

java springboot restful 28 AlbumController – Build Photo Upload for Album

In AlbumController.java -> photos method -> Updated like below

```
@PostMapping(value = "{album_id}/upload_photos", consumes = { "multipart/form-data" })
@Operation(summary = "Upload Photo into album")
@SecurityRequirement(name = "studyeasy-demo-api")
@ApiResponse(responseCode = "400", description = "Please check the payload of token")
public ResponseEntity<List<String>> photos(@RequestPart(required = true)
MultipartFile[] files,
    @PathVariable long album_id, Authentication authentication) {
    String email = authentication.getName();
    Optional<Account> optionalAccount = accountService.findByEmail(email);
    Account account = optionalAccount.get();
    Optional<Album> optionalAlbum = albumService.findById(album_id);
    Album album;
    if (optionalAlbum.isPresent()) {
        album = optionalAlbum.get();
        if (account.getId() != album.getAccount().getId()) {
            return ResponseEntity.status(HttpStatus.BAD_GATEWAY).body(null);
        }
    } else {
        return ResponseEntity.status(HttpStatus.BAD_GATEWAY).body(null);
    }

    List<String> fileNamesWithSuccess = new ArrayList<>();
    List<String> fileNamesWithError = new ArrayList<>();
    Arrays.asList(files).stream().forEach(file -> {
        //Checking the type of the file is correct or not
        String contentType = file.getContentType();

        if(contentType.equals("image/png")
        || contentType.equals("image/jpg")
        || contentType.equals("image/jpeg")){
            fileNamesWithSuccess.add(file.getOriginalFilename());
            // When we are storing the file in the database, there is a
possibility that file name from the user
            // is repeating and if that happens, then file from the user will
get repeated and in server will
            // get replaced.
            // In order to prevent that, we need to create a random string.;
            int length = 10;
            boolean useLetters = true;
```

```

        boolean useNumbers = true;
        try {
            String fileName = file.getOriginalFilename();
            String generatedString = RandomStringUtils.random(length,
useLetters, useNumbers);
            String final_photo_name = generatedString + fileName;
            String absolute_fileLocation =
AppUtil.get_photo_upload_path(final_photo_name, album_id);
            Path path = Paths.get(absolute_fileLocation);
            Files.copy(file.getInputStream(), path,
StandardCopyOption.REPLACE_EXISTING);
        } catch (Exception e) {
            // TODO: handle exception
        }
    }
});
return null;
}

```

Add below code in AlbumService.java

```

public Optional<Album> findById(long id){
    return albumRepository.findById(id);
}

```

Create a util -> AppUtils -> Inside that create file -> AppUtil.java

```

package org.studyeasy.SpringRestdemo.util.AppUtils;

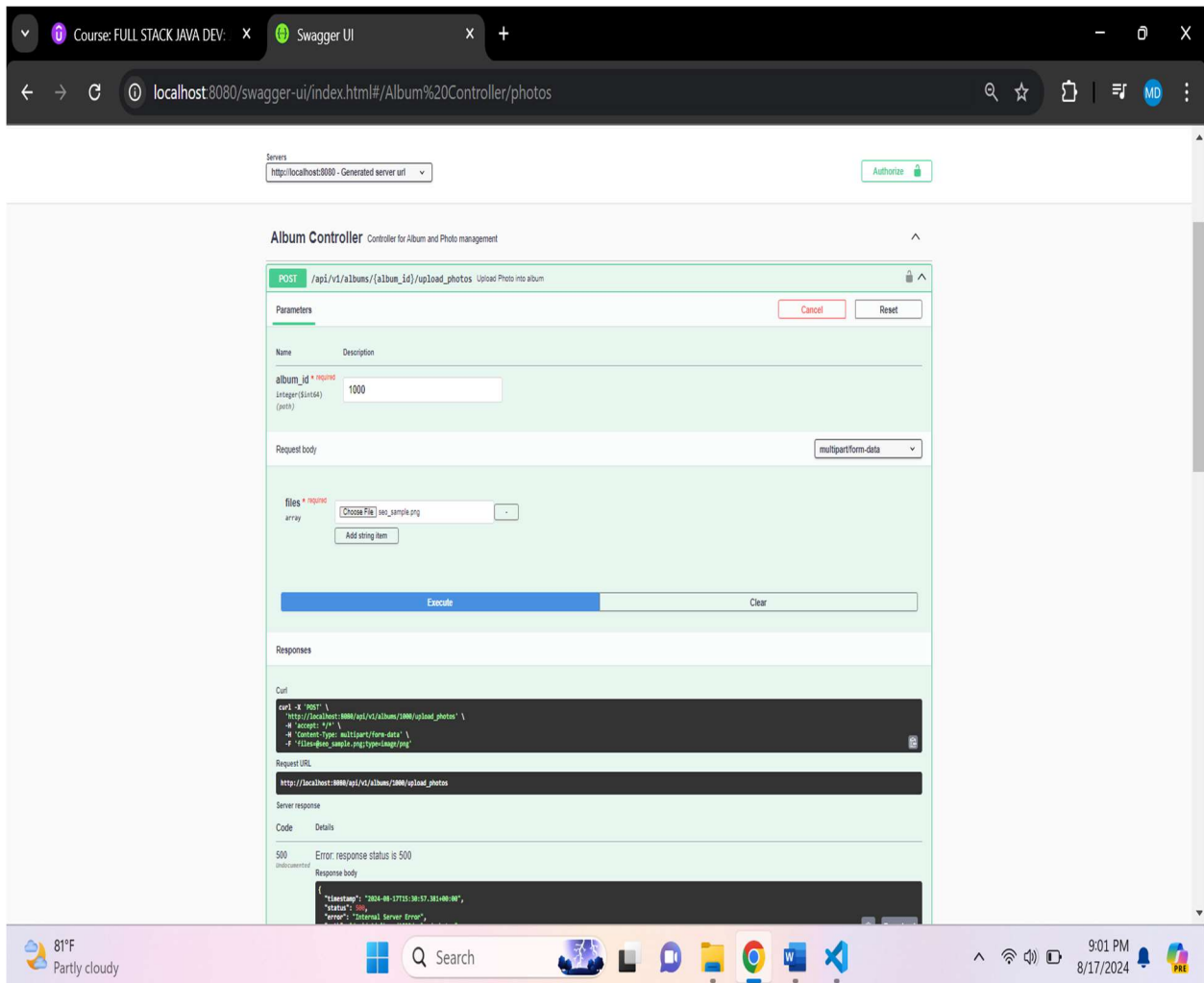
import java.io.File;
import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Paths;

//Receiving the filename and also the album_id
public class AppUtil {
    public static String get_photo_upload_path(String fileName, long album_id)
throws IOException {
        Files.createDirectories(Paths.get("src\\main\\resources\\static\\uploads\\
\" + album_id));
        return new File("src\\main\\resources\\static\\uploads\\" +
album_id).getAbsolutePath() + "\\\" + fileName;
    }
}

```

Output

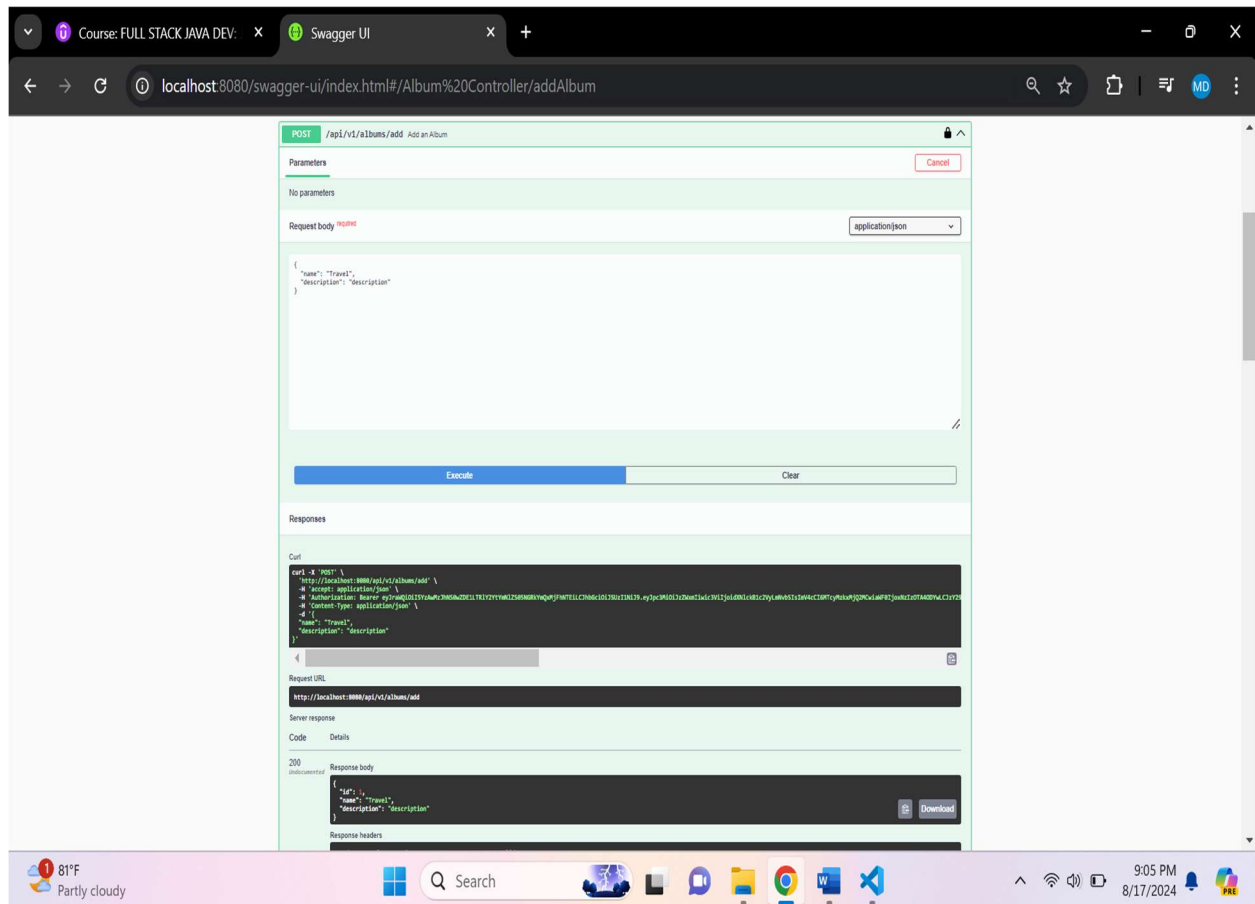
When you try to upload the photo with an invalid id like 100, you get 500 error



And you also get error in console like below

java.lang.NullPointerException: Cannot invoke
"org.springframework.security.core.Authentication.getName()" because "authentication" is null

Add Album



The id is 1 and use it in Api for uploading folders

The next output is in the pdf form