ИНСТРУКЦИЯ ПО УСТАНОВКЕ ЭКЗЕМПЛЯРА ПО **FINIST-DIGITALBANK.DBO**

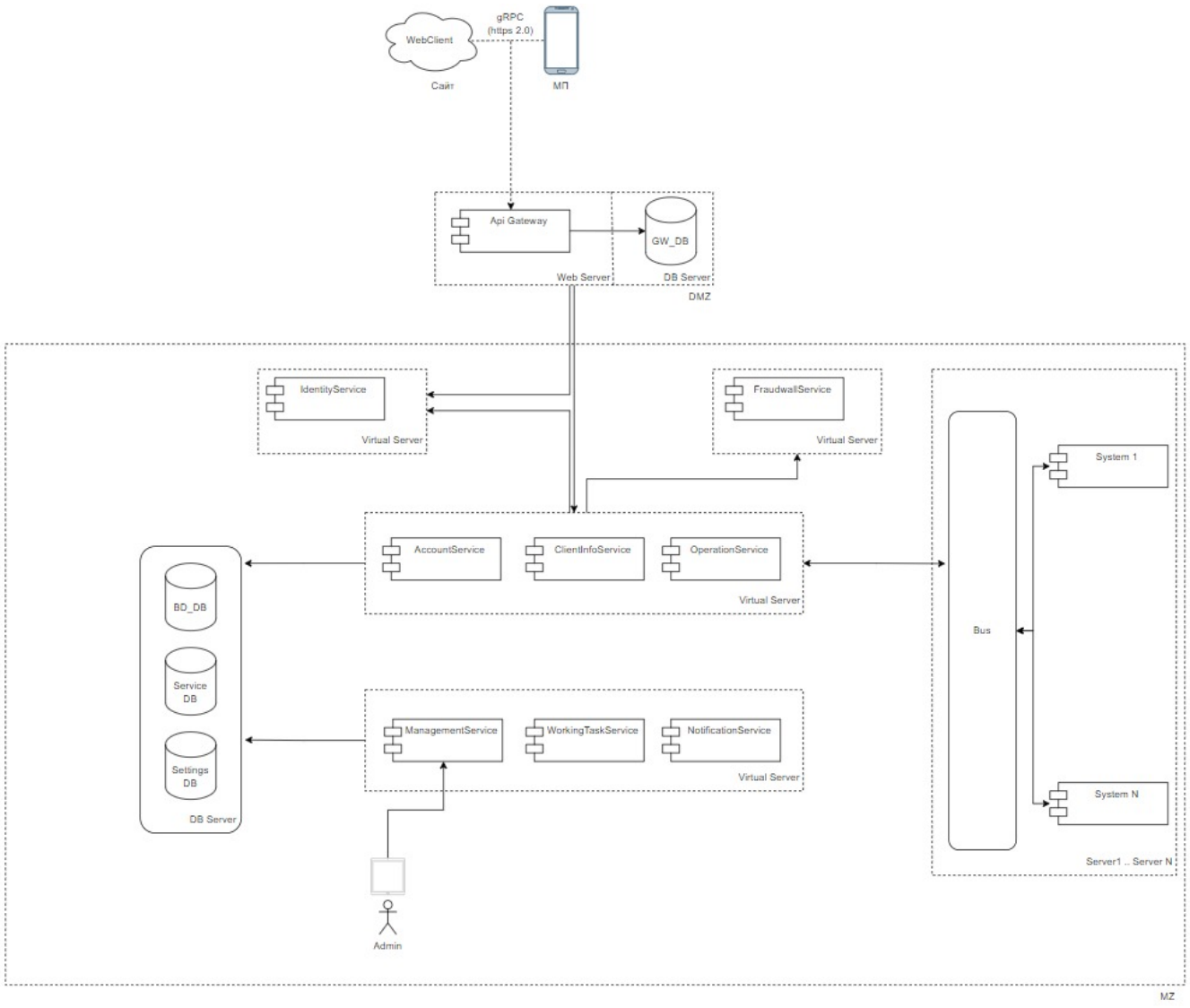
г. Казань, 2024

Оглавление

Оглавление 2
1 Архитектура решения 3
2 Поставка 4
3 Предварительная настройка 4
4 Конфигурирование сервисов 4
4.1 Appsettings.json 5
4.1.1 HTTP/HTTPS 6
4.2 LDAP 7
4.3 Настройка веб-сервера, HTTPS 8
5 Проверка работоспособности 9
6 Проблемы, которые могут возникнуть ..... **Ошибка! Закладка не определена.**
6.1 При авторизации по корректному логину/паролю ничего не происходит. ..... **Ошибка! Закладка не определена.**
6.2 При авторизации возникает ошибка 502 Bad gateway .. **Ошибка! Закладка не определена.**
6.3 При авторизации возникает ошибка 408 Request timeout **Ошибка! Закладка не определена.**
7 Технические требования 11
7.1 Требования к СУБД 11
7.1.1 Программное обеспечение СУБД 11
7.1.2 Аппаратное обеспечение сервера: 11
7.2 Требования к конфигурации сервера приложений 11
7.2.1 Программное обеспечение сервера 11
7.2.2 Аппаратное обеспечение сервера 11
7.3 Рабочая станция: ..... **Ошибка! Закладка не определена.**
7.3.1 Программное обеспечение ОС рабочей станции ... **Ошибка! Закладка не определена.**
7.3.2 Аппаратное обеспечение рабочей станции ..... **Ошибка! Закладка не определена.**

1 Архитектура решения

Система Дистанционного Банковского Обслуживания Finist-DigitalBank.DBO (далее по тексту - ДБО) реализована на базе микросервисной архитектуры, с возможностью динамического подключения сервисов. Кажый из сервисов может быть горизонтально масштабирован для увеличения пропускной способности, а общий набор сервисов может быть масштабирован как вертикально, так и горизонтально для обеспечения отказоустойчивости и равномерного распределения возрастающей нагрузки.



Архитектура системы Дистанционного Банковского Обслуживания (Интернет-банк)

2 Поставка

В поставке участвуют не менее девяти архивов:

- **ApiGateway** – прокси сервис принимающий запросы от МП и Веб-клиента. Исходя из типа запроса, маршрутизирует его на нужный сервис. Выполняет роль балансировщика.
- **IdentityService** – сервис для выполнения идентификации. Выполняет роль обеспечения защиты между несанкционируемыми запросами из ДМЗ в МЗ.
- **AccountService** – сервис для получения информации по счетам и привязанными к ним картам клиента. Синхронизация данных между рассматриваемой системой и действующими системами может происходить как прямая, так и обратная.
- **ClientInfoService** – сервис для получения информации по клиенту и его данным. Синхронизация данных между рассматриваемой системой и действующими системами может происходить как прямая, так и обратная.
- **OperationService** – сервис для проведения финансовых операций. Каждая финансовая операция перед фактическим исполнением обращается к сервису FraudwallService для формирования риск индикатора операции.
- **FraudWallService -** сервис для предотвращения проведения мошеннических операций.
- **WorkingTaskService** - сервис для исполнения периодичных задач в фоновом режиме. Пример - синхронизация данных, установка финального статуса операции и пр.
- **NotificationService** - сервис рассылки уведомлений. Уведомление может быть отправлено 3 путями - SMS, PUSH, почта. Для уведомления администраторов предусмотрена интеграция с Zabbix. Также, предусмотрена работа с SIEM системой MaxPatrol для уведомления о нештатном поведении системы.
- **ManagementService** - сервис администрирования. Администраторы и бизнес пользователи системы могут иметь доступ к интерфесам управления процессами системы. Для разграничения доступа используется ролевая модель. Аутентификация происходит по протоколу Kerberos. Получение данных о пользователе системы из AD происходит через LDAP.

3 Предварительная настройка

Необходимо настроить учётные записи, под которыми будет осуществляться работа веб-приложения.

Учётная запись должна обладать правами уровня Администратора.

4 Конфигурирование сервисов

Каждым сервис конфигурируется настройками в БД и файлом настроек ***appsettings.json***.

4.1 Appsettings.json

```
{
  "AllowedHosts": "*",
  "Kestrel": {
    "EndpointDefaults": {
      "Protocols": "Http1AndHttp2"
    },
    "Endpoints": {
      "gRPC": {
        "Url": "https://localhost:5075",
        "Protocols": "Http1AndHttp2"
      }
    }
  },
  "MethodConfig": {
    "RetryMaxAttempts": 3,
    "MinDelayBackoff": 1,
    "MaxDelayBackoff": 5,
    "BackoffMultiplier": 1.5
  },
  "JwtBearerTokenSettings": {
    "SecretKey": "secretKey@xxxxxxxxxx@KJHA)UDYAS__",
    "Audience": "http://localhost:5143",
    "Issuer": "http://localhost:5143",
    "ExpiryTimeInSeconds": 360,
    "RefreshTokenExpiryInDays": 7
  },

  // Провайдер работы с БД. PsSQL - PostgreSQL; MsSQL - Microsoft SQL Server;
  "DatabaseProvider": "PsSQL",

  // InfrastructureDB - единая БД для работы с логами, метриками и настройками сервисов.
  "ConnectionStrings": {
    "InfrastructureDB":
"Host=dev.domen.com;Port=5432;Database=Bank_InfrastructureDB;Username=postgres;Password=xxxxxx"
  }
}
```

Секция ***AllowedHosts***:

Указываются адреса, с которых сервер принимает запросы.

Секция ***Kestrel***:

EndpointDefaults – системная секция. Версия протокола общения gRPC.

ServerOptions / Limits / MaxRequestBodySize – Максимальный размер тела запроса, в байтах.

4.1.1 HTTP/HTTPS

В секции Endpoints находятся настройки выставленных адресов для gRPC. **Url** – адрес, по которому будет осуществляться общение с сервером.

- **Http** – вариант адреса без сертификата.
- **Https** – вариант адреса с сертификатом. Настройки сертификата указываются в секции Certificate. В секции Certificate указывается путь (Параметр Path) до pfx-сертификата и пароль от сертификата (Параметр Password).

Секция ***MethodConfig*** – секция управления повторными вызовами при неуспешном обращении к другим сервисам.

- **RetryMaxAttempts –** Максимальное количество попыток вызова, включая исходную попытку
- **MinDelayBackoff -** Минимальное значение задержки
- **MaxDelayBackoff -** Максимальное значение задержки
- **BackoffMultiplier -** Задержка будет умножена на это значение после каждой повторной попытки и будет увеличиваться экспоненциально

Секция ***JwtBearerTokenSettings*** – секция настроек для обращения к IdetityService.

- **SecretKey –** секретный ключ на основе которого генерируется токен
- **Audience -** задает допустимое значение аудитории для полученого токена безопасности
- **Issuer -** строка, которая идентифицирует принципала, выдавшего JWT
- **ExpiryTimeInSeconds –** время жизни токена в секундах
- **RefreshTokenExpiryInDays –** Период обновления токена (в днях)

Секция ***DatabaseProvider*** – указывается тип провайдера для работы с БД.

Секция ***ConnectionStrings*** – указывается строка соединения с БД шлюза.

4.2 LDAP

В настройках, хранящихся в БД IdentityService есть возможность настроить LDAP сервер, для аутентификации пользователей через доменные учётные данные. LDAP серверов может быть несколько.

Пример:

```
{
        "Ldap": {
                "Server": [
                        {
                                "GEstAuthentication": {
                                        "Bind": {
                                                "_dynamic": "true",
                                                "_dn": "uid=%%login%%,ou=Users,dc=some_server,dc=com",
                                                "_password": "some_password"
                                        },
                                        "Search": {
                                                "_root": "dc=some_server,dc=com",
                                                "_filter": "(&(uid=%%login%%)(objectClass=posixAccount))"
                                        }
                                },
                                "_key": "some_server",
                                "_host": "some_server.com",
                                "_port": "389",
                                "_useSsl": "false",
                                "_trustServerCertificate": "false",
                                "_repeatedAuthenticationInterval": "0:10:0",
                                "_authenticationType": "Basic",
                                "_protocolVersion": "3",
                                "_displayName": "Some server"
                        },
                        {
                                "GEstAuthentication": {
                                        "Bind": {
                                                "_dynamic": "true",
                                                "_dn": "%%login%%@domen.com"
                                        },
                                        "Search": {
                                                "_root": "dc=domen,dc=com",
                                                "_filter":
"(&(sAMAccountName=%%login%%)(objectCategory=person)(objectClass=user)(!(userAccountControl:1.2.840.113556.1.4.803:=
2)))"
                                        }
                                },
                                "_key": "finist",
                                "_host": "bighead.domen.com",
```

```
                        "_port": "389",
                        "_useSsl": "false",
                        "_trustServerCertificate": "false",
                        "_repeatedAuthenticationInterval": "0:10:0",
                        "_authenticationType": "Basic",
                        "_protocolVersion": "3",
                        "_displayName": "EsterDev (LDAP)"
                },
                {
                        "GEstAuthentication": {
                                "Bind": {
                                        "_dynamic": "true",
                                        "_dn": "uid=%%login%%,ou=Users,dc=ldaptest,dc=com"
                                },
                                "Search": {
                                        "_root": "dc=ldaptest,dc=com",
                                        "_filter": "(&(uid=%%login%%)(objectClass=posixAccount))"
                                }
                        },
                        "_key": "ldaptest",
                        "_host": "192.168.1.114",
                        "_port": "389",
                        "_useSsl": "false",
                        "_trustServerCertificate": "false",
                        "_repeatedAuthenticationInterval": "0:10:0",
                        "_authenticationType": "Basic",
                        "_protocolVersion": "3"
                }
        ]
}
```

4.3 Настройка веб-сервера, HTTPS

Для подписания веб-сайта сертификатом потребуется связка публичный сертификат и секретный ключ.

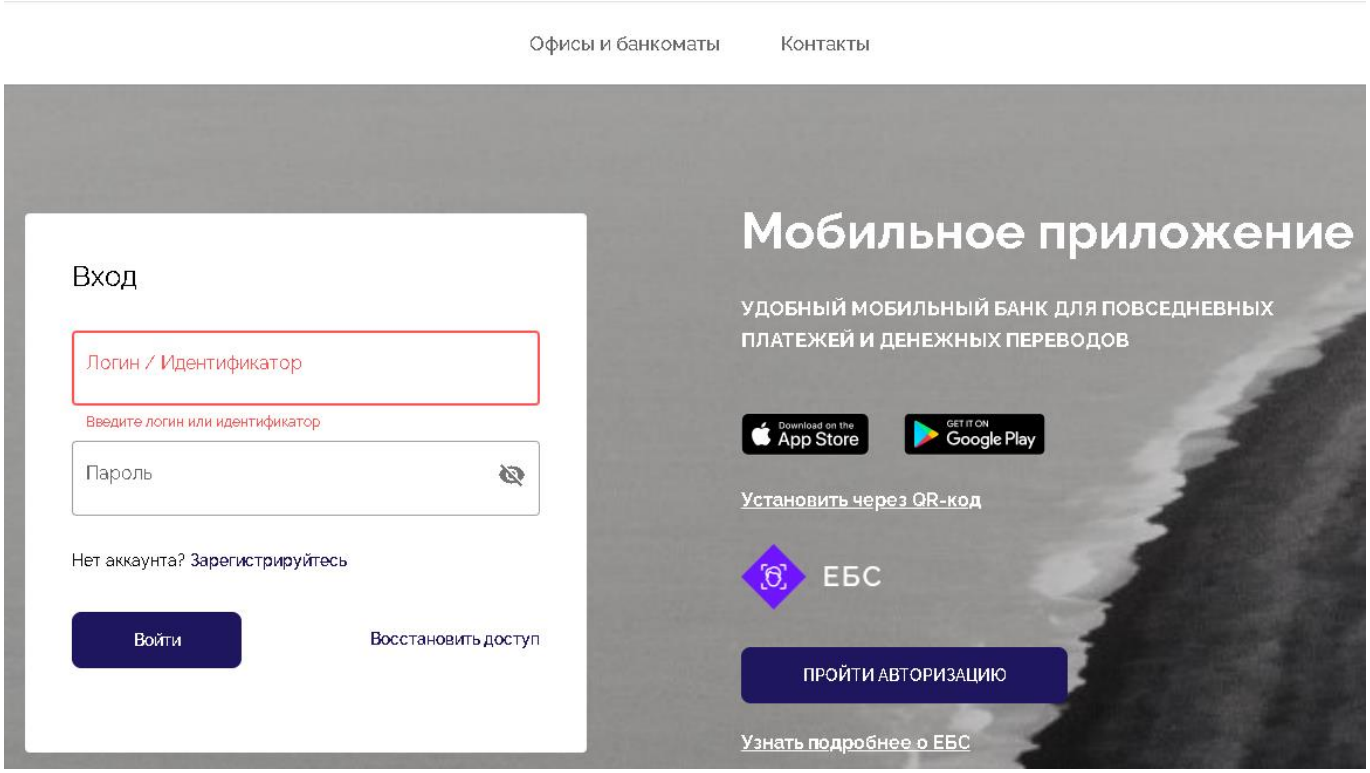
```
ssl_certificate     www.example.com.crt;
ssl_certificate_key www.example.com.key;
```

Секретный ключ (ssl_certificate_key) необходимо хранить в файле с ограниченным доступом, но Nginx'y необходимы права для чтения на этот файл.

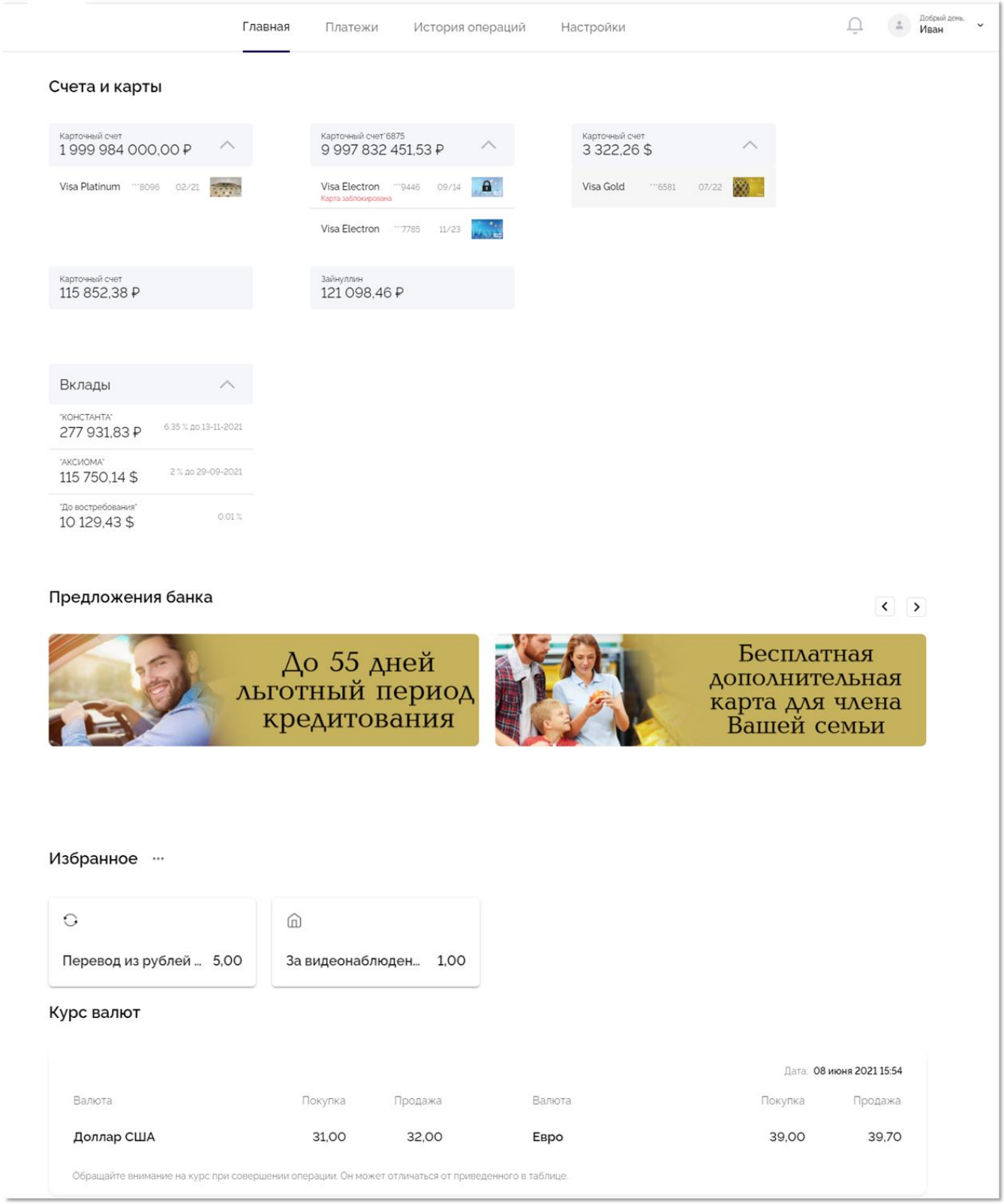
5 Проверка работоспособности

Открыть веб-браузер, ввести адрес машины/глобальный адрес машины, на которой развёрнуто ДБО.

При корректной настройке веб-сервера/указания адреса – откроется страница авторизации.



На странице указываем логин/пароль пользователя ДБО. Если всё настроено корректно, то откроется окно основной страницы системы ДБО:



6 Технические требования

6.1 Требования к СУБД

6.1.1 Программное обеспечение СУБД

- PostgreSQL - рекомендуется устанавливать только для ОС на базе Linux:
- PostgreSQL от версии 15;
- pgAdmin IV

6.1.2 Аппаратное обеспечение сервера:

- Процессор: x86-совместимый 64-разрядный (Intel, AMD); 8 ядер с частотой от 2 ГГц.
- Оперативная память: 64ГБ.
- Дисковая подсистема: от 400 ГБ.

Канала связи: от 100 Mbit/sec, при кол-ве пользователей более 50 рекомендуется от 1000 Mbit/sec.

6.2 Требования к конфигурации сервера приложений

6.2.1 Программное обеспечение сервера

- Linux, поддерживающий .Net 6.0:
- Nginx версия не ниже 1.12
- Необходимо установить пакеты libc6-dev и libgdiplus
- .Net 6.0

6.2.2 Аппаратное обеспечение сервера

- Основные требования к аппаратному обеспечению вытекают из требований используемой версии ОС и СУБД.
- Процессор: x86-совместимый 64-разрядный (Intel, AMD); 8 ядер с частотой от 2 ГГц.
- Оперативная память: 32ГБ.
- Дисковая подсистема: от 200 ГБ.