Fra idé til virkelig med Azure Mobile Services

Niels Ladegaard Beck
Holion
nlb@holion.dk
@nielslbeck

Windows Developers in Denmark

Azure App Service Mobile App
Introduktion til ~~Azure Mobile Services~~

Platform tiltænkt app-udviklere der ikke gider bøvle med backendhalløj

Overbygning til WebAPI

- Push
- Login
  - Facebook, Twitter, LinkedIn, Google, Microsoft Account og Active Directory
- Storage
  - SQL, Table Storage og MongoDB
  - Offline data
- Scheduler
  - Ikke direkte understøttet for Mobile Apps
- SDK til mange klientplatforme
  - Windows Store, Windows Phone, iOS, Android, Xamarin, HTML, PhoneGap, Sencha, Appcelerator

JavaScript backend / .NET backend – what to choose?

In the beginning there was…

…Node.js

JavaScript backenden

- Super let at komme i gang med
  - Alt kan styres via portalen: scripts, scheduleres, database, konfiguration, skalering, logs osv.
- Ingen lokal debugging

.NET backenden

- Lokal debugging
- Knap så let at komme i gang med
  - Masser af små forhindringer, som kræver at man følger tutorials / læser dokumentationen / selv finder på en løsning / ser denne præsentation ;-)

Deployment

HOLION

JavaScript backenden

- Git – credentials opsættes via portalen, URL findes på konfigurationssiden
- Ofte fristes man til at benytte online editoren
  - `Ctrl+s` – klar til test
  - FULDSTÆNDIG broken commit history

Committing update from request /scmvfs/service/api/storeevent.js	13 feb 2014 9:53	windowsazure <windowsazure>
Committing update from request /scmvfs/service/api/list.js	13 feb 2014 9:52	windowsazure <windowsazure>
Committing update from request /scmvfs/service/api/list.js	13 feb 2014 9:46	windowsazure <windowsazure>
Committing update from request /scmvfs/service/api/device.js	13 feb 2014 9:46	windowsazure <windowsazure>
Committing update from request /scmvfs/service/api/list.js	13 feb 2014 9:42	windowsazure <windowsazure>
Committing update from request /scmvfs/service/api/list.js	13 feb 2014 9:41	windowsazure <windowsazure>
Committing update from request /scmvfs/service/api/list.js	13 feb 2014 9:38	windowsazure <windowsazure>
Committing update from request /scmvfs/service/api/list.js	13 feb 2014 9:37	windowsazure <windowsazure>

.NET backenden

- Samtlige videoer, tutorials osv: publish direkte fra VS
- Git – URL findes på konfigurationssiden
- Tip: Brugernavn/kodeord findes i publish profile (userName/userPWD), der kan downloades fra portalen

Lokal debugging

IIS Express og LocalDB fungerer fint til formålet

Debugging via WP-emulatoren (eller et device) kræver adgang via netværket

Tilføjelse til `%userprofile%\My Documents\IISExpress\config\applicationhost.config`:

```
<binding protocol="http" bindingInformation="*:8080:*" />
```

http.sys-magi via kommandoprompten (som administrator):

```
netsh http add urlacl url=http://*:8080/ user=everyone
```

Åben port 8080 i din firewall eller tillad IIS Express

Kræver administratorrettigheder at starte IIS Express med en port åben mod netværket!

Lokal debugging og login

Server side / client side authentication

Server side

- LoginAsync kræver HTTPS – UA kræver validt certifikat

Client side

- I november
  - Kun AD-authentication understøttet at .NET backenden
  - Facebook og Microsoft Account authentication understøttet af JS backenden
- I dag – de rykker hurtigt!
  - .NET: Facebook, Google, Microsoft Account og AD understøttet
  - Node: Facebook, Google og Microsoft Account understøttet

Custom authentication

- Nedarv fra `LoginProvider`, overskriv relevante metoder og kald CreateLoginResult på denne

Login i Windows Phone XAML baseret app

Login – og intet sker

Husk følgende (godt gemte) code snippet:

```
protected override void OnActivated(IActivatedEventArgs args) {
#if WINDOWS_PHONE_APP
  if (args.Kind == ActivationKind.WebAuthenticationBrokerContinuation)
    MobileService.LoginComplete(args as WebAuthenticationBrokerContinuationEventArgs);
#endif

  base.OnActivated(args);
}
```

Dataadgang (1/2)

HOLION

Udgangspunktet for dataadgang: Entity Framework Code First

Tip: Brug migrations

- Slå migrations til via PowerShell (f.eks. NuGet Package Manager Console)
  `Enable-Migrations`
  Herved oprettes bl.a. **`Migrations/Configuration.cs`**
  I `WebApiConfig` udskiftes
  ```
  Database.SetInitializer(new MobileServiceInitializer());
  ```
  med
  ```
  var migrator = new DbMigrator(new Configuration());
  migrator.Update();
  ```
- Efter hver ændring til modellen
  `Add-Migration`
- Opdater databasen (gøres automatisk når backenden tilgås)
  `Update-Database`

Dataadgang (2/2)

Din `DbContext.OnModelCreating` bør sætte skemaet til noget statisk

```
modelBuilder.HasDefaultSchema("mitSkema");
```

Modelklasser bør nedarve fra `EntityData`

- Tilføjer `Id`, `CreatedAt`, `UpdatedAt`, `Version` og `Deleted`

Opret kald på backenden

Ønskes "direkte" adgang til data nedarves fra `TableController<TData>`
Api-kald implementeres ved at nedarve fra `ApiController`
Scheduled jobs skal nedarve fra `ScheduledJob`

Dekorer klasse eller metode med `AuthorizeLevel` attributten

- Alle kan kalde: `[AuthorizeLevel(AuthorizationLevel.Anonymous)]`
- Dine apps kan kalde: `[AuthorizeLevel(AuthorizationLevel.Application)]`
- Autentificerede brugere kan kalde: `[AuthorizeLevel(AuthorizationLevel.User)]`
- Kun admins kan kalde: `[AuthorizeLevel(AuthorizationLevel.Admin)]`

Kald af backenden fra browseren

HOLION

Controllers der nedarver fra

- `ApiController` kan kaldes på `http://localhost:{port}/api/{navn}`
- `TableController<TData>` kan kaldes på `http://localhost:{port}/tables/{navn}`
- `ScheduledJob` kan kaldes på `http://localhost:{port}/jobs/{navn}`

Servicen tilgås uden brugernavn/kodeord lokalt – også selvom `AuthorizeLevel` attributten er sat til andet end `AuthorizationLevel.Anonymous`

Ved adgang til servicen i skyen, kan master key'en benyttes som password

Ønskes samme opførsel lokalt som i skyen indsættes følgende i `WebApiConfig`

```
config.SetIsHosted(true);
```

MVVM - Model View ViewModel

HOLION

Hurtigt i gang: MVVM Light Toolkit
Design time view bør altid fungere

Tip: Tilføj følgende i **`ViewModelLocator.cs`** og undlad at registrere dependencies, hvis `ViewModelBase.IsInDesignModeStatic` er `true`

```
static T ViewModel<T>() where T : class {
  if (SimpleIoc.Default.IsRegistered<T>())
    return ServiceLocator.Current.GetInstance<T>();

  return Activator.CreateInstance<T>();
}

public MainViewModel Main {
  get { return ViewModel<MainViewModel>(); }
}
```

Tilføj en constructor uden argumenter i `MainViewModel` og opret en anden constructor, der tager klassens dependencies. Angiv denne til at være `[PreferredConstructor]`.

Kald af backenden fra app

HOLION

Tilføj `WindowsAzure.MobileServices` via NuGet

`MobileServiceClient` instantieres med URL til backenden og application key'en

Controllers der nedarver fra

- `ApiController` kan kaldes med `InvokeApiAsync`
- `TableController<TData>` kan tilgås via `GetTable<TData>`
  - Data kan sorteres og filtreres
  - Paging foregår automatisk
- `ScheduledJob` kan ikke kaldes direkte

Konflikthåndtering

Vi forsøger at indsætte et element med en ID der allerede eksisterer
`MobileServiceConflictException`

Vi forsøger at ændre et element der er ændret af en anden
`MobileServicePreconditionFailedException`

Løsning

- Ønskes serverens version – ignorer
- Ønskes egen version - sæt det lokale versionsnummer lig serverens og prøv igen

Hvad er en app uden mulighed for brug offline?

SyncContext på `MobileServiceClient` skal initialiseres før brug

Tilføj `WindowsAzure.MobileServices.SQLiteStore` via NuGet

Tip: Portable Class Library for SQLite hentes automatisk via NuGet men selve databasemotoren skal downloades separat

- Downloades fra `https://sqlitepcl.codeplex.com/documentation`
- VSIX indeholdende databasemotoren skal hentes og installeres for både Windows Runtime og Windows Phone Runtime

Opsætning af offline muligheder

HOLION

SyncContext kan nu initialiseres

```
var store = new MobileServiceSQLiteStore("sync.db");
store.DefineTable<TData>();
await mobileServiceClient.SyncContext.InitializeAsync(store);
```

I stedet for at kalde `GetTable<TData>` kaldes nu `GetSyncTable<TData>`

`IMobileServiceSyncTable<TData>` har nogle ekstra muligheder i forhold til `IMobileServiceTable<TData>`

- `PullAsync` trækker data ned i den lokale database
- `PurgeAsync` fjerner data fra den lokale database

`PushAsync` findes direkte på din SyncContext

Opdatering af ændrede data

`PullAsync` henter alle forespurgte data ned hver gang!

Ønskes kun ændrede data hentet, kan `queryId` angives

- Tip: Hold øje med netværksforbindelsen

Konflikthåndtering – offline data

Implementer `IMobileServiceSyncHandler` indeholdende
`ExecuteTableOperationAsync` og `OnPushCompleteAsync`

Håndter samme exceptions som tidligere: `MobileServiceConflictException` og `MobileServicePreconditionFailedException`

Giv implementationen med til SyncContext.InitializeAsync

Opgradering til App Service Mobile App

HOLION

Pt. en manuel proces

JavaScript-backends kan endnu ikke opgraderes!

Scheduled jobs er ikke understøttet (brug f.eks. Web Jobs i stedet)

1. Opret en Mobile App
   - Connection strings og notification hub kan genbruges
2. Migrer bruger id'er
3. Opdater klienter
   - Opdater NuGets
   - Peg på ny backend
   - Opdater til nyt push notification API
4. Slet den gamle Mobile Service

Opsummering

HOLION

Introduktion til Azure Mobile Services
Deployment
Lokal debugging
Custom authentication
Entity Framework - migrations
Udstilling af API-kald og tabeller
Sikring af kald
Kald fra browser og app
Konflikthåndtering
Offline-opsætning
Konflikthåndtering
Opgradering til App Service Mobile App

?