## МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

Санкт-Петербургский национальный исследовательский университет информационных технологий механики и оптики

Мегафакультет трансляционных информационных технологий Факультет информационных технологий и программирования Лабораторная работа №5

По дисциплине «Web-программирование»

Выполнил:

Студент группы №М33091 Сабитов Искандэр

САНКТ-ПЕТЕРБУРГ

2022

Устанавливаем необходимые зависимости и привязываем ValidationPipe в main.ts:

```
app.useGlobalPipes(new ValidationPipe())
```

Добавляем проверки в dto:

```
import { ApiProperty } from '@nestjs/swagger';
import { IsEmail, IsNotEmpty, IsString, MinLength } from "class-validator";

export class CreateUserDto {
    @IsNotEmpty()
    @IsString()
    @MinLength( min: 3)
    @ApiProperty()
    name: string;

@IsEmail()
    @ApiProperty()
    email: string;

@IsNotEmpty()
    @MinLength( min: 5)
    @ApiProperty()
    password: string;
}
```

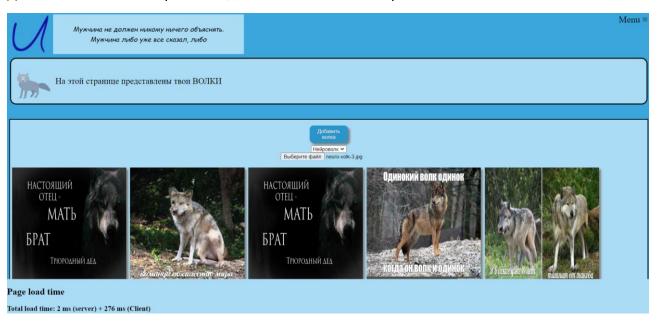
Создаем http-exepction.filter и добавляем их в контроллеры:

```
@ApiResponse( options: {
    status: 200,
    description: 'User updated',
})
@ApiResponse( options: {
    status: HttpStatus.BAD_REQUEST,
    description: 'Bad request',
})
@ApiResponse( options: {
    status: HttpStatus.NOT_FOUND,
    description: 'User not found',
})
@ApiResponse( options: {
    status: HttpStatus.FORBIDDEN,
    description: 'You can not change this user',
})
@ApiResponse( options: {
    status: HttpStatus.INTERNAL_SERVER_ERROR,
    description: 'Internal unknown error',
})
@UseFilters(new HttpExceptionFilter())
@ApiOperation( options: {summary: 'Update info about user by id'})
@Patch( path: ':id')
async update(@Param( property: 'id') id: number, @Body() updateUserDto: UpdateUserDto) {
    return await this.usersService.update(id, updateUserDto);
}
```

Пробуем вызвать исключение:



## Добавляем новое изображение для пользователя example1:



## Изображение добавлено:



	驔 id 🕏	📭 authorId 🕏	.⊞ imageURL ÷	<b>I</b> wolfType
1	5		data:image/jpeg;base64,/9j/4AAQSkZJRgA	common
2	6		data:image/jpeg;base64,/9j/4AAQSkZJRgA	neuro
3	7		data:image/jpeg;base64,/9j/4AAQSkZJRgA	common
4	8		data:image/jpeg;base64,/9j/4AAQSkZJRgA	common
5	9		data:image/jpeg;base64,/9j/4AAQSkZJRgA	neuro
6	10		data:image/jpeg;base64,/9j/4AAQSkZJRgA	neuro

Get запросы возвращают изображения в соответствии с фильтрами:

