

- 1) Gruppe
 - Pierre Pötter Matr. Nr.
 - Martin Ott Matr. Nr.
- 2) Git
 - Github-link
- 3) Thema und Ziel der App
 - Diese App richtet sich an Fans des Fußballvereins Alemannia Aachen und ermöglicht es, besondere Momente aus dem Stadiongeschehen festzuhalten und zu teilen. Die aufgenommenen Videos werden dem jeweiligen Stadionblock zugeordnet, sodass Spielszenen aus verschiedenen Blickwinkeln betrachtet werden können. Ziel der App ist es, ein interaktives Fan-Erlebnis zu schaffen und die Gemeinschaft im Stadion digital zu vernetzen.

Anwendungsfälle (Use Cases) und Anforderungen (Requirements)

Aus diesen Anwendungsfällen leiten sich die Anforderungen an die App ab:

Fans nehmen während eines Spiels besondere Momente auf und laden diese hoch
 Der Upload ist nur in einem bestimmten Zeitraum nach dem Scannen eines QR-Codes im Stadion möglich
 Die Videos werden einem bestimmten Spiel zugeordnet und auf einer zeitbasierten Timeline dargestellt
 Nutzerinnen und Nutzer können gezielt zu bestimmten Spielszenen springen (z. B. Tor, Elfmeter)
 Es gibt eine Übersicht aller Spiele inklusive der zugehörigen Fan-Videos

Funktional

Fans können Videos aufnehmen und nach dem Scannen eines stadioninternen QR-Codes hochladen
 Die QR-Codes aktivieren für einen kurzen Zeitraum (z. B. 2 Minuten) die Upload-Funktion
 Jedes Video wird einem konkreten Spiel und einer Zeit auf der Spiel-Timeline zugeordnet
 In der Timeline lassen sich markierte Schlüsselmomente (z. B. Tore, Karten) direkt anwählen
 Die App verhindert doppelte Uploads identischer Videos durch zeitlich begrenzte QR-Zugriffe

Nicht-Funktional

Die App reagiert innerhalb von 2 Sekunden auf einen QR-Code-Scan
 Videos werden komprimiert und optimiert, um schnelle Upload-Zeiten zu gewährleisten
 Das Interface ist so gestaltet, dass auch bei schlechter Netzabdeckung möglichst viele Inhalte lokal gespeichert und später hochgeladen werden können
 Datenschutz und Urheberrecht werden durch Nutzungsbedingungen und Upload-Beschränkungen berücksichtigt

4) Aufgaben

1. Video-Service (nicht bewertet)

- Aufbau des Backends zur Speicherung und Verwaltung der Videos
- Anbindung an eine Datenbank (z. B. Firebase, Supabase oder CloudKit)
- API zur Video-Zuordnung nach Spiel und Zeit

2. QR-Code Scanner – Basisfunktionalität

- Implementierung der QR-Code-Scanner-Logik
- Verknüpfung mit Spiel-ID und zeitlich begrenztem Upload-Zugriff
- **3. UI – QR-Code Scanner**
- Gestaltung und Implementierung der Scan-Oberfläche
- Benutzerfeedback bei erfolgreichem oder ungültigem Scan
- **4. UI – Uploadbereich**
- Kamera-/Dateiauswahl
- Upload-Button mit Ladeanzeige und Bestätigung
- Verbindung zur QR-Freigabe
- **5. UI – Startseite**
- Übersicht über letzte Spiele, Navigation zu Upload/Spiele/Profil
- Grundlegendes App-Layout (Navigation Bar, Icons, etc.)
- **6. UI – Spielübersicht**
- Liste aller Spiele (Vergangenheit und Zukunft)
- Verlinkung zu jeweiligen Spiel-Timelines
- **7. UI – Spiel (Timeline)**
- Zeitstrahl mit markierten Ereignissen
- Möglichkeit zur Auswahl und Wiedergabe der hochgeladenen Fan-Videos
- **8. Feinschliff und Zusatzfeatures (optional)**
- Kommentarfunktion oder Likes
- Favoriten oder Markierungen
- Moderations-Tools
- **9. Testing und Optimierung**
- Technische Tests (Upload, QR, Timeline-Darstellung)
- UI/UX-Tests unter realen Bedingungen
- Performance-Optimierung und Bugfixes