Общество с ограниченной ответственностью «Делитек»

Инструкция по установке экземпляра программного обеспечения «Forkly»

2025 г.

ОГЛАВЛЕНИЕ

1 Функциональное описание и средства разработки

Back-end	– Язык программирования C# – Платформа Microsoft .NET 9.0 (открытые исходные коды)
Front-end	– Язык программирования JavaScript – Платформа для создания rich internet applications Angular (открытые исходные коды).

Система построена на основе сервисной архитектуры. Основные блоки системы реализованы как веб-сервисы и взаимодействуют друг с другом по протоколам, используемым в Интернет- и Интранет- сетях.

Базовые компоненты взаимодействуют по REST API. REST API описывается с использованием технологии OpenAPI 3.0, что позволяет предоставить как человеко-, так и машиночитаемую документацию по программному интерфейсу системы.

В качестве формата сериализации данных при вызове REST API используется JSON.

Публикация общего REST API осуществляется с использованием технологии reverse proxy. Прокси-сервер – инфраструктурный компонент, который используется для распределения нагрузки при горизонтальном масштабировании системы, сведения API-вызовов к одному доменному адресу, осуществления кеширования, авторизации и аутентификации запросов. В качестве прокси-сервера поддерживаются NGINX, HAProxy или YARP. В инструкции описан рекомендуемый метод по настройке прокси-сервера NGINX.

REST API и сервера очередей также могут использоваться для интеграции с внешними системами клиента.

Хранение данных сервисов осуществляется с использованием СУБД с открытым исходным кодом PostgreSQL.

2 Технические требования

Сервер «Forkly»:

- ПО – Ubuntu Server 24.04;
- 4 Гб ОЗУ или более;
- Процессор 2,6 Ггц, 4 ядра или более;
- Жесткий диск (SSD) 250Гб свободного места и более;
- Наличие Интернета;
- Внешний IP-адрес для подключения мобильного приложения;
- Доступ к портам сервера 80 и 443 из Интернета по внешнему IP;
- Зарегистрированное и привязанное к внешнему IP доменное имя;

– SSL-сертификат для организации HTTPS-подключения по зарегистрированному имени.

Примечание:

На сервере может быть развернута исключительно внедряемая система, всё необходимое для нее ПО. Наличие и функционирование на сервере иного ПО не допускается.

3 Установка и настройка серверной части

3.1 Требования к системе

Для установки серверной части системы «Forkly» необходим физический или виртуальный сервер с установленной ОС Ubuntu Server 24.04.

Для работы сервера должны быть установлены следующие компоненты:

– NGINX
– ASP.NET Core Runtime 9.0
– PostgreSQL 16

Все дальнейшие действия производятся от пользователя root (администратор) в консоли сервера.

3.2 Предварительная настройка системы

Для корректной установки пакетов в системе необходимо обновить репозитории следующей командой:

```
sudo apt update
```

3.3 Установка ASP.NET Core Runtime

Для установки ASP.NET Core Runtime версии 9 необходимо выполнить следующие действия:

– Добавить .NET backports репозиторий:
  `sudo add-apt-repository -y ppa:dotnet/backports`
– Обновить apt-репозитории:
  `sudo apt-get update`
– Установить ASP.NET Core Runtime:
  `sudo apt-get install -y aspnetcore-runtime-9.0`

3.4 Установка и настройка PostgreSQL

Для установки PostgreSQL версии 16 необходимо ввести следующую команду:

```
sudo apt install -y postgresql
```

Далее, необходимо убедиться, что была установлена необходимая версия ПО при помощи команды:

```
psql --version
```

В консоли должен отобразиться текст подобного содержания:

```
psql (PostgreSQL) 16.10 (Ubuntu 16.10-0ubuntu0.24.04.1)
```

Версия после «(PostgreSQL)» должна отобразиться начиная с «16…»

После установки PostgreSQL, необходимо провести её конфигурацию. Нужно ввести следующую команду для открытия консоли PostgreSQL от имени пользователя «postgres»:

```
sudo -u postgres psql template1
```

В консоли должно отобразиться сообщения подобного содержания:

```
psql (16.10 (Ubuntu 16.10-0ubuntu0.24.04.1))
Type "help" for help.

template1=#
```

Далее, нужно поменять пароль пользователю «postgres» следующей командой:

```
ALTER USER postgres with encrypted password 'НОВЫЙ_ПАРОЛЬ';
```

Где «НОВЫЙ_ПАРОЛЬ» - пароль для пользователя postgres. Его необходимо запомнить – далее он будет необходим в конфигурации ПО.

После ввода команды должно отобразиться сообщение «ALTER ROLE», после чего можно выйти из программы «psql» комбинацией клавиш «Ctrl+D» или вводом команды «quit».

Далее, необходимо содержимое файла «/etc/postgresql/16/main/pg_hba.conf» заменить на следующие строки:

```
# TYPE  DATABASE        USER            ADDRESS                 METHOD

# "local" is for Unix domain socket connections only
local   all             all                                     md5
# IPv4 local connections:
host    all             all             127.0.0.1/32            md5
# IPv6 local connections:
host    all             all             ::1/128                 md5

host    all             all             0.0.0.0/0               md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
local   replication     all                                     md5
host    replication     all             127.0.0.1/32            md5
host    replication     all             ::1/128                 md5
```

После сохранения файла необходимо перезапустить сервис PostgreSQL следующей командой:

```
sudo systemctl restart postgresql.service
```

3.5 Установка сервера «Forkly»

Для установки сервера «Forkly», необходимо разместить файлы сервера «Forkly» на машине в любом месте. После размещения сервера необходимо сделать исполняемый файл «Forkly.Web» доступным к выполнению. Для этого, находясь в папке с исполняемым файлом, необходимо ввести следующую команду:

```
sudo chmod +x Forkly.Web
```

3.6 Настройка базы данных в PostgreSQL

Для настройки базы данных в PostgreSQL, необходимо отредактировать файл «appsettings.json» и запустить сервер «Forkly». В файле «appsettings.json» нужно изменить данные для подключения к PostgreSQL – ввести пароль под ключом «Password=...» от пользователя «postgres», который был указан во время конфигурации PostgreSQL. Указать пароль необходимо в двух полях - «hangfire» и «database»:

```
//настройка hangfire
  "hangfire": {
    //строка подключения
    "connection":
"Username=postgres;Database=forkly;Password=ПАРОЛЬ;Host=localhost;Pool
ing=true"
  },

  //настройки бд
  "database": {
    //строка подключения к бд

"connectionString":"Username=postgres;Database=forkly;Password=ПАРОЛЬ;
Host=localhost;Pooling=true",
    //автоматическое выпонление миграций при старте, если они
необходимы
    "autoMigrate": true
  }
```

Где «ПАРОЛЬ» - пароль, указанный при настройке PostgreSQL ранее.

После указания пароля БД необходимо запустить сервер. ПО автоматически настроит таблицы в БД PostgreSQL. Для запуска сервера, в папке с исполняемым файлом, необходимо выполнить следующую команду:

```
./Forkly.Web
```

При успешном запуске сервера и настройке БД, в консоль должно вывестись следующее сообщение в конце вывода, без ошибок:

```
info: Microsoft.Hosting.Lifetime[0]
```

```
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Production
info: Microsoft.Hosting.Lifetime[0]
      Content root path: /home/user/publish
```

После настройки базы данных сервером, в таблице с пользователями был добавлен пользователь-администратор.

3.7 Настройка API-ключа ДаДаты

В файле «appsettings.json» необходимо указать API-ключ для работы с «ДаДатой». API-ключ указывается под полем «dadata»:

```
// Параметры для обращения к сервису DaData
"dadata": {
  "apiKey": "API-КЛЮЧ"
},
```

Где «API-КЛЮЧ» - API-ключ для работы с «ДаДатой». API-ключ можно получить по инструкции в базе знаний «ДаДаты» - «Как получить API-ключ».

3.8 Настройка автоматического запуска ПО через systemd

Для автоматического запуска сервера «Forkly» необходимо создать systemd-сервис. Для этого нужно создать файл по пути «/etc/systemd/system/forkly.service» со следующим содержанием:

```
[Unit]
Description=Forkly
After=network.target
Wants=network-online.target

[Service]
Restart=on-failure
Type=notify
ExecStart=ПУТЬ К ИСПОЛНЯЕМОМУ ФАЙЛУ Forkly.Web
WorkingDirectory=ПАПКА С ИСПОЛНЯЕМЫМ ФАЙЛОМ Forkly.Web
Environment=

[Install]
WantedBy=multi-user.target
```

Где «ПУТЬ К ИСПОЛНЯЕМОМУ ФАЙЛУ Forkly.Web» - полный путь к исполняемому файлу « Forkly.Web», «ПАПКА С ИСПОЛНЯЕМЫМ ФАЙЛОМ Forkly.Web» - полный путь к папке, в котором находится исполняемый файл « Forkly.Web».

Далее, необходимо перезагрузить конфигурацию systemd. Для этого необходимо ввести следующую команду:

```
sudo systemctl daemon-reload
```

После этого необходимо активировать и включить созданный сервис. Делается это следующими командами:

```
sudo systemctl enable forkly
sudo systemctl start forkly
```

3.9 Установка NGINX

Для установки NGINX необходимо ввести следующую команду:

```
sudo apt install -y nginx
```

3.10 Настройка NGINX

Для настройки сайта в NGINX, необходимо редактировать стандартную конфигурацию сайта. Предварительно на сервер нужно загрузить HTTPS сертификаты сайта в виде двух файлов – файла сертификата и файла приватного ключа. Затем, необходимо редактировать файл, находящийся по пути «/etc/nginx/sites-available/default», заменив содержимое файла на следующий код:

```
server {
        listen                  443 ssl http2;
        listen                  [::]:443 ssl http2;
        server_name             ДОМЕН;
        ssl_certificate         ПУТЬ К СЕРТИФИКАТУ;
        ssl_certificate_key     ПУТЬ К ПРИВАТНОМУ КЛЮЧУ;
        ssl_session_timeout     1d;
        ssl_protocols           TLSv1.2 TLSv1.3;
        ssl_prefer_server_ciphers off;
        ssl_ciphers             ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-
RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-
GCM-SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-
POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384;
        ssl_session_cache       shared:SSL:10m;
        ssl_session_tickets     off;
        ssl_stapling            off;

        add_header X-Frame-Options DENY;
        add_header X-Content-Type-Options nosniff;

        #Redirects all traffic
        location / {
            proxy_pass http://127.0.0.1:5000;
```

```
    }
}
```

Где «ДОМЕН» - домен, используемый для входа в приложение, «ПУТЬ К СЕРТИФИКАТУ» - полный путь к файлу сертификата, «ПУТЬ К ПРИВАТНОМУ КЛЮЧУ» - полный путь к приватному ключу.

После сохранения файла необходимо убедиться, что сайт включён – для этого в папке «/etc/nginx/sites-enabled» должна находиться ссылка на сайт «default». Это можно проверить следующей командой:

```
ls /etc/nginx/sites-enabled -al
```

В консоль должна отобразиться информация схожего содержания:

```
drwxr-xr-x 2 root root 4096 Jan  1 00:00 .
drwxr-xr-x 8 root root 4096 Jan  1 00:00 ..
lrwxrwxrwx 1 root root   34 Jan  1 00:00 default -> /etc/nginx/sites-available/default
```

После этого, необходимо убедиться в отсутствии ошибок в файле, который был отредактирован. Это можно сделать следующей командой:

```
nginx -t
```

Должна отобразиться информация схожего содержания:

```
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

При отсутствии ошибок необходимо перезапустить NGINX для активации новой конфигурации. Делается это следующей командой:

```
nginx -s reload
```

В консоли должно либо ничего не отобразиться, либо должна отобразиться строчка схожего содержания:

```
2025/01/01 00:00:00 [notice] 8284#8284: signal process started
```

После запуска сервиса и настройки PostgreSQL и NGINX, ПО должно быть доступно по адресу, введённому при конфигурации NGINX, а также должна быть доступна локальная версия по адресу «http://127.0.0.1:5000».

Для входа в систему в роли пользователя-администратора необходимо использовать логин «admin» и пароль «1234567890». После входа в систему необходимо изменить пароль пользователя-администратора на новый.

3.11 Ограничения в процессе эксплуатации

В процессе эксплуатации категорически не рекомендуется:

- Производить какие-либо изменения параметров в файле конфигурации **appsettings.json**. Это может привести к потере связей между компонентами системы.

- Производить апгрейд или даунгрейд компонентов системы, указанных в пункте Технические требования.
- Переносить рабочую систему на другое оборудование или виртуальные машины. Это приведёт к потере лицензий, привязанных к учетным записям выездных сотрудников.
- Производить замену комплектующих частей компьютера (или компонентов виртуальной машины), на котором установлен сервер «Forkly» – материнских плат, процессоров, памяти, жестких дисков и т.п. Это может привести к потере лицензий, привязанных к учетным записям выездных сотрудников.
- Для развертывания других копий сервера потребуется приобретение дополнительных персональных лицензий.