Flexkids - Hiring exercise

React/JS developer

Introduction

Flexkids builds software to help day-care organizations succeed. One of the features is that we offer daycare employees with an overview of children that are scheduled to attend on a given day. We call every occurrence of a child that will attend on a certain time a "session". In this exercise we'll ask you to implement such an overview yourself.

Instructions

You've received a file **ExerciseCode.zip**. It contains a javascript application built with React, React Redux+Sagas and React Router; the same libraries we use to develop our React applications.

Before you begin:

1. Make sure you have NodeJS and yarn installed. Run **yarn install** to install all required modules.

2. Run **yarn start** in one window and **yarn run json-server-2** in another. This starts two servers: one webserver for the webapplication and a JSON webservice that acts as our API.
3. The application provides some sample functionality: listing news items. Read through **README.md** and take a look at the provided diagram to get a feel for how this works.
4. Implement the functionality requested described in the **Exercise** section below.

When you're done:

1. Before moving on, make sure things work!
2. Create a ZIP file with all your code minus the node_modules!
3. Send the ZIP file back to us. We'll review it and take the next step in the process.

Tips:

- Consider real world problems like latency and failed HTTP requests. Provide the user with some feedback.
- Feel free use to libraries
- Show how you think code should be written. Think about code-styling, patterns, comments, if you have time left some tests would be nice, etc. Make it pretty.
- If you're new to React or Redux/Sagas don't be afraid to ask a question.

Exercise:

The API server you started with **yarn start json-server-2** offers the following endpoints:

- http://localhost:3001/sessions/ - This contains sessions per day.
    - You can find sessions for a certain day by filtering on the day property: http://localhost:3001/sessions?day=2018-06-02 .
    - Per session it will have a child_id to refer to the child for whom the session was booked.
- http://localhost:3001/children/ - This contains a list of children.
- http://localhost:3001/news/ - This contains a list of news items.

Attention: the API server will sometimes intentionally delay or fail requests!

First, create a new page 'Sessions overview' and add it to the menu. the requirements for this page are as follows:

1. The sessions overview page by default shows all the sessions available for a certain date. Don't worry about styling (CSS) too much but do think a little about what would be a useful view for a day care employee.
2. If no date is provided the **default date** will be 2018-06-02.
3. There will be two buttons: 'Next day' and 'Previous day'. When clicked it will show the sessions for that date.
4. For each session show the following information:
    1. The start and end times for this session.
    2. The name and avatar for the child
    3. The group name associated with that session
    4. The current presence status for that session
    5. A button that updates the presence status according to the following rules. Make sure the button shows a useful label.
        1. If the presence status is 'unknown' it becomes 'present'.
        2. If the presence status is 'present' it becomes 'picked up'.
        3. If the presence status is 'picked up' it becomes unknown' again.
5. Allow the user to filter the session list to only show the sessions for one specific group.

Flow Diagram for the 'news' feature: