

Abstract

Spikee ist eine Anwendung zur professionellen Trainings- und Fitnessanalyse. Spikee integriert aktuelle medizinische Erkenntnisse, intelligente Algorithmen und die langjährige Erfahrung von Leistungssportlern zu einem leistungsfähigen Analysewerkzeug. Aufgabe innerhalb dieses Projektes war es die bestehende Web Applikation um die sogenannten Wellness oder Aktivitätsdaten des Users zu ergänzen. Es werden zum Beispiel die Schritte gemessen die der User an einem Tag zurückgelegt oder die Kalorien die er verbrannt hat. Diese werden nun in einer extra Kalenderansicht visualisiert um Rückschlüsse über das allgemeine Bewegungsprofil zu geben.

Beschreibung

Die Applikation Spikee beschäftigt sich im Kern mit der Analyse von Leistungsdaten wie sie bei Sportlern erstellt werden können. Neben den Leistungsdaten aus Trainingszyklen werden von den verschiedenen Fitnessgeräten jedoch auch weitere Daten gesammelt die Rückschlüsse darüber geben können wieviel sich ein Nutzer am Tag bewegt hat. Zum Beispiel die Gesamtzahl der zurückgelegten Schritte oder die verbrannten Kalorien. Inhalt war es diese Daten zuerst über eine REST Schnittstelle von den Anbietern Polar und Garmin abzufragen und die Daten in einem JSON Format zu serialisieren. Dabei wurden sowohl zusammenfassende Daten pro Tag abgefragt als auch Daten die kürzere Zeitabstände beschreiben. Diese Informationen wurden anschliessend in einem angelegten Datenbankmodell der Datenbank MongoDB eingetragen und in einem zweiten Schritt innerhalb einer Django Webapplikation visualisiert. Zur Visualisierung wurde ein HTML Template genutzt das eine spezielle Kalenderansicht darstellt und mithilfe von HTML5 Canvas um Grafiken bereichert wird.

Wichtige Schritte des Projektes

1. Schnittstelle zur REST API von Garmin und Polar erstellen
2. JSON Format zur Ablage der jeweiligen API Daten definieren
3. Vereinheitlichendes Datenbankmodell in MongoDB definieren
4. Algorithmus erstellen um die JSON Daten in die DB zu parsen
5. Template in Django erstellen das die DB Daten darstellt
6. Spikee Web Applikation um das Template ergänzen und anpassen
7. JavaScript Rendering mit HTML5 Canvas nutzen für Grafiken

Benutzte Technology

Die Umsetzung erfolgte mithilfe der PyCharm IDE und in den Programmiersprachen Python, JavaScript, HTML5 und CSS. Die Authentifizierung der REST APIs erfolgte mit dem OAuth 1.0 und 2.0 Verfahren. Serialisiert wurden die Daten im JSON Format. Es kamen diverse Python Libraries zum Einsatz, auf eine Auflistung dieser wird verzichtet.

Beispiel Ausschnitt der JSON Datenstruktur

```

"summary": {
  "active-calories": 0,
  "active-steps": 972,
  "calories": 1680,
  "created": "2017-05-25T13:09:32.000",
  "date": "2017-05-21",
  "duration": "PT29M",
  "id": "1431XXXX",
  "polar-user": "https://www.polaraccesslink.com/v3/users/3XX",
  "spikee_data_type": "polar_wellness_activity",
  "spikee_user_id": 2,
  "transaction-id": 113869214
}
    
```

Entwicklungsstufen der Ansicht

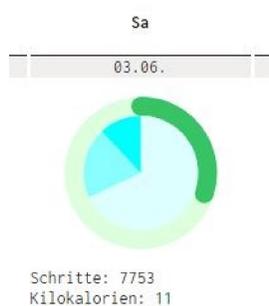


Figure 1: Ein erster Ansatz mit Kuchendiagramm



Figure 2: Detailansicht der Aktivitätsdarstellung

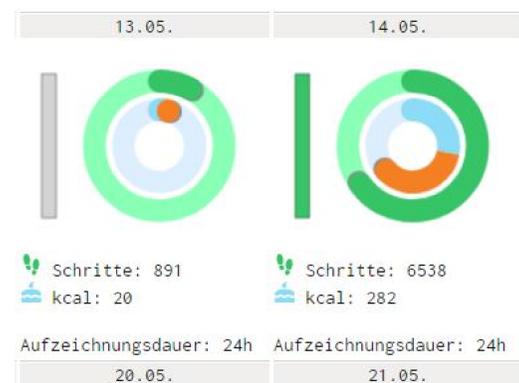


Figure 3: Integration der Morgenmessungen

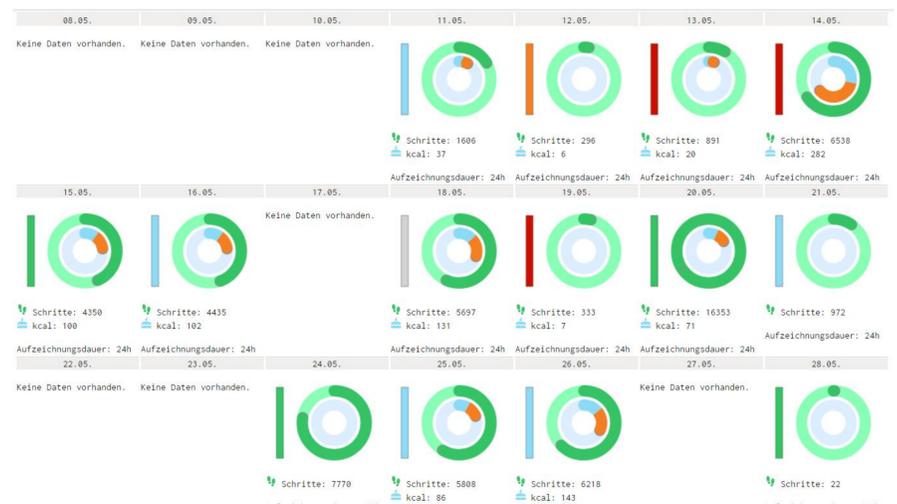


Figure 4: Aktivitätsdaten im Kalender integriert

Weiterführende Ideen

- Die Entkopplung von Daten und Visualisierung kann leicht genutzt werden um eine reine textbasierte Darstellung für spezielle Endgeräte bereitzustellen
- Die Detailansicht eines Tages wäre möglich welche genauer anzeigt zu welchem Zeitpunkt der Nutzer am aktivsten war

Danksagung

Mein Dank gilt Herrn Professor Maucher für die Betreuung des Projektes und Frau Annette Landmesser für die technische Betreuung beim Aufsetzen der Entwicklungsumgebung und bei Problemen während der Entwicklung.