

Руководство по развертыванию приложения

Наименование проекта: СМАРТ-КАСКО

Наименование продукта: «Терминал Продавца»

Версия продукта: XXX.XXX

Дата: 16.05.2022

Руководство по развертыванию

1	Введение	3
2	Архив поставки	4
2.1	Структура	4
2.1.1	Конфигурационные данные	5
2.1.2	Веб-приложение	8
2.1.3	База данных	8
2.1.4	Исходный код	8
2.2	Инструкция по развертыванию	9

1 Введение

Документ представляет собой описание структуры архива поставки приложения «Терминал Продавца: КАСКО» и ключевых шагов по его развертыванию в среде, сконфигурированной в соответствии с требованиями, представленными в документе «Целевая архитектура проекта», с целью обеспечения возможности использования его функциональности.

2 Архив поставки

2.1 Структура

Архив поставки представляет собой ZIP-архив, который содержит ряд артефактов, с помощью которых можно осуществить запуск приложения на сервере приложений, создание базы данных и сборку исходного кода проекта.

Данный архив имеет следующий формат наименования:

AIPAS_DELIVERY_UNIT-XXX.XXX-YY-YY-YYYY.zip,

где XXX.XXX – номер версии продукта, YY-YY-YYYY – дата создания архива.

Архив поставки имеет следующую структуру (см. Рис.1):

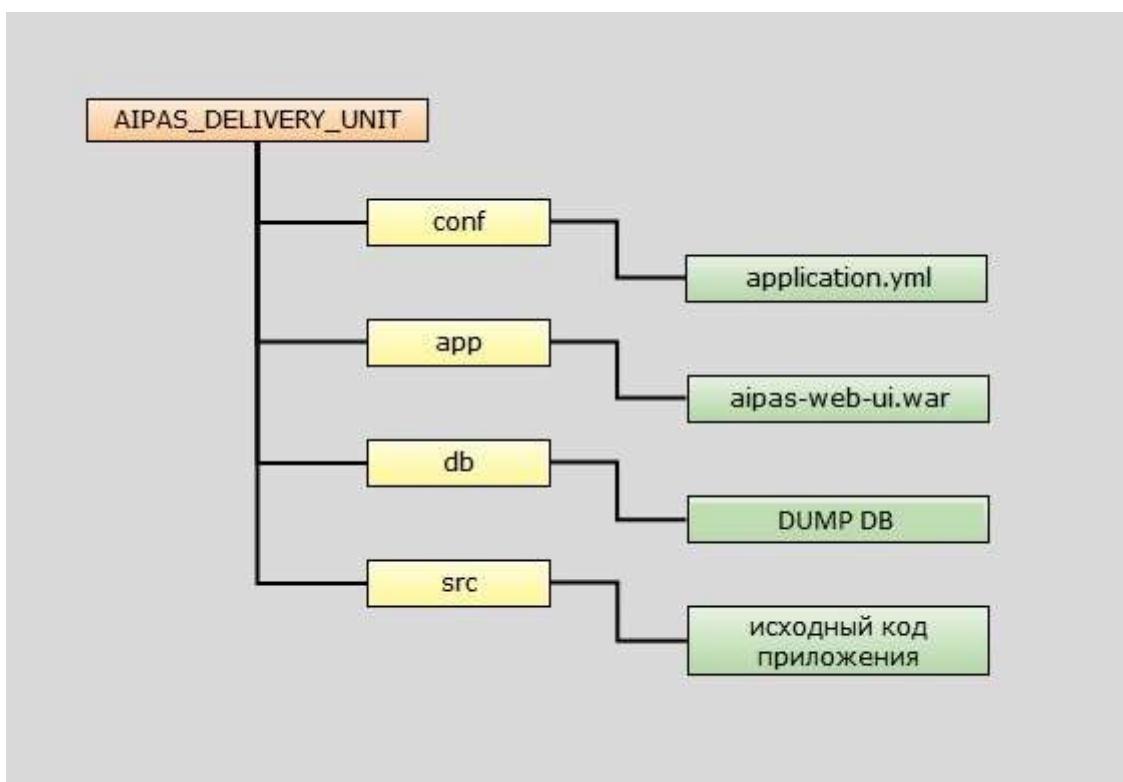


Рис.1. Структура архива поставки

2.1.1 Конфигурационные данные

Директория /conf содержит конфигурационный файл application.yml, в котором содержатся настройки приложения. Эти настройки необходимы для корректного функционирования приложения и должны использоваться при его запуске, разработке или создании базы данных.

Краткая спецификация свойств конфигурационного файла application.yml:

Наименование свойства			Описание
spring	profiles.active		задает spring-профиль, который будет активным по умолчанию старте приложения
spring	datasource	*	настройки соединения с базой данных
spring	jpa	*	настройки уровня и механизмов доступа приложения к базе данных
spring.	mail	*	настройки соединения с почтовым сервером
logging	level	*	настройки уровней логирования для различных артефактов и библиотек проекта
scheduler	integrationTo1CScheduler		
		scheduling.enabled	
		doIntegrationTask	флаг активации/деактивации задачи
			сcheduling.value
	deletePolicyScheduler		
		scheduling.enabled	
		doDeletePolicyTask	флаг активации подзадачи по удалению
			сcheduling.value
	installmentAssigneeNotifyScheduler		
		scheduling.enabled	
		notifyAssigneeTask	флаг активации подзадачи по оповещению
			сcheduling.value
	updateExpiredPolicyStatusScheduler		
		scheduling.enabled	
		updateStatusTask	флаг активации подзадачи по обновлению статуса
			сcheduling.value
			временной параметр запуска подзадачи на исполнение

Наименование свойства			Описание
scheduler	expiringPolicyAssigneeNotifyScheduler		задача по оповещению ответственного пользователя об истекающих полисах
		scheduling.enabled	флаг активации/деактивации задачи
		notifyAssigneeTask	scheduling.enabled
			scheduling.value
liquibase			настройки системы версионирования структуры базы данных
		change-log	путь к родительскому скрипту изменений
		drop-first	флаг активации удаления базы данных перед применением файлов изменений
	parameters		дополнительные настройки
		admin.mail	настройка почты пользователя ADMIN (при изменении значения данной настройки с последующим выполнением команды по сборке базы данных, у пользователя ADMIN будет обновлено значение электронной почты)
application			группа настроек приложения Смарт-Каско
			подгруппа настроек пользовательского интерфейса приложения Смарт-Каско
	web	server-path	базовый URL приложения, развернутого на сервере (используется в качестве основы пути при генерации ссылок в приложении)
		welcome-page	путь к стартовой странице приложения (используется для генерации путей перехода и ссылок в шаблонах писем)
		policies-page	путь к странице поиска полисов (используется для генерации путей перехода)
		page-size	настройка количества записей на одной странице при отображении списков
		preview-url	путь для формирования адреса перехода при просмотре документов
		calc-url	используется для генерации путей перехода к странице калькулятора
		policy-url	используется для генерации путей перехода к странице оформления
		policy-search	настройка количества записей на одной странице при отображении списка на форме поиска полисов
		page-size	

Наименование свойства			Описание
db		frozen-period	период «заморозки» состояния полиса в миллисекундах
		policy-number-generator	настройка количества кэшируемых номеров при генерации номера расчета/полиса
		agent-report-number-generator	настройка количества кэшируемых номеров при генерации номера отчета агента
		selected.rows.number	подгруппа настроек уровня взаимодействия с базой данных приложения Смарт-Каско
	mail		
		escape-character	ограничение на количество записей, выбираемых из базы данных (используется при загрузке списка со сброшенными фильтрами на форме поиска полисов)
integration	1c		
		*	настройки интеграционной части приложения для взаимодействия со сторонними системами и приложениями
		ftp	подгруппа настроек интеграции с 1C
			настройки FTP сервера
			директория для хранения файлов с данным для интеграции
			директория для хранения ответов о результатах интеграции
			директория для хранения успешно обработанных ответов о результатах интеграции
		stub	директория для хранения ответов, в результате обработки которых возникли ошибки
			эмулятор-заглушка ответа от 1С
			флаг активации заглушки
			тип ответа (успешная/неуспешная интеграция)

Наименование свойства		Описание
spring	profiles	задает spring-профиль, который в котором могут быть добавлены или переопределены специфичные для него настройки. Все настройки, определенные вне конкретного профиля, будут наследоваться каждым профилем.
	default	профиль, который использовался при разработке приложения
	aipas_test	набор профилей, использовавшихся при тестировании приложения
	aipas_preat	
	aipas_uat	профиль, подготовленный для использования в окружении заказчика
	test	профиль, содержащий настройки для запуска модульных и интеграционных тестов
spring	jmx	default_domain настройка системы управления расширениями, позволяющая запускать на одном сервере веб-приложений несколько инстансов приложения Смарт-Каско с разными активными профилями

2.1.2 Веб-приложение

В директории /app находится файл aipas-web-ui.war, содержащий основные артефакты приложения «Терминал Продавца: КАСКО», необходимые для его запуска на сервере веб-приложений.

2.1.3 База данных

В директории /db находится дамп базы данных приложения «Терминал Продавца: КАСКО».

2.1.4 Исходный код

В директории /src находится исходный код приложения «Терминал Продавца: КАСКО». Для работы с ним могут быть использованы следующие команды, предполагающие использование заранее установленного и сконфигурированного инструмента для автоматизации сборки проектов Apache Maven:

- `maven clean install` – сборка проекта. Для сборки всего исходного кода проекта должна быть выполнена из корневой директории /src, в которой расположен родительский конфигурационный файл проекта pom.xml;
- `mvn spring-boot:run -Dspring.profiles.active=default` – команда для запуска приложения. Выполняется из директории /src/applications/aipas-web-ui.

Для дальнейшего сопровождения и доработки исходного кода проекта можно использовать специализированные программные комплексы, такие как интегрированные среды разработки (IntelliJ IDEA, IDE Eclipse, NetBeans).

2.2 Инструкция по развертыванию

Перед тем, как начать развертывание приложения, необходимо выполнить все ключевые шаги по подготовке и конфигурации среды, в которой оно будет функционировать. В документе «Целевая архитектура проекта» содержатся основные положения о том, какие программные средства и аппаратные необходимы для запуска приложения «Терминал Продавца: КАСКО».

После выполнения всех необходимых условий, можно приступить к выполнению следующих шагов:

- 1) скопировать конфигурационный файл application.yml, находящийся в директории /conf архива поставки, в директорию /bin сервера веб-приложений Apache Tomcat. Изменить в этом файле значение настройки spring.profiles.active на имя профиля, содержащего все необходимые и специфичные для конкретного окружения настройки, в котором будет развернуто приложение. По умолчанию в качестве значения этой переменной может быть использован профиль aipas_uat. Данный профиль поставляется вместе с приложением и содержит все необходимые настройки под окружения заказчика актуальные на момент поставки;
- 2) создать в этой же директории /bin файл, который будет использоваться для настройки необходимых переменных окружения. Для операционных систем семейства UNIX этот файл будет называться setenv.sh. В него необходимо добавить две инструкции:

```
CLASSPATH="$CATALINA_HOME"/bin/application.yml  
JAVA_OPTS="$JAVA_OPTS -Duser.language=ru -Duser.region=RU -  
Dspring.config.location='$CATALINA_HOME/bin/application.yml' -  
Duser.timezone=Europe/Moscow"
```

- CLASSPATH="\$CATALINA_HOME"/bin/application.yml.
- JAVA_OPTS="\$JAVA_OPTS -Duser.language=ru -Duser.country=RU -
Dspring.config.location=\$CATALINA_HOME/bin/application.yml"
-
- echo AIPAS customization:
- echo JAVA_OPTS: \$JAVA_OPTS
echo CLASSPATH: \$CLASSPATH

Здесь важно учесть то, что все приведенные выше переменные окружения, а именно \$CATALINA_HOME (путь к родительской директории с исходниками Apache Tomcat), \$JAVA_OPT (параметры запуска Java Virtual Machine), \$JAVA_HOME (путь к родительской директории Java Development Kit версии не ниже 1.8.0) должны быть заданы в системе на этапе конфигурации среды. По аналогии можно сделать аналогичный файл для операционных систем семейства Windows, но с учетом специфичных данному виду систем синтаксических особенностей и расширением .bat;

- 3) скопировать файл aipas-web-ui.war, находящийся в директории /app архива поставки, в директорию /webapps Apache Tomcat;
- 4) развернуть базу из дампа;
- 5) запустить сервер веб-приложений Apache Tomcat можно с помощью скрипта startup.sh или startup.bat, для операционных систем семейства UNIX и Windows соответственно, находящихся в директории /bin;

- 6) после запуска приложения, оно будет доступно в браузере по адресу `http://адрес_сервера:номер_порта/aipas-web-ui/dashboard/main.zul`;
- 7) остановить сервер веб-приложений Apache Tomcat с помощью скрипта `shutdown.sh` или `shutdown.bat`, для операционных систем семейства UNIX и Windows соответственно, находящихся в директории `/bin`.

После успешного запуска приложения, оно будет полностью готово для использования с учетом начальных настроек, которые заложены в нем в соответствии с требованиями заказчика.