# Leitfaden für Azure DevOps für Power Apps und Dynamics 365 Apps Entwicklung

Dank moderner Entwicklungsdienste wie Azure DevOps können Sie besser planen, einfacher im Team zusammenarbeiten und Ihren Code schneller ausliefern. In diesen Leitfaden zeigen wir anhand einer konkreten Power App «Market Insight» den Aufbau eines Azure DevOps Projektes für Power Apps und Dynamics 365 Apps schrittweise auf.

## DevOps in Kürze

Ziel von DevOps ist viele stabile Releases zu ermöglicht durch einen hohe Automatisierung Grad der Entwicklungs- (Dev) und Betriebs- (Ops) Aufgaben. Automatisiert ablaufen sollen zum Beispiel der Build aus dem Repository, statische und dynamische Code-Analysen sowie Unit-, Integrations-, System- und Performance-Tests. Ein kontinuierliches, möglichst automatisiertes Monitoring überwacht die sogenannte Deployment Pipeline.

A picture containing playing, game

Description automatically generated

DevOps mit Continuous-Integration- und Continuous-Delivery-Werkzeuge ermöglichen den erforderlichen hohen Grad an Automatisierung der «Deployment Pipeline».

## Azure DevOps und Werkzeuge

|  |  |  |
| --- | --- | --- |
| Bereich | Beschreibung | Verwendete Werkzeuge |
| Plan | Kanban-Boards mit Backlogs und Planungstools für Agiles Vorgehen | Azure Boards |
| Code | Code-Entwicklung und Code-Review, Werkzeuge zur Versionskontrolle, Zusammenfügen von Code (Merge) | Power Apps Studio (make.powerapps.com)  Azure Repos Git (dev.azure.com)  Visual Studio 2019 |
| Build | Werkzeuge zur kontinuierlichen Integration und Erstellung eines «Build Status»  Package Manager zum Ausliefern von binären Formaten (ZIP, DLL, etc.) | Azure Pipeline (dev.azure.com)  Power Apps BuildTools  Solution Packager  Package Deployer |
| Test | Statische und dynamische Code-Analysen und Tests | Azure Test Plans (dev.azure.com) |
| Release | Change-Management, z. B. nach ITIL, Freigabe von Releases | Azure Pipeline (dev.azure.com)  Power Apps BuildTools |
| Deploy | Verteilen eines Releases in Umgebungen (Development-Test-Produktion)  Konfiguration von Umgebungen (Infrastructure as Code-Werkzeuge) | Azure Pipeline (dev.azure.com)  Power Apps BuildTools |
| Operate | Betrieb einer Applikation | Power Platform Admin Portal  Power Platform Admin Connectors |
| Monitor | Überwachung der Applikation mit automatisierter Fehlermeldung in die Planung für die Behebung. | Power Platform Admin Portal  Power Platform Admin Connectors |

## Azure Dev Ops für Power Apps aufsetzen

Der Leitfaden beinhaltet die schrittweisen Anweisungen zur Erstellung des folgenden Szenarios:

Teil 1 – Einrichten der Umgebung

1. Einrichten der Entwicklungs-, Test- und Produktionsumgebungen.
2. Einrichten des DevOps Projektes für «Market Insights»

Teil 2 – Einrichten der Azure DevOps Pipelines

1. Exportieren des Lösungs-Paketes, welches die Applikation aus einer Entwicklungsumgebung enthält.
2. Entpacken des Lösungs-Paketes.
3. Übernehmen des Lösungs-Paketes in das Repository (Quellcode Verwaltung).
4. Importieren des Lösungs-Paketes in die Build Umgebung.
5. Erstellen des Build Artefakts (verwaltete Lösung).
6. Bereitstellen der Lösung in einer Downstream Umgebung (Test und/oder Produktion).

## Teil 1 - Einrichten der Umgebungen

Wir verwenden eine vereinfachte Form mit drei Common Data Service (CDS) Umgebungen zusammen mit Azure DevOps mit dem Ziel die Bereitstellungen zu automatisieren.

Umgebungen

* Entwicklungsumgebung (Power Apps Dev)
* Testumgebung (Test)
* Produktion Umgebung (Production)

A screenshot of a cell phone

Description automatically generated

### Umgebungen je nach Applikationstyp

* Power Apps - Bei reiner Power Apps Entwicklung (Canvas, Flow, Model Driven) OHNE Dynamics 365 Applikationen (Marketing, Sales, Customer Services), sollte die Entwicklungsumgebung und die Testumgebung ohne Dynamics 365 Applikationen bereitgestellt werden!
* Dynamics 365 Apps – Bei Power Apps Entwicklung, die auf Dynamics 365 Apps basieren sollten in den Entwicklungsumgebung und Testumgebung nur die Dynamics 365 Apps vorinstalliert werden, die für die Applikation verwendet werden!

Mit dieser Trennung werden Abhängigkeiten ihrer Applikationen vermieden, bzw. sichergestellt die für Ihre Applikation relevant sind.

Für die «Market Insigths» Applikation verwenden wir eine reine Power Apps Entwicklung Umgebung, da diese aus im Wesentlichen aus Custom Entitäten und Power Automate Flows besteht.

### Erste Umgebung einrichten

1. Web Browser mit <https://admin.powerplatform.microsoft.com/> öffnen
2. Zu «Environment» gehen
3. «New» auswählen und die Umgebungsangaben eingeben
   1. Name «Power Apps Dev»
   2. Type «Sandbox»
   3. Region «Europe» oder eine andere Datenzentrum Region
4. Wichtig «Create a Database» auswählen, damit eine CDS Datenbank angelegt wird

A screenshot of a social media post

Description automatically generated

1. «Next» auswählen
   1. Sprache «English»
   2. Währung «CHF» oder eine andere Standardwährung
2. Wichtig «Enable Dynamics 365 apps» und «Deploy sample apps and data» nicht auswählen
3. Berechtigung Gruppe bei Bedarf setzen, damit nur die gewünschten Benutzer zugriff auf diese Umgebung haben
4. «Save» auswählen

A screenshot of a social media post

Description automatically generated

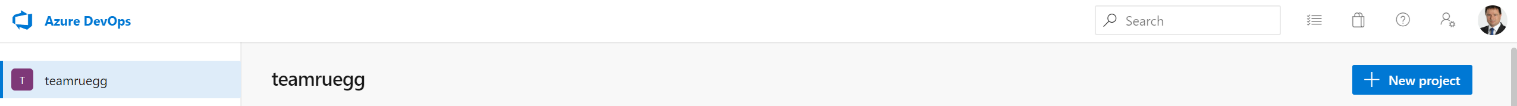
1. Jetzt wird die Umgebung angelegt und die Detailangaben der Umgebung können überprüft werden

A screenshot of a cell phone

Description automatically generated

1. Die Schritte 3 – 8 wiederholen wir zweimal für die zwei weiteren Umgebungen
   1. Test Umgebung «Dynamics 365 Test»
   2. Produktiv Umgebung «Dynamics 365 Prod»
2. Jetzt sind die drei Umgebungen vorbereitet

### Einrichten des DevOps Projektes für «Market Insights»

1. Web Browser mit <https://dev.azure.com> öffnen
2. «New project» auswählen
3. 
4. «Project settings» asuwählen

A screenshot of a cell phone

Description automatically generated

1. «Service connections» und «New connection» auswählen

A screenshot of a computer

Description automatically generated

1. Die Verbindungsangaben eingeben
   1. Server URL «<dev>.crm4.dynamics.com» der Entwicklungsumgebung
   2. Benutzername und Password
   3. Service connection name «Power Apps Dev»
2. Wichtig «Grant access permission to all pipelines» muss gesetzt sein
3. «Save» auswählen
4. Die Schritte 5 – 8 wiederholen wir zweimal für die zwei weiteren Umgebungen
   1. Test Umgebung «Dynamics 365 Test»
   2. Produktiv Umgebung «Dynamics 365 Prod»
5. Jetzt sind die Verbindungen zu den drei Umgebungen vorbereitet und können innerhalb des Projektes verwendet werden.

Der Teil 1 abgeschlossen. Die drei Umgebungen sind eingerichtet und das Azure DevOps Projekt für «Market Insights» vorbereitet.

Nun kann die «Market Insights» applikation in der «Power Apps Dev» Umgebung erstellt werden. Das Erstellen der «Market Insights» Applikation behandeln wir in einem separaten Artikel.