|  |  |  |
| --- | --- | --- |
| Prüfungsphase: Sommer 2024 |  | Name, Vorname: Gregory Maleterre Ausbildungsberuf: Fachinformatiker für Anwendungsentwicklung   Prüflingsnummer: Id.Nr.: |

|  |
| --- |
| Titel  der Dokumentation |

|  |  |  |
| --- | --- | --- |
| Projektgeber: |  | Ausbildungsbetrieb: BFW Berlin-Brandenburg e. V. Kastanienallee 25 16567 Mühlenbeck  Ansprechpartner: Name Telefon: +4933056 862xxx  Mail: [xxx@bfw-berlin-brandenburg.de](mailto:xxx@bfw-berlin-brandenburg.de) |

|  |  |  |
| --- | --- | --- |
| Eidesstattliche Erklärung:  Hiermit erkläre ich, Gregory Maleterre, die hier vorliegende Arbeit (Dokumentation) selbstständig und nur unter Zuhilfenahme der aufgeführten Quellen angefertigt zu haben. Der aufgeführte zeitliche Rahmen wurde eingehalten. | | |
| Ort/Datum |  | Unterschrift |
|  |  |  |

IHK Bewilligter Antrag

Zeitmitschreibung der Projektarbeit

Prüfungsteilnehmer

Name: Gregory

Vorname: Maleterre

Login: (Login des Online Portals)

|  |  |  |
| --- | --- | --- |
| Datum | Tätigkeit | Zeit in Stunden |
|  | Planungsphase |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  | Summe Planungsphase |  |
|  |  |  |
|  | Durchführungsphase |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  | Summe Durchführungsphase |  |
|  |  |  |
|  | Qualitätsmanagement |  |
|  |  |  |
|  |  |  |
|  | Summe Qualitätsmanagement |  |
|  |  |  |
|  | Übergabe-, Auswertungsphase |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Inhaltsverzeichnis:

1. Projektdokumentationshinweise Verzeichnisse 1

1.1. Verwendungsvermerk 1

1.2. Dokumentation zur Projektarbeit 1

1.3. Abbildungsverzeichnis 1

1.4. Tabellenverzeichnis 1

1.5. Abkürzungen 1

1.6. Quellenverzeichnis 1

2. Planungsphase 1

2.1. Ausgangssituation 1

2.1.1. Projektumfeld 1

2.1.2. IST-Zustand 2

2.1.3. SOLL-Zustand 2

2.1.4. Projektziele 3

2.1.5. Personellen Schnittstellen 4

2.1.6. Technische Schnittstellen 4

2.1.7. Projektabgrenzung 4

2.1.8. Projektbegründung 4

2.1.9. Projektrisiken 4

2.2. Projektablauf 4

2.2.1. Technische Planung des Projektzieles 4

2.2.2. Analyse des IST-Zustandes 4

2.2.3. Erstellung des SOLL-Konzeptes 4

3. Ressourcen- und Ablaufplan 6

3.1. Personalplanung 6

3.2. Sachmittelplanung 6

3.3. Zeitliche Planung 6

3.4. Kostenplanung 6

4. Durchführungsphase 6

4.1. Eventuelle Beschaffung gemäß Planung 6

4.2. Herstellen der Arbeitsumgebung 6

4.3. Eigentlich Umsetzung 6

4.4. Testen 6

4.5. Fehler, Fehlerbehebung und Anpassung 6

5. Übergabe und Auswertung 6

5.1. Projektübergabe 6

5.2. Mitarbeiterschulung, Bedienungsanleitung 7

5.3. SOLL-IST-Vergleich (inhaltlich, zeitlich, kostentechnisch) 7

5.4. Technische Auswertung und Ausblick 7

5.5. Persönliche Auswertung 7

Deckblatt

VORWORT

**gehen Sie in Ihrer Dokumentation bitte genauer auf den Punkt "Datenextraktion von den Webseiten der Lieferanten" ein. Warum wurde sich für diesen Weg entschieden und warum nicht für eine definiertere Kommunikation über z.B. Schnittstellen?**

***Wird noch ergaenzt***

*Es gibt einen klaren Unterschied zwischen den Webseiten der Hersteller und denen der Lieferanten. Während es bei den Herstellern oft die Möglichkeit einer definierten Kommunikation über Schnittstellen gibt, ist dies bei den Lieferanten nicht der Fall. Diejenigen, die diese Option anbieten, stellen in der Regel nur begrenzte Informationen zur Verfügung, die eher für ein Fachpublikum gedacht sind.  
Das SEO-Team ist möglicherweise nicht mit dem Spezialgebiet der Putzmittel vertraut und benötigt daher umfassendere Informationen, die es auch für ein breiteres Publikum verständlich aufbereiten kann. In Anbetracht dieser Anforderungen erwies sich die Extraktion von Informationen und Daten mit Python als die bessere Lösung.  
Die Verwendung von Python ermöglichte eine flexible und anpassungsfähige Extraktion direkt von den Webseiten der Lieferanten. Auf diese Weise konnten wir sicherstellen, dass das SEO-Team über die notwendigen Informationen verfügte, um die Produktbeschreibungen optimal zu gestalten und gleichzeitig die Anforderungen des Kunden zu erfüllen. Diese Herangehensweise bot uns die Möglichkeit, eine umfassende Datengrundlage zu schaffen, die sowohl für interne als auch externe Zwecke geeignet war.*

# Projektdokumentationshinweise Verzeichnisse

## Verwendungsvermerk

darf man Sachen von der Firma benutzeb ist vorher zu klären

Es wurde in Absprache mit dem Projektgeber, Pixelding, und dem Kunden, Putzmittelshop24, eine Vereinbarung getroffen und genehmigt, dass ich ihre Daten für die Dokumentation meines Projekts verwenden darf.

Ich bestätige, dass der Projektantrag dem Ausbildungsbetrieb vorgelegt und vom Ausbildenden genehmigt wurde. Der Projektantrag enthält keine Betriebsgeheimnisse. Soweit diese für die Antragstellung notwendig sind, wurden nach Rücksprache mit dem Ausbildenden die entsprechenden Stellen unkenntlich gemacht.

Mit dem Absenden des Projektantrages bestätige ich weiterhin, dass der Antrag eigenständig von mir angefertigt wurde. Ferner sichere ich zu, dass im Projektantrag personenbezogene Daten (d. h. Daten über die eine Person identifizierbar oder bestimmbar ist) nur verwendet werden, wenn die betroffene Person hierin eingewilligt hat.

Bei meiner ersten Anmeldung im Online-Portal wurde ich darauf hingewiesen, dass meine Arbeit bei Täuschungshandlungen bzw. Ordnungsverstößen mit „null“ Punkten bewertet werden kann. Ich bin weiter darüber aufgeklärt worden, dass dies auch dann gilt, wenn festgestellt wird, dass meine Arbeit im Ganzen oder zu Teilen mit der eines anderen Prüfungsteilnehmers übereinstimmt. Es ist mir bewusst, dass Kontrollen durchgeführt werden.

## Dokumentation zur Projektarbeit

Das Projekt hat das Ziel, einen reibungslosen Prozess für die Integration neuer Produkte in einen bestehenden Online-Shop auf Basis von WordPress WooCommerce zu schaffen. Der Kunde plant, Produkte von neuen Lieferanten in seinen Shop aufzunehmen, wobei sämtliche Informationen zu den Produkten, einschließlich Name, SKU, Fotos, Booklet-PDFs, Preise und erfasst werden sollen.

Der Workflow beginnt mit der automatischen Erstellung einer strukturierten Liste aller Produkte von den neuen Lieferanten, wobei die Benutzung einer definierten Kommunikation über z.B. Schnittstellen nicht möglich ist. Diese Liste bildet die Grundlage für den weiteren Prozess. Fotos und PDF-Dateien werden in einem systematisch angelegten Ordner gespeichert, wodurch eine effiziente Organisation und Auffindbarkeit gewährleistet sind. Parallel dazu werden zwei separate Excel-Dateien vorbereitet: eine für das SEO-Team, um Produktbeschreibungen zu aktualisieren, und eine für das Grafikdesign-Team, um bei Bedarf Bilder zu aktualisieren. Eine zusätzliche Excel-Datei wird speziell für den Kunden erstellt, in der er Preise, Versandkosten und Lagerbestände bequem eingeben kann.

Die entwickelte Lösung gewährleistet nicht nur, dass der gesamte Prozess effizient und benutzerfreundlich abläuft, sondern bietet auch die automatische Generierung der ersten Produktliste, um menschliche Eingriffe auf ein Minimum zu reduzieren. Auf diese Weise kann der Kunde mühelos neue Produkte in seinem Online-Shop präsentieren.

## Abbildungsverzeichnis

Alle bilder von der Dok

## Tabellenverzeichnis

alle Tabellen von der Dok vielleicht alle in Bilder

## Abkürzungen

nur abk schreiebn die in der Dok sind

## Quellenverzeichnis

nur wen welche komme, Ki Bilder generieren

# Planungsphase

## Ausgangssituation

### Projektumfeld

Was macht die Firma/wie gross ist die Firma

Pixelding ist ein verlässlicher Partner in Sachen Web- und Grafikdesign aus der Medien-Hauptstadt Berlin. Mit einem Team von 2–10 Beschäftigten vereinen sie Expertise, Leidenschaft und effektive Kommunikation, um die Visionen ihrer Kunden zum Leben zu erwecken.

Ihre Kernkompetenzen umfassen Webdesign, Suchmaschinenoptimierung (SEO), die Entwicklung virtueller Touren sowie die Gestaltung und Umsetzung von Online-Shops und Buchungssystemen. Zusätzlich bieten sie professionelle Werbekampagnen und unterstützen bei der Gestaltung von Printmedien.

Weitere Informationen über Pixelding finden Sie auf unserer Website unter: <https://www.pixelding.de>

In welcher Abteilung wird das Projekt durchgeführt:

Mitarbeiter und Vorgesetze mit denen Sie interagieren im Projekt:

Das Projekt umfasst die Erweiterung eines Online-Shops und wird in verschiedenen Abteilung verarbeitet. Alexander Wolobuew, der Eigentümer von Pixelding, ist der verantwortliche Leiter des Projekts, Welches größtenteils von der Abteilung für Programmierung durchgeführt wird. Die Anpassung von Bildern liegt in der Zuständigkeit des Grafikteams, während das SEO-Team für die Textanpassungen verantwortlich ist.

KundenFirma:

Putzmittelshop24 ist ein etablierter Online-Shop für Reinigungsprodukte mit Sitz in Berlin. Der Shop wird von Thomas Pries geleitet und bietet eine breite Palette hochwertiger Reinigungsmittel für private Haushalte und gewerbliche Kunden.

Weitere Informationen über Putzmittelshop24 finden Sie auf unserer Website unter: https://putzmittelshop24.de/

### IST-Zustand

Der gegenwärtige Zustand des Online-Shops basiert auf einem manuellen Prozess zur Erstellung der Produktliste. Dabei werden Namen und Informationen von den Websites der Großhändler kopiert und eingefügt. Sowohl der Download von PDFs als auch von Bildern erfolgt manuell und beansprucht erheblich viel Zeit. Die manuelle Benennung und Strukturierung der Ordner für die Speicherung von Bildern und PDFs führt zu potenziellen Tippfehlern und erschwert die Auffindbarkeit des richtigen Pfads erheblich.

Zusätzlich ist der Kunde gezwungen, seine Produkte einzeln in die WooCommerce-Plattform über die langsame grafische Benutzeroberfläche von WordPress einzugeben. Dieser mühsame Prozess erfordert nicht nur das manuelle Verknüpfen von Bildern, sondern auch das Kopieren und Einfügen von Preisen und anderen Informationen. Die resultierende Arbeitsweise ist äußerst zeitaufwendig und ineffizient.

Die manuelle Natur dieser Abläufe birgt das Risiko von menschlichen Fehlern, was zu ungenauen Produktinformationen und Verzögerungen bei der Aktualisierung der Produktliste führen kann. Insgesamt ist der aktuelle Prozess nicht nur zeitintensiv, sondern auch anfällig für Fehler und stellt eine erhebliche Herausforderung für eine effiziente Verwaltung und Aktualisierung des Produktbestands dar.

### SOLL-Zustand

das koperien von Antrag

Unser Kunde benötigt sämtliche tausende neue Produkte mit Fotos, PDF-Downloads und Preisen und Beschreibungen auf seiner Webseite. Darüber hinaus benötigt er eine unkomplizierte Infrastruktur zur Preisgestaltung sowie die Möglichkeit, die Texte aller Produkte für SEO-Zwecke zu aktualisieren.

Die optimierte Lösung automatisiert den Prozess der Produktintegration in den Online-Shop erheblich. Ein automatischer Algorithmus generiert eine strukturierte Liste aller neuen Produkte von den Lieferantenwebseiten. Diese Liste bildet die Grundlage für sämtliche Aktualisierungen im System.

Fotos und PDFs werden automatisch heruntergeladen und in einem systematisch organisierten Ordner abgelegt. Dadurch wird eine fehleranfällige manuelle Handhabung vermieden, und die Auffindbarkeit der Dateien wird optimiert.

Zwei separate Excel-Dateien werden automatisch erstellt: Eine für das SEO-Team zur Aktualisierung von Produktbeschreibungen und eine für das Grafikdesign-Team zur optionalen Aktualisierung von Bildern. Die Excel-Dateien werden im Anschluss automatisch in das erforderliche CSV-Format für WooCommerce konvertiert.

Dies schließt auch die spezielle Excel-Datei des Kunden ein, um Preise, Versandkosten und Lagerbestände unkompliziert zu verwalten. Diese automatisierte Lösung gewährleistet nicht nur eine erhebliche Zeitersparnis, sondern minimiert auch das Risiko von Fehlern.

Der gesamte Prozess ist effizienter und benutzerfreundlicher, sodass der Kunde mühelos neue Produkte in seinem Online-Shop präsentieren kann.

### Projektziele

soll zu stand als Stichpunkte

Aufgrund der fehlenden Option einer definierten Kommunikation über beispielsweise Schnittstellen, beinhaltet der Prozess die Verwendung von Python zur Extraktion von Daten von den Webseiten der Lieferanten. Die gesammelten Daten werden in einer JSON-Datei gespeichert. Die Bilder und PDFs jedes Produkts werden in einem Ordner abgelegt, dessen Name sich aus dem Produktnamen ableitet, und der Pfad dazu wird in der JSON-Datei referenziert. JSON-File und Folder werden nicht nur auf Google Drive gesichert, sondern auch auf zwei separaten Festplatten für zusätzliche Redundanz abgelegt. Anschließend erfolgt die Umwandlung der JSON-Daten in das notwendige WooCommerce CSV-Format mittels Python. Die Dateien werden daraufhin auf den Test-WooCommerce-Shop im lokalen Host hochgeladen.

Das Grafikerteam erhält eine autogenerierte XLSX-Liste der Produktfotos zur eingehenden Prüfung. Gleichzeitig wird dem SEO-Team eine XLSX-Liste bereitgestellt, um Texte umzuschreiben und zu ergänzen. Parallel dazu wird eine Excel-Tabelle für den Kunden erstellt, um die Verwaltung seiner Preise und Lagerbestände zu erleichtern.

Schließlich wird ein System implementiert, das die Excel-Dateien wiederum in das WooCommerce CSV-Format konvertiert. Die CSV-Dateien werden lokal auf WordPress hochgeladen, um umfassende Tests durchzuführen. Bei erfolgreicher Durchführung werden die CSV-Dateien vom lokalen Shop exportiert und in den Online-Shop des Kunden übertragen. Nach erneuten Tests zur Sicherstellung eines reibungslosen Ablaufs werden sämtliche Projektdetails, Probleme und Fortschritte in Asana und Google Drive dokumentiert.

### Personellen Schnittstellen

Die Menschen mit der wir im Unterhenm sind nennen oder sagen sie möchten nicht bennant werden nicht mehr als 2, ein reicht aus bzw der Auftraggeber

In diesem Projekt habe ich unter der Verantwortung von Alexander Wolobuew gearbeitet, parallel dazu auch mit den Grafikern und SEO-Spezialisten.

### Technische Schnittstellen

könne muessen nicht sein, wenn nicht nötig server und was fuer den Project wichtig istbzw daten bank die benutz wird oder bereits exoistiert, api aus dem internet

* **Python**: Eine Programmiersprache, die häufig für die Automatisierung, Datenanalyse und Webentwicklung verwendet wird.
* **WordPress**: Eine Open-Source-Plattform zur Erstellung und Verwaltung von Websites und Blogs.
* **WooCommerce**: Ein E-Commerce-Plugin für WordPress, das es ermöglicht, einen Online-Shop zu erstellen und zu betreiben.
* **ONOS Server:** Ein Open Network Operating System, das zur Steuerung und Verwaltung von Netzwerken verwendet wird.
* **Windows:** Ein Betriebssystem von Microsoft, das auf PCs weit verbreitet ist.
* **XAMPP:** Eine kostenlose und Open-Source-Software, die einen lokalen Server für die Entwicklung von Webanwendungen bereitstellt.
* **GitHub:** Eine Plattform für die Versionsverwaltung und Zusammenarbeit bei der Softwareentwicklung.
* **Visual Studio Code:** Ein plattformübergreifender Code-Editor, der von Entwicklern für die Softwareentwicklung verwendet wird.
* **Google Drive**: Ein cloudbasierter Speicherdienst von Google, der die Speicherung und gemeinsame Nutzung von Dateien ermöglicht.
* **Asana**: Eine webbasierte Anwendung zur Teamkollaboration und Aufgabenverwaltung.
* **Slack**: Ein Messaging-Tool für Teams, das die Kommunikation und Zusammenarbeit unterstützt.
* **FileZilla**: Eine kostenlose und Open-Source-FTP-Software zur Dateiübertragung zwischen Computern und Servern.
* **phpMyAdmin**: Ein webbasiertes Administrationswerkzeug für die Verwaltung von MySQL-Datenbanken.

### Projektabgrenzung

ist der project allein stehen punkte weg lassen  
wenn Project ein Teil Projekt von einem Grösserem Project

Dieses Projekt ist Teil eines größeren Vorhabens, das darauf abzielt, den Online-Shop von Putzmittel24 zu erweitern und verschiedene zusätzliche Funktionen und Inhalte bereitzustellen. Innerhalb dieses größeren Projektrahmens sind bestimmte Grenzen und Abgrenzungen definiert, um den Fokus und die Ziele klar zu halten. Zu den wichtigsten Aspekten der Projektabgrenzung gehören:

**Erweiterung des Online-Shops:** Das Hauptziel des Projekts ist die Integration neuer Produkte in den bestehenden Online-Shop von Putzmittel24. Dabei sollen sämtliche Informationen zu den Produkten, einschließlich Name, SKU, Fotos, Booklet-PDFs und Preise, erfasst werden.

**Bereitstellung von Reinigungstipps**: Putzmittel24 plant die Einrichtung eines Informationsblocks auf seiner Website, der Reinigungstipps enthält. Dieser Block soll den Kunden zusätzliche nützliche Informationen bieten und das Engagement auf der Website erhöhen.

**Video-Kanal auf YouTube**: Um die Reichweite zu erhöhen und Kunden anzusprechen, wird auch die Einrichtung eines Video-Kanals auf YouTube geplant. Hier sollen Videos mit Anwendungstipps, Produktvorstellungen und anderen relevanten Inhalten veröffentlicht werden.

Präsenz in sozialen Medien: Zusätzlich zur YouTube-Plattform strebt Putzmittel24 eine Präsenz auf Facebook und Instagram an. Durch regelmäßige Beiträge und Interaktionen mit Kunden soll die Bekanntheit der Marke gesteigert und die Kundenbindung verbessert werden.

**Verbesserung der Website für Barrierefreiheit**: Ein weiteres wichtiges Ziel des Projekts ist die Verbesserung der Website von Putzmittel24 im Hinblick auf Barrierefreiheit. Dies umfasst die Anpassung der Website, um sie für Menschen mit Behinderungen zugänglicher zu machen und sicherzustellen, dass sie die Website ohne Einschränkungen nutzen können.

### Projektbegründung

unterschied zwishen ist und soll ??

Die Projektbegründung basiert auf einem klaren Ist-Zustand, der die ineffizienten und fehleranfälligen manuellen Prozesse bei der Verwaltung des Produktbestands des Online-Shops verdeutlicht. Der gegenwärtige Zustand zeigt, dass die manuelle Extraktion von Produktinformationen von den Websites der Großhändler zeitaufwendig ist und ein hohes Risiko für menschliche Fehler birgt. Dies führt zu ungenauen Produktinformationen und Verzögerungen bei der Aktualisierung der Produktliste.

Im Soll-Zustand wird deutlich, dass der Kunde eine umfassende Lösung benötigt, um den Prozess der Produktintegration erheblich zu optimieren und zu automatisieren. Die Anforderungen umfassen die automatisierte Extraktion von Produktinformationen von den Lieferantenwebseiten, das Herunterladen und Speichern von Fotos und PDFs, die automatische Generierung von Excel-Dateien zur Aktualisierung von Produktbeschreibungen und Bildern sowie die Konvertierung in das erforderliche CSV-Format für WooCommerce. Diese Lösung bietet nicht nur eine erhebliche Zeitersparnis, sondern minimiert auch das Risiko von Fehlern und verbessert die Effizienz der Produktverwaltung erheblich.

Die Entscheidung, Python für die Implementierung dieser Lösung zu verwenden, ergibt sich aus der Tatsache, dass eine definierte Kommunikation über Schnittstellen nicht möglich ist, da keine Zugangsdaten vorhanden sind(wird ergaenzt). Somit ist Python die geeignete Plattform, um die automatisierten Prozesse effektiv umzusetzen und die Anforderungen des Kunden zu erfüllen.

### Projektrisiken

Kranksein? mit ansprechpartenr darueber sprechen, etwas nicht rechtzeitig geliefert

Punkt wirfd ausfallen

## Projektablauf

### Technische Planung des Projektzieles

??

Die technische Planung des Projektziels beinhaltet die detaillierte Analyse und Planung der Schritte, die erforderlich sind, um das definierte Ziel des Projekts zu erreichen. Dies umfasst die Auswahl der geeigneten Technologien, Tools und Methoden sowie die Festlegung des Arbeitsablaufs und der Ressourcen.

Im vorliegenden Projektziel, die Automatisierung des Produktintegrationsprozesses in den Online-Shop des Kunden, stehen folgende technische Schritte im Vordergrund:

1. **Auswahl der Programmiersprache und Frameworks:** Die Wahl fällt auf Python als Hauptprogrammiersprache aufgrund seiner Vielseitigkeit und der verfügbaren Bibliotheken für Web-Scraping und Datenverarbeitung. Als Framework wird BeautifulSoup für das Web-Scraping und pandas für die Datenverarbeitung verwendet.
2. **Implementierung des Web-Scraping-Algorithmus**: Entwicklung eines Python-Programms, das automatisch die Webseiten der Lieferanten durchsucht, die benötigten Produktinformationen extrahiert und in einer strukturierten Form speichert.
3. **Automatisierte Dateiverwaltung:** Einrichtung eines Systems zur automatischen Speicherung von heruntergeladenen Fotos und PDFs in einem ordnungsgemäß organisierten Ordnerstrukturen, um die Auffindbarkeit und Organisation zu optimieren.
4. **Entwicklung von Excel-Exportfunktionen:** Implementierung von Funktionen zur automatischen Generierung von Excel-Dateien für das SEO-Team und das Grafikdesign-Team, die für die Aktualisierung von Produktbeschreibungen und Bildern zuständig sind.
5. **Integration mit WooCommerce:** Konvertierung der Excel-Dateien in das erforderliche CSV-Format für WooCommerce und automatische Aktualisierung des Produktbestands im Online-Shop des Kunden.
6. **Testen und Validierung:** Durchführung von umfassenden Tests, um sicherzustellen, dass der entwickelte Algorithmus fehlerfrei funktioniert und den Anforderungen des Kunden entspricht. Validierung des Systems durch den Kunden, um sicherzustellen, dass es seinen Anforderungen entspricht.

### Analyse des IST-Zustandes

??

Das habe ich doch bereits in 2.1.2 Ausganglage/Istzustand

### Erstellung des SOLL-Konzeptes

soll zu stand als Stichpunkte wie bereits oben oder weglassen

**# Zeitplanung**

## Technische Analyse des Ist-Zustandes: 5 Stunden

- Untersuchung des aktuellen manuellen Prozesses zur Integration neuer Produkte: 2 Stunden

- Evaluierung der bestehenden Datenextraktionsmethoden und -werkzeuge: 2 Stunden

- Meeting: 1 Stunde

## Herstellen der Arbeitsumgebung: 3 Stunden

- Aufsetzen einer geeigneten Entwicklungsumgebung, einschließlich der erforderlichen Softwaretools wie Python, BeautifulSoup oder Selenium, Xampp: 3 Stunden

## Datenextraktion von 4 Webseiten: 31 Stunden

- Python Script schrieben: 10 Stunden

- Durchführung von Modultests auf die Datenextraktion mittels Python: 5 Stunden

- Durchführung eines Systemtests, um die gesamte Datenextraktionspipeline zu überprüfen: 5

Stunden

- Script anpassen und herunterladen: 9 Stunden

- Sicherung auf Google Drive sowie auf zwei separaten Festplatten für zusätzliche Redundanz:

1 Stunde

- Meeting: 1 Stunde

## Datenkonvertierung für WooCommerce: 11 Stunden

- Umwandlung der JSON-Daten in das erforderliche WooCommerce CSV-Format mittels

Python: 7 Stunden

- WordPress Localhost-Shop installieren: 2 Stunden

- Tests: 1 Stunde

- Hochladen der Dateien auf den Test-WooCommerce-Shop im lokalen Host: 1 Stunde

## XLSX-Liste der Produktfotos für das Grafikerteam: 1 Stunde

- Bereitstellung einer XLSX-Liste der Produktfotos zur eingehenden Prüfung: 1 Stunde

## XLSX-Liste der Texte für das SEO-Team: 1 Stunde

- Bereitstellung einer XLSX-Liste für das SEO-Team zur Überarbeitung und Ergänzung von Texten: 1 Stunde

## Erstellung einer Excel-Tabelle für den Kunden: 1 Stunde

- Erstellung einer Excel-Tabelle zur erleichterten Verwaltung von Preisen und Lagerbeständen durch den Kunden: 1 Stunde

## Implementierung eines Systems für die Datenrückkonvertierung: 2 Stunden

- Implementierung eines Systems zur Konvertierung der bearbeiteten Excel-Dateien zurück in das WooCommerce CSV-Format: 2 Stunden

## Lokale Tests auf WordPress: 7 Stunden

- Hochladen der CSV-Dateien lokal auf WordPress: 1 Stunde

- umfassende Tests: 6 Stunden

## Export und Übertragung in den Online-Shop des Kunden: 2 Stunden

- Export der CSV-Dateien vom lokalen Shop und Übertragung in den Online-Shop des Kunden: 2 Stunden

## Erneute Tests zur Sicherstellung eines reibungslosen Ablaufs: 6 Stunden

- Durchführung erneuter Tests, um die ordnungsgemäße Funktion sicherzustellen: 6 Stunden

## Dokumentation von Projektdetails, Problemen und Fortschritten: 10 Stunden

- Dokumentation sämtlicher Projektdetails, auftretender Probleme und erzielter Fortschritte in Asana und Google Drive. 9 Stunden

- Meeting: 1 Stunde

## Gesamt: 80 Stunden

# Ressourcen- und Ablaufplan

## Personalplanung

nur ich

An der Programmierung der Python CSV- und XLSX-Funktionen arbeite ich alleine.

## Sachmittelplanung

was gebraucht wird

In der Sachmittelplanung werden die benötigten Ressourcen für das Projekt identifiziert, beschafft und verwaltet. Dazu gehören physische Gegenstände wie Computer, für die bereits vorhandene Geräte verwendet werden, sowie Softwarelizenzen wie WooCommerce und Visual Studio Code, die bereits bezahlt wurden oder kostenlos verfügbar sind.

## Zeitliche Planung

erst mal leer lassen

Das Projekt ist geplant, vom 19. Februar 2024 bis zum 12. April 2024 abgeschlossen zu werden.

## Kostenplanung

was ich verdiene im echten leben pro Stunde \*80 bei meeting schaeten was der ansprechpartenr kostet energie pro schal 6 euro am Tag home office

# Durchführungsphase

## Eventuelle Beschaffung gemäß Planung

## Da kommt nichts

## Herstellen der Arbeitsumgebung

wi ich mich zurecht finde und arbeitumbgebung fertig mach

Die Einrichtung der Arbeitsumgebung ist ein wesentlicher Schritt, um effizient und produktiv am Projekt zu arbeiten. Hier sind einige Schritte, um deine Arbeitsumgebung vorzubereiten:

1. **Installation von Software und Tools:** Stelle sicher, dass du alle benötigten Softwareanwendungen und Tools installiert hast, die für das Projekt erforderlich sind. Dazu gehören Python, BeautifulSoup, pandas, ein Texteditor wie Visual Studio Code und möglicherweise andere Bibliotheken oder Frameworks, je nach den spezifischen Anforderungen des Projekts.
2. **Einrichten des Projektverzeichnisses:** Erstelle ein dediziertes Verzeichnis für das Projekt auf deinem Computer. Organisiere die Dateien und Ordner in einer logischen Struktur, um eine einfache Navigation und Auffindbarkeit zu gewährleisten.
3. **Version Control einrichten:** Falls du Git verwenden möchtest, um Änderungen am Code zu verfolgen und zu verwalten, richte ein Git-Repository ein und initialisiere es in deinem Projektverzeichnis.
4. **Dokumentation erstellen:** Beginne mit der Erstellung von Dokumentationen für das Projekt, einschließlich eines Projektplans, einer technischen Spezifikation und eventuell einer Benutzerdokumentation. Dies hilft dir, den Überblick über den Fortschritt des Projekts zu behalten und anderen Teammitgliedern bei der Orientierung zu helfen.
5. **Testdaten vorbereiten:** Wenn Testdaten erforderlich sind, um die Funktionalität deiner Anwendung zu überprüfen, stelle sicher, dass du über ausreichende Testdaten verfügst und diese entsprechend vorbereitest.
6. **Kommunikationskanäle einrichten:** Stelle sicher, dass du Zugang zu den erforderlichen Kommunikationskanälen hast, um mit anderen Teammitgliedern, dem Kunden und anderen relevanten Parteien in Kontakt zu treten. Dies kann E-Mail, Slack, Asana oder andere Projektmanagement- und Kommunikationstools umfassen.
7. **Systemkonfiguration überprüfen:** Überprüfe, ob dein System ordnungsgemäß konfiguriert ist und alle erforderlichen Einstellungen vorgenommen wurden, um reibungslos arbeiten zu können. Dies kann die Konfiguration von Proxy-Einstellungen, Netzwerkeinstellungen oder anderen Systemeinstellungen umfassen.

Indem du diese Schritte befolgst, kannst du sicherstellen, dass deine Arbeitsumgebung optimal eingerichtet ist und du effizient am Projekt arbeiten kannst.

e

## Eigentlich Umsetzung

**Daten von den Webseiten holen**

Das gilt für alle 4 Webseiten:

* Zuerst wird ein Python-Skript geschrieben, das die URLs der Hauptseiten und Unterseiten erfasst und in einem Protokolltextdokument namens "Protokoll.txt" speichert.
* Das Protokoll.txt-Dokument enthält auch eine Nummerierung der Seiten, die mit der erwarteten Anzahl der Seiten verglichen werden kann, die aus der Navigation berechnet werden kann.
* Es werden Seiten herausgefiltert, die mehrmals auftreten, um Redundanzen zu vermeiden.
* Anschließend wird eine Ordnerstruktur für jedes Produkt basierend auf dem Namen der jeweiligen Seite erstellt.
* Die Namen der Ordner werden auf ihre Gültigkeit überprüft, um sicherzustellen, dass sie nur erlaubte Zeichen enthalten.
* Das Skript wird zunächst nur auf einer Seite getestet, um das URL und das h1-Element in einer JSON-Datei aufzulisten.
* Das Protokolldokument enthält nun die URLs der Seiten sowie das h1-Element der getesteten Seite.
* Falls das Auflisten des h1-Elements erfolgreich ist, wird das Skript für alle Seiten ausgeführt und im Protokolldokument vermerkt.
* Bei auftretenden Problemen wird im Protokolldokument die URL der Seite angegeben, bei der das Problem aufgetreten ist. Das Skript wird entsprechend angepasst. Das Testverfahren wird im nächsten Kapitel erklärt.
* Nach erfolgreicher Ausführung des Skripts auf einer Seite wird versucht, das nächste Element zu extrahieren.

Nachdem das Skript erfolgreich das h1-Element auf einer Seite extrahiert hat, wird der

gleiche Ansatz auf andere Elemente angewendet, einschließlich Beschreibung,

Kurzbeschreibung, PDFs, Bilder, Kategorien, Tags usw. Jedes Element wird nach dem

gleichen Verfahren im JSON-Format aufgelistet.

Falls BeautifulSoup das Element nicht finden kann, wird das Skript auf Selenium

umgeschaltet, um eine genauere Suche durchzuführen. Jegliche Probleme oder Fehler werden

im Protokolldokument vermerkt und das Skript wird entsprechend angepasst, um das Problem

zu lösen.

Sobald das Skript erfolgreich für einzelne Elemente funktioniert, wird es auf alle Seiten

angewendet, um die gesamten Elemente zu extrahieren. Dieser Prozess wird systematisch

durchgeführt, um sicherzustellen, dass alle relevanten Informationen von jeder Seite erfasst

werden. Jegliche auftretende Probleme werden im Protokolldokument festgehalten und

entsprechende Anpassungen am Skript vorgenommen, um diese zu beheben.

Das Skript, das JSON-Datei und das Protokoll werden im persönlichen Git-Repository

gespeichert, das auf Google Drive bei Pixelding, auf dem Arbeitscomputer von Pixelding und

auf meinem Heimcomputer gespeichert ist. Die PDF- und Bilderordner werden ebenfalls in

diesem Repository abgelegt, um eine konsistente und zugängliche Speicherung

sicherzustellen.

Damit ist der erste Teil fertig

**Daten für WordPress WooCommerce csv, and xlsx umbauen**

Um die Daten für die Integration in den WordPress WooCommerce-Shop vorzubereiten, wurde

ein Python-Skript entwickelt. Dieses Skript liest Daten aus einer JSON-Datei und wandelt sie in

das erforderliche CSV- und XLSX-Format um. Die JSON-Datei enthält strukturierte Informationen

über Produkte, einschließlich Namen, Beschreibung und andere relevante Details. CSV und

Excel haben zwar die gleichen Werte, aber CSV ist für Menschen so unübersichtlich.

Das Skript beginnt damit, einen ersten Index der JSON-Daten zu lesen und in die beiden

verschiedenen Formate zu konvertieren, um zu testen, ob der Skript funktioniert. Wenn dieser

Test erfolgreich ist, wird das Skript auf den gesamten JSON-Datensatz angewendet.

Durch die Entwicklung dieses Skripts wurde sichergestellt, dass die Daten für den Online-Shop in

den erforderlichen Formaten vorliegen und einfach in das WooCommerce-System integriert

werden können. Dies trägt dazu bei, den Prozess der Produktintegration zu rationalisieren und

die Effizienz bei der Verwaltung des Online-Shops zu verbessern.

Speicher in GitHub und auf den Computers

**CSV in WordPress WooCommerce lokal einbinden**

Zuerst wird für jede Webseite eine lokale Wordpress-Instanz eingerichtet. Anschließend wird

WooCommerce auf dieser lokalen Wordpress-Instanz installiert, gefolgt von der Installation des

Plugins WooCommerce Germanized.

Dann wird die CSV-Datei mit den Produktdaten über die WordPress WooCommerce-GUI

hochgeladen. Im Falle eines Fehlers in der CSV-Datei wird dies hier gemeldet.

Nach erfolgreicher Integration wird überprüft, ob die Produkte auf der Website ordnungsgemäß dargestellt werden und ob alle Informationen korrekt angezeigt werden.

**Daten für WooCommerce vorbereiten**

## Testen

ganz wichtig

## Fehler, Fehlerbehebung und Anpassung

wenn kein Problem weg lassen

# Übergabe und Auswertung

## Projektübergabe

wie das project abgegeben wurde

## Mitarbeiterschulung, Bedienungsanleitung

?

## SOLL-IST-Vergleich (inhaltlich, zeitlich, kostentechnisch)

wieveil zeit fuer jeden punkt eine ganze seite

## Technische Auswertung und Ausblick

was hat mir das gebracht?  
ausblcik was passiert mit diesem Programm jetzt?

## Persönliche Auswertung

Wie es war fuer mich

Max zeilen abstand 1,5

**gehen Sie in Ihrer Dokumentation bitte genauer auf den Punkt "Datenextraktion von den Webseiten der Lieferanten" ein. Warum wurde sich für diesen Weg entschieden und warum nicht für eine definiertere Kommunikation über z.B. Schnittstellen? was kann ich dazu sagen??**

NUR WEIL ICH DAS KANN, Preis von der Lizenz und Zeit zu lernen Webharvey oder Visual Web Ripper, ist es am Ende nicht günstiger.

Wenn ich selber programmiere, ist es lighter, ich habe einen gutem Überblick und wenn es nicht geht bin ich auch dran selber schuld. Die Seiten sind auch schlecht gebaut und es ist viel anzupassen damit es am Ende klappt

**wie hätte es sonst funktioniert mit einer definierten Kommunikation über z.B. Schnittstellen? kannst erklaeren?**

Wenn eine definierte Kommunikation über Schnittstellen oder APIs gewählt worden wäre, hätte das bedeutet, dass wir direkt mit den Systemen der Lieferanten kommunizieren würden, um die benötigten Daten abzurufen. Hier sind einige Schritte, wie es funktionieren könnte:

1. **Schnittstellenintegration**: Zunächst müssten wir mit jedem Lieferanten individuell Schnittstellen oder APIs integrieren. Dies könnte bedeuten, dass wir uns mit verschiedenen Systemen und Datenformaten auseinandersetzen müssen, je nach den Systemen, die von den Lieferanten verwendet werden.
2. **Authentifizierung und Autorisierung**: Wir müssten sicherstellen, dass wir die erforderlichen Zugriffsberechtigungen haben, um auf die Daten der Lieferanten zuzugreifen. Dies könnte die Implementierung von Authentifizierungsmechanismen wie API-Schlüsseln oder OAuth umfassen.
3. **Datenabfrage**: Nachdem die Integration abgeschlossen ist, könnten wir über die Schnittstellen oder APIs Anfragen an die Systeme der Lieferanten senden, um die benötigten Daten abzurufen. Dies würde normalerweise die Definition spezifischer Abfragen oder Endpunkte für verschiedene Datentypen umfassen.
4. **Datenverarbeitung**: Nachdem die Daten abgerufen wurden, müssten wir sie möglicherweise transformieren oder anpassen, um sie in das gewünschte Format für unsere Zwecke zu bringen. Dies könnte das Mapping von Feldern, die Konsolidierung von Daten aus verschiedenen Quellen oder die Bereinigung von Daten umfassen.
5. **Regelmäßige Aktualisierung**: Da sich Daten kontinuierlich ändern können, müssten wir sicherstellen, dass unsere Integrationen regelmäßig aktualisiert werden, um die neuesten Daten von den Lieferanten zu erhalten. Dies könnte die Implementierung von automatisierten Aktualisierungsprozessen oder Zeitplänen umfassen.

*Obwohl die Verwendung von Schnittstellen oder APIs eine standardisierte und möglicherweise robustere Lösung sein kann, kann sie auch mit einigen Herausforderungen verbunden sein, insbesondere wenn die Lieferanten unterschiedliche Systeme verwenden oder keine umfassenden Schnittstellen anbieten. Die direkte Extraktion von Webseiteninhalten bietet in solchen Fällen oft eine flexiblere und kosteneffizientere Lösung.*

comment cela aurait-il fonctionné avec une communication définie via des interfaces par exemple ? tu peux expliquer ?

Si une communication définie via des interfaces ou des API avait été choisie, cela aurait signifié que nous aurions communiqué directement avec les systèmes des fournisseurs pour récupérer les données nécessaires. Voici quelques étapes sur la manière dont cela pourrait fonctionner :

1. intégration des interfaces : nous devrions d'abord intégrer individuellement des interfaces ou des API avec chaque fournisseur. Cela pourrait signifier que nous devons nous occuper de différents systèmes et formats de données, en fonction des systèmes utilisés par les fournisseurs.

2) Authentification et autorisation : nous devrions nous assurer que nous disposons des droits d'accès nécessaires pour accéder aux données des fournisseurs. Cela pourrait impliquer la mise en œuvre de mécanismes d'authentification tels que les clés API ou OAuth.

3) Requête de données : une fois l'intégration terminée, nous pourrions envoyer des requêtes aux systèmes des fournisseurs via les interfaces ou les API afin de récupérer les données requises. Cela impliquerait normalement la définition de requêtes ou de points de terminaison spécifiques pour différents types de données.

4) Traitement des données : une fois que les données ont été récupérées, nous pourrions avoir besoin de les transformer ou de les adapter afin de les mettre dans le format souhaité pour nos besoins. Cela pourrait inclure le mappage des champs, la consolidation des données provenant de différentes sources ou le nettoyage des données.

5) Mise à jour régulière : étant donné que les données peuvent changer en permanence, nous devrions veiller à ce que nos intégrations soient régulièrement mises à jour afin d'obtenir les données les plus récentes auprès des fournisseurs. Cela pourrait inclure la mise en œuvre de processus de mise à jour automatisés ou de calendriers.

Bien que l'utilisation d'interfaces ou d'API puisse être une solution standardisée et potentiellement plus robuste, elle peut aussi présenter certains défis, notamment si les fournisseurs utilisent des systèmes différents ou ne proposent pas d'interfaces complètes. Dans de tels cas, l'extraction directe du contenu des pages web offre souvent une solution plus flexible et plus rentable.

**gehen Sie in Ihrer Dokumentation bitte genauer auf den Punkt "Datenextraktion von den Webseiten der Lieferanten" ein. Warum wurde sich für diesen Weg entschieden und warum nicht für eine definiertere Kommunikation über z.B. Schnittstellen? was kann ich dazu sagen??**

Intro: Eine präzisere Kommunikation über Schnittstellen oder APIs stellt oft eine verlockende Option für die Extraktion von Daten von Webseiten dar. Im ersten Abschnitt werden wir erkunden, was diesen Ansatz besonders attraktiv macht. Anschließend werden wir die potenziellen Herausforderungen betrachten, die sich dabei ergeben können. Im dritten Abschnitt teile ich meine persönlichen Erfahrungen und Einsichten zu diesem Thema.

Vorteile einer definierteren Kommunikation über Schnittstellen: Die Nutzung von Schnittstellen oder APIs bietet klare Vorteile. Erstens ermöglichen sie eine standardisierte und strukturierte Datenübertragung, wodurch die Integration mit den Systemen der Lieferanten erheblich vereinfacht wird. Zweitens kann die direkte Anbindung an die Systeme der Lieferanten die Aktualität und Zuverlässigkeit der Daten verbessern, da sie in Echtzeit abgerufen werden können. Drittens ermöglicht die Verwendung von Schnittstellen eine effizientere und automatisierte Datenübertragung, was Zeit und Ressourcen spart. Darüber hinaus erfordert dieser Ansatz keine Programmierung und kann schnell implementiert werden. Zudem stehen dem Kunden Online-Hilfe und Support zur Verfügung, um den Integrationsprozess weiter zu erleichtern.

Nachteile einer definierteren Kommunikation über Schnittstellen: Trotz der Vorteile können bei der Verwendung von Schnittstellen auch potenzielle Nachteile auftreten. Erstens kann die Integration mit verschiedenen Lieferanten aufgrund der Notwendigkeit individueller Anpassungen für jede Schnittstelle komplex und zeitaufwändig sein. Zweitens sind Schnittstellen möglicherweise nicht immer verfügbar oder bieten nicht alle benötigten Daten an, was zu Lücken in den Produktinformationen führen kann. Darüber hinaus erfordert die Nutzung von Schnittstellen auch Zeit und Ressourcen, um die Bedienung zu erlernen, geeignete Schnittstellen zu finden und diese zu testen, was den Integrationsprozess weiter verkomplizieren kann. Zudem besteht eine Abhängigkeit vom Kundenservice, wenn Probleme auftreten, was die Effizienz und den Betrieb der Integrationen beeinträchtigen kann.

Meine eigene Meinung: Ich sehe die Verwendung von Python und Web-Scraping-Bibliotheken wie BeautifulSoup und Selenium als eine effektive und kostengünstige Alternative zur Nutzung von Schnittstellen. Diese Methode ermöglicht eine flexible und anpassungsfähige Datenextraktion direkt von den Webseiten der Lieferanten. Durch die Entwicklung eigener Skripte habe ich die Kontrolle über den Extraktionsprozess und kann schnell auf Änderungen oder Probleme reagieren. Darüber hinaus können wir Kosten senken und eine effiziente Datenintegration sicherstellen. Datenextraktion mittels Web-Scraping stellt keine große Herausforderung dar, und selbst wenn es einmal schwierig wird, bietet es die Möglichkeit, neue Fähigkeiten zu erlernen und den eigenen Wissensschatz zu erweitern. Und es macht unheimlich viel Spaß.

Schlussatz: Insgesamt bieten sowohl die Nutzung definierter Schnittstellen als auch die Datenextraktion mit Python verschiedene Vor- und Nachteile. Die Wahl des geeigneten Ansatzes hängt von den spezifischen Anforderungen des Projekts sowie von Faktoren wie Kosten, Flexibilität und Kontrolle, und persönlichen Präferenzen ab. Eine kontinuierliche Bewertung und Anpassung dieser Methoden im Verlauf des Projekts ist jedoch entscheidend, um sicherzustellen, dass sie den sich ändernden Anforderungen und Gegebenheiten gerecht werden. Ich bin mit der Nutzung von Python für diesen Projekt sehr zufrieden.

**Aktikitaets diagram**

**Netzplan**

Hier sind einige mögliche Interpretationen von "Schnittstellen" in deinem Kontext:

Web Scraping: Die Schnittstelle zwischen deinem Programm und den Webseiten der Lieferanten. Du könntest erklären, wie dein Programm die Webseiten analysiert, um Informationen zu extrahieren.

Dateiübertragung: Die Schnittstelle zwischen deinem Programm und den Dateispeichern (z.B. Google Drive, lokale Festplatten). Du könntest beschreiben, wie dein Programm Daten speichert und abruft.

Datenkonvertierung: Die Schnittstelle zwischen den verschiedenen Datenformaten (z.B. JSON, CSV). Du könntest erklären, wie dein Programm Daten in das richtige Format konvertiert.

Kommunikation mit anderen Teams: Die Schnittstelle zwischen deinem Programm und anderen Teams, z.B. dem Grafikdesign-Team oder dem SEO-Team. Du könntest beschreiben, wie dein Programm Daten austauscht oder Informationen bereitstellt.

In deinem Dokument könntest du also unter dem Punkt "Schnittstellen" genau erklären, wie dein Programm mit diesen verschiedenen Aspekten interagiert und wie die Kommunikation erfolgt. Dies könnte beinhalten, wie Daten gesammelt, gespeichert, verarbeitet und weitergegeben werden.

WebHarvy ist eine weitere Software, die für das Web-Scraping verwendet werden kann. Es ist eine Desktop-Anwendung, die speziell für das Extrahieren von Daten aus Webseiten entwickelt wurde. Mit WebHarvy kannst du einfach die gewünschten Webseiten aufrufen, die Daten selektieren, die du extrahieren möchtest, und dann die extrahierten Daten in einem gewünschten Format speichern, z.B. als CSV- oder Excel-Datei.

Der Hauptvorteil von WebHarvy liegt in seiner Benutzerfreundlichkeit und seiner Fähigkeit, Daten aus komplexen Webseiten ohne Programmierung zu extrahieren. Es bietet eine intuitive Benutzeroberfläche, die es auch weniger technisch versierten Benutzern ermöglicht, Web-Scraping-Aufgaben durchzuführen.

Wenn dir WebHarvy angeboten wurde, könnte es eine gute Option sein, besonders wenn du keine Programmierkenntnisse hast oder eine schnellere Lösung wünschst. Allerdings solltest du beachten, dass Desktop-Anwendungen wie WebHarvy normalerweise eine einmalige Lizenzgebühr erfordern und möglicherweise nicht so flexibel oder anpassbar sind wie Python-basierte Lösungen mit Bibliotheken wie BeautifulSoup und Selenium.

Top of Form

Testing: Die And retry

Wenn das Programm beim ersten Mal schnell und gut läuft, ist es sicher ein gravierender Fehler drinen.

Module testing: nur einzeln elemnte testen

Global testing: das ganze program laufen lassen aber berentzt auf nur ein paar Seiten

Global global Testing : das ganze Program laufen

Neben von der json schreibt python ein protokole was gemacht wurde

White box test, black box tests

Protokoll sieht so aus:

1 url from 2 main urls

https://www.noelle-profi-brush.de/produkte/fussmatten/fussmatten.html

getMainSnippet

- url to open https://www.noelle-profi-brush.de/produkte/fussmatten.html

- - url\_to\_find https://www.noelle-profi-brush.de/produkte/fussmatten/fussmatten.html

- opening 1 from 1 https://www.noelle-profi-brush.de/produkte/fussmatten/fussmatten.html

...........Links.......

...........Links.......

...........Links.......

...........Links.......

...........Links.......

...........Links.......

...........Links.......

- FOUND Links: 1

- - https://www.noelle-profi-brush.de/produkte/fussmatten/fussmatten.html

- - - FOUND Produkts pages: 7

- - - - https://www.noelle-profi-brush.de/produkte/fussmatten/fussmatten/produktdetails/allwetter-grasmatte-911.html

- - - - https://www.noelle-profi-brush.de/produkte/fussmatten/fussmatten/produktdetails/gummi-ringmatte-941.html

- - - - https://www.noelle-profi-brush.de/produkte/fussmatten/fussmatten/produktdetails/gummi-wabenmatte-922.html

- - - - https://www.noelle-profi-brush.de/produkte/fussmatten/fussmatten/produktdetails/kokos-drahtgittermatte-1024.html

- - - - https://www.noelle-profi-brush.de/produkte/fussmatten/fussmatten/produktdetails/kokos-veloursmatte-1488.html

* **Selenium**: Selenium is a web browser automation tool that allows you to control web browsers programmatically. It is often used for tasks such as filling out web forms, clicking buttons, and navigating through web pages. Selenium can interact with dynamic web content and JavaScript-heavy websites, making it a powerful tool for web scraping. It simulates user interaction with the web page, allowing you to extract data as if you were using a web browser yourself.
* **BeautifulSoup**: BeautifulSoup is a Python library for parsing HTML and XML documents. It provides simple methods and Pythonic idioms for navigating, searching, and modifying the parse tree. BeautifulSoup is often used in conjunction with other libraries like Requests to scrape data from web pages. While BeautifulSoup itself does not interact with web browsers like Selenium does, it is very effective for extracting data from static web pages and parsing the HTML structure.

**WordPress**: WordPress ist eine Open-Source-Software, die ursprünglich als Blogging-Plattform entwickelt wurde, sich aber im Laufe der Zeit zu einem leistungsstarken Content-Management-System (CMS) entwickelt hat. Es wird weltweit von Millionen von Websites verwendet und bietet eine benutzerfreundliche Oberfläche sowie eine Vielzahl von Plugins und Themes für die Erstellung verschiedener Arten von Websites, darunter Blogs, Unternehmenswebsites, Online-Shops und mehr.

**WooCommerce**: WooCommerce ist ein beliebtes E-Commerce-Plugin für WordPress. Es ermöglicht es Benutzern, schnell und einfach einen Online-Shop zu erstellen und zu verwalten, indem es Funktionen wie Produktlisten, Warenkörbe, Kassenseiten und Zahlungsabwicklung integriert. WooCommerce bietet eine Vielzahl von Erweiterungen und Themes, mit denen Benutzer ihren Online-Shop an ihre individuellen Bedürfnisse anpassen können. Es ist eine kostengünstige und skalierbare Lösung für kleine bis mittlere Unternehmen, die einen Online-Verkaufskanal aufbauen möchten.

**WooCommerce Germanized:** WooCommerce Germanized ist eine Erweiterung für WooCommerce, die speziell für den deutschen Markt entwickelt wurde. Sie erleichtert die Einhaltung deutscher Gesetze und Richtlinien für den Online-Handel, einschließlich des deutschen Datenschutzrechts (DSGVO), des Widerrufsrechts und der Impressumspflicht. WooCommerce Germanized bietet Funktionen wie eine optimierte Checkout-Seite für deutsche Kunden, die Integration von Rechtstexten und die Anpassung von E-Mail-Benachrichtigungen an deutsche Rechtsvorschriften. Es ist eine wertvolle Erweiterung für deutsche Online-Händler, die sicherstellen möchten, dass ihr Online-Shop den lokalen gesetzlichen Anforderungen entspricht.

**Woocommerce API**: un produkte zu loeschen auf dem Local host

**Wooocommerce info page from git**

Logo erlaubnis:

wordpress,

woocomerce,

**WooCommerce Germanized**

python,

visualstudio code

pixelding

putzmittelshop24

phpmyadmin

bfw

wordpress gui ist sooo langsam

<https://codex.wordpress.org/Database_Description>

A computer screen shot of a computer

Description automatically generated

By default, WordPress uses the MyISAM storage engine for database tables, which does not support foreign keys or enforce referential integrity. MyISAM tables are optimized for read-heavy workloads and offer better performance for certain types of queries.

However, while foreign keys are not enforced at the database level in WordPress, developers can still implement referential integrity and establish relationships between tables at the application level. This means that developers need to manually manage relationships between tables and ensure data consistency through their WordPress plugins, themes, or custom code.

For example, if you're creating a custom plugin that extends WordPress functionality and involves multiple database tables with relationships, you would need to write code to handle data validation, insertion, and updates in a way that maintains referential integrity.

Alternatively, developers can choose to use the InnoDB storage engine for their WordPress database tables, which does support foreign keys and referential integrity. InnoDB tables provide stronger data consistency guarantees compared to MyISAM tables, but they may come with a slight performance overhead, especially for write-heavy workloads.

Ultimately, the choice between MyISAM and InnoDB depends on the specific requirements and performance characteristics of your WordPress application, as well as your preference for data consistency versus performance.

WordPress installation doesn’t enforce referential integrity. WordPress uses the MyISAM storage engine, which gives more functionality and better performance but does not support foreign keys.

ChatGPT

**ChatGPT**

Yes, that's correct. By default, WordPress uses the MyISAM storage engine for database tables, which does not support foreign keys or enforce referential integrity. MyISAM tables are optimized for read-heavy workloads and offer better performance for certain types of queries.

However, while foreign keys are not enforced at the database level in WordPress, developers can still implement referential integrity and establish relationships between tables at the application level. This means that developers need to manually manage relationships between tables and ensure data consistency through their WordPress plugins, themes, or custom code.

For example, if you're creating a custom plugin that extends WordPress functionality and involves multiple database tables with relationships, you would need to write code to handle data validation, insertion, and updates in a way that maintains referential integrity.

Alternatively, developers can choose to use the InnoDB storage engine for their WordPress database tables, which does support foreign keys and referential integrity. InnoDB tables provide stronger data consistency guarantees compared to MyISAM tables, but they may come with a slight performance overhead, especially for write-heavy workloads.

Ultimately, the choice between MyISAM and InnoDB depends on the specific requirements and performance characteristics of your WordPress application, as well as your preference for data consistency versus performance.

https://webappick.com/woocommerce-database-schema-explained/

### Find WooCommerce Products in the WP Database

[WooCommerce product data](https://woocommerce.com/document/bundles/bundles-data-structures-storage/) can be found in the following tables:

* wc\_product\_meta\_lookup (product ID or product\_variation ID)
* wc\_order\_product\_lookup (product ID or product\_variation ID)
* posts (product ID or product\_variation ID)
* postmeta (product or product\_variation)
* posts (product ID or product\_variation ID)
* wc\_product\_meta\_lookup (product ID or product\_variation ID)

A screenshot of a computer

Description automatically generated

* wc\_order\_product\_lookup (product ID or product\_variation ID)
* A screenshot of a computer

  Description automatically generated
* posts (product ID or product\_variation ID)
* postmeta (product or product\_variation)
* posts (product ID or product\_variation ID)

https://www.webhat.in/article/woocommerce-tutorial/decoding-product-data-part-1-of-the-woocommerce-db-series/

A computer screen shot of a computer

Description automatically generated

Anhang