

Тестовое задание: Система выдачи кредитов

Описание

Мы ожидаем, что вы продемонстрируете навыки проектирования кода с использованием архитектурных принципов:

- **DDD (Domain-Driven Design)**,
- **Clean Architecture**,
- **SOLID**,
- модульность и разделение ответственности.

Не требуется жесткое следование одной конкретной методологии — важно показать структурированность кода, его читаемость и удобство сопровождения.

Реализовать функциональность, чтобы оценить ваше понимание архитектурных подходов и умение проектировать системы.

Задание

Реализовать базовую функциональность для системы выдачи кредитов с учетом правил и условий, используя **Symfony 7 / Laravel 11** framework.

Требуемый функционал

Приложение должно поддерживать следующие сценарии:

1. **Создание нового клиента.**
2. **Проверка возможности выдачи кредита. (Условия выдачи кредита)**
В этом пункте важно проявить подход с учетом масштабирования условий и гибкого добавления новых правил проверки.
3. **Выдача кредита**, включая уведомление клиента (заглушка).

Условия выдачи кредита

1. **Кредитный рейтинг клиента (Score) > 500.**
2. **Доход клиента \geq \$1000/мес.**
3. **Возраст клиента** от 18 до 60 лет.
4. Кредит выдается только клиентам из регионов:
 - **Прага (PR)**,
 - **Брно (BR)**,
 - **Острава (OS)**.
5. Клиентам из **Праги** рандомно отказывать.
6. Клиентам из **Острава** увеличивать процентную ставку на **5%**.

Сущности

Клиент:

- ФИО,
- возраст,
- адрес (город, регион),
- pin (Personal Identification Number),
- score (кредитный рейтинг),
- email,
- телефон.

Кредит:

- название,
- срок,
- процентная ставка,
- сумма.

Допущения и ограничения

- Использовать **Symfony 7 / Laravel 11**.
- Код должен быть написан с использованием PHP 8.3 и соблюдать стандарты **PSR**.
- Данные клиентов и кредитов можно хранить статически (в массиве или конфигурационном файле).
- Уведомление клиента реализовать через запись в лог (например: [Дата/время] Уведомление клиенту [Имя клиента]: Кредит одобрен/отклонен.).
- Наличие тестов будет плюсом.
- Применение инструментов статического анализа и Docker тоже будет плюсом.

Пример данных

Чтобы упростить задачу, можно использовать такие данные:

Пример клиента:

json

Copy code

```
{  
  "name": "Petr Pavel",  
  "age": 35,  
  "region": "PR",
```

```
"income": 1500,  
"score": 600,  
"pin": "123-45-6789",  
"email": "petr.pavel@example.com",  
"phone": "+420123456789"  
}
```

Пример кредита:

json

Copy code

```
{  
  "name": "Personal Loan",  
  "amount": 1000,  
  "rate": "10%",  
  "start_date": "2024-01-01",  
  "end_date": "2024-12-31"  
}
```

Ожидания от реализации

- **Базовые сценарии:** минимальный набор REST API (например, эндпоинты для создания клиента, проверки кредита, выдачи кредита).
- **Читаемость кода:** использовать понятные названия классов, методов и переменных.
- **Логичное разделение кода**
- **Простота использования:** достаточно README-файла с инструкцией по запуску.

Важно!

Тестовое задание служит исключительно для оценки ваших навыков и не предполагает использования в коммерческих или некоммерческих проектах.