



Fakultät Informatik

Betreuerin: Dr. Iris Braun

**Dokumentation
Gruppe 3**

Service and Cloud Computing

Eingereicht von

Name	Email	Matrikelnummer	Studium
Julis Döbelt	julus.doebelt @mailbox.tu-dresden.de	4962026	Wirtschafts- informatik
Christoph Sense	christoph.sense @mailbox.tu-dresden.de	4965109	Wirtschafts- informatik
Christoph Lommatzsch	christoph.lommatzsch @mailbox.tu-dresden.de	4949731	Wirtschafts- informatik

Inhaltsverzeichnis

Team.....	3
Vorgehensweise.....	3
Schematischer Aufbau der Microservices.....	4
Technologien.....	4
Verwendete Software und Bibliotheken.....	4
Sicherheit.....	4
Deployment.....	5
Schnittstellenbeschreibung.....	5
Bedienungsanleitung für Clients.....	5
Feedback und Kritik am Praktikum.....	7

Team

Das Projektteam besteht aus Julius Döbelt, Christoph Sense und Christoph Lommatzsch. Die Aufgabenverteilung war nicht strikt geregelt, sondern wurde situationsabhängig für den nächsten Arbeitsschritt gemeinsam abgestimmt.

Vorgehensweise

Zu Beginn des Projekts stellte sich die Frage, welches Anwendungsfeld unser Programm bedienen soll. Gemeinsam wurde sich für ein Notenmanagementsystem für Schüler entschieden, indem Nutzer ihre neuen Schulnoten mit dem jeweiligen Fach eintragen können und eine entsprechende Auflistung mit Auswertung abrufbar ist.

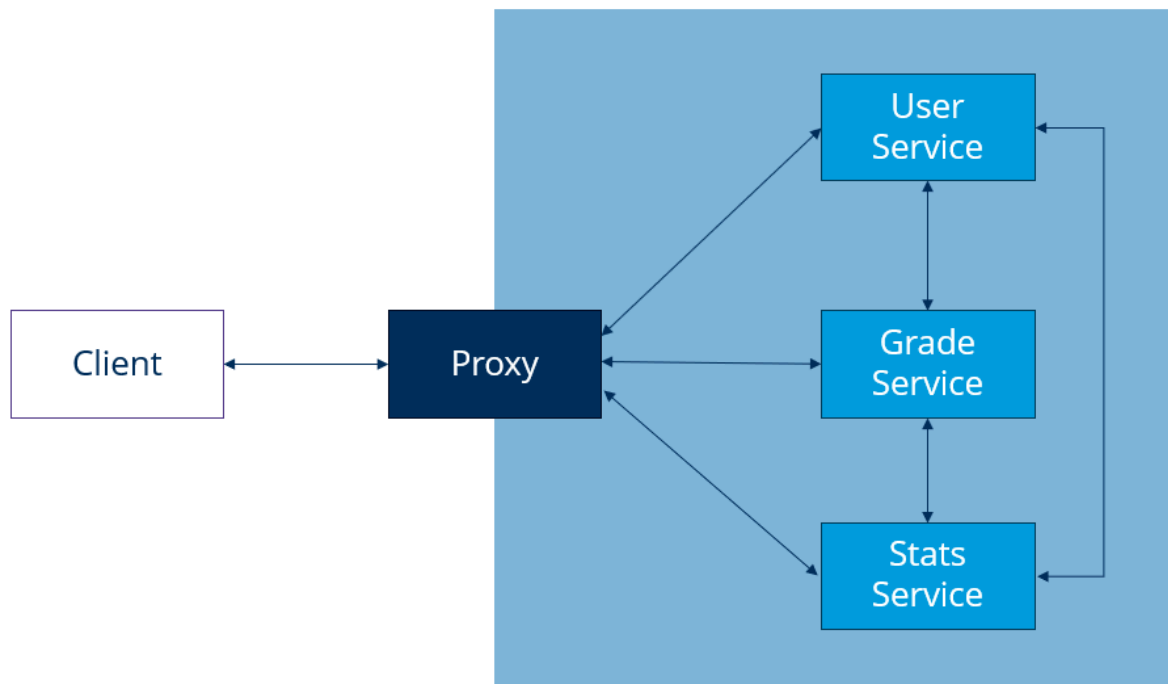
Als ersten Schritt wurde versucht, die einzelnen Services parallel und von verschiedenen Gruppenmitgliedern entwickeln zu lassen. Diese Vorgehensweise hat sich recht schnell als unpraktisch erwiesen. In dieser Zeit entstanden von verschiedenen Mitgliedern jeweils unabhängige Teilprogramme, die nie den gesamten Funktionsumfang abdeckten. Julius Döbelt entwickelte beispielsweise ein Programm, welches zwar alle geplanten Services umfasste, jedoch nicht den Anforderungen einer REST-Anwendung genügte, sondern als monolithische Anwendung aufgebaut war.

Die Herausforderung bestand folglich darin, die verschiedenen Ansätze zu kombinieren. In diesem Zug wurde ein Proxy-Service erstellt, der es möglich macht, die Anfragen an unterschiedliche Endpunkte der Microservices zu routen. Der Proxy-Service ermöglicht somit eine konsolidierte Schnittstelle für den Client, wodurch dieser nicht direkt mit den individuellen Microservices interagieren muss. Dies erleichtert die Wartung, Skalierbarkeit und Erweiterbarkeit der Anwendung, da Änderungen in den Microservices transparent für den Client erfolgen können, solange die Schnittstelle des Proxys beibehalten wird.

Im nächsten Arbeitsschritt wurden die Funktionalität des Grade- und Stats-Service aus der zuvor entwickelten, monolithischen Anwendung extrahiert und als einzelne Microservices implementiert.

Schwierige Fragestellungen, z.B. beim Debugging oder die Umsetzung der Funktionalität *Löschen einer Note*, wurden gemeinsam im Team analysiert und bearbeitet.

Schematischer Aufbau der Microservices



Technologien

Verwendete Software und Bibliotheken

- Flask als Webservice-Framework für Python
- Docker zur Containerization
- GitLab als Versionsverwaltungstool
- SQLAlchemy als Python-Bibliothek zur Datenbankbindung
- JWT (JSON Web Token) zur Authentifizierung
- Matplotlib zur Erstellung von Histogrammen
- Requests zur Kommunikation mit HTTP

Sicherheit

Die Kommunikation zwischen Client und Proxy erfolgt über https. Das Zertifikat ist selbst signiert, weshalb beim Öffnen über den Browser eine Sicherheitswarnung ausgespielt wird. Das Zertifikat wurde mithilfe von OpenSSL selbst generiert. In einer Produktionsumgebung wäre es notwendig, ein Zertifikat von einer vertrauenswürdigen Zertifizierungsstelle zu erwerben.

Deployment

Zum Deployment der Anwendung auf der virtuellen Maschine von Cloud&Heat wurde das vorhandene Betriebssystem Linux (Debian) genutzt. Dafür wurde zuerst die aktuelle Version von Docker auf der virtuellen Maschine installiert. Durch die Containerisierung unserer Anwendung wurden anschließend nur die Docker Container gestartet.

Schnittstellenbeschreibung

Die Swagger-Schnittstellenbeschreibungen sind als yaml-Dateien auf GitLab zu finden.

Bedienungsanleitung für Clients

Rufen Sie zunächst die Anwendung über 185.128.118.134 über Ihren Browser auf. Es öffnet sich die Startseite. Über *Login* erreichen Sie die Anmeldemaske, über die Sie sich mit Ihrem Benutzernamen und Ihrem Passwort anmelden können. Besitzen Sie noch keinen Account, so können Sie über den unteren Link einen Account registrieren.

[Login](#) [Register](#)

Register for Scholarum

Username:

Password:

Confirm Password:

[Register](#)

Already have an account? [Login](#)

Sie können sich mit folgenden Login-Daten anmelden:

Username: Testuser1

Password: Password123

Nach dem Login werden Sie zum Dashboard weitergeleitet, wo Sie Ihre aktuelle Notenübersicht einsehen können. Durch Eingabe einer neuen Note mit Fach und anschließendem Eintragen, wird die neue Note dem Schulfach der Tabelle hinzugefügt.

Overview

Subject:

Add new subject

New Subject:

physics

Grade:

2

Add grade

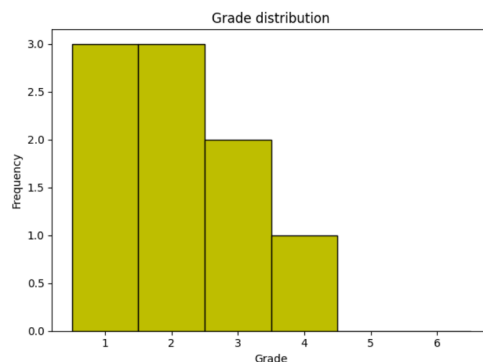
Sollte das Fach neu sein, wird es automatisch angelegt und steht Ihnen zur Auswahl bereit. Noten werden in einer Liste untereinander angezeigt und können durch einen Klick auf das Kreuz auf der rechten Seite gelöscht werden

Wechseln Sie die Seite von *Home* auf *Dashboard*, wird Ihnen Ihre Notenstatistik angezeigt. Sie sehen Ihren aktuellen Notendurchschnitt je Fach und Ihre Notenverteilung über alle Noten als Histogramm. Die Farbe des Histogramms ändert sich je nach Gesamtdurchschnitts des Nutzers. Ist der Durchschnitt geringer als 2.0, ist das Histogramm grün, ab 4.0 rot und dazwischen gelb.

Dashboard

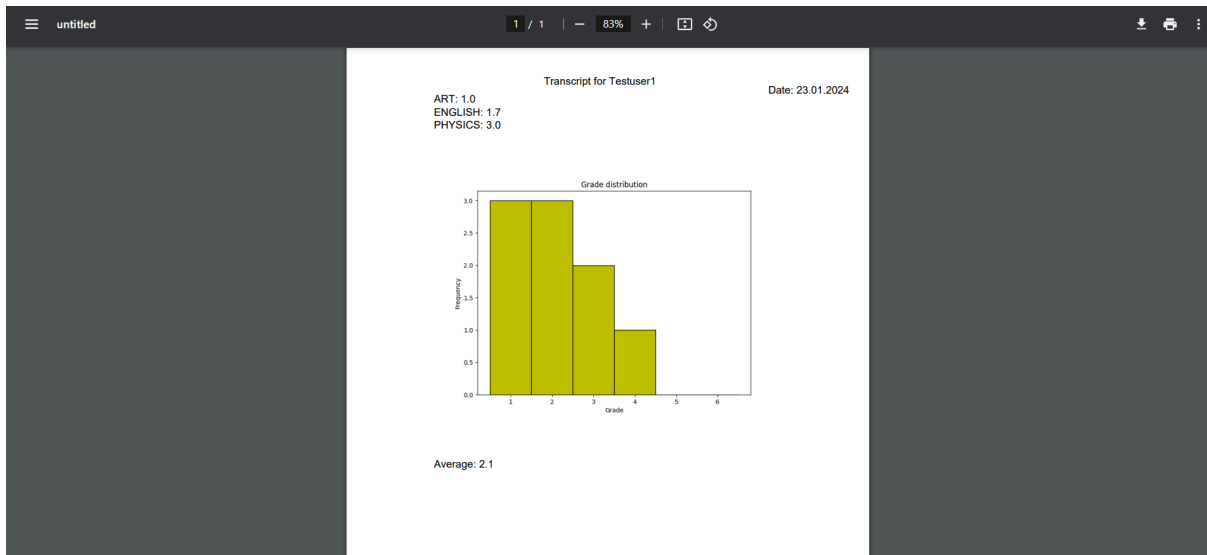
Subject	Average Grade
ART	1.0
ENGLISH	1.7
PHYSICS	3.0

Grade distribution



Create transcript

Zusätzlich dazu können Sie sich ein Transcript als PDF generieren lassen. Diese gibt Ihnen zusätzlich Auskunft über Ihren gesamten Notendurchschnitt.



Auf der Seite Account können Sie durch zweimalige Eingabe ein neues Passwort für Ihr Konto vergeben und Ihr Konto aus dem System löschen.

Home Dashboard Account Logout

Account Data

Change Password

New Password:

Confirm new Password:

Change Password

Delete Account

☐ I want to delete my account

Delete Account

Über den Menüpunkt *Logout* können Sie sich abmelden und werden zur Startseite weitergeleitet.

Feedback und Kritik am Praktikum

Das Praktikum hat einen sinnvollen und guten Überblick über die Entwicklung einer eigenen REST-Anwendung gegeben. Das praktische Arbeiten im Team und die gemeinsame Entwicklung einer Lösung hat uns sehr gefallen. Die Einarbeitung in die Tools wie Docker, GitLab fiel den allen Teilnehmern aufgrund mangelnder Vorerfahrung zunächst etwas schwer. Hier hätte die einführende Übung vielleicht etwas ausführlicher und langsamer gestaltet werden können. Im Laufe der Einarbeitung funktionierte die Kollaboration und das Nutzen der Tools aber immer besser. Das theoretische Wissen aus der Vorlesung konnte mit Hilfe des Praktikums in praktisches Wissen transferiert werden. Auch schön war, dass die Aufgabenstellung relativ offen war, sodass eine eigene Idee kreativ umgesetzt werden konnte. Allerdings wäre es dafür hilfreich gewesen, wenn die

Vorlesung das REST-Thema gerade im Vergleich zu klassischen Web-Services mit SOAP stärker behandeln würde.