VIDEO ANALYTICS - DSAID - GOVTECH

FULLSTACK DEVELOPER INTERVIEW – TECHNICAL TEST

Create a video upload web application using React and Node.js that does the following:

1. A multi-step wizard form to allow user to upload a video to a server;
2. In the first step of the form:
    - Allow user to select a video, and display a thumbnail of the selected video, as well as the show the filename of the video;
    - Allow user to enter the following metadata of the video
        - video title (required),
        - video start date time (required),
        - video location (i.e. postal code) (optional);
    - All data entries must be validated;
3. In the second step of the form:
    - Display some Terms and Conditions [you can specify] with a tick box for user to indicate that he/she agrees to them;
    - Allow user to navigate back and forth between the first and second steps of the form to make amendments before upload;
4. In the third step:
    - Begin uploading the video to a back-end server, showing a progress bar to indicate the percentage process based on the no. of bytes uploaded;
    - The backend should store the video in the file system on the server-side and store the video metadata a relational database (can be SQLite, PostgreSQL or MySQL);
    - The backend should validate the posted metadata before storing to DB;
    - At the end of upload, show a success message or failure message based on response from the back-end server;

Additional requirements:

1. You can use existing React frameworks, but do not use any prebuilt upload component or library in React or jQuery. You should build the upload component from scratch.
2. You can use Webpack or any other compiling tool to pack your code.
3. You can use any CSS pre-processors or CSS frameworks (like Bootstrap), any web fonts, icons or images to style up the components.
4. Design and develop your component with usability in mind. Display your controls and content with proper style and layout, so that the component is friendly to the user.
5. For the server-side code, you can develop from scratch using Node.js, Express or any Node.js back-end frameworks.
6. You can make use of ORM frameworks if you know how to do so.
7. Make use of TypeScript to properly define the types and validate your data on server-side if you know how to do so.