

# Фидбек по тестовому заданию для Владимира

November 11, 2020

В целом, тестовое задание выполнено правильно. Но, с учетом, ошибок допущенных ниже. Мы ищем кандидата с большим опытом разработки на **FastAPI**.

```
4
5 class Task(BaseModel):
6     taskid: str = Field(..., title='TaskID')
7     title: str = Field(..., title='Title')
8     params: Optional[Dict[str, str]] = Field(None, title='Params')
```

---

1. В данном случае, не особо целесообразно использование Field и не особо понятно зачем она здесь примененна

```
o
9 connection = None
10 while True:
11     try:
12         sleep(5)
13         connection = pika.BlockingConnection(pika.ConnectionParameters(host='rabbit'))
14         break
15     except:
16         pass
17 channel = connection.channel()
--
```

2. Подобная конструкция подключения к RabbitMQ очереди, является опасной при работе в нескольких потоках и может повлечь за собой бесконечное заикливание в случае проблем с Rabbit. Рекомендую посмотреть как создавать коннект к каналу с конструкцией with  
По поводу защиты, RabbitMQ желательно всегда инициализировать с логином и паролем

3.

Не обязательно выполнять объявление принимаемых параметров через Body и выписывать title. Бизнес логику различных тасков (по типу как, здесь публикация в канал) лучше не писать в о  
вьяхах, а вынести в отдельный utils файл.

Так же не особо понятна передача task в string формате, если для адекватной сериализации в обработчике, лучше было бы использовать Json

```
@app.post('/add_task', status_code=status.HTTP_201_CREATED,)
def hello(task: Task = Body(..., title='The phone number to be checked')):
    channel.basic_publish(exchange='', routing_key='task_queue', body=task.to_string())
```

---