Frontend Test

Test Explanation

Test
- Please implement the front-end based on the problem description design.

Supplement
- Please develop and assume that you will connect to the API server
- For an overview of the design, please refer to【[Application Specifications and Overview](#)】

Testing Hour

Please follow these guidelines
- Estimated working time: 8 hours
- Max Working Time: 12 hours

Submission

Submit a zipped archive of the Git repository folder.

When archiving zip, Please pay attention at some points as below:
- Include the .git folder in the zip
- Delete any files that are not under git control (.node_modules is not required)

Design Resource
- Design
    - Please Refer Figma below
    - https://www.figma.com/file/7qqT3dvv5OagaRlUFK01vB/HealthApp_1203
- Imagine Resource
    - Specification of image files in the following folder
    - Imagine

Development Environment

There are no restrictions on the development environment and editor. Please use what you are familiar with. However, you have to specify the build environment and package.

Operation/ Build Environment	Version
Node.js	16.13.1
npm	^6.14.15
react	^11.4.1

※ The above package list is the minimum structure.
You can add the appropriate packages as needed.

Application Specification and Overview

A web application for health care.

Top Page

This is the Top Page after login.

The outline of each display is as follows.
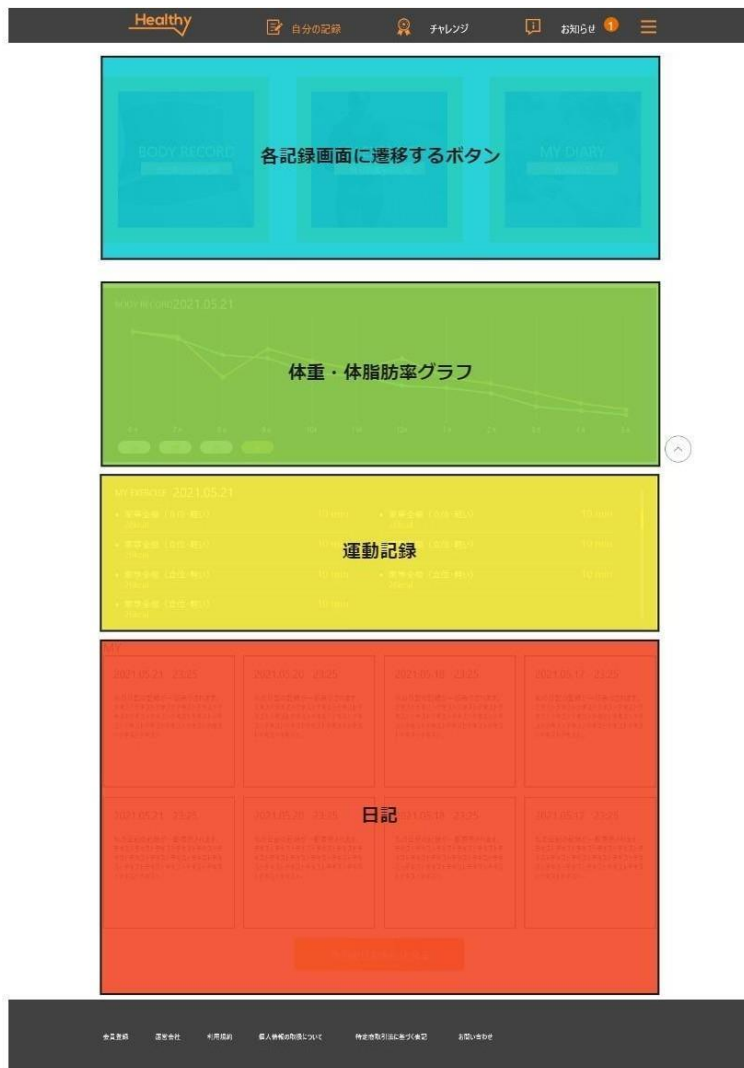
My Record

This is the page where you want to view your records
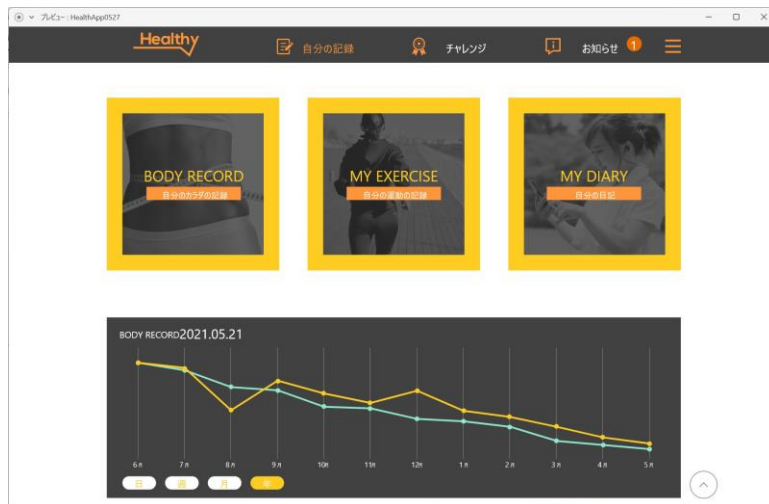
The outline of each display is as follows.

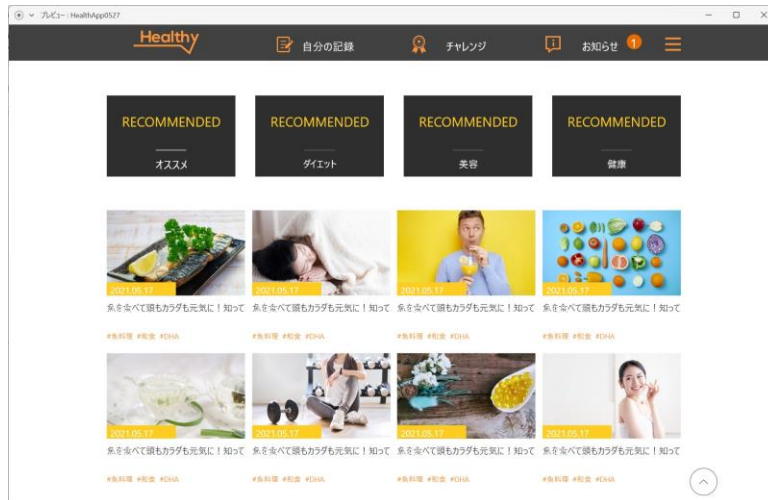****Note:*
Blue collum: Button to move to each entry screen
Green collum: Body fat percentage graph
Yellow collum: Exercise record
Red collum: Diary

Column page

This is the column page about health. You can watch it without login.

Evaluation Criteria

We will review your submission by focusing on the following criteria:

- Fidelity
  - Is the design reproduced?
- Clarity
  - Does the README clearly and concisely describe the problem, solution, and operation verification procedure?
- Architecture
  - Is there a separation between the data and the UI?
- Code Quality
  - Is the code simple, easy to understand, and maintainable?
  - Does your coding style match the language guidelines? ?
  - Is it consistent in your codebase?