

# ГАЛАКТИКА АММ.WEB

## СИСТЕМА

## ГАЛАКТИКА АММ.WEB 1.0

ИНСТРУКЦИЯ ПО УСТАНОВКЕ И ОБНОВЛЕНИЮ  
СИСТЕМЫ

---



# Аннотация

В данном документе описан процесс установки, настройки и обновления системы *Галактика АММ.WEB 1.0*.

**Издание 07.2021**

# Оглавление

<b>1</b>	<b>Развертывание системы .....</b>	<b>4</b>
<b>2</b>	<b>Настройка сервера приложений и WEB- приложения .....</b>	<b>6</b>
2.1	Настройка и запуск сервера приложений под ОС Astra Linux .....	6
2.2	Настройка и запуск сервера приложений под ОС Microsoft Windows.....	6
2.3	Настройка WEB-приложения.....	7
2.4	Рекомендации по настройке виртуальных хостов для веб-сервера Apache .	7
2.5	Рекомендации по настройке виртуальных хостов для веб-сервера Nginx....	9
<b>3</b>	<b>Настройка системы лицензирования (на текущий версии не используется) ...</b>	<b>12</b>
<b>4</b>	<b>Настройка файла конфигурации.....</b>	<b>13</b>
4.1	Строка для взаимодействия с web-приложением .....	13
4.2	Строка соединения с базой данных.....	13
4.3	Настройка подключения дополнительных модулей.....	15
4.4	Строка с пользовательским отображением названия базы данных .....	15
4.5	Настройка протоколирования .....	16
4.6	Настройка аудита действий пользователей (в текущей версии не используется).....	17
4.7	Строка подключения к серверу лицензий (на текущей версии не используется).....	18
<b>5</b>	<b>Генерация базы данных .....</b>	<b>19</b>
<b>6</b>	<b>Вход в систему .....</b>	<b>20</b>
<b>7</b>	<b>Установка обновлений .....</b>	<b>21</b>

# 1 Развертывание системы

Комплект поставки системы Галактика АММ.WEB состоит из:

1. Сервер приложений - «Галактика АММ.WEB» на базе системы «Галактика АММ». Сервер приложений, обеспечивающие его работу и организацию взаимодействия, обработки и передачи данных между БД и Web-приложением
2. Web-приложение - обеспечивает работу и организацию взаимодействия, обработки и передачи данных между сервером приложений «Галактика АММ.WEB» и различным ПО для просмотра сайтов в Интернете (браузерах).

Под разные операционные системы (используемые операционные системы см. в документе «ТРЕБОВАНИЯ К АППАРАТНОМУ И ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ») выпускаются разные комплекты поставки, состоящие из 2-х архивов формата «zip».

## Для ОС Astra Linux

№ п/п	Имя файла	Комментарии к файлу
1	Galaktika.AMM.WEB-X.X.Y.Y-ГГГГММДД_Linux.zip, где X.X – номер версии Y.Y - сборки системы ГГГГММДД – дата выпуска	Сервер приложений «Галактика АММ.WEB» на базе системы «Галактика АММ».
2	Galaktika.Applicationt.WEB-X.Y.Y-ГГГГММДД.zip, где X.X – номер версии Y.Y - сборки системы ГГГГММДД – дата выпуска	WEB-приложение

## Для ОС Microsoft Windows:

№ п/п	Имя файла	Комментарии к файлу
1	Galaktika.AMM.WEB-X.X.Y.Y- ГГГГММДД_Windows.zip, где  X.X – номер версии  Y.Y - сборки системы  ГГГГММДД – дата выпуска	Сервер приложений «Галактика AMM.WEB» на базе системы «Галактика АММ».
2	Galaktika.Applicationt.WEB-X.Y.Y- ГГГГММДД.zip, где  X.X – номер версии  Y.Y - сборки системы  ГГГГММДД – дата выпуска	WEB-приложение

**Для развертывания системы необходимо:**

- 1) Развернуть при необходимости сервер лицензий (в соответствии с разделом 3 «Настройка системы лицензирования» - *на текущей версии не используется*).
- 2) Настроить и запустить сервер приложений и WEB-приложение (в соответствии с разделом 2 «Настройка сервера приложений и WEB-приложения»).
- 3) Сообщить пользователям адрес веб-страницы для работы с системой через браузер.

## 2 Настройка сервера приложений и WEB- приложения

---

Для установки сервера приложений и WEB-приложения необходимо извлечь содержимое из архивов в папки сервера.

### 2.1 Настройка и запуск сервера приложений под ОС Astra Linux

Для работы сервера приложений под ОС Astra Linux требуется наличие фреймворка .NET Core 3.1 версии не ниже 3.1.17. При его отсутствии требуется провести работу по его установке и настройке.

В файле **appsettings.json** (находится в папке с сервером приложений) в разделе «*Kestrel:Endpoints:Http:Url*» необходимо указать **адрес сетевого интерфейса и порт** для взаимодействия с web-приложением (см. ниже пример).

```
"Kestrel": {
  "Endpoints": {
    "Http": {
      "Url": "http://0.0.0.0:5000"
    }
  }
}
```

Перед запуском сервера приложений необходимо настроить файл конфигурации (в соответствии с разделом 4 «Настройка файла конфигурации») кроме строки для взаимодействия с web-приложением.

Для запуска сервера приложений необходимо воспользоваться командой из папки сервера приложений «**dotnet Galaktika.AMM.Server -u**» (см.картинку).

```
root@itsm-21:~/web_server# ./Galaktika.AMM.Server -u
Galaktika AMM Server 5.5.515.95

Starting services
info: Microsoft.AspNetCore.DataProtection.KeyManagement.XmlKeyManager[0]
      User profile is available. Using '/root/.aspnet/DataProtection-Keys' as key repository; keys will not be encrypted at rest.
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: http://0.0.0.0:5000
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: /root/web_server
```

### 2.2 Настройка и запуск сервера приложений под ОС Microsoft Windows

Для работы сервера приложений под ОС Microsoft Windows требуется наличие фреймворка .Microsoft .NET Framework 4.5.2 и выше. При его отсутствии требуется провести работу по его установке и настройке.

Перед запуском сервера приложений необходимо настроить файл конфигурации (в соответствии с разделом 4 «Настройка файла конфигурации»)

Для запуска сервера приложений необходимо воспользоваться командой из папки сервера приложений «./Galaktika.AMM.Server -u» или запустить файл *Galaktika.AMM.Server.exe* из папки сервера приложений.

## 2.3 Настройка WEB-приложения

Для работы WEB-приложения необходимо, чтобы был установлен и настроен Веб-сервер. При его отсутствии требуется провести работу по его установке и настройке. Рекомендуемые к использованию веб-сервера **Apache** или **Nginx**.

Рекомендации по настройке виртуальных хостов для веб-серверов приведены в разделах 2.4 и 2.5.

Для использования web-приложения необходимо в файле **main.js** (находится в папке с web-приложением) в секции **var environment** в параметре **pathApi** указать **адрес сетевого интерфейса и порт** для взаимодействия с сервером приложений (см. ниже пример).

```
"use strict";
__webpack_require__r(__webpack_exports__);
/* harmony export (binding) */ __webpack_require__.d(__webpack_exports__, {
  "environment": () => environment
});
var environment = {
  production: false,
  /* pathApi: 'http://localhost:8080/api', */
  pathApi: 'http://api.amm-web.galaktika.local/api',
};
```

Отдельного запуска web-приложение не требует. При установке обновлений они отображаются в браузере после обновления окна пользователем.

Если обновления не отображаются, то необходимо почистить кэш (cache) браузера.

## 2.4 Рекомендации по настройке виртуальных хостов для веб-сервера Apache

Apache может поддерживать сотни сайтов на одном компьютере и выдавать для каждого из них правильное содержимое. Для этого используются виртуальные хосты. Сервер определяет к какому домену приходит запрос и отдает нужное содержимое из папки этого домена.

Настройки хостов Apache расположены в папке */etc/apache2/sites-available/*. Для создания нового хоста достаточно создать файл с любым именем (лучше конечно с именем хоста) и заполнить его нужными данными. Обернуть все эти параметры нужно в директиву **VirtualHost**. Кроме рассмотренных параметров здесь будут использоваться такие:



- **ServerName**- основное имя домена
- **ServerAlias**- дополнительное имя, по которому будет доступен сайт
- **ServerAdmin**- электронная почта администратора
- **DocumentRoot**- папка с документами для этого домена

Например:

```
vi /etc/apache2/sites-available/reborn.galaktika.ru
```

Обязательные параметры:

```
<VirtualHost *:80>
ServerName reborn.galaktika.ru //доменное имя
ServerAlias www.reborn.galaktika.ru //дополнительное описание имени
ServerAdmin webmaster@localhost //администратор домена контакты
DocumentRoot /var/www/ reborn.galaktika.ru/public\_html //Папка с расположением
фронтенд части
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

Виртуальные хосты, как и модули нужно активировать. Для этого есть специальные утилиты. Чтобы активировать наберите:

```
sudo a2ensite reborn.galaktika.ru
```

Здесь [reborn.galaktika.ru](http://reborn.galaktika.ru) - имя файла виртуального хоста. Для отключения тоже есть команда: `sudo a2dissite reborn.galaktika.ru`

Настройка виртуальных хостов Apache завершена и на публичном сервере это все бы уже работало, но если вам нужна настройка Apache на домашней машине, то вы ваш новый сайт не откроется в браузере. Браузер не знает такого сайта. DNS службы не могут ничего сообщить об этом доменном имени. Но в системе Linux мы можем сами указать ip адреса для доменных имен в файле /etc/hosts. Поэтому добавляем в конец файла такие строки:

```
vi /etc/hosts
```

Добавляем:

```
127.0.0.1 reborn.galaktika.ru
127.0.0.1 www.reborn.galaktika.ru
```

Для корректной работы необходимо настроить реврайт на index.html, так как используется SPA, чтобы работал роутинг внутренней системы, а не роутинг apache, для этого необходимо в настройках домена прописать реврайт:

```
<VirtualHost *:80>
    ServerName my-app
    DocumentRoot /path/to/app
    <Directory /path/to/app>
        RewriteEngine on
```

```
# Don't rewrite files or directories
RewriteCond %{REQUEST_FILENAME} -f [OR]
RewriteCond %{REQUEST_FILENAME} -d
RewriteRule ^ - [L]
# Rewrite everything else to index.html
# to allow html5 state links
RewriteRule ^ index.html [L]
</Directory>
</VirtualHost>
```

Либо добавить в корень приложения файл `.htaccess` в котором прописать те же настройки для ревайта:

```
<IfModule mod_rewrite.c>
  RewriteEngine on
  # Don't rewrite files or directories
  RewriteCond %{REQUEST_FILENAME} -f [OR]
  RewriteCond %{REQUEST_FILENAME} -d
  RewriteRule ^ - [L]
  # Rewrite everything else to index.html
  # to allow html5 state links
  RewriteRule ^ index.html [L]
</IfModule>
```

## 2.5 Рекомендации по настройке виртуальных хостов для веб-сервера Nginx

На сервере может размещаться несколько сайтов. Все запросы приходят на ip сервера, а nginx уже определяет на основе домена какой контент нужно выдать. Для того чтобы nginx знал что к какому домену относится нужно настроить виртуальные хосты. Каждый хост принято размещать в отдельном файле. Настройка хоста находится в секции `server`, но поскольку все файлы из `sites-enabled` импортируются в секцию `http`, то логика структуры конфигурационного файла не нарушается.

Рассмотрим пример настройки:

```
vi /etc/nginx/sites-enabled/reborn.galaktika.ru.conf
```

- **listen 80**- указывает, что нужно ожидать подключения на порту 80, может также содержать опцию **default-server**, которая означает, что этот домен будет открываться если домен не был задан в запросе.
- **root /var/www/html**- директория, в которой находятся файлы сайта.
- **index index.html**- страница, которая будет открываться по умолчанию.
- **server\_name**- доменное имя сайта.
- **access\_log** - файл для записи лога запросов к серверу, может использоваться как глобально в секции http, так и для определенного типа файлов в location.
- **error\_log**- лог ошибок веб-сервера, может принимать дополнительный параметр, указывающий подробность лога. warn - максимум, crit - только критические ошибки.

Это все основные настройки виртуального хоста, после них он уже будет работать. Но тут есть еще секция location, которая позволяет настроить поведение сервера для определенных директорий и файлов. Синтаксис location такой:

**location адрес {}**

В качестве адреса может использоваться как прямой запрос относительно корня сервера, так и регулярные выражения. Для использования регулярных выражений перед ним ставится символ "~". Примеры рассмотрим ниже, а пока рассмотрим возможные директивы:

- **allow**- разрешить доступ к местоположению для пользователей, all - всех, также можно указать ip или подсеть.
- **deny**- запретить доступ к местоположению, all - для всех.
- **try-files**- пытается открыть файлы в определенном порядке, открывает первый обнаруженный файл. Например, такая конструкция: \$uri \$uri/index.html \$uri.html =404; сначала пытается открыть \$uri, затем index.html, если не найден \$uri.html, и аж потом, если ни одного из предложенных файлов не существует, выдает ошибку 404.
- **expires**- задает время кэширования браузером отданного элемента, например, 1d - один день, 2h - два часа, 30s - 30 секунд.

Кроме этих главных директив, здесь могут использоваться и другие. Чтобы получить больше подробностей, смотрите официальную документацию.

После того, как установка и настройка nginx будет завершена проверяем конфигурацию на ошибки:

```
nginx -t
```

Затем перезагружаем сервер:

```
sudo service nginx reload
```

```
sudo service nginx restart
```

Если изменялись незначительные параметры можно использовать reload, тогда будет просто обновлена конфигурация без перезагрузки, если же изменяли глобальные опции, нужно перезагрузить программу полностью с помощью restart.

Для корректной работы необходимо настроить реврайт на index.html, так как используется SPA, чтобы работал роутинг внутренней системы, а не роутинг nginx, для этого необходимо в настройках домена прописать реврайт это необходимо, чтобы после перезагрузки страницы не выдавало 404 ошибку:

```
server {  
listen    80;  
server_name reborn.galaktika.ru www.reborn.galaktika.ru;  
root /usr/share/nginx/html;  
location / {  
try_files $uri$args $uri$args/ /index.html;  
}  
}
```

## 3 Настройка системы лицензирования (на текущий версии не используется)

---

Программное обеспечение сервера лицензий поставляется в составе сборки системы «Галактика АММ.WEB» и находится в папке *LicenseServer* основной папки приложения.

Установка, удаление и настройка сервера системы лицензирования описаны в *АММ.WEB 1.0 Инструкция по системе лицензирования*.

# 4 Настройка файла конфигурации

---

Все параметры запуска и работы сервера приложения настраиваются в файле конфигурации *Galaktika.AMM.Server.dll.config*, находящемся в основной папке сервера приложения.

Ключевые параметры, настраиваемые в файле конфигурации:

- 1) строка для взаимодействия с web-приложением;
- 2) строка соединения с базой данных;
- 3) строка подключения дополнительных модулей;
- 4) строка с пользовательским отображением названия базы данных, с которой работает пользователь;
- 5) параметры протоколирования (место хранения и уровень детализации);
- 6) параметры подсистемы аудита действий пользователей;
- 7) строка подключения к серверу лицензий (на текущей версии не используется).

## 4.1 Строка для взаимодействия с web-приложением

Для взаимодействия сервера приложения и **web-приложения** в разделе `<connectionStrings>` файла конфигурации необходимо указать **адрес сетевого интерфейса и порт**:

```
<add key="ApiServerHost" value="http://localhost" />
```

```
<add key="ApiServerPort" value="8888" />
```

## 4.2 Строка соединения с базой данных

Настройки подключения к базе данных находятся в разделе `<connectionStrings>` файла конфигурации.

Приложение поддерживает работу со следующими системами баз данных:

- 1) MS SQL;
- 2) PostgreSQL;

Строка соединения с базой данных приложения называется *ConnectionString*, и указывается в формате:

```
<add name="ConnectionString" connectionString="[параметры подключения]" />
```

Описание ключевых параметров подключения приведено ниже.

Параметр	Описание
Data Source Server Address Addr Network Address	Имя или сетевой адрес сервера, к которому необходимо подключиться, в формате: <i>[имя сервера]\[имя экземпляра]</i> или <i>[имя сервера],[номер порта]</i>
Initial Catalog Database	Имя базы данных
Integrated Security Trusted_Connection	При значении, равном <i>false</i> , в строке соединения необходимо указать параметры <i>User ID</i> и <i>Password</i> . Если указаны параметры <i>User ID</i> и <i>Password</i> , и при этом значение параметра <i>Integrated Security</i> равно <i>true</i> , параметры <i>User ID</i> и <i>Password</i> будут проигнорированы. Значение по умолчанию: <i>false</i> . Допустимые значения: <i>true, false, yes, no</i> и <i>SSPI</i> (эквивалентен <i>true</i> ).
User ID UID	Имя учетной записи пользователя для подключения к серверу базы данных
Password PWD	Пароль к учетной записи пользователя для подключения к серверу базы данных

**Примечание** – Не рекомендуется использовать в строке соединения параметры *User ID* и *Password*. Для поддержания высокого уровня безопасности настоятельно рекомендуется вместо этого использовать ключевые слова *Integrated Security* или *Trusted\_Connection*.

Пример настройки строки соединения для MS SQL:

```
<connectionStrings>
  <add name = "ConnectionString"
    connectionString = "XpoProvider=AMMSqlProvider;
    Integrated Security=SSPI; Data Source=(local)\ammInstance;
    Initial Catalog=Galaktika.AMM.WEB; Pooling=false;
    MultipleActiveResultSets=True" />
</connectionStrings>
```

Пример настройки строки соединения для PostgreSQL (настраивается две строки):

```
<connectionStrings>
```

```

<add name = "ConnectionString"
  connectionString = "XpoProvider=AmmPostgres;
  Server=127.0.0.1;User ID=postgres; Password=postgres;
  Database=Galaktika.AMM.WEB; Persist Security Info=true;
  Pooling = false; Max Auto Prepare = 20" />
<add name="EFConnectionString"
  connectionString="Server=127.0.0.1;Database= Galaktika.AMM.WEB;
  User Id=postgres;Password=postgres;
  Max Auto Prepare = 20;" providerName="Npgsql" />
</connectionStrings>

```

## 4.3 Настройка подключения дополнительных модулей

Система Галактика АММ.WEB может использоваться как в базовой поставке (без подключения дополнительных модулей), так и в поставке с отдельными подключаемыми модулями. Как правило, это модули, в которых реализована специфическая для конкретного заказчика функциональность (интерфейсы, действия, отчеты и прочее).

Функциональность, реализованная в подключаемом модуле, является приоритетной над функциональностью базовой поставки системы.

Дополнительные модули подключаются администратором системы в конфигурационном файле самостоятельно. Для этого в разделе `<appSettings>` файла конфигурации следует указать имена подключаемых модулей в формате:

```

<add key="Modules" value="[имена модулей через ';']" />

```

Пример подключения дополнительных модулей:

```

<appSettings>
  ...
  <add key="Modules" value="Galaktika.AMM.WEB.
special;Galaktika.Local.Module" />
  ...
</appSettings>

```

## 4.4 Строка с пользовательским отображением названия базы данных

Пользовательское название базы данных, с которой работает пользователь задается в разделе `<appSettings>` файла конфигурации в параметре:

```

<add key="DBName" value="АММ.WEB" />

```



Задаваемое название в интерфейсе пользователя отображается в верхнем правом углу



## 4.5 Настройка протоколирования

Более подробная информация приведена в документации *XAF*.

В лог-файл *eXpressAppFramework.log* записываются операции, выполняемые системой и конечным пользователем во время работы приложения. Например, загрузка модели, выбор действия на панели инструментов или протокол выполнения бизнес-операции. Если операция приводит к ошибке или к исключению, соответствующая информация также записывается в лог-файл. Таким образом, если ошибка случается, лог-файл облегчает поиск причины ошибки.

При невозможности самостоятельно разобраться с ошибкой следует отправить лог-файл разработчику системы.

Уровень детализации информации, записываемой в лог-файл, настраивается в разделе `<system.diagnostics>` файла конфигурации *Galaktika.AMM.Win.exe.config*:

```
<system.diagnostics>
  <switches>
    <add name="eXpressAppFramework" value="3" />
    <add name="XafariManagedOperationsLogLevel" value="3" />
    <!--<add name="XPO" value="3" />-->
    ...
  </switches>
</system.diagnostics>
```

Приведенные выше параметры позволяют указать различные уровни детализации для отдельных модулей системы:

- 1) параметр *eXpressAppFramework* определяет уровень протоколирования инструментария XAF;
- 2) параметр *XafariManagedOperationsLogLevel* определяет уровень протоколирования управляемых операций из инструментария Xafari;
- 3) параметр *XPO* определяет уровень протоколирования действий с бизнес-объектами.

Уровни детализации информации приведены в таблице.

Уровень	Описание
0	Протоколирование отключено
1	Протоколируются только сообщения об ошибках
2	Протоколируются сообщения об ошибках и предупреждения
3	В дополнение к сообщениям об ошибках и предупреждениям протоколируются системные сообщения и действия пользователей
4	Протоколируются все возможные события

Лог-файлы хранятся в папке сервера приложений.

## 4.6 Настройка аудита действий пользователей (в текущей версии не используется)

Настройки подсистемы аудита задаются в файле конфигурации:

```
<configSections>
...
  <section name="xafariAudit"
    type="Xafari.BC.Audit.Configuration.AuditConfiguration, Xafari.BC.Audit"
  />
</configSections>
...
<xafariAudit auditLogOn="False" auditLogOnFail="False" auditLogOff="False"
auditActions="False" auditBusinessOperations="False" auditReportsCreation="False"
numberReturningObjects="1000" sqlCommandTimeout="60"/>
```

Приведенные параметры позволяют определить перечень аудируемых событий и ограничения при обращении к базе данных:

- 4) *auditLogOn* – аудит входа пользователя (в журнал аудита включаются записи об удачных попытках входа пользователя в систему). По умолчанию выключен (равен *False*);
- 5) *auditLogOnFail* — аудит неудачных попыток входа пользователя (в журнал аудита включаются записи о неудачных попытках входа пользователя в систему). По умолчанию выключен (равен *False*);
- 6) *auditLogOff* — аудит выхода пользователя (в журнал аудита включаются записи об удачных попытках выхода пользователя из системы). По умолчанию выключен (равен *False*);

- 7) *auditActions* — аудит действий (в журнал аудита включаются данные о запуске пользователем действий). Для контекстных действий в запись включается информация об объекте контекста. По умолчанию выключен (равен *False*);
- 8) *auditBusinessOperations* — аудит бизнес-операций (в журнал аудита включаются данные о запуске и отмене пользователем бизнес-операций). Для контекстных бизнес-операций в запись включается информация об объекте контекста. По умолчанию выключен (равен *False*);
- 9) *auditReportsCreation* — аудит генерации отчетов (в журнал аудита включаются данные о генерации пользователем отчетов). По умолчанию выключен (равен *False*);
- 10) *numberReturningObjects* — количество возвращаемых объектов (количество записей, которое будет отображаться при просмотре журнала аудита). По умолчанию равен 1000;
- 11) *sqlCommandTimeout* — таймаут выполнения SQL-запросов. По умолчанию равен 60.

## 4.7 Строка подключения к серверу лицензий (на текущей версии не используется)

Предварительно должна быть настроена система лицензирования и установлена служба сервера лицензий (в соответствии с разделом 3 «Настройка системы лицензирования»).

В разделе `<appSettings>` файла конфигурации следует указать адрес сервера лицензий.

Пример настройки адреса сервера лицензий:

```
<appSettings>
...
  <add key="XafariLicenseServer" value="127.0.0.1:33444" />
...
</appSettings>
```

## 5 Генерация базы данных

---

В файле конфигурации должна быть настроена строка соединения с базой данных. Если база данных с таким названием существует, то будет произведено обновление ее структуры, в противном случае будет создана новая база данных.

Пользователь, от имени которого производится подключение к серверу базы данных, должен обладать правами на создание базы данных и на создание и изменение объектов базы данных (таблиц, индексов, процедур, триггеров и прочего).

Для запуска генерации или обновления структуры базы данных необходимо запустить сервер приложений Галактика АММ.WEB с параметром «-u» (или «-update»). Например, с помощью команды:

```
C:\Program Files (x86)\Galaktika\Galaktika АММ v5.5\Galaktika.AMM.Server.exe -u
```

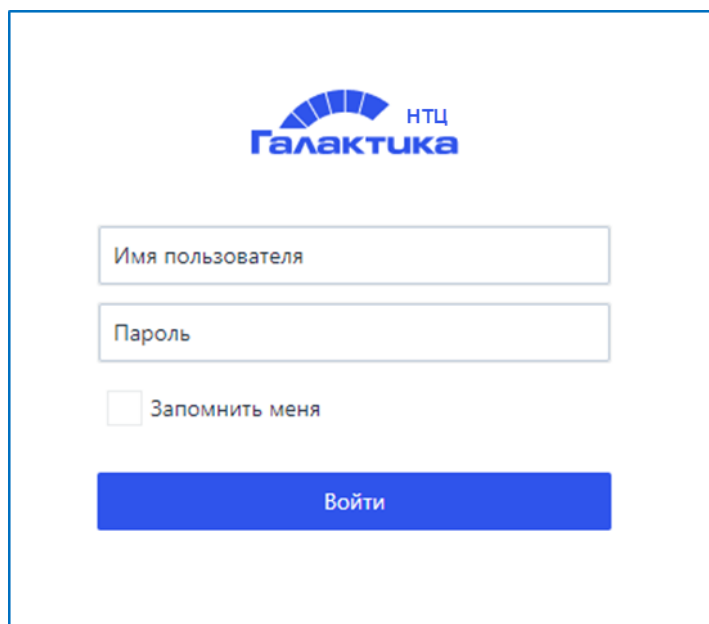
Генерация БД будет осуществлена приложением в момент запуска. При успешном выполнении операции система будет запущена без отображения сообщений об ошибках.

## 6 Вход в систему

---

Для входа в систему Галактика АММ.WEB со стороны клиента необходимо в браузере указать адрес, указанный при настройке Веб-сервера.

На стартовой странице заполните поля **Имя пользователя** и **Пароль** и нажмите [Войти].



The image shows a login form for the 'Галактика' system. At the top center is the logo, which consists of a blue arc above the word 'Галактика' and the letters 'НТЦ' to its right. Below the logo are two text input fields: the first is labeled 'Имя пользователя' and the second is labeled 'Пароль'. Underneath these fields is a checkbox followed by the text 'Запомнить меня'. At the bottom of the form is a solid blue button with the white text 'Войти'.

При успешном входе откроется основное рабочее окно приложения.

## 7 Установка обновлений

---

Перед установкой обновлений необходимо:

- 1) Сделать резервную копию базы данных.
- 2) Сделать резервную копию сервера приложений, включая файлы конфигурации и файлы модели приложения.
- 3) Сделать резервную копию WEB-приложения.

Обновление системы поставляется в виде новых архивов сервера приложения WEB-приложения. Для установки обновления системы следует произвести действия в соответствии с разделом 2 «Установка системы» текущего документа.

После установки обновлений необходимо:

- 4) Сделать резервную копию файлов конфигурации и файлов модели приложения, поставляемых вместе с обновлением системы.
- 5) Скопировать из резервной копии сервера приложений в папку сервера приложений файл конфигурации *Galaktika.AMM.Serwer.exe.config*, файл *appsettings.json* под ОС Astra Linux, файлы модели приложения (*Model.xafml* и *Model\_ru.xafml*) и модели рабочих мест (если используются), заменив имеющиеся одноименные файлы.
- 6) Скопировать из резервной копии web-приложения в папку web-приложения файл *main.js*, заменив имеющийся одноименный файл
- 7) Запустить сервер приложений в режиме автоматического обновления – с параметром «-u» (или «-update»);