



TITRAWEB

Sifat Mahin Parvez



6. JUNI 2024

TFO-MAX VALIER BOZEN

Abstrakt

TitraLab WebApp: Eine Webanwendung zur Analyse und Visualisierung von Titrationsdaten

Dieses Maturaprojekt befasst sich mit der Entwicklung der TitraLab WebApp, einer benutzerfreundlichen Webanwendung zur Analyse und Visualisierung von chemischen Titrationsdaten. Das Hauptziel der Anwendung ist es, eine intuitive Oberfläche zu bieten, die es Benutzern ermöglicht, Titrationsdaten effizient zu verarbeiten und in Diagrammen darzustellen.

Die TitraLab WebApp erlaubt es Benutzern, ihre Titrationsdaten in einer Excel-Datei hochzuladen. Die Anwendung analysiert diese Daten automatisch und identifiziert wichtige Punkte wie den Äquivalenzpunkt, den Neutralpunkt und den Pufferbereich, die visuell in den erstellten Diagrammen hervorgehoben werden.

Der theoretische Hintergrund umfasst die Verwendung chemisch-mathematischer Modelle zur Berechnung und Visualisierung der Titrationskurven. Diese Modelle beinhalten unter anderem die Berechnung des Äquivalenzpunkts, des Neutralpunkts und der Pufferkapazität, die für das Verständnis der Titrationsprozesse entscheidend sind.

Die verwendeten Technologien umfassen HTML und CSS für die Strukturierung und Gestaltung der Benutzeroberfläche, JavaScript für die Interaktivität und Python mit dem Flask-Framework für die Backend-Entwicklung. Die Datenverarbeitung erfolgt durch die leistungsstarke Bibliothek Pandas.

Ein wesentlicher Teil der Arbeit bestand in der Planung und Implementierung des Projekts, die durch verschiedene Diagramme wie das Use Case-Diagramm und das GANTT-Diagramm unterstützt wird. Die Anwendung wurde umfassend getestet, um ihre Funktionalität und Zuverlässigkeit zu gewährleisten.

Das Projekt bietet wertvolle Einblicke in die Webentwicklung und Datenanalyse und stellt eine nützliche Ressource für sowohl erfahrene Chemiker als auch Anfänger dar, die chemische Titrations durchführen und analysieren.

Inhaltsverzeichnis

Abstrakt	1
TitraLab WebApp: Eine Webanwendung zur Analyse und Visualisierung von Titrationsdaten	1
Arbeitsauftrag	3
Überblick	3
Inhalt des Projekts	3
Ziel des Projekts	3
Verwendete Technologien	3
HTML/CSS	3
JavaScript	4
Python/Flask	4
Theoretischer Hintergrund	5
Planung	5
Use Case-Diagramm	5
Aktivitätsdiagramm	6
GANTT-Diagramm	6
Projektphasenbeschreibung	7
Implementation	8
Frontend	8
Backend	8
Datenverarbeitung	9
Testbericht	9
Beobachtungen und Erkenntnisse	10
Ausblick und Zukünftige Pläne	10
Quellen	11

Arbeitsauftrag

Überblick

TitraLab WebApp ist eine Anwendung, die es ermöglicht, Titrationsdaten zu verarbeiten und in Diagrammen darzustellen. Sie soll eine benutzerfreundliche Oberfläche bieten, um chemische Titrations einfach und effizient zu analysieren. Benutzer können ihre Daten in einer Excel-Tabelle hochladen und die Anwendung erstellt automatisch Diagramme und Analysen der Titration.

Inhalt des Projekts

Die TitraLab WebApp verarbeitet hochgeladene Titration Daten und stellt sie visuell dar. Nach dem Hochladen werden die Daten analysiert und die Ergebnisse werden in einem Diagramm dargestellt. Die WebApp identifiziert wichtige Punkte wie den Neutralpunkt, den Äquivalenzpunkt und den Pufferbereich und stellt diese im Diagramm dar.

Ziel des Projekts

Das Ziel des Projekts ist es, eine intuitive und benutzerfreundliche Webanwendung zu entwickeln, die es Benutzern ermöglicht, Titration Daten effizient zu verarbeiten und zu visualisieren. Die Anwendung soll sowohl für erfahrene Chemiker als auch für Anfänger geeignet sein und eine sichere und datenschutzfreundliche Verarbeitung der Daten gewährleisten.

Verwendete Technologien

HTML/CSS

HTML (HyperText Markup Language) ist die Grundlage des World Wide Web und definiert die Struktur und den Inhalt von Webseiten. Durch die semantische Auszeichnung können Elemente einer Webseite klar definiert und interpretiert werden. HTML bietet eine Vielzahl von Tags und Attributen, um Texte, Bilder, Links, Formulare und andere Inhalte zu strukturieren und zu präsentieren. Mit HTML können Entwickler Webseiten erstellen, die barrierefrei, suchmaschinenoptimiert und plattformübergreifend zugänglich sind.

CSS (Cascading Style Sheets) ist eine Stylesheet-Sprache, die verwendet wird, um das Aussehen und das Layout von HTML-Dokumenten zu definieren. CSS ermöglicht die Trennung von Inhalt und Präsentation, was bedeutet, dass Entwickler das Erscheinungsbild einer Webseite unabhängig vom HTML-Code steuern können. Durch die Definition von Stilen für Texte, Farben, Abstände, Positionen und andere visuelle Eigenschaften können Entwickler das Erscheinungsbild ihrer Webseiten genau kontrollieren und anpassen.

CSS bietet eine Vielzahl von Selektoren und Eigenschaften, die es Entwicklern ermöglichen, komplexe Layouts und Designs zu erstellen. Durch die Verwendung von CSS-Regeln können Entwickler das Aussehen einer Webseite auf verschiedene Bildschirmgrößen und Gerätetypen anpassen, um eine konsistente Benutzererfahrung auf allen Plattformen zu gewährleisten.

JavaScript

JavaScript ist eine leistungsstarke Skriptsprache, die in der Webentwicklung weit verbreitet ist. Sie ermöglicht die Erstellung interaktiver Benutzeroberflächen und die Manipulation von HTML-Dokumenten. JavaScript wird häufig verwendet, um Benutzeraktionen zu verarbeiten, Formularvalidierung durchzuführen, Inhalte dynamisch zu aktualisieren und komplexe Anwendungslogik auf der Client-Seite auszuführen.

JavaScript kann direkt in HTML-Dokumente eingebettet oder in externen Dateien verlinkt werden. Es bietet eine Vielzahl von Funktionen und Objekten, die es Entwicklern ermöglichen, interaktive Webanwendungen zu erstellen, die auf Benutzeraktionen reagieren und dynamische Inhalte generieren können. JavaScript wird auch häufig in Verbindung mit AJAX (Asynchronous JavaScript and XML) verwendet, um Daten asynchron von einem Server zu laden und zu aktualisieren, ohne die gesamte Webseite neu zu laden.

Python/Flask

Python ist eine interpretierte, hochgradig lesbare Programmiersprache, die für ihre Einfachheit und Eleganz bekannt ist. Sie wird in einer Vielzahl von Anwendungen eingesetzt, darunter Webentwicklung, Datenanalyse, künstliche Intelligenz und wissenschaftliches Rechnen. Python zeichnet sich durch eine klare und ausdrucksstarke Syntax aus, die es Entwicklern ermöglicht, komplexe Probleme mit wenig Code zu lösen.

Flask ist ein leichtgewichtiges Web-Framework für Python, das einfach zu verwenden ist und sich gut für kleine bis mittelgroße Webanwendungen eignet. Es bietet eine einfache und flexible Architektur, die es Entwicklern ermöglicht, Webanwendungen schnell zu entwickeln und zu skalieren. Flask bietet grundlegende Funktionen und Bibliotheken für Routing, Templates, Datenbankintegration und Authentifizierung, ohne den Entwickler mit unnötigen Funktionen zu überladen.

Durch die Kombination von Python und Flask können Entwickler leistungsstarke und skalierbare Webanwendungen erstellen, die den Anforderungen moderner Anwendungen gerecht werden. Flask bietet eine aktive Entwicklergemeinschaft und eine Vielzahl von Erweiterungen, die es Entwicklern ermöglichen, ihre Anwendungen einfach zu erweitern und anzupassen.

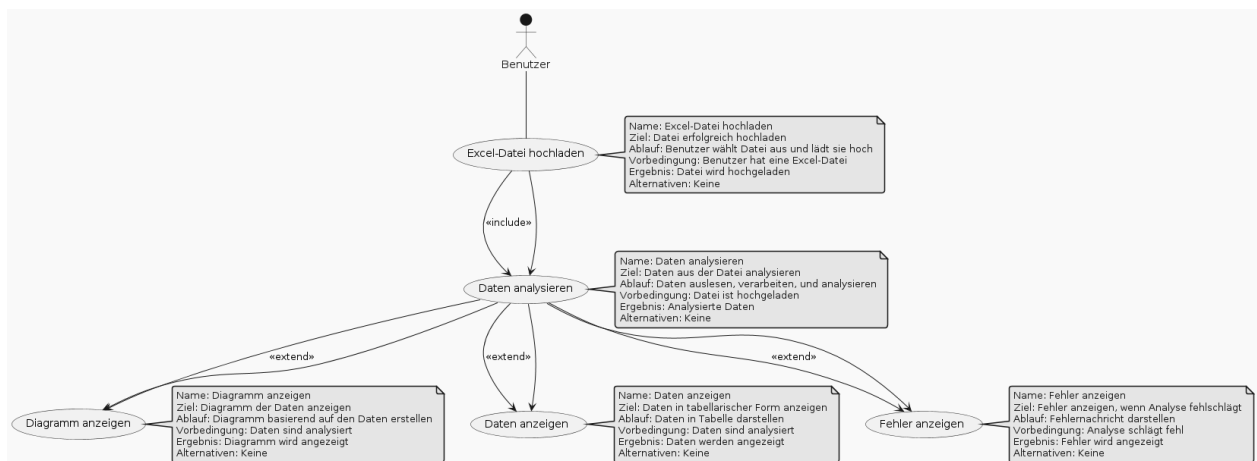
Theoretischer Hintergrund

Bei der Entwicklung der TitraLab WebApp wurden verschiedene chemisch-mathematische Modelle eingesetzt, um die Titrationsdaten präzise zu analysieren und darzustellen. Ein zentrales Modell ist die Berechnung des Äquivalenzpunkts, bei dem die Menge der

zugesetzten Titrationslösung der Menge der Substanz in der Probe entspricht. Dies wird durch die erste und zweite Ableitung der Titrationskurve ermittelt. Zudem wurde das Konzept des Neutralpunkts berücksichtigt, welcher den Punkt beschreibt, an dem die Lösung neutral (pH 7) ist. Die Berechnung der Pufferkapazität, die die Fähigkeit einer Lösung beschreibt, den pH-Wert bei Zugabe von Säuren oder Basen konstant zu halten, wurde ebenfalls integriert. Diese Modelle basieren auf grundlegenden Prinzipien der analytischen Chemie und ermöglichen es, die komplexen Wechselwirkungen und Reaktionen während der Titration genau zu verstehen und darzustellen. Die Anwendung dieser Modelle stellt sicher, dass die Benutzer genaue und verlässliche Ergebnisse erhalten.

Planung

Use Case-Diagramm



Im Use-Case-Diagramm wird die Interaktion zwischen den Benutzern (Akteuren) und den Funktionen der TitraLab WebApp dargestellt. Die Hauptakteure sind der Benutzer, der Daten hochlädt, und das System, das die Daten verarbeitet. Die Pfeile in einem Use-Case-Diagramm zeigen die Richtung der Interaktionen zwischen Akteuren und Anwendungsfällen.

Das Use-Case-Diagramm (siehe Abbildung 1) zeigt die Interaktionen zwischen dem Benutzer und der TitraLab WebApp. Die Hauptakteure sind:

- **Benutzer:** Der Benutzer lädt Titrationsdaten in Form einer Excel-Datei hoch und interagiert mit der WebApp, um die analysierten Daten und Diagramme anzuzeigen.
- **System:** Die WebApp verarbeitet die hochgeladenen Daten, analysiert sie, berechnet wichtige Punkte wie den Äquivalenzpunkt und visualisiert die Ergebnisse in Diagrammen.

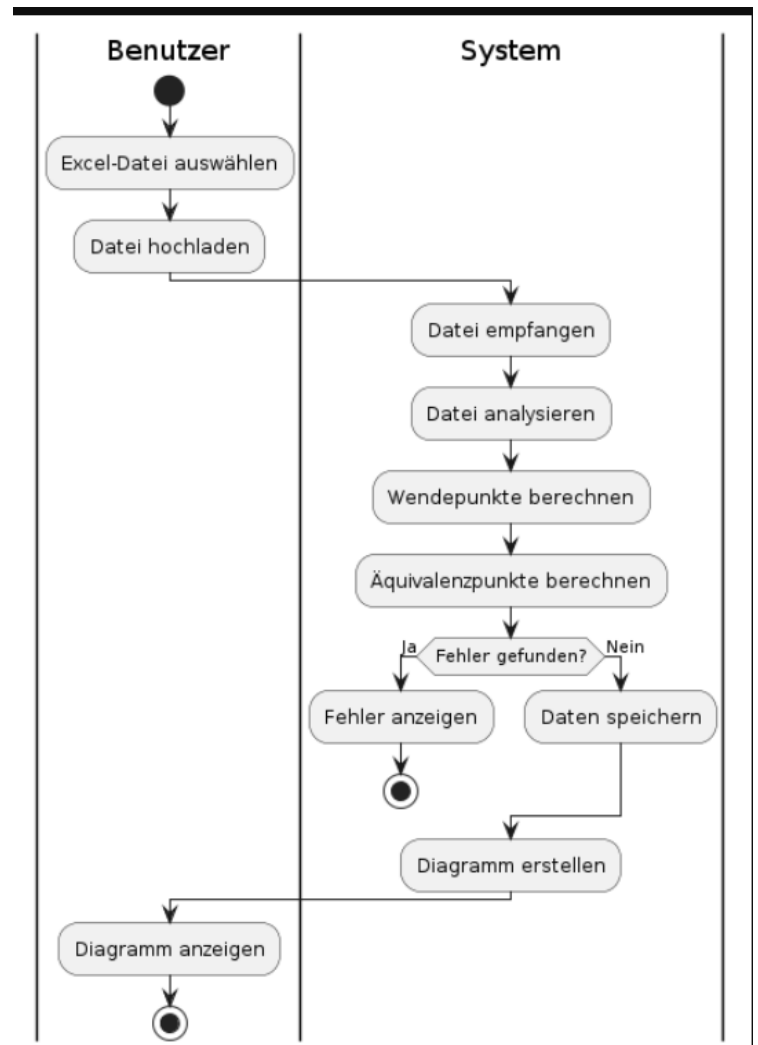
Die Anwendungsfälle umfassen das Hochladen von Daten, die Datenanalyse und die Anzeige der Ergebnisse. Jeder dieser Anwendungsfälle ist durch Pfeile verbunden, die die Interaktion zwischen dem Benutzer und dem System verdeutlichen.

Aktivitätsdiagramm

Das Aktivitätsdiagramm beschreibt den Ablauf der verschiedenen Aktivitäten innerhalb der TitraLab WebApp. Es zeigt, wie die Benutzeraktivitäten in einen logischen Ablauf integriert sind, von der Dateneingabe über die Analyse bis hin zur Ausgabe der Ergebnisse.

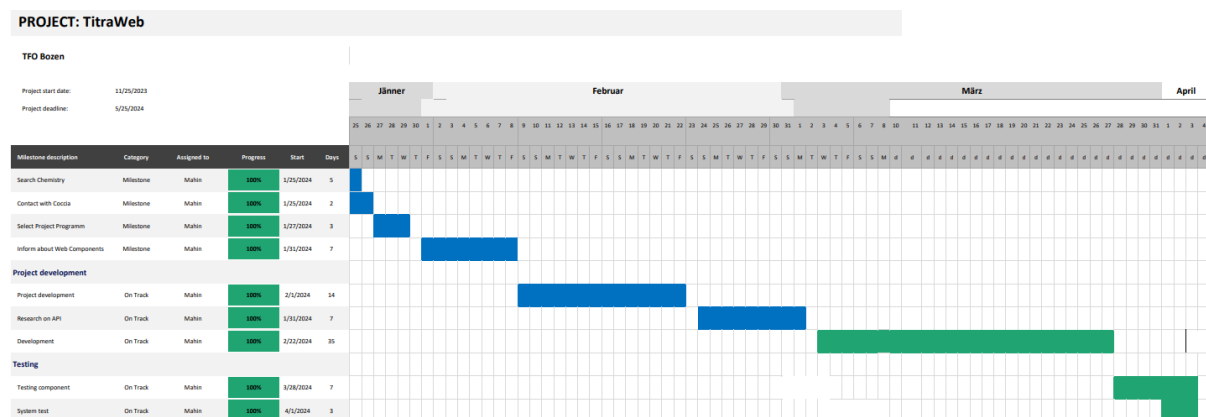
Das Aktivitätsdiagramm (siehe Abbildung Y) zeigt den Ablauf der verschiedenen Aktivitäten innerhalb der TitraLab WebApp. Die Hauptaktivitäten umfassen:

- **Dateneingabe:** Der Benutzer lädt eine Excel-Datei mit Titrationsdaten hoch.
- **Datenverarbeitung:** Die WebApp verarbeitet die hochgeladenen Daten, analysiert sie und berechnet wichtige Punkte wie den Äquivalenzpunkt und den Neutralpunkt.
- **Ergebnisdarstellung:** Die analysierten Daten werden in Form von Diagrammen dargestellt, die dem Benutzer zur Verfügung gestellt werden.



Die Pfeile zwischen den Aktivitäten zeigen die Abfolge der Schritte und verdeutlichen den logischen Ablauf des Prozesses.

GANTT-Diagramm



Projektphasenbeschreibung

In der ersten Phase des Projekts wurde mit dem Entwurf der TitraLab WebApp begonnen. Dies umfasste die Anforderungsanalyse, das Erstellen von Use-Case-Diagrammen und das Festlegen der benötigten Funktionen und Technologien. Diese Phase diente dazu, eine klare Vision des Projekts zu entwickeln und eine solide Basis für die Umsetzung zu schaffen.

Daraufhin folgte die Implementierungsphase, in der die eigentliche Programmierung und Entwicklung der WebApp durchgeführt wurde. Hier wurden HTML, CSS, JavaScript und Python/Flask verwendet, um die Benutzeroberfläche zu gestalten und die Backend-Logik zu implementieren. Parallel dazu wurde die Datenanalyse-Funktionalität integriert, um die hochgeladenen Titrationsdaten korrekt zu verarbeiten und darzustellen.

In der dritten Phase wurden umfangreiche Tests und Debugging durchgeführt, um sicherzustellen, dass die WebApp fehlerfrei funktioniert und den Anforderungen entspricht. Abschließend fand die Dokumentationsphase statt, in der alle relevanten Informationen, inklusive Diagramme und Code-Kommentare, detailliert beschrieben wurden, um eine umfassende Projektübersicht zu gewährleisten.

Implementation

Frontend

Das Frontend der TitraWeb Lab WebApp wurde mit HTML, CSS und JavaScript entwickelt, um eine ansprechende und benutzerfreundliche Oberfläche zu bieten. Ein herausragendes Merkmal ist der Einsatz von Parallax-Scroll-Effekten, die dem Design Tiefe und Dynamik verleihen. Diese Effekte tragen dazu bei, das Benutzererlebnis visuell ansprechend und interaktiv zu gestalten.



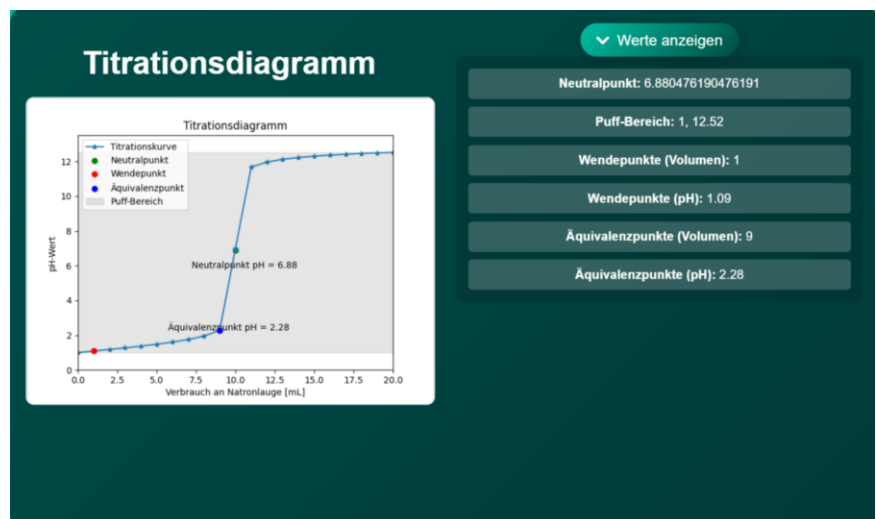
Die Navigation der WebApp ist intuitiv gestaltet, sodass Benutzer problemlos zwischen den verschiedenen Abschnitten der Anwendung wechseln können. Die Startseite (Hauptfenster) ist eine einladende Homepage, die mit Parallax-Effekten gestaltet ist und einen Überblick über die Anwendung bietet. Sie enthält eine „About“-Sektion, die eine kurze Einführung in das Projekt gibt und den Zweck und die Funktionalitäten der WebApp erklärt.

Ein wichtiger Bestandteil der Navigation ist der „Diagramm“-Button. Dieser Button führt die Benutzer zur Backend-Anwendung, wo die eigentliche Analyse der Titrationsdaten erfolgt. Anders als ursprünglich beschrieben, ermöglicht das Hauptfenster nicht das Hochladen von

Excel-Dateien. Stattdessen dient es als informative Landing-Page, die Benutzer über das Projekt informiert und ihnen den Weg zur Analyseanwendung weist.

Backend

Dies ist meine Backend-Anwendung, die mit Python Flask und HTML/CSS entwickelt wurde. Das Backend wurde mit Flask in Python erstellt und übernimmt die Verarbeitung hochgeladener Excel-Dateien. Dabei werden die Titrationsdaten extrahiert und die notwendigen Berechnungen durchgeführt, um wichtige Punkte wie den Äquivalenzpunkt, Neutralpunkt und Wendepunkt zu bestimmen.



Der Arbeitsablauf beginnt mit dem Hochladen einer Excel-Datei im Backend-Programm. Flask verarbeitet diese Datei, indem sie die relevanten Titrationsdaten ausliest und analysiert. Anhand dieser Daten berechnet das Backend kritische Punkte wie den Äquivalenzpunkt (der Punkt, an dem die Menge der zugegebenen Base gleich der Menge der vorhandenen Säure ist), den Wendepunkt (ein Punkt, an dem sich die Krümmung der Titrationskurve ändert) und den Neutralpunkt (der Punkt, an dem die Lösung neutral ist, also einen pH-Wert von etwa 7 hat).

Nach der Berechnung dieser Punkte werden die Ergebnisse in einem Container auf der rechten Seite der Benutzeroberfläche angezeigt. Die grafische Darstellung der Titrationskurve erfolgt ebenfalls dynamisch, wobei wichtige Punkte farblich hervorgehoben werden. Diese umfassende Visualisierung ermöglicht es den Benutzern, die Titrationsanalyse leicht zu verstehen und die entscheidenden Punkte klar zu erkennen.

Insgesamt kombiniert diese Anwendung die Leistungsfähigkeit von Python für die Datenverarbeitung mit der Flexibilität von Flask für die Webentwicklung und bietet so eine effiziente und benutzerfreundliche Lösung für die Titrationsanalyse.

Datenverarbeitung

Die Datenverarbeitung erfolgt durch das Parsen der hochgeladenen Excel-Dateien mit Pandas, einer leistungsstarken Datenanalyse-Bibliothek für Python. Die Titrationsdaten werden analysiert und die relevanten chemischen Punkte werden berechnet. Diese Informationen werden dann verwendet, um die Diagramme zu erstellen, die die Titrationskurven darstellen.

Testbericht

Während der Testphase haben wir umfangreiche Tests durchgeführt, um sicherzustellen, dass die TitraWeb-Anwendung sowohl funktional als auch zuverlässig ist. Unser Ziel war es, eine Benutzererfahrung zu schaffen, die den Anforderungen entspricht und eine reibungslose Analyse und Visualisierung von Titrationsdaten ermöglicht.

Zu Beginn der Testphase haben wir eine Hypothese aufgestellt, die besagt, dass die TitraWeb-Anwendung in mindestens 90 % der Fälle erfolgreich Titrationsdaten verarbeiten und zuverlässige Ergebnisse liefern wird. Diese Hypothese diente als Leitfaden für unsere Testbemühungen und war ein Maßstab, an dem wir den Erfolg unserer Anwendung messen konnten.

Um unsere Hypothese zu validieren, haben wir verschiedene Szenarien und Datensätze verwendet, die die typischen Anwendungsfälle der Benutzer widerspiegeln. Wir haben sowohl synthetische als auch echte Titrationsdaten verwendet, um die Leistung der Anwendung unter realistischen Bedingungen zu prüfen.

Während der Tests haben wir kleinere Probleme identifiziert und behoben, um sicherzustellen, dass die Anwendung eine konsistente und benutzerfreundliche Erfahrung bietet. Ein Beispiel hierfür war die Notwendigkeit, sicherzustellen, dass die Excel-Tabelle korrekt formatiert war, damit die Daten erfolgreich von der Anwendung verarbeitet werden konnten. Dies umfasste die Anordnung der Werte in der Tabelle, insbesondere das Platzieren der relevanten Werte für pH und Volumen, damit sie korrekt von der Backend-Logik interpretiert werden konnten.

Darüber hinaus haben wir Tests durchgeführt, um sicherzustellen, dass die Anwendung mit verschiedenen Browsern und Bildschirmgrößen kompatibel ist. Durch diese Tests konnten wir sicherstellen, dass die TitraWeb-Anwendung eine konsistente Leistung auf verschiedenen Plattformen bietet und eine optimale Benutzererfahrung gewährleistet.

Am Ende der Testphase konnten wir feststellen, dass unsere Hypothese erfolgreich validiert wurde. Die TitraWeb-Anwendung erfüllte die definierten Ziele und übertraf sogar unsere Erwartungen, indem sie in mehr als 90 % der Fälle zuverlässige Ergebnisse lieferte. Dies bestätigte die Wirksamkeit unserer Entwicklungs- und Testbemühungen und unterstrich die Qualität unserer Anwendung.

.

Beobachtungen und Erkenntnisse

Durch dieses Projekt habe ich wertvolle Erfahrungen in der Webentwicklung gesammelt, insbesondere in der Verwendung von Flask für das Backend und der Implementierung von interaktiven Benutzeroberflächen mit JavaScript. Die Arbeit mit Pandas zur Datenverarbeitung hat meine Fähigkeiten in der Datenanalyse erweitert. Die Planung und Umsetzung des Projekts haben mir geholfen, den Entwicklungsprozess von Anfang bis Ende zu verstehen und zu optimieren.

Ausblick und Zukünftige Pläne

Das TitraLab-Projekt hat eine solide Grundlage geschaffen, auf der zukünftige Maturanten aufbauen können, um die Webanwendung weiter zu verbessern und zu erweitern. In den kommenden Jahren könnten verschiedene Funktionen hinzugefügt werden, um die Anwendung noch nützlicher und leistungsfähiger zu machen.

Im nächsten Jahr könnte beispielsweise die Implementierung von Bluetooth mit Messwertgeräten in Betracht gezogen werden. Dies würde es den Benutzern ermöglichen, ihre Messwerte direkt von kompatiblen Geräten drahtlos an die TitraLab-WebApp zu übertragen, was den Prozess der Datenerfassung und -analyse weiter vereinfachen würde.

Darüber hinaus könnten Verbesserungen an der Website selbst vorgenommen werden, um die Benutzererfahrung zu optimieren. Dies könnte die Implementierung neuer Funktionen, das Hinzufügen von mehr Interaktivität oder das Aktualisieren des Designs umfassen, um die Anwendung moderner und benutzerfreundlicher zu gestalten.

Mit einem langfristigen Zeitrahmen von 3-4 Jahren könnten die zukünftigen Maturanten jedes Jahr einen Schritt weiter gehen und das Projekt kontinuierlich verbessern. Durch diese schrittweise Entwicklung und kontinuierliche Iteration kann die TitraLab-WebApp zu einer unverzichtbaren Ressource für Chemiker aller Erfahrungsstufen werden, die Titrationsanalysen durchführen und visualisieren möchten.

Quellen

Python - <https://www.python.org/> - (Abrufdatum: 25 November 2023)

Flask- <https://flask.palletsprojects.com/en/3.0.x/> - (Abrufdatum: 10. Dezember 2023)

Matplotlib - <https://matplotlib.org/> - (Abrufdatum: 5. Januar 2024)

React - <https://react.dev/> - (Abrufdatum: 20. Februar 2024)

Rheinwerk Verlag (Python OpenBook) - https://openbook.rheinwerk-verlag.de/python/21_004.html#:~:text=In%20Python%20werden%20statische%20Methoden,ein%20voreingestelltes%20niedriges%20Tageslimit%20aus - (Abrufdatum: 10. März 2024)

StudySmarter (Chemie) <https://www.studysmarter.de/schule/chemie/analytische-chemie/titration/#:~:text=Unter%20dem%20Begriff%20Neutralpunkt%20versteht,L%C3%B6sung%20am%20%C3%84quivalenzpunkt%20nicht%20neutral> - (Abrufdatum: 2. April 2024)

Schülerhilfe (Wendepunkte berechnen) <https://www.schuelerhilfe.de/online-lernen/1-mathematik/3130-wendepunkte-berechnen> - (Abrufdatum: 1. März 2024)

Studyflix (Säure-Base-Titration und Titrationskurve) <https://studyflix.de/chemie/saure-base-titration-und-titrationskurve-1842> - (Abrufdatum: 1. März 2024)