UDACITY

‹ Return to Classroom

Image Processing API

REVIEW

CODE REVIEW 9

HISTORY

▸ src/file.ts 4

▸ src/processing.ts 2

▸ README.md 1

▾ src/routes/api/images.ts 1

```
import express from 'express';
import control from './../../file';
// query segments
interface ImageQuery {
    filename?: string;
    width?: string;
    height?: string;
  }

  /**
   * Validate query.
   * @param {ImageQuery} query Query object passed by express.
   * @return {null|string} Null if valid or error message.
   */
  const validate = async (query: ImageQuery): Promise<null | string> => {
```

AWESOME

Really an amazing job using a validate function before sending the parameters for image processing.

```
  // Check if requested file is available
  if (!(await control.isImageAvailable(query.filename))) {
    const availableImageNames: string = (
      await control.getAvailableImageNames()
    ).join(', ');
    return `Please pass a valid filename in the 'filename' query segment. Available
  }

  if (!query.width && !query.height) {
    return null; // No size values
  }

  // Check for valid width value
  const width: number = parseInt(query.width || '');
  if (Number.isNaN(width) || width < 1) {
    return "Please provide a positive numerical value for the 'width' query segment
  }

  // Check for valid height value
  const height: number = parseInt(query.height || '');
  if (Number.isNaN(height) || height < 1) {
    return "Please provide a positive numerical value for the 'height' query segmen
  }

  return null;
};

const images: express.Router = express.Router();

images.get('/',async (request: express.Request, response: express.Response): Promis
    const validationMessage: null | string = await validate(request.query);
    if (validationMessage) {
      response.send(validationMessage);
      return;
    }

    let error: null | string = '';

    // Create thumb if not yet available
    if (!(await control.isThumbAvailable(request.query))) {
      error = await control.createThumb(request.query);
    }

    // Handle image processing error
    if (error) {
      response.send(error);
      return;
    }

    // Retrieve appropriate image path and display image
    const path: null | string = await control.getImagePath(request.query);
    if (path) {
      response.sendFile(path);
    } else {
      response.send('something went wrong');
    }
  }
);

export default images;

```

▸ src/routes/index.ts 1

▸ dist/file.js

▸ dist/index.js

▸ dist/processing.js

▸ dist/routes/api/images.js

▸ dist/routes/index.js

▸ dist/tests/fileSpec.js

▸ dist/tests/helpers/reporter.js

▸ dist/tests/indexSpec.js

▸ dist/tests/routes/api/imagesSpec.js

▸ dist/tests/routes/indexSpec.js

▸ src/index.ts

▸ src/tests/fileSpec.ts

▸ src/tests/helpers/reporter.ts

▸ src/tests/indexSpec.ts

▸ src/tests/routes/api/imagesSpec.ts

▸ src/tests/routes/indexSpec.ts

RETURN TO PATH