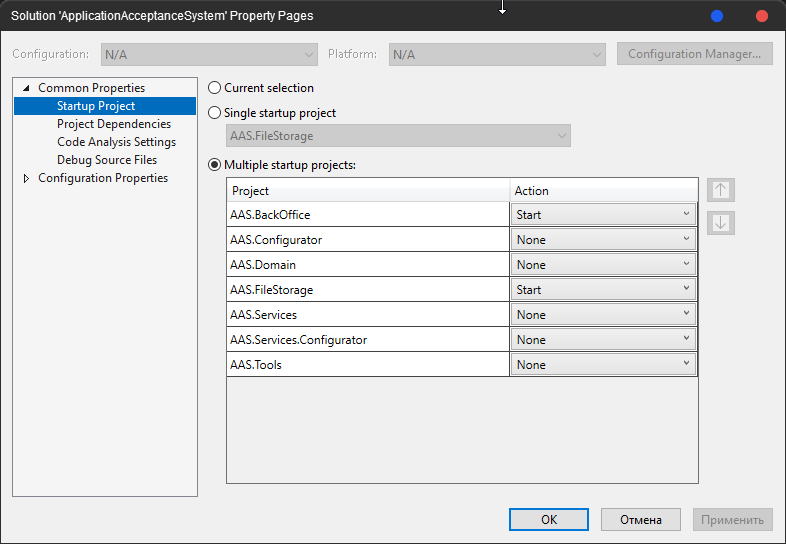
На компьютере должен быть установлен .NET 7.0, node.js 19.0.1+, PostgreSQL 15+

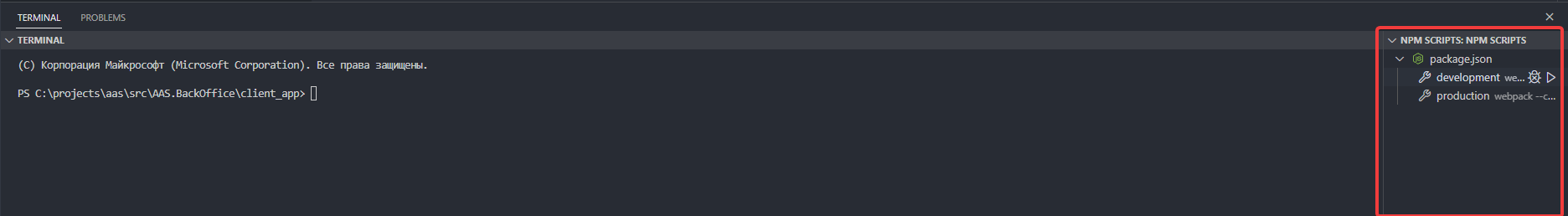
1. Скачать проект с github (<https://github.com/sivank0/aas>).
2. Создать папку FileStorage на диске C. Внутри этой папки создать папку AAS.

**Для запуска локально:**

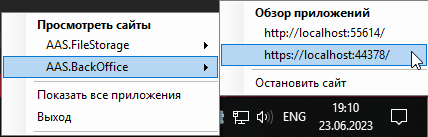
1. Сделать восстановление резервной копии БД через DBeaver. Файл бекапа есть на GitHub.
2. Открыть серверную часть проекта.
3. Необходимо поменять connectionString. Местонахождение в проекте AAS.Configurator => Configs
4. Зарегистрировать почту Gmail для отправки верификационных писем.
5. Зайти в AAS.Configurator => Configurations.cs в классе EmailVerification изменить Login и Password.
6. Нажать ПКМ на решении => Свойства => Выбрать пункты, аналогичные скриншоту. Запустить проект.



1. Зайти на клиентскую часть через VS Code. File => Open Folder. (aas/src/AAS.BackOffice/client\_app)
2. Написать в терминале npm i
3. Если есть меню выделенное красным прямоугольником, то запустить development, если нет, то написать в терминале run development.



1. В трее найти значок IISExpress (), нажать по нему ПКМ и выбрать то, что на скриншоте

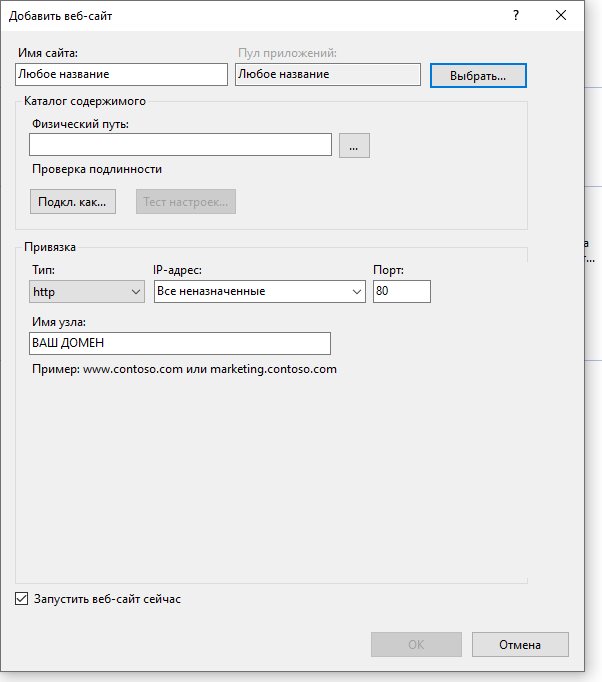


**Для запуска на сервере:**

1. Сделать восстановление резервной копии БД через DBeaver. Файл бекапа есть на GitHub.
2. Получить домен (например site.college.ru) и сделать к нему субдомен (например file.college.ru). Субдомен будет нужен для FileStorage. Так же надо получить сертификаты SSL на оба домена.
3. Необходимо поменять connectionString. Местонахождение в проекте AAS.Configurator => Configs
4. Зарегистрировать почту Gmail для отправки верификационных писем.
5. Зайти в AAS.Configurator => Configurations.cs в классе EmailVerification изменить Login и Password. В классах FileStorage и BackOffice поменять значения в полях Host на соответствующие. Проверить соответствие поля UploadFolder в классе FileStorage папке созданной в п.2.
6. Запустить следующие команды в PowerShell.

* dotnet publish **"Местоположение AAS.FileStorage.csproj**" --configuration Production --output “C:/projects/AAS.FileStorage” /p:environmentName=Production
* dotnet publish " **Местоположение** **AAS.BackOffice.csproj**" --configuration Production --output “C:/projects/AAS.BackOffice” /p:environmentName=Production

1. Перекинуть папку C:/projects на сервер в тоже место.
2. Зайти в IIS. Добавить новый сайт с настройками, как на скриншоте. В пункте физический путь указать путь до папки с сайтом (C:/projects/AAS.BackOffice). Повторить тоже самое, только с AAS.FileStorage. Добавить привязку для https.



ПРИМЕЧАНИЕ

Если невозможно сделать восстановление резервной копии БД через DBeaver, то в приложении будут SQL запросы на создание БД.

При таком методе база данных будет пустой для того, чтобы можно было хотя бы зайти на сайт необходимо сделать следующее:

* Пройти все пункты, описанные выше;
* Зайти на сам сайт и поменять URL на следующий: <https://localhost:44378>/updates/create\_default\_role .
* После успешного выполнения запроса необходимо зарегистрироваться в системе.

В данном случае будет создана роль, которая будет позволять делать всё возможное в системе

Приложение

Таблица Bids

CREATE TABLE public.bids (

id bytea NOT NULL,

title varchar NOT NULL,

description varchar NOT NULL,

denydescription varchar NULL,

status int4 NOT NULL,

acceptancedate date NULL,

approximatedate date NULL,

createddatetimeutc timestamp NOT NULL,

createduserid bytea NOT NULL,

modifieddatetimeutc timestamp NULL,

modifieduserid bytea NULL,

isremoved bool NOT NULL,

"number" int4 NOT NULL,

filepaths \_varchar NULL,

CONSTRAINT bids\_pk PRIMARY KEY (id)

);

CREATE INDEX bids\_number\_idx ON public.bids USING btree (number);

Таблица EmailVerifications

CREATE TABLE public.emailverifications (

userid bytea NOT NULL,

"token" varchar NOT NULL,

isverified bool NOT NULL,

createddatetimeutc date NOT NULL,

CONSTRAINT emailverifications\_pk PRIMARY KEY (userid, token)

);

Таблица UserPermissions

CREATE TABLE public.userpermissions (

userid bytea NOT NULL,

roleid bytea NOT NULL,

CONSTRAINT userpermissions\_pk PRIMARY KEY (userid)

);

Таблица UserRoles

CREATE TABLE public.userroles (

id bytea NOT NULL,

"name" varchar NOT NULL,

accesspolicies \_int4 NOT NULL,

createddatetimeutc timestamp NOT NULL,

createduserid bytea NOT NULL,

modifieddatetimeutc timestamp NULL,

modifieduserid bytea NULL,

isremoved bool NOT NULL,

CONSTRAINT userroles\_pk PRIMARY KEY (id)

);

Таблица Users

CREATE TABLE public.users (

id bytea NOT NULL,

firstname varchar NOT NULL,

middlename varchar NULL,

lastname varchar NOT NULL,

email varchar NOT NULL,

passwordhash varchar NOT NULL,

phonenumber varchar NOT NULL,

createddatetimeutc timestamp NOT NULL,

createduserid bytea NOT NULL,

modifieddatetimeutc timestamp NULL,

modifieduserid bytea NULL,

isremoved bool NOT NULL,

photopath varchar NULL,

CONSTRAINT users\_pk PRIMARY KEY (id)

);

Таблица UserTokens

CREATE TABLE public.usertokens (

userid bytea NOT NULL,

"token" varchar NOT NULL,

datetimeutc timestamp NOT NULL,

CONSTRAINT usertokens\_pk PRIMARY KEY (userid)

);