

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Дисциплина:** Бек-энд разработка

**Отчет**

**Лабораторная работа 6**

**Выполнил:**

**Илья Чебан**

**К3441**

**Проверил:  
Добряков Д. И.**

**Санкт-Петербург**

**2025 г.**

## Задача

Добавить очередь в проект

## Ход работы

подключаем Bull в [app.module.ts](#)

```
BullModule.forRoot({
    redis: {
        host: 'localhost',
        port: 6379,
    },
}) ,
```

Имитируем очередь

```
import { Process, Processor } from '@nestjs/bull';
import bull from 'bull';

@Processor('user')

export class UserProcessor {
    @Process('processUser')
    async handleProcessUser(job: bull.Job<{ userId: string }>) {
        console.log('Обработка квиза:', job.data.userId);

        await new Promise(resolve => setTimeout(resolve, 3000));

        return { status: 'completed', userId: job.data.userId };
    }
}
```

описываем логику в сервисе и в контроллере

```
import { Injectable } from '@nestjs/common';
import { InjectQueue } from '@nestjs/bull';
import bull from 'bull';

@Injectable()
export class userService {

    constructor(@InjectQueue('user') private userQueue: bull.Queue) {}

    async adduserToQueue(userId: string) {
        await this.userQueue.add('processuser', { userId }, {
            attempts: 3,
            backoff: 1000,
        });
    }
}
```

```
@Post(':id/process')
async function processuser(@Param('id') id: string) {
    await this.userService.adduserToQueue(id);
    return { message: 'Задача добавлена в очередь' };
}
```

Вывод: Добавили через микросервис и обрабатываем асинхронные задачи. Разобрались с работой Bull