

Proof of Concept

Matchingsystem

Das Matching findet zwar auf dem Server statt, setzt aber auch unterem Daten vom Client voraus, um ein für den Benutzer zufriedenstellendes Ergebnis zu liefern.

Konkret bedeutet dies, der Matching-Algorithmus (im folgenden “Anwendungslogik” genannt) serverseitigen Stammdaten der Benutzer und die aktuellen clientseitigen Fahrprofile zurückgreift, um relevante Radfahrpartner zu ermitteln.

Exit: Die Anwendungslogik kann auf alle benötigten Daten (das aktuelle Fahrprofil des Benutzers und die Stammdaten sämtlicher Benutzer) zugreifen. Das Matching liefert anschließend für den Benutzer zufriedenstellende Ergebnisse.

Sprich: Der Benutzer und die vorgeschlagenen Radfahrpartner treiben die selbe Radsportart(en), fahren etwa gleich schnell und leben vorzugsweise in der selben Stadt.

Fail: clientseitige Statistiken und/oder die Stammdaten sämtlicher Benutzer können nicht aufgerufen werden. Das Matching liefert keine und nicht zufriedenstellende Ergebnisse.

Beispiel: Es werden nur Benutzer in der näheren Umgebung gefunden, die jedoch nicht die gleiche Radsportart treiben.

Fallback: ?

Erstellung der Fahrprofile

Zu den strategischen Zielen gehört das Finden gleichwertiger Radfahrpartner. Um die Gleichwertigkeit zweier Radfahrpartner zu bestimmen, werden die Fahrprofile miteinander verglichen. Dies setzt voraus, dass die Fahrprofile korrekt und vollständig sind.

Exit: Ein aussagekräftiges Fahrprofil vom Benutzer liegt vor, wenn es seine durchschnittliche Fahrgeschwindigkeit und die Radsportarten beinhaltet.

Fail: Aufgrund einer fehler- oder lückenhaften Aufzeichnung der GPS-Daten kann kein korrektes Fahrprofil erstellt werden.

Fallback: Aufzeichnungslücken schätzen.

Verarbeitung der GPS-Daten

Mit Hilfe der GPS Daten soll die Erstellung eines Fahrerprofils ermöglicht werden.

Exit: Das System liefert korrekte Ergebnisse: Sämtliche vorgeschlagene Events finden im tatsächlichen Aufenthaltsort statt.

Fail: Ein oder mehrere Events finden anderswo statt.

Fallback: Es gibt keine Alternative zum GPS. Ansonsten muss der Benutzer die erforderlichen Daten per Hand eingeben.

Asynchrone Kommunikation mit GCM

Benachrichtigung von Events in der Umgebung oder wenn ein Matching stattgefunden hat, mit Hilfe unter Verwendung eines Push Benachrichtigungsdienstes.

Exit: Der Nutzer wird automatisch benachrichtigt wenn ein Matching vorliegt.

Fail: Auf Grund technischer Störungen wird die Push Nachricht nicht erfolgreich ermittelt.

Fallback: Als Alternative kommen entweder andere Dienste in Frage oder die Auswahl eines anderen Messaging-Pattern wie zum Beispiel das Publish/Subscribe Prinzip, nach dem das node.js faye arbeitet.