

# Proof-of-Concept

Mit unserem Proof-of-Concept (kurz: PoC) werden wir feststellen, ob wir in das System APIs (=Application Programming Interface), also externe Web-Services, einbinden können. Ein mögliches Beispiel für unser System könnte eine Wetterdaten-API sein, mit der man über entsprechende Koordinaten eines Orts die entsprechenden Informationen zur Wetterlage aufrufen kann. Wir möchten bei unserem PoC vor allem einsehen, ob die richtigen Informationen ausgewertet und dem späteren Endnutzer angezeigt werden und ob das zukünftige System jederzeit mit den aktuellen und korrekten Daten aktualisiert wird.

## Exit Kriterien

- Der Endnutzer sollte zu jeder Zeit die Anwendung bzw. das System aktualisieren können, je nach Input/Information, sollten sich die Daten/Output entsprechend automatisiert ändern.
- Der Benutzer soll auf aktuelle Informationen zugreifen können.
- Der Benutzer soll nicht länger als 5 Sekunden auf Informationen/Daten warten.
- Der Benutzer kann auf aktuelle Informationen zugreifen.
- Der Benutzer soll mit unterschiedlichen Optionen interagieren können.

## Fail Kriterien

- Die Seite lässt sich nicht aktualisieren und es werden keine aktuellen Informationen übertragen.
- Der Benutzer bekommt falsche bzw. ungültige Informationen, falls diese sich nicht auf den aktuellen Webservices beziehen.
- Der Benutzer kann keine aktuellen Informationen aufrufen, wenn sich zwischenzeitig die Verbindung trennt.
- Es werden keine angepassten Daten ausgewertet, falls diese nicht in der (eigenen/externen) Datenbank vorhanden sind.

## Fallbacks

- Zur Unterstützung werden APIs, also externe Web-Services benutzt
- Es könnte eine alternative Software zu Unterstützung einbezogen werden
- Zeitplan neu aufgestellt.
- Externe Datenbank zu Unterstützung