# **Разработчик .NET Core**

## Постановка задачи

Разработать два взаимодейтвующих между собой REST сервиса для реализации возможности приема заявки на кредит и получения статуса рассмотрения этой заявки.

Необходимо спроектировать структуру БД и разработать сервисы с открытым API.

## Сервис заявки на кредит

REST сервис принимает заявку на кредит, сохраняет ее в базу данных. Обращается к другому сервису (см. ) с данными этой заявки. Сервис возвращает truе или false на эту заявку (скоринг), а первый меняет у себя в базе статус одобрения.

Далее, сотрудник банка, по номеру заявки может сделать запрос и получить результат одобрена или не одобрена заявка (true или false).

api/application/create – принимает заявку в формате json. Возвращает уникальный номер и Id принятой заявки в формате json.

api/application/status – возвращает результат рассмотрения заявки с некоторыми деталями по заявке по ее номеру в формате json (статус одобрения кредита)

*Пример входящего запроса:*

{

"ApplicationNum": "КД2135554",

"ApplicationDate": "2021-08-24T13:00:00",

"BranchBank": "Арбат",

"BranchBankAddr": "ул. Новая, 21",

"CreditManagerId": 678543,

"Applicant": {

"FirstName": "Иванов",

"MiddleName": "Иван",

"LastName": "Диванович",

"DateBirth": "1974-03-03T00:00:00",

"CityBirth": "Москва",

"AddressBirth": "Место рождения",

"AddressCurrent": "Прописка",

"INN": "7747456",

"SNILS": "12345567",

"PassportNum": "4508 8345567"

},

"RequestedCredit": {

"CreditType": 1,

"RequestedAmount": 10000000.0,

"RequestedCurrency": "rur",

"AnnualSalary": 14400000.0,

"MonthlySalary": 120000.0,

"CompanyName": "IBM LLC",

"Comment": "Комментарий к кредитной заявке"

}

}

*Пример получения статутса по заявке:*

{

"Id": 10010,

"ApplicationNum": "КД2135554",

"ApplicationDate": "2021-08-24T13:00:00",

"BranchBank": "Арбат",

"BranchBankAddr": "ул. Новая, 21",

"CreditManagerId": 678543,

"Applicant": {

"FirstName": "Иванов",

"MiddleName": "Иван",

"LastName": "Диванович"

},

"RequestedCredit": {

"CreditType": 1,

"RequestedAmount": 10000000.0,

"RequestedCurrency": "rur"

},

"ScoringStatus": true,

"ScoringDate": "2021-08-24T16:00:00"

}

## Сервис оценки возможности выдачи кредита

Сервис эмуляции работы скорринга – оценки возможности выдачи кредита. True или false на этапе скоринга можно возвращать случайным образом 😊

api/scoring/evaluate – мок сервиса скоринга, который возвращает решение по заявке.

*Пример результата скорринга:*

{

"ScoringStatus": false

}

## Технический челлендж

*Уровень: middle и выше*

Доработать сервис оценки возможности выдачи кредита таким образом, чтобы взаимодействие с ним могло осуществлялось в асинхронном режиме. Сервис принимает сообщение и возвращает статус 200 (или любой другой индикатор, который вы будете воспринимать, как то, что запрос принят системой).

Через настраиваемый промежуток времени, например 10 или 40 секунд, сервис отдает ответ в режиме callback первому сервису заявки на кредит, который поместит результат скоринга в свою БД.

## **Технологии:**

* . NET Core
* Microsoft SQL Server, PostgreSQL или Oracle

*Плюсами будут являться:* использование Dapper или EF, подключенный swagger, наличие логирования и наличие Unit тестов.

## Длительность:

* от 4 часов разработки и 60 мин. на обсуждение результатов.

## Ожидаемый результат:

* Работающий сервис, запущенный на локальном компьютере или опубликованный в публичном доступе.
* Исходный код, опубликованный на GitHub или отправленный по email.