Запуск приложения «Баланс калькулятор»

Требования:

- кластер Microsoft ServiceFabric версии 10.1.2175.9590 или новее
 - нужно выполнить настройку нод кластера согласно инструкции https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-clusterstandalone-deployment-preparation
 - о развернуть кластер Microsoft ServiceFabric
 - либо можно развернуть кластер на локальной машине: https://learn.microsoft.com/en-us/azure/service-fabric/service-fabric-get-started
- на нодах кластера Microsoft ServiceFabric должны быть установлены:
 o dotnet SDK версии 8.0.400
- инстанс PostgreSQL Server
- инстанс RabbitMQ
- инстанс Redis

Подготовка инстанса RabbitMQ

На сервере *RabbitMQ* необходимо создать два *vhost*-а с именами *localdev_app* и *localdev_audit*. Строки подключения подразумевают использование пользователя по умолчанию – *guest* с паролем *guest*. В этом случае логин и пароль можно не указывать в строке подключения.

Подготовка инстанса Redis

Сервер *Redis* не требует какой-либо дополнительной настройки. Достаточно указать строку подключения в конфигурационных файлах сервисов.

Подготовка инстанса PostgreSQL

База и необходимые таблицы создаются с помощью миграций. Для их запуска нужно перейти в папку *Migrations* и из командной строки выполнить скрипт *limeMigrate.cmd*. Перед запуском скрипта, нужно открыть его редактором и указать свою строку подключения к базе *PostgreSQL*:

Пример запуска:

Пример запкуска и вывода PS D:\BalanceCalculator\Migrations> .\limeMigrate.cmd

Prd.Model... Apply 20250314101327_Initial migration. Apply 20250317033916_AddSettings migration. Seed 20250317033916_AddSettings migration data.

Подготовка конфигурационных файлов приложений

- .\BalancaCalculator-Application\ApplicationParameters\Local.1Node.xml в этом файле указываются пользовательские значения параметров, объявленных в файле "\BalancaCalculator-Application\ApplicationPackageRoot\ApplicationManifest.xml". Можно указывать только те параметры, которые есть в ApplicationManifest.xml, иначе будет ошибка.
 - Здесь нужно обязательно указать:
 - Audit_QueueConnection
- .\BalancaCalculator-Application\ApplicationPackageRoot\ApplicationManifest.xml этот файл формируется на этапе разработки, в нём ничего менять нельзя.
- .\BalancaCalculator-API\ApplicationParameters\Local.1Node.xml в этом файле указываются пользовательские значения параметров, объявленных в файле "\BalancaCalculator-API\ApplicationPackageRoot\ApplicationManifest.xml". Можно указывать только те параметры, которые есть в ApplicationManifest.xml, иначе будет ошибка. Здесь нужно обязательно указать:
 - Audit_QueueConnection
- .\BalancaCalculator-API\ApplicationPackageRoot\ApplicationManifest.xml этот файл формируется на этапе разработки, в нём ничего менять нельзя.

Подготовка конфигурационных файлов микросервисов

Поставка приложений в кластер *ServiceFabric* происходит с помощью **пакетов**. В пакете содержатся микросервисы и их манифесты. В нашем случае пакет расположен по пути:

- .\BalancaCalculator-Application\pkg
- .\BalancaCalculator-API\pkg

Для установки своих строк подключения нужно зайти в каждую папку с микросервисом, затем в папку *Code*, затем открыть редактором файл *ConnectionStrings.config* и внести свои значения. Пример полного пути относительно корня архива:

.\BalancaCalculator-

Application\pkg\Release\Prd.Behavior.ImplementationPkg\Code\ConnectionStrings.config

Поставка приложений в кластер

ПО Баланс калькулятор состоит из двух приложений: основного и АРІ для него.

Для деплоя приложений в кластер *ServiceFabric* используется скрипт .\Deploy-FabricApplication.ps1

Сначала нужно подключиться к кластеру ServiceFabric. Для локального кластера достаточно использовать командлет Connect-ServiceFabricCluster без параметров:

PS D:\BalanceCalculator≻ Connect-ServiceFabricCluster True	
ConnectionEndpoint : FabricClientSettings : { ClientFriendlyName PartitionLocationCacheLimit PartitionLocationCacheBucketCount ServiceChangePollInterval ConnectionInitializationTimeout KeepAliveInterval ConnectionIdleTimeout HealthOperationTimeout HealthReportSendInterval HealthReportRetrySendInterval NotificationGatewayConnectionTimeout NotificationCacheUpdateTimeout AuthTokenBufferSize	: PowerShell-9316e15b-3452-4e37-87bb-f20039d704ac : 100000 : 1024 : 00:02:00 : 00:00:02 : 00:00:20 : 00:00:00 : 00:02:00 : 00:02:00 : 00:00:00 : 00:00:30 : 00:00:30 : 4096
GatewayInformation : { NodeAddress NodeId NodeInstanceId NodeName }	: devops.lime.local:19000 : bf865279ba277deb864a976fbf4c200e : 133869435295564143 : _Node_0

Для поставки основного приложения нужно запустить *PowerShell* и перейти в папку .*BalancaCalculator-Application*. Теперь запускаем скрипт:

PS D:\BalanceCalculator\BalancaCalculator-Application> .\Scripts\Deploy-FabricApplication.ps1 -ApplicationPackagePath .\pkg\Release\ -PublishProfileFile .\PublishProfiles\Local.1Node.xml -OverwriteBehavior Always -OverrideUpgradeBehavior VetoUpgrade

Для поставки *API* для основного приложения нужно запустить *PowerShell* и перейти в папку .*BalancaCalculator-API*. Теперь запускаем скрипт:

PS D:\BalanceCalculator\BalancaCalculator-API> .\Scripts\Deploy-FabricApplication.ps1 - ApplicationPackagePath .\pkg\Release\ -PublishProfileFile .\PublishProfiles\Local.1Node.xml - OverwriteBehavior Always -OverrideUpgradeBehavior VetoUpgrade

Service Fabric Explorer Service Fabric Troublesh Application fabric:/App Ок 🔇 Error 0 Sort by Health A Warning Essentials Details Deployments Manifest Commands Events Search list Application Type ∨ Cluster ✓ Applications Version 1.0.0 😭 Services ✓ АррТуре 个 Ready 🕐 c fabric:/App > X fabric:/App/ProductBehaviorServiceActorService > X fabric:/App/ProductService Error 0 BalanceCalculatorAppType fabric:/BalanceCalculatorApp Warning 0 Replicas > X fabric:/BalanceCalculatorApp/BalanceCalculatorApiService > Nodes Healthy 2 > System

Приложения в кластере выглядят так: