

# Website-Erstellung: Konzeption, Backend, Frontend

Um die in der LVA erlernten Konzepte anzuwenden ist im Rahmen der LVA ein gesamtes Projekt zu entwickeln, um folgendes Szenario zu realisieren. – „KWM Padlet“. „Padlet“ ist eine Software, mit der eine digitale Pinnwand entsteht, auf der Texte, Bilder, Videos, Links, Sprachaufnahmen, Bildschirmaufnahmen und Zeichnungen abgelegt werden können (siehe <https://de.padlet.com/>)

## Szenario Beschreibung

In der Übung sollen Sie ein vereinfachtes „Padlet-Service“ implementieren. Wir wollen für dieses Szenario eine webbasierte Applikation erstellen. Dabei soll es möglich sein, Padlets zu erstellen, User dazu einzuladen und diesen Rechte zuzuweisen (nur lesen, editieren, neu anlegen und löschen).

## Backend mit Laravel

- Implementierung des DB-Layers und der Businesslogik
- Implementierung einer entsprechenden REST-Schnittstelle

## Frontend mit Angular

- Erstellung eines entsprechenden Frontends
  - Erstellen von neuen (öffentlichen oder privaten) Padlets
  - Einladen von Usern zu einem Padlet + Rechtevergabe
  - Übersicht über die „eigenen“ Padlets
  - Detailansicht und Hinzufügen/Editieren von Inhalten (je nach Rechte) zu den Padlets
    - Text reicht in der Grundversion, gerne können Sie aber auch das Hochladen von Bilder/Video ermöglichen
  - Ratings und Kommentare zu einzelnen Padlet-Elementen

Erstellen Sie eine neue Laravel und Angular Applikation.

## Funktionale Anforderungen Backend

- Padlet Grundfunktionalität
  - Padlet mit zumind. Namen und Erstelldatum
  - User und Rechtezuordnung zu Padlet
  - Einträge in Padlet (Text reicht, Bilder/Videos optional)
  - Informationen zu User (Vorname/Nachname, Bild, ....)
  - Bewertung und Kommentar zu einem Eintrag.

- Benutzer
  - Authentifizierung und Login
  - Rechtevergabe auf Padlet-Ebene

## **Funktionale Anforderungen REST Service**

- CRUD Methoden für sämtliche Elemente in der DB
- Benutzer anlegen und verwalten (hierfür muss keine UI erstellt werden - User können über Seeders oder das REST Service angelegt werden)
- Padlets und Einträge anlegen, editieren und löschen
- Hinzufügen, entfernen von Usern \* Rechtevergabe durch Padlet-Ersteller

## **Funktionale Anforderungen Frontend**

- Öffentlicher Zugang
  - Auflistung und Einsicht in alle öffentlichen Padlets
  - Erstellen von öffentlichen Padlets
- Login-Möglichkeit
  - Erstellen von privaten Padlets (nur eingeladene User können Inhalte sehen)
  - Einladen von anderen Usern - diese müssen Einladung im System akzeptieren um Zugriff auf Padlet zu halten
  - Rechtevergabe
- Allgemeine Funktionalität (je nach Rechte)
  - Möglichkeiten zum Hinzufügen von beliebig vielen Einträgen
    - Bilder/Video optional
  - Bewertung und Kommentare zu einzelnen Padlet-Einträgen (nur eingeloggte User)
  - Nur Ersteller von Padlet kann User hinzufügen/löschen bzw. gesamtes Padlet löschen

## **Nicht-Funktionale Anforderungen**

- Achten Sie auf eine entsprechende Usability.
- Sämtliche Businesslogik / Persistierung / Datenverwaltung findet ausschließlich über entsprechende REST-Services statt -> Anwendung ist ausschließlich via Angular Frontend steuerbar
- Entsprechende Validierung und Datenprüfung

## **Abgabe**

Die Implementierung des BE muss auf dem zur Verfügung gestellten-Server erfolgen und muss dort lauffähig sein. Damit wir den Code begutachten können, checken Sie diesen in ein Git-Repository ein und geben Sie die URL an. Das FE sollte ebenfalls zugänglich sein - bei

lokaler Entwicklung, durch ein entsprechendes Deployment auf dem Server, damit wir das FE begutachten können. Checken Sie den Code ebenfalls in ein Git-Repository ein und geben Sie die URL (des Git-Repo und der Startseite) bekannt.

Erstellen Sie eine kurze Dokumentation. Geben Sie dabei einen Überblick wie Sie die Aufgabe gelöst haben (Überblick über die Seiten, Navigationspfade, usw z.B. in Form einfacher Mockups) und diskutieren Sie wichtige Entscheidungen (zB. Datenbank-Schema, Überblick über REST-API, Methoden, Routingkonzept,...). Die Dokumentation ist online im Moodle Kurs abzugeben.

Die Anwendung soll kurz während der Prüfung erläutert und präsentiert werden. Wir werden dazu (theoretische) Fragen stellen, die von Ihnen erklärt werden müssen.

Wichtig: Achten Sie bei der Implementierung der Logik und Views (Design) auf eine eigenständige Lösung - Copy / Paste wird entsprechend geahndet.

**Prüfungstermin ist der Fr. 26. Mai. Die Abgabe der Übung muss bis spätestens Mi. 24. Mai, 23.59 Uhr erfolgen.**