



Planung, Entwicklung und Einsatz eines leicht weiterentwickelbaren Systems für Persönlichkeitstests

Bearbeiter: Nicolai Wolfrom

Betreuer: Prof. Dr. Lutz Prechelt

Gliederung



Thema

- Entstehung
- Systemanforderungen



Aufgaben

- Aufgabengebiete
- Schwerpunkte



Lösungsansatz

- Aktueller Stand
- Technologien

Thema

- ▶ Entstehung
- ▶ Systemanforderungen

- ▶ Bekannte arbeitet mit Persönlichkeitstests
 - ▶ IT-System zur Unterstützung
- ▶ Projekt mit Betreuer in passendes Format gebracht
 - ▶ Umfang angepasst
 - ▶ Schwerpunkte gesucht

- ▶ Unzureichende „Fertiglösung“ ablösen
 - ▶ Aktuell keine Autorisierung, Auswertung oder Ergebnisdarstellung
- ▶ Leicht wart- & weiterentwickelbar
 - ▶ Neue Anforderungen können folgen

Aufgaben

- ▶ Aufgabengebiete
 - ▶ Technologie-Stack & Grundaufbau
 - ▶ Deployment & VCS
 - ▶ (Agiles) Entwickeln
 - ▶ Weiterentwickelbarkeit
- ▶ Schwerpunkte

Aufgaben Aufgabengebiete

- ▶ Technologie-Stack auswählen & Grundaufbau erstellen
 - ▶ Cloud Service: SaaS, PaaS oder IaaS
 - ▶ Datenbank System
 - ▶ Backend
 - ▶ Programmiersprache, Framework, ...
 - ▶ Frontend
 - ▶ Survey Dienst
- ▶ Deployment & VCS
- ▶ (Agiles) Entwickeln
- ▶ Weiterentwickelbarkeit

Aufgaben Aufgabengebiete

- ▶ Technologie-Stack auswählen & Grundaufbau erstellen
- ▶ Deployment & VCS
 - ▶ Version Control System (VCS): git & github
 - ▶ Cloud Service (SaaS / PaaS / IaaS) einrichten
 - ▶ Deployment ermöglichen – z.B. über VCS
- ▶ (Agiles) Entwickeln
- ▶ Weiterentwickelbarkeit

Aufgaben Aufgabengebiete

- ▶ Technologie-Stack auswählen & Grundaufbau erstellen
- ▶ Deployment & VCS
- ▶ (Agiles) Entwickeln
 - ▶ Grundfunktionalitäten (Autorisierung, Auswertung, Ergebnisdarstellung, ...)
 - ▶ Integration Survey Dienst
 - ▶ REST Schnittstellen
 - ▶ Feedback & Tests
 - ▶ Regelmäßige Gespräche & Praxistests
- ▶ Weiterentwickelbarkeit

Aufgaben Aufgabengebiete

- ▶ Technologie-Stack auswählen & Grundaufbau erstellen
- ▶ Deployment & VCS
- ▶ (Agiles) Entwickeln
- ▶ Weiterentwickelbarkeit
 - ▶ Test der Weiterentwickelbarkeit
 - ▶ Externe Testperson(en)
 - ▶ Ergebnisauswertung
 - ▶ Verbesserungen umsetzen

- ▶ Gute Technologien wählen
 - ▶ Cloud Service (Provider), Programmiersprachen, Framework, Bibliotheken, ...
- ▶ Entwickeln einer ganzheitlichen Lösung
 - ▶ Von Planung, Entwicklung & Testen bis Deployment
 - ▶ (Agiles) Entwickeln mit Bekannten
- ▶ Weiterentwickelbarkeit
 - ▶ Weiterentwickelbarkeit durch Code Organisation, Dokumentation, ...
 - ▶ Test durch externe Testperson(en)

Lösungsansatz

- ▶ Aktueller Stand
- ▶ Technologien

Lösungsansatz

Aktueller Stand

- ▶ Bachelorarbeit geplant
 - ▶ Mit Betreuer in passendes Format gebracht
 - ▶ Exposé / Outline geschrieben
- ▶ Über mögliche Technologien informiert
 - ▶ Cloud Service Dokumentation
 - ▶ „Hello World“ Applikation mit möglichen Technology-Stack

- ▶ Cloud Service Provider: Heroku als PaaS
 - ▶ Stellt Betriebssystem & Laufzeit bereit
 - ▶ Deployment mit git
- ▶ Datenbank: PostgreSQL
 - ▶ Ebenfalls von Heroku bereitgestellt
- ▶ Backend: Python
 - ▶ Gunicorn (Webserver) & Flask (Webframework)
- ▶ Frontend: HTML, JavaScript (VanillaJS) & CSS
 - ▶ Dynamische Testgenerierung: SurveyJS (externe Bibliothek)