see is the list of existing food entries

Calorie limit warning per day
- The daily threshold limit of calories should be 2.100.
    - Ensure the users can see for which day they reached that limit. Also, ensure it is easy to change that limit in the code, per user. You don't have to create an interface for this purpose.

Admin role with a simple reporting
- Implement an admin role
- Admin can see a screen with all added food entries of all users and manage existing food entries (read, update, create, delete)
- Admin should also see the report screen with the following information
    - Number of added entries in the last 7 days vs. added entries the week before that. Please Include the current day in those stats
    - The average number of calories added per user for the last 7 days
    - A regular user should not be able to access this reporting screen or access its data

User authentication/authorization
- **Please use a token authentication method.** You don't have to implement a signup and login process. You can manage everything using a predefined user-specific token in the backend, however, ensure the token can be changed easily during your next interview

Filtering by dates
- Users should be able to filter food entries by entry date (date from / date to)
- Place filter fields on the same screen where the list of previously added food entries is

Allow adding a photo for each food entry
- Allow users to take a photo of each food entry

Assignment guidelines

- For this project, you need to provide functionality based on REST or GraphQL API. Ensure the API can be tested and checked during your next call.
  - If you do not feel comfortable building an API, feel free to use Firebase to build the API. Using Firebase is allowed, however, this would limit you in getting backend engagements in Toptal.
- **PIck technologies you are most comfortable with.** Some developers decide to use new technologies for this assignment, however, we are trying to evaluate your skills working with a client. Using new technologies for a short project, for your new client, and during a short period can be a risk that cannot guarantee success.
- During your next call, demonstrate the project in your local environment. We do not want you to spend time deploying this home-take assignment.
- Prepare sample data for a project.
- Ensure data quality of entered information in the system.

Milestones and task delivery

- The deadline to submit your completed project is **1 week from the moment you received the project requirements**.
- It means project code must be submitted within 1 week from the moment that was delivered to you by email.
- If you schedule your final interview after the 1-week deadline, make sure to submit your completed project and all code to the private repository before the deadline
- Everything that is submitted after the deadline will not be taken into consideration.
- Please do not commit any code at least 6 hours before the meeting time so that it can be reviewed. Anything that is submitted after this time will not be taken into consideration.
- **Short project video** - after you are done, please provide us with a simple and short video recording of the functionality and submit it to the GIT repository. Please do not spend too much time on this step. Our goal is to see your project presentation as soon as possible and how you would provide a quick intro for the client. So, please pick any free or preinstalled software that can help you do this.
- Please join the meeting room for this final interview on time. If you miss your interview without providing any prior notice, your application may be paused for six months.