 **Notiere** in dieser Tabelle jeweils **zum Stundenanfang und -ende** die Planung, Durchführung, Abweichungen und aufgetretene Probleme.

| **Dat.** | **Grobplanung**[[1]](#footnote-1) | **Feinplanung**(Stundenbeginn):  „Am Ende der Sitzung (werde ich/soll)…“ | **auftretende Probleme  (und ihre Lösung)** | **Meine/unsere Erfolge** | **Was ist** (für nächstes Mal) **noch offen geblieben?** |
| --- | --- | --- | --- | --- | --- |
| **7.12.21-14.1.22** | *Schul - Navi* | Schul – Navi mithilfe AppInventor programmiert | - | - | - |
| **18.01.** | Neue Idee eines Vokabeltrainers | Ausarbeitung der Beschreibung vom Vokalbeltrainer | - | Idee gefunden und gut Ausgebaut | Wie können wir unsere Ideen umsetzen? Wer übernimmt welchen Teil des Projektes? |
| **21.01.** | Einteilung von Arbeit | Jeder bekommt einen Aufgabenbereich zugeteilt | Interessenkonflikt | Einteilung der Arbeit   * Meine Aufgabe: Programmierung + Implementierung der DB | Anfang mit dem eigenen Aufgabenbereich |
| **25.01.** | Einarbeitung in die Programmierung | Einarbeitung grobe Syntax von Python | - | Ich kenne „die Basics“ von Python | Anfang der Programmierung mithilfe von Tkinter |
| **28.01.** | - (Zeugnisse) | - (Zeugnisse) | - (Zeugnisse) | - (Zeugnisse) | - (Zeugnisse) |
| **08.02.** | Beginn Einarbeitung Tkinter | Wichtige Befehle von Tkinter lernen (Buttons, Text,…) |  | Erstes Fenster für den Vokabeltrainer mit den gröbsten Funktionen steht | „Merken“ Button hinzufügen für gezieltes Abfragen |
| **11.02.** | Ausbau des Fensters | „Merken“ Button hinzufügen + Vokabeln zufällig abfragen | „Merken“ funktioniert noch nicht | Vokabeln werden aus einer Liste (manuell erstellt) zufällig ausgewählt | „merken“ Funktion implementieren + erstellte DB implementieren |
| **15.02** | Implementierung | erstellte DB implementieren | Python selbst kann nicht mit SQL kommunizieren -> Bibliothek „sqllite3“ installiert | Geklärt, wie Python mit SQL DB kommunizieren kann | Erstellung eines Skripts, mit dem Python mit der DB kommunizieren kann |
| **18.02** | Erstellung eines Skripts, mit dem Python mit der DB kommunizieren kann | Erstellung „DB – Kommunikation“ | Probleme in der Syntax von sqllite3 | „DB – Kommunikation“ erstellt + funktioniert | Implementierung „DB – Kommunikation“ ins Hauptprogramm |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 22.02 | Implementierung „DB – Kommunikation“ ins Hauptprogramm | 2 Skripts verknüpfen | Wie verknüpft man mehrere Skripts in Python? -> per „import“ | 2 verknüpfte Skripts erstellt | Vokabelliste im Hauptprogramm implementieren |
| 25.02 | Vokabelliste im Hauptprogramm implementieren | Neues Fenster „Vokabelliste“ programmieren („wie ein Wörterbuch“) mit Suchfunktion | Wie programmiert man eine Suchfunktion in Tkinter? | Es kann nach lateinischen + deutschen Vokabeln gesucht werden | Verschönerung des Programms |
| 1.03 | Verschönerung des Programms Programm weniger fehleranfälliger machen | Verschönerung des Programms mithilfe von Klassen | Einarbeitung OOP und Klassen in Python | Pro Fenster 1 Skript mit 1 Klasse | Programm weniger fehleranfälliger machen + neue Sprachen |
| 4.03 | Neue Sprachen implementieren | Englisch, Französisch und Altgriechisch implementieren | Verschiedene „Übersetzungsrichtungen“ in z.B. Latein und Englisch -> Variablen „Übersetzungsrichtung“, „Ausgangssprache“, „Zielsprache“ erstellt + als Parameter jeder Funktion übergeben | Funktionierendes Programm mit verschiedenen Sprachen + Anleitung für das Programm geschrieben | Auswahlmenü der Sprachen programmieren |
| 8.3. | Auswahlmenü der Sprachen programmieren | Auswahlmenü braucht 4 Buttons (pro Sprache 1) | - | Auswahlmenü programmiert + funktioniert | Programm zum besseren Verständnis kommentieren |
| 11.3. – 1.4. | Programm kommentiert | - | - | Programm vollständig kommentiert | - |

1. [↑](#footnote-ref-1)