JavaScript test assignment

Hotel Safe Deposit Box [M02-t]

January 6th, 2020
Computer Rock

COMPUTER ROCK

Introduction

Goal of this project is to simulate real world problems and to determine how you would approach solving it as a Computer Rock JavaScript developer. We are not only focused on checking your programming skills, but also how are you going to set up the project and organize your work. We expect strong programming skills, so give us your best.

Some rules before we start

- Project development time should not exceed 5 days. Starting date is the day of receiving this assignment. Deadline is emphasized in the assignment email.
- Uncompiled code should be submitted before the deadline in the form of GitHub repository. Please send us the link to the repository. Compiled (clickable) web-application should be hosted on the GitHub Pages.
- Haven't managed to implement all the requirements by the deadline? Don't worry. Submit your code, as we grade all the effort put into the assignment.
- Project may have constraints, such as which libraries may or may not be used. Follow the project requirements closely.

Project requirements

Your task is to create a web simulation of a safe deposit box usually found in hotel rooms. User interface is made simple to avoid the need for issuing mechanical keys or pin codes by hotel's staff. What follows are User's manual and then the control panel of the safe deposit box.

Locking:
1. **Before closing doors enter any 6 or more digit passcode.**
2. **Close the door and press lock button [L] to lock the safe.**

Unlocking:
1. **Enter your passcode.**
2. **Please leave the door open before checking out from the hotel.**

Simulation should be implemented using React (reactjs.org). Please use these libraries and APIs:

- redux,
- Fetch API,
- Sass/SCSS.

You may use create-react-app to jumpstart the project. You may use redux-saga, redux-thunk or similar redux middleware as you see fit.

General look & feel should be approximate to given design in **Appendix A** at the end of this document. Control panel is composed of a body, number keypad and backlit screen. Backlit screen has two textual segments. Top left segment indicates if the door is locked. Main segment displays status messages during the locking/unlocking process. Screen's backlight is turned off when idle for more than 5 seconds. You can find all possible values for the textual segments in **Appendix B**.

Locking/unlocking sequence is described in User's manual for the safe. You can also check out this video to see one variation of real world safe implementation: https://youtu.be/qNPJqtGSXuE

Number keypad doesn't have submit/cancel keys except the lock/unlock (L) button. For all other cases there is input timeout instead. User enters a sequence of button presses, then if idle for 1.2s, control panel processes given input. Mechanical process of locking/unlocking lasts 3 seconds.

Locked safe should also be unlockable with master unlock code:

1. upon entering 6 zeros, safe goes into "service" mode, so secret master code of unknown length can be entered. It is made up from any keypad character (eg. '4L5336*987**L01576823').
2. when input is completed, master code should be sent to this validation endpoint: https://9w4qucosgf.execute-api.eu-central-1.amazonaws.com/default/CR-JS_team_M02a?code=456R987L0123
3. if response, eg. {sn:123456}, matches the serial number of the safe deposit box - door unlocks. Serial number is predefined and marked with [S/N] and placed on the door of the safe box. (check the design!)
4. Response from the master code validation endpoint will be random for invalid master codes.

Appendix A - design

Colors:

- background: #434343
- panel body: #7d7d7f
- button: #63636e
- button text: #f3f3f3
- screen text: #434343
- screen, backlight off: #47b2b2
- screen, backlight on: #7fffff

Fonts:

- Roboto Mono Normal
- Roboto Mono Medium

Appendix B - screen messages

Top left segment values:

- Locked
- Unlocked

Main segment values:

- blank (no value)
- Error
- Ready
- Locking...
- Unlocking...
- Service
- Validating...