### Anforderungsanalyse für den "Simple Graph Calculator and Plotter"

#### 1. Hintergrund & Motivation

* **Ziel:** Entwickeln einer Windows-Anwendung zur Eingabe und Visualisierung von trigonometrischen Funktionen (sin und cos) sowie der Sinc-Funktion.
* **Nutzen:** Benutzer sollen die Auswirkungen von Parameteränderungen an den Funktionen leicht erkennen und visuell nachvollziehen können.

#### 2. Funktionale Anforderungen

**2.1. Benutzeroberfläche**

* **Eingabebereich:**
  + Textfelder oder Schieberegler zur Eingabe der Parameter für die Funktionen.
  + Dropdown-Menü oder Auswahlfeld zur Auswahl des Funktionstyps (sin, cos, sinc).
  + Eingabefelder zur Anpassung des Darstellungsbereichs der Funktionen.
* **Anzeigebereich:**
  + 2D-Plotbereich zur Darstellung der ausgewählten Funktion.
  + Anzeige der letzten verwendeten Parameter nach dem Start der Anwendung.
* **Speicheroptionen:**
  + Möglichkeit, die Visualisierung im Vektorformat (z.B. SVG, PDF) zu speichern.

**2.2. Funktionalität**

* **Funktionstypen:** Implementierung von sin, cos und sinc Funktionen.
* **Parameteranpassung:** Dynamische Anpassung und Echtzeit-Aktualisierung des Plots bei Änderung der Parameter.
* **Darstellungsbereich:** Anpassbare Achsenbereiche für die Funktionen.
* **Speichern:** Möglichkeit, die generierte Grafik als Vektordatei zu speichern.
* **Letzte Parameter:** Speicherung und Wiederherstellung der letzten verwendeten Parameter beim Öffnen der Anwendung.

#### 3. Nicht-funktionale Anforderungen

**3.1. Benutzerfreundlichkeit**

* Intuitive und leicht verständliche Benutzeroberfläche.
* Echtzeit-Feedback bei der Anpassung von Parametern.

**3.2. Leistung**

* Schnelle Berechnung und Darstellung der Funktionen.
* Effizientes Speichern der Vektorgrafiken.

**3.3. Zuverlässigkeit**

* Sicherstellen, dass die Anwendung stabil läuft und korrekt speichert.
* Speicherung der Parameter auch bei unerwartetem Schließen der Anwendung.

**3.4. Wartbarkeit**

* Klare und dokumentierte Code-Struktur.
* Einfache Erweiterung für zukünftige Funktionen.

#### 4. Technische Anforderungen

**4.1. Entwicklungsumgebung**

* Windows 11
* Programmiersprache: C#
* Framework: WPF (Windows Presentation Foundation)

**4.2. Bibliotheken**

* Verwendung von graphischen Bibliotheken zur Darstellung und Speicherung der Vektorgrafiken (z.B. OxyPlot, LiveCharts).

**4.3. Datenspeicherung**

* Lokale Speicherung der letzten Parameter (z.B. in einer Konfigurationsdatei).

#### 5. Akzeptanzkriterien

* Die Anwendung öffnet sich mit den zuletzt verwendeten Parametern.
* Benutzer können zwischen sin, cos und sinc Funktionen wählen und deren Parameter anpassen.
* Der Plot aktualisiert sich in Echtzeit bei Parameteränderungen.
* Der Darstellungsbereich ist anpassbar.
* Die Visualisierung kann als Vektorgrafik gespeichert werden.

### Relevante Teilaufgaben

1. **Projektplanung und -management**
   * Projektziel und -umfang definieren
   * Zeitplan und Meilensteine festlegen
   * Ressourcenplanung (z.B. Teammitglieder, Werkzeuge)
2. **Anforderungsdefinition**
   * Sammlung und Dokumentation der Anforderungen
   * Priorisierung der Anforderungen
   * Anforderungserhebung durch Interviews, Workshops oder Umfragen
3. **Analyse der Funktionalen Anforderungen**
   * Identifikation der Kernfunktionen (Eingabeparameter, Auswahl der Funktion, Plotten)
   * Spezifikation der Benutzeroberfläche
   * Bestimmung der Interaktionsmöglichkeiten (z.B. Echtzeit-Update der Grafik)
4. **Analyse der Nicht-funktionalen Anforderungen**
   * Usability (Benutzerfreundlichkeit)
   * Performance (Reaktionszeiten, Effizienz der Berechnungen)
   * Zuverlässigkeit (Stabilität der Anwendung)
   * Wartbarkeit (Code-Qualität, Dokumentation)
5. **Technische Analyse**
   * Auswahl der Entwicklungsumgebung und Programmiersprache
   * Bestimmung der benötigten Bibliotheken und Frameworks
   * Analyse der Anforderungen an die Datenhaltung und Speicherung
6. **Prototyping und Benutzerfeedback**
   * Erstellung von Mockups oder Prototypen der Benutzeroberfläche
   * Einholen von Benutzerfeedback und Anpassung der Anforderungen
7. **Risikomanagement**
   * Identifikation von potenziellen Risiken (z.B. technische Herausforderungen, Zeitverzug)
   * Entwicklung von Risikominimierungsstrategien
8. **Erstellung eines Pflichtenhefts**
   * Detaillierte Beschreibung der Anforderungen und Spezifikationen
   * Abstimmung und Freigabe durch alle relevanten Stakeholder

### Weitere relevante Aspekte einer Anforderungsanalyse

1. **Stakeholder-Analyse**
   * Identifikation aller relevanten Stakeholder (z.B. Endbenutzer, Projektmanager, Entwickler)
   * Verständnis der Bedürfnisse und Erwartungen der Stakeholder
2. **Zielgruppenanalyse**
   * Definition der Zielgruppe der Anwendung
   * Analyse der Benutzeranforderungen und -fähigkeiten
3. **Wettbewerbsanalyse**
   * Untersuchung ähnlicher bestehender Anwendungen
   * Bestimmung der Alleinstellungsmerkmale der eigenen Anwendung
4. **Kriterien für den Erfolg**
   * Definition messbarer Erfolgsfaktoren (z.B. Benutzerzufriedenheit, Funktionsumfang)
   * Festlegung von KPIs (Key Performance Indicators)
5. **Anforderungskonsistenz und -vollständigkeit**
   * Überprüfung, ob alle Anforderungen konsistent und vollständig sind
   * Sicherstellung, dass keine wesentlichen Anforderungen fehlen
6. **Kommunikation und Dokumentation**
   * Regelmäßige Kommunikation mit allen Stakeholdern
   * Laufende Dokumentation des Anforderungsprozesses und der Ergebnisse

Diese Teilaufgaben und Aspekte stellen sicher, dass die Anforderungsanalyse gründlich und umfassend durchgeführt wird, um eine erfolgreiche Entwicklung des "Simple Graph Calculator and Plotter" zu gewährleisten.

*OK, das erschlägt mich erst mal und ich weiß wieder mehr, was ich alles nicht weiß.*