**Dokumentation - NeoNotes**

**1. Einleitung:**

NeoNotes wird im Rahmen eines Schulprojektes im Fach Informatik Systeme entwickelt und soll bis zum 13.03.2023 fertig gestellt werden. Es wird entwickelt um den Schulalltag, für Schüler, die abseits von Word und Co Digital arbeiten, zu erleichtern und besser mit ihren Mitschriften lernen zu können.

**2. Anforderungsprofil:**

**2.1 Ist-Situation:**

Zurzeit schreibe ich meine Mitschriften in einem Simplen Texteditor dabei fällt mir immer wieder auf das es schwer und zeitaufwendig sein kann diese zu organisieren und zu verwalten.

**2.2 Soll-Situation:**

NeoNotes soll ein Programm werden womit Schüler ihre Mitschriften niederschreiben können. Diesen Mitschriften sollte man „Tags“ hinzufügen können, um diese Kategorisieren und organisieren zu können.

Notizen sollten in einer Datenbank , welche ohne Server fungiert, gespeichert werden damit auf diese schnell zugegriffen werden kann und diese auch zuverlässig gespeichert werden.

Das Design der Benutzeroberfläche sollte simpel gehalten werden um mögliche Ablenkungen während des Unterrichts zu vermeiden.

**2.3 Systemanforderungen:**

Um NeoNotes optimal zu nutzen sollte man ein 64-Bit basiertes Betriebssystem zur Verfügung haben. CPU und RAM dürften dabei keine Rolle spielen.

**2.6 Abnahmekriterien:**

Vollständige Lauffähigkeit: Das Programm sollte ohne Einschränkungen und Fehler ausgeführt werden können. Alle Funktionen und Features des Programms müssen ordnungsgemäß funktionieren und es sollte möglich sein, neue Notizen zu erstellen und zu bearbeiten sowie diese zu organisieren, kategorisieren und zu teilen.

Fehlerfreiheit: Das Programm sollte frei von Fehlern. Es sollten keine unerwarteten Abstürze oder Fehlermeldungen auftreten, wenn das Programm verwendet wird.

Test-Datenbestand: Das Programm sollte mit einem bereits angefertigten Test-Datenbestand versehen sein. Dieser Datenbestand sollte realitätsnah sein und verschiedene Szenarien und Anwendungsfälle abdecken, um sicherzustellen, dass das Programm in verschiedenen Situationen und Bedingungen funktioniert.

**2.7 Benutze Ressourcen:**

NeoNotes wurde von mir allein entwickelt und hatte keine finanziellen Mittel zur Verfügung, da diese zur Entwicklung nicht nötig waren. Stattdessen habe ich meine vorhandenen Kenntnisse und Fähigkeiten im Bereich der Softwareentwicklung genutzt. Ich habe Pascal und SQLite ausgewählt, da diese mir vertraut sind und ich bereits Erfahrung damit habe.

Als Entwickler habe ich die Pascal IDE Lazarus gewählt um die Oberfläche zu entwerfen und zu implementieren. Zur Entwicklung der Funktionen habe ich jedoch Neovim als PDE und den JEDI Code formatierer der IDE Lazarus genutzt. Um den Stand der Entwicklung zu dokumentieren habe ich Git und den Host Anbieter Github genutzt.

Obwohl ich alleine an dem Projekt gearbeitet habe, hatte ich auch Unterstützung von Freunden, um Fragen zu stellen oder Feedback zu erhalten.

**3. Produktbegleitende Dokumentation:**

Ich habe das Projekt Neonotes nach dem Vier-Phasen-Modell entwickelt, um sicherzustellen, dass das Programm alle Anforderungen erfüllt und fehlerfrei läuft.

In der Analysephase habe ich zunächst die Anforderungen an das Programm ermittelt und dokumentiert. Hierbei habe ich mir Gedanken darüber gemacht, welche Funktionen das Programm haben sollte und welche Datenstrukturen und Algorithmen dafür notwendig wären.

In der Entwurfsphase habe ich dann das Konzept und die Architektur des Programms entwickelt. Hierbei habe ich die Funktionen und Datenstrukturen aus der Analysephase in ein logisches Gesamtkonzept integriert. Dazu gehörte auch die Festlegung der Schnittstellen und der Datenbankstruktur, die das Programm benötigt. Parallel dazu habe ich eine Oberflächenplanung durchgeführt und ein Design erstellt, das dem gewünschten aussehen des Programms entsprach. Hierbei habe ich mich an eine Simple Oberfläche gehalten, um diese benutzerfreundlich wie möglich zu gestalten. Zudem habe ich eine Zeitplanung erstellt, die die verschiedenen Schritte des Projekts umfasste und die Dauer sowie die Reihenfolge der einzelnen Aufgaben festlegte.

In der Implementierungsphase habe ich schließlich den Programmcode auf Basis des Prototypings geschrieben. Ich habe den Code modular aufgebaut und dabei auch auf eine sorgfältige Dokumentation und Kommentierung geachtet, um die Wartbarkeit und Erweiterbarkeit des Programms zu gewährleisten. Im ersten Prototype habe ich zunächst alle Grund Funktionen entwickelt, wie das erstellen von Notizen und deren Sammlungen. Ich habe diese Version genutzt, um fehlende Funktionen zu finden und diese auszuarbeiten.

Auf Basis der ersten Version habe ich dann weitere Funktionen hinzugefügt und das Programm verbessert. In jeder Iteration habe ich das Programm getestet und evaluiert, um sicherzustellen, dass es meine Anforderungen erfüllt.

Nach mehreren Iterationen hatte ich schließlich ein funktionsfähiges Programm erstellt, das alle Funktionen enthielt, die ich benötigte. Ich habe das Programm dann ausgiebig getestet, um sicherzustellen, dass es vollständig lauffähig und fehlerfrei ist.

Zuletzt habe ich das Programm In der Testphase intensiv getestet, um Fehler und Probleme aufzudecken und zu beheben. Hierbei habe ich manuelle Tests durchgeführt und alle Funktionalitäten des Programms ausgiebig überprüft. Dies habe ich mit einem bereits angefertigten Test-Datenbestand durchgeführt, um sicherzustellen, dass es in der Praxis einwandfrei funktioniert.

**4. Programm:**

Im Anschluss wird die Installation und Nutzung des Programms erklärt.

**4.1 Installation:**

Um NeoNotes nutzen zu können, muss man es auf seinem System kompilieren. Dazu benötigt man Lazarus und dessen LCL Package (Standardmäßig in Lazarus enthalten). Ebenfalls benötigt man die SQLite Bibliothek auf dem System

Als erstes sollte man die Pascal IDE Lazarus und die SQLite Bibliothek installieren. Anschließend muss man die main.lpr datei im src Verzeichniss mit Lazarus öffnen und den Source Code Kompilieren.

Wenn alles erfolgreich ablief sollte man ein voll Funktionsfähiges Programm haben.

**4.2 Nutzung:**