**Prozessdokumentation: Morse-Maschine mit Raspberry Pi**

**Projektname:** Morse-Maschine mit Raspberry Pi  
**Fach:** Informatik  
**Projektzeitraum:** 25.10.2024 bis 06.01.2025

**1. Ausgangslage und Zielsetzung**

In unserem Informatikunterricht haben wir den Auftrag erhalten, eine Morse-Maschine mit einem Raspberry Pi zu entwickeln. Ziel war es, eine Anwendung zu programmieren, die Morsezeichen eingibt und entschlüsselt. Die Ausgabe sollte auf einem LCD-Display erscheinen und zusätzlich per E-Mail versendet werden können. Das Projekt bot uns die Möglichkeit, unsere Programmierfähigkeiten zu vertiefen und ein praktisches Anwendungsszenario umzusetzen.

**2. Zeitplan und Arbeitsprozess**

**2.1 Planung (25.10.2024 - 31.10.2024)**

* Zieldefinition: Festlegung der Funktionalitäten der Morse-Maschine.
* Aufteilung der Aufgaben: Zuständigkeiten für Codierung und Dokumentation wurden verteilt.
* Konzeptentwicklung: Grobes Design der Software-Architektur und Überlegungen zur Morsecode-Erkennung.

**2.2 Codierung (01.11.2024 - 20.12.2024)**

**01.11.2024 - 07.11.2024:**

* Einrichtung der Entwicklungsumgebung auf dem Raspberry Pi.
* Erstellung der Grundstruktur des Python-Programms:
  + Definition der Hardware-Schnittstellen (Taster, LED, Buzzer).
  + Initialisierung der GPIO-Pins und Festlegung der Morsecode-Timings (Punkt, Strich, Pausen).

**08.11.2024 - 14.11.2024:**

* Implementierung der Funktion zur Morsecode-Erkennung:
  + Erfassen von Tastendrücken und deren Dauer.
  + Zuordnung der Dauer zu Morsezeichen (Punkt oder Strich).
  + Speicherung der erfassten Zeichen in einer Variablen.

**15.11.2024 - 21.11.2024:**

* Entwicklung der Übersetzungsfunktion:
  + Erstellung des Morsecode-zu-Text-Dictionaries.
  + Implementierung einer Logik zur Erkennung von Buchstaben- und Wortpausen.
  + Test der Übersetzung mit verschiedenen Morsezeichen-Kombinationen.

**22.11.2024 - 28.11.2024:**

* Implementierung der LCD-Ausgabe:
  + Ausgabe der entschlüsselten Nachrichten auf dem LCD-Display.
  + Integration einer Farbsteuerung für visuelles Feedback (z. B. grün bei erfolgreicher Entschlüsselung).

**29.11.2024 - 05.12.2024:**

* E-Mail-Benachrichtigungsfunktion:
  + Entwicklung einer Funktion zum Versand der entschlüsselten Nachrichten per E-Mail.
  + Verwendung einer externen YAML-Datei zur Speicherung der Zugangsdaten.
  + Sicherstellung der Verschlüsselung und Sicherheit beim SMTP-Server-Zugriff.

**06.12.2024 - 12.12.2024:**

* Integration und Optimierung:
  + Zusammenführung der Module für Morsecode-Erkennung, Textübersetzung, LCD-Ausgabe und E-Mail-Versand.
  + Überarbeitung der Benutzerführung und Verbesserung der Code-Kommentare.

**13.12.2024 - 20.12.2024:**

* Fehlerbehebung und Feintuning:
  + Anpassung der Morsecode-Timings zur Erhöhung der Erkennungsgenauigkeit.
  + Überprüfung aller Funktionen auf Zuverlässigkeit und Stabilität.

**2.3 Abschlussphase (06.01.2025)**

* Vervollständigung der Prozessdokumentation.
* Vorbereitung der Präsentation des Projekts.

**3. Herausforderungen und Lösungen**

* **Fehler bei der Morsecode-Erkennung:** Einige Punkt- und Strichlängen wurden anfangs falsch erkannt. Dies wurde durch eine genaue Anpassung der Zeitwerte gelöst.
* **E-Mail-Versandprobleme:** Schwierigkeiten bei der Konfiguration des SMTP-Servers wurden durch die Nutzung einer externen YAML-Konfigurationsdatei behoben.
* **Fehlerhafte LCD-Ausgabe:** Probleme mit der Anzeige auf dem LCD-Display traten auf, wenn Sonderzeichen in den Morsezeichen enthalten waren. Der Code wurde angepasst, um nur gültige Zeichen zu verarbeiten.

**4. Fazit**

Das Projekt war ein voller Erfolg. Die Morse-Maschine erfüllt alle geplanten Anforderungen: Sie kann Morsezeichen über einen Button erfassen, diese entschlüsseln und die Nachricht auf einem LCD-Display anzeigen sowie per E-Mail versenden. Durch dieses Projekt konnten wir nicht nur unsere Programmierkenntnisse, sondern auch unsere Problemlösungsfähigkeiten erheblich verbessern.

Die strukturierte Planung und Zusammenarbeit im Team waren entscheidend für den erfolgreichen Abschluss des Projekts.