

WayMatcher

1. Einleitung

1.1. Ziel des Dokuments

Dieses Pflichtenheft beschreibt die technischen und funktionalen Anforderungen an die Fahrgemeinschafts-Applikation WayMatcher. Es dient als Grundlage für die Entwicklung und Implementierung der Anwendung und stellt sicher, dass die Anforderungen des Auftraggebers erfüllt werden.

1.2. Projektbeteiligte

- **Auftraggeber:** Hr. Pausch
- **Entwicklungsteam:** Hr. Rudigier, Hr. Wonnebauer

1.3. Zielgruppe

- Jugendliche
 - Studierende
 - Personen ohne Führerschein
 - Sparsame Personen
-

2. Systembeschreibung

2.1. Anwendungsfall

WayMatcher ist eine Webanwendung, die die Vermittlung von Mitfahrgelegenheiten im regionalen Bereich ermöglicht. Benutzer können Reiseterrine mit detaillierten Informationen erstellen und nach passenden Mitfahrgelegenheiten suchen.

2.2. Systemarchitektur

Die Anwendung besteht aus folgenden Komponenten:

- **Frontend:** React mit TypeScript
- **Backend:** ASP.NET mit C#
- **Datenbank:** Azure (relational)

2.3. Schnittstellen

- **REST-API** für die Kommunikation zwischen Frontend und Backend
 - **Google Maps API** für die Kartenansicht
-

3. Funktionale Anforderungen

3.1. Reiseterrine

- **Erstellen:**
 - Typ (Passagier/Pilot)
 - Zielort
 - Datum & Uhrzeit (wiederkehrend möglich)
 - Strecke (Start & Ziel mit Radius)
 - Plätze
 - Inhaber
 - Preis/km (automatische Berechnung)
 - One-Shot/Seriell
 - Beschreibung
 - Fortbewegungsmittel
 - Status (offen, limitiert, abgebrochen, erledigt)
- **Beitreten/Erstellen:**
 - **Offene Reiset termine:** Direktes Beitreten
 - **Limitierte Reiset termine:** Angebot schreiben oder Einladung erhalten
- **Abbrechen:**
 - Mit Begründung (und evtl. Chatnachricht bzw. E-Mail Benachrichtigung)
- **Chat:**
 - Integrierter Chat innerhalb eines Reiset ermins mit anderen Benutzern

3.1.1. Zuständige Arbeitspakete

PSP-Code	PSP-Name
[1.1.4]	Event
[1.2.4]	Event-Service
[1.3.2]	Event-Planning
[1.3.3]	Event-Page

3.2. Einladungen

- Einladung zu Reiset erminen mit kurzer Nachricht
- Pflicht zum Ausfüllen des **Passagierformulars** nach Annahme

3.2.1. Zuständige Arbeitspakete

PSP-Code	PSP-Name
[1.1.1]	User
[1.1.4]	Event
[1.2.2]	E-Mail Service

PSP-Code	PSP-Name
[1.3.2]	Event-Planning
[1.3.5]	Messages/Notifications

3.3. Passagierformular

- Zielort, Startort (inkl. Suchradius)
- Datum & Uhrzeit
- Passagiere
- Beschreibung
- Angebot an Piloten erstellen
- Abbrechen mit Begründung

3.3.1. Zuständige Arbeitspakete

PSP-Code	PSP-Name
[1.1.4]	Event
[1.2.4]	Event-Service
[1.3.2]	Event-Planning
[1.3.3]	Event-Page

3.4. Benutzerverwaltung

- **Registrierung und Login** (E-Mail, Passwort, Benutzername)
- **Profilverwaltung:**
 - Führerschein
 - Fahrzeugbesitz
 - Profilbild
 - Beschreibung
 - Terminverlauf
 - Offene Termine
 - Statistik
 - Bewertungen (1-5 Sterne)
- **Login/Registrierung/Bearbeiten/Löschen Funktionen**
- **Liste von Fortbewegungsmitteln**

3.4.1. Zuständige Arbeitspakete

PSP-Code	PSP-Name
[1.1.1]	User
[1.2.3]	User-Service
[1.3.1]	User-Management

3.5. Suche/Filter

- Orte/Strecke
- Uhrzeit/Datum (Hin- und Rückfahrt)
- Anzahl freie Plätze
- Bestimmte Benutzer
- Beigetretene Reiset termine

3.5.1. Zuständige Arbeitspakete

PSP-Code	PSP-Name
[1.1.1]	User
[1.1.2]	Address
[1.1.3]	Vehicle
[1.1.4]	Events
[1.1.5]	Schedule
[1.1.10]	Views & Stored Procedures
[1.2.2]	E-Mail-Service
[1.2.3]	User-Service
[1.2.4]	Event-Service
[1.3.7]	Search

3.6. Karte

- Anzeige der Strecke (Google Maps)
- Start- und Endpunkt
- Öffnen/Schließen Funktionen

3.6.1. Zuständige Arbeitspakete

PSP-Code	PSP-Name
[1.3.4]	Map-View

4. Nichtfunktionale Anforderungen

4.1. Benutzerfreundlichkeit

- Die Anwendung soll mittels übersichtlichen Schaltflächen sowie ansprechende Namen benutzerfreundlicher gestaltet werden
- Das Design wird auf Basis von [Balsamiq Wireframe Mockup](#) basiert
- Die Farbpalette besteht aus diesen Farben:

- Primär(Akzent)-Farbe: #861388
- Sekundär-Farbe: #EEABC4
- Text-Farbe: #1D191B
- Erfolg-Farbe: #27873A
- Warnung-Farbe: #DEBB40
- Fehler-Farbe: #861322
- Hintergrund-Farbe: #EEDAE1

861388

EEABC4

1D191B

27873A

DEBB40

861322

EEDAE1

Wird über eine oder mehrere ausenstehende Personen evaluiert.

4.2. Performance

- Die hohe Performance der Anwendung soll durch [Design Patterns](#) und eine ausreichend normalisierte relationelle Datenbank ermöglicht werden.

4.2.1. Test

Wird von ausenstehenden Personen auf angemessene Performance geprüft

4.3. Sicherheit

- Die hohe Sicherheit soll erzielt werden, indem User-Daten verschlüsselt versendet werden.
- Der Webserver soll via HTTPS über Let's Encrypt Zertifikate aufrufbar sein.
- Packages und Frameworks werden auf dem neuesten stabilen Stand gehalten.

4.3.1 Test

Interne Penetration-tests sollen gewährleisten, dass die Sicherheit gegeben ist.

4.4. Zuverlässigkeit

- Das Hosting des Front- und Backends sowie auch die Datenbank werden an Cloud-Provider (Ionos & Azure) ausgelagert.

4.4.1 Test

Die Website wird auf Erreichbarkeit geprüft.

4.5. Kompatibilität

- Die Anwendung soll auf verschiedenen Geräten (Desktop, Tablet, Smartphone) gleichermaßen gut funktionieren. Wir zielen dabei übliche Desktop-Auflösungen an (z.B. Full-HD, 4k, ...) und marktführende Smartphones (z.B. iPhone 14 Pro Max, Pixel 7, Samsung Galaxy S20 Ultra, ...)

4.5.1 Test

Tests erfolgen über Dev-Tools (Chromium) im Browser. Es werden auch private Geräte für die Tests verwendet.

4.6. Wartbarkeit

- Die Anwendung verwendet [Design Patterns](#) und verfolgt die Objektorientierte-Programmierung
- Die technische und [projektorganisatorische Dokumentation](#) erfolgt in GitHub
 - Die Doku des Backends geschieht im [Backend Repository](#)
 - Die Doku des Frontends geschieht im [Frontend Repository](#)
 - Die Doku sowie auch Diagramme der Datenbank geschehen im [Datenbank Repository](#)

4.6.1 Test

Die Wartbarkeit wird von einer aussentehenden Person via Stichproben geprüft.

5. Rahmenbedingungen

5.1. Technische Anforderungen

5.1.1. Frontend

- React mit TypeScript
- Responsive Design (Mobil und Desktop)

5.1.2. Backend

- ASP.NET mit C#
- REST-API

5.1.3. Datenbank

- Azure (relational)

5.1.4. Hosting

- Lokaler Server

5.1.5. Projektorganisations-Tool

- GitHub Organisation inkl. Repositories

5.2. Zeitplan

- **Projektstart:** 05.02.2025
 - **Lastenheft abgeschlossen:** 07.02.2025
 - **Pflichtenheft abgeschlossen:** 21.02.2025
 - **Umsetzung:** 24.02.2025 - 18.03.2025
 - **Testphase:** 18.03.2025 - 24.03.2025
 - **Projektpräsentation:** 04.04.2025
 - **Projektabschluss:** 04.04.2025
-

6. Abnahmebedingungen & Abnahmetests

6.1. Abnahmezeitpunkt

Die Abnahme erfolgt mit Ende des Projektabschlusses: **01.04.2025**

6.2. Projektabnehmer

- **Verantwortliche Person:** Hr. Pausch
- **Qualitativer Projektabnehmer:** Hr. Pausch

6.3 Abnahmetests

6.3.1 Funktionale Tests

- siehe Testkonzept

6.3.2 Nichtfunktionale Anforderungen

- siehe Punkt [4. Nichtfunktionale Anforderungen](#)
-

7. Erweiterungen (Optional)

- **Karte:**
 - Ansicht der Strecke (**Google Maps**) mit Start- und Endpunkt.
 - Öffnen/Schließen Funktionen für die Kartenansicht.
- **Intelligente Pfadfindung der Reiseterrine** (z.B. Suche in einem Radius am Weg)
- **Chat zwischen Benutzern**
- Verwendung von **Designpattern**
- **Detaillierte Benutzerprofile** (Bewertungen, Automodell, Führerschein)
- **Multifaktor Authentifizierung**
- **Github Dokumentation** via Markdown und Projektmanagement
- **Clouddienste** (App-Service)
- **E-Mail Versand** (Passwort zurücksetzen & Benachrichtigungen)