**Projektdokumentation**

1. **Projektziel:**  **Entwicklung eines Lead-Magneten für die KI-gestützte Strategiefindung**

*Das Ziel dieses Projekts ist die Erstellung einer interaktiven Webseite, die es ermöglicht, grundlegende Fragen zum Thema Social-Media-Strategie von Nutzern zu sammeln, um eine maßgeschneiderte Strategie zu Entwickeln. Durch die gezielte Befragung der Nutzer sollen deren Bedürfnisse und Ziele präzise erfasst und der KI übermittelt werden. Diese soll Bedarfsgerecht eine dem Thema und Auswahlmöglichkeiten bedingte Antwort dem Nutzer übermitteln.*

1. **Lösungsansatz**

**2.1 Frageseite gestalten:**

* *Wir planen eine benutzerfreundliche Umfrage, die die Besucher Schritt für Schritt durch die Fragen führt.*
* *Die Umfrage wird 20 spezifische Fragen beinhalten, die sich auf verschiedene Themen wie Unternehmensziele, Zielgruppen und Ressourcen konzentrieren.*

**2.2 Fragen und Antworten:**

* *Jede Frage wird klare Antwortmöglichkeiten bieten, und wir werden sowohl einfache als auch mehrere Auswahloptionen anbieten.*
* *Diese Struktur hilft uns, ein umfassendes Bild von den Bedürfnissen der Nutzer zu bekommen.*

**2.3 Autonome Strategiedesign durch KI:**

* *Die gesammelten Antworten werden direkt von der KI, in diesem Fall ChatGPT, analysiert. Basierend auf den Informationen wird die KI eigenständig personalisierte Strategien entwickeln, die auf die spezifischen Bedürfnisse und Ziele der Nutzer abgestimmt sind.*
* *Dadurch können wir schnell und effizient relevante Vorschläge generieren, die den Nutzern helfen, ihre Social-Media-Präsenz zu optimieren.*

**2.4 Benutzerfreundlichkeit im Fokus:**

* *Durch die klare und strukturierte Gestaltung der Umfrage wollen wir sicherstellen, dass die Nutzer schnell und einfach ihre Antworten eingeben können.*

1. **Teilaufgaben**

* **Meilensteine**
* **Konzeptionierung und Planung des Projekts ✔️**
* **Aufbau des Grundgerüsts der Seiten ✔️**
* **Implementierung der Fragen/ Verlinkungen der Seiten ✔️**
* **Funktionserstellungen für die Abfragen und Weiterleitungen ✔️**
* **XML-Integration und Aufbau ✔️**
* **Speicherung der Dateien ✔️**
* **Integration des API-Keys für die Schnittstelle zur KI ✔️**
* **KI-Einstellung und Ausgabe ✔️**
* **Debugging und Testphase**

**Grobplanung:**

* *Anfertigung von drei Html Seiten*
  + *Index Startseite*
  + *Fragenseite*
  + *Endseite*
* *Gerüstaufbau*

**Verbindung:**

* *Verbinden und Weiterleitung zwischen den Seiten*
  + *Mittels PHP-Codierung*

**Implementierungen:**

* *Implementierung einer xml.php zur Vorbereitung der Datenkopie und Weitergabe*
* *Zum Kopieren und Speichern des Datentyps*

**Datenschnittstelle zur KI herstellen:**

* *Xml.php mittels API-Key zu ChatGPT weiterleiten*
* *Verarbeitung durch Framing im KI-Bot vorbereiten*
* *Übergabe der ausgeworfenen Ergebnisse an die Endseite weiterleiten zur Ergebnissichtung für den Nutzer*

**Strukturierte Teilaufgaben:**

**Erstellung der index.html Seite**

* *Gerüstaufbau der Html Seite*
* *Definierung der Container-Class zur optisch verbesserten Sichtbarkeit und Lesbarkeit der Seite*
* *Erstellung eines kurzen Einführungstextes*
* *Footer Erstellung mit Copy tags*
* *Erstellung der styles\_1.css Datei*
* *Entwicklung eines CSS-Layouts für die Startseite zur optischen Verbesserung*
* *Verlinken des CSS-Layouts durch href=“css/styles\_1.css“*
* *Optimieren des Layouts durch probieren*
* *Vorbereiten des Weiterleiten-Buttons an die Frageseite*
* *Entwicklung des Labels mit Jetzt Starten*
* *Vorbereiten der Verlinkung mittels onclick=“window.location.href=‘….‘“*
* *Vorbehalt des Einfügens des Links*

**Erstellung der frageseite.html Seite**

* *Zunächst Erstellung des Html Gerüsts*
* *Sammlung der Fragen und Einbindung als Label mit Radio & Checkbox Funktion via html Integrierung zur Auswahl der antworten*
* *Erstellung eines Vor- und Zurück-Buttons*
* *Erstellung eines CSS als Layout damit die Fragen vernünftig eingebettet sind*
* *Erstellung einer Java-Skript Funktion damit die Fragen nacheinander angezeigt werden und durch Klicken der Buttons „Weiter“ und Zurück“ durchgespielt werden können*

**Erstellung der Endseite erfolgsseite.html**

* *Gerüstaufbau der Html Struktur*
* *Container Klasse integrieren für die „Dankes-Abschluss-Box“*
* *Integrierung eines „Go Back“- Buttons um zur Startseite zurück zu gelangen (via href- Funktion)*
* *CSS-Gestaltung der Seite*

**Verlinkungen der Seiten**

* *Verlinkung der Seiten via href= Funktion und dem Seitenlink der Zielseiten*
* *Integrierung eines Absende Buttons auf der frageseite.html damit nicht wieder auf die 1.Frage zurückgeworfen wird*
* *Java-Skript Einstellung und CSS-Anpassung damit flüssig weitergeleitet werden kann*
* *Counter für die Fragen eingebaut via JS damit die Frage 1 Nur einen „Weiter“- Button hat und keinen „Zurück“-Button*

**Umbau der Frageseite.html**

* *Die Frageseite.html muss auf php umgeschrieben werden und die html Elemente ebenso auf php umgeschrieben werden, damit die Fragen und Antworten mittels Post Funktion dynamisch übermittelt werden können*
* *Erstellung eines Fragen- und Antworten- Array über php*
* *Erstellung eines Mehrdimensionalen Arrays*
* *ID, Text, type & Options*
* *Einfachere Zuordnung*
* *Dynamisch*
* *Besseres abrufen*
* *Erstellung einer Fragen- und Antworten- Schleife, damit dynamisch das Array abgefragt werden kann*
* *Als Vorbereitung auf das XML*
* *Anpassung des Java-Scripts an die Veränderung durch Festlegung der Variablen zur besseren Übermittlung der einzelnen Strings*

**Zwischenänderungen**

* *Anpassung der href-Links an die Veränderung das aus der Frageseiten.html eine frageseite.php wurde*

**Erstellung einer xml.php**

* *Übernahme des Arrays von der frageseite.php in die xml.php für leichteres abfragen in der nachfolgenden Funktion*
* *Erstellung einer XML doc Funktion via ‚new SimpleXMLElement‘*
* *Erstellung einer Schleife mit php die das Fragen-Array scannt*
* *Erstellung einer php Funktion, die die Options hinzufügt (Antwortmöglichkeiten)*
* *Dadurch wird das XML-Dokument aufgebaut*
* *Zum Abschluss wird mit $\_POST die Antwort der jeweiligen Frage hinzugefügt*
* *Dies läuft ebenfalls durch den Scan, der das Array nach Abgabe der Antwort durchläuft*
* *Somit werden zwei Scan-Schleifen durchlaufen um daraus das XML-Dokument zu erstellen*
* *Unterschieden wird beim zweiten Scan noch durch eine Funktion, die zwischen Mehrfachauswahl und Einfachauswahl unterscheidet*
* Erstellung einer Funktion die jeden Schritt speichert während man die Fragen durchwählt und auch abändert beim Zurückgehen in den Fragen
* Speicherortablage wurde erstellt
* Zeitstempel und UniqID sichern die Daten ab
* Kontrollversuche funktionieren

Vorbereitung für die Eingliederung des API-Keys

* Erstellung eines Komplettstrings für die bessere Aufnahme des Antworten-Strings aus dem XML-Format an die KI

Schnittstelleneinrichtung

* Abwicklungsseite eingerichtet worüber die Rückgabe des Prompts ausgelesen wird (Ausgabe)
* Eingliederung des API-Keys via Request Method und Post Übergabe
* Einstellung der Parameter samt Übergabe via „$ki\_input\_string“
* Curl Anfrage an OpenAI mit Bearer Authorization
* Antwort Rückgabe via json\_decode und json\_encode

Funktional ist damit die KI fertig, der Rest sind Quality Verbesserungen, Debugging und Testphase (Dauertest mit Großer Fehleranalyse).

Erste Tests zeigen aktuell keine Fehler an, jedoch muss das Framing vorgenommen werden um passendere antworten zu enthalten. Anschließend kann die Feinabstimmung und das Debugging beginnen.

**-Bis hierhin bereits erbracht- Stand 21.11.24**

**-Ab hier aktuell:-**

**Überarbeitungsphase**

* *Überarbeitung des CSS und des Gesamtstylings der Seiten und Unterseiten*
* *Button der Endseite zum Abschicken der Anfrage über die API mit dem Button Absenden der Frageseite verknüpfen um doppeltes Klicken/Absenden zu vermeiden*
* *Promptfenster erstellen für die Ausgabe für die bessere Sichtbarkeit*

**Nächste Arbeitsschritte:**

* *Framing der KI*
* *Sicherheitslücken überprüfen und debuggen (ggf. Style anpassen)*
* *Evtl. Sicherheitssysteme einbauen*
* *Testphase/ Kontrollphase*
* *Abgabe*