



Fiori App Entwicklung

Aufgabe 7 (20 Punkte):

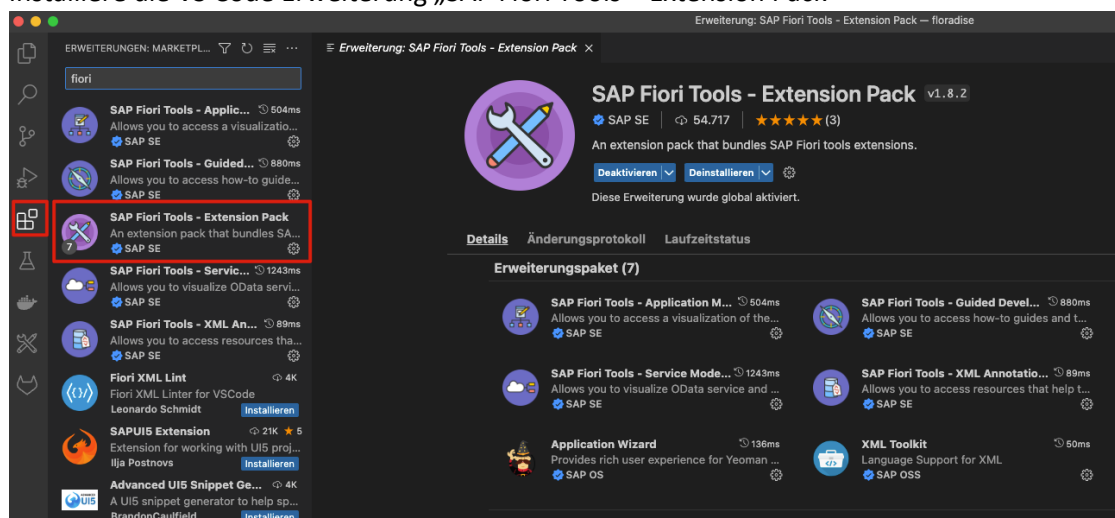
In dieser Aufgabe soll es darum gehen, den ersten Teil des Floradise-Straußkonfigurators zu entwickeln.

Entwicklungsumgebung einrichten

1. Richte Node.js ein
 - a. Lade node.js herunter und installiere es auf deinem Rechner: <https://nodejs.org/>

Tipp: Verwende die neueste Version. Zum Zeitpunkt der Erstellung dieser Aufgabe ist das v19.2.0.

2. Richte Visual Studio Code ein
 - a. Lade Visual Studio Code herunter und installiere es auf deinem Rechner: <https://code.visualstudio.com/download>
 - b. Installiere die VS Code Erweiterung „SAP Fiori Tools – Extension Pack“



3. Richte Git ein. Folgt den Installationsanweisungen auf <https://git-scm.com/>

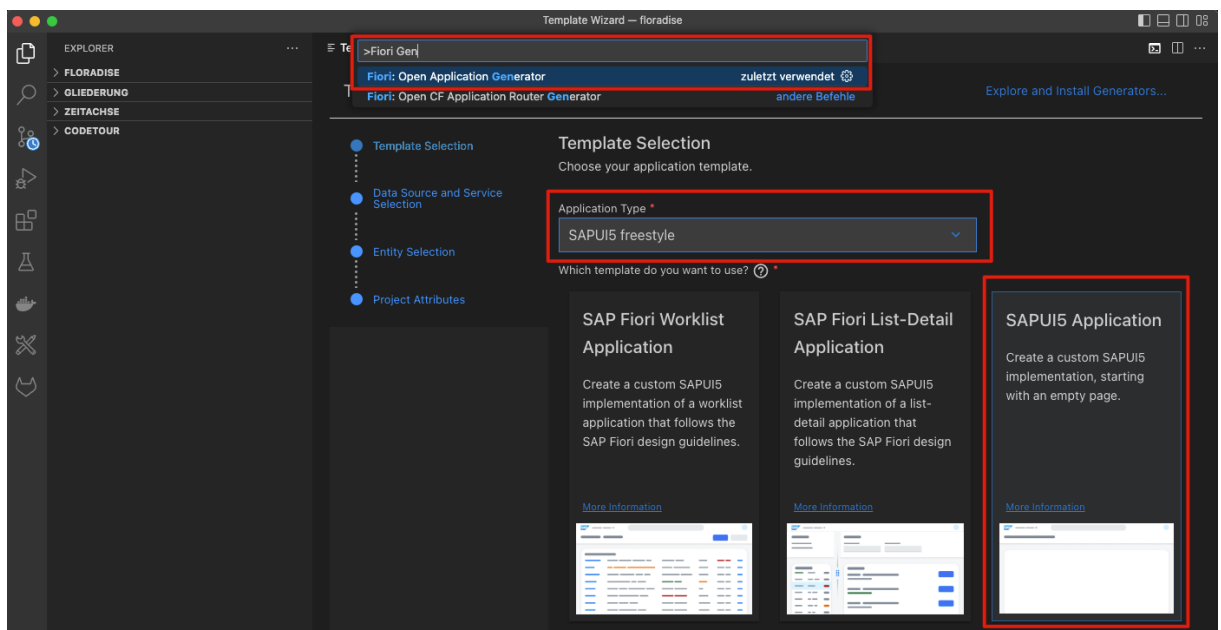


Floradise Straußkonfigurator entwickeln

1. Rufe in VS Code die Aktion „Fiori: Open Application Generator“ auf und wähle:

Application Type: SAPUI5 freestyle

Template: SAPUI5 Application



2. Als Datenquelle wählt:

Data source: Connect to an OData Service

OData service URL: <https://floradise.octavia.de/floradise>

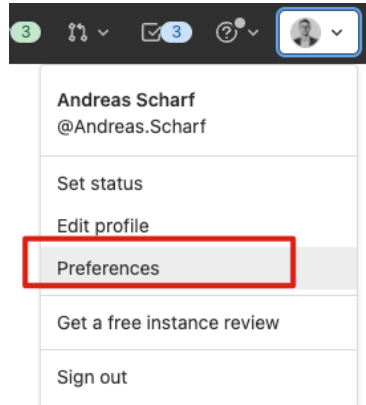
3. Wählt für alle weiteren Daten auf den nächsten Wizard-Seiten sinnvolle Werte wie in der Vorlesung gezeigt.

4. Implementiert den Floradise Straußkonfigurator so weit, wie in der Vorlesung gezeigt

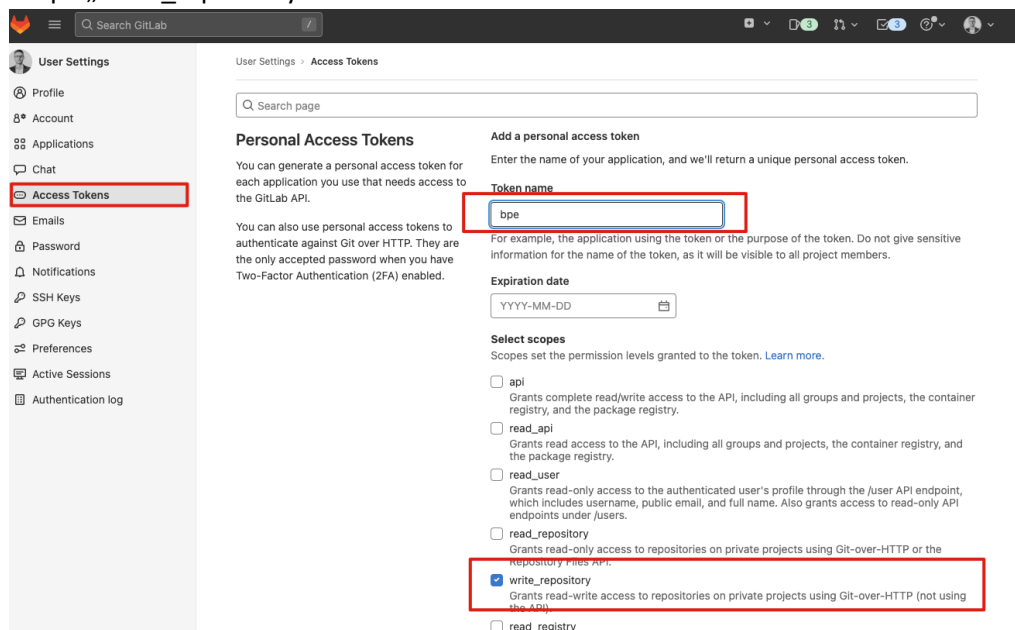
- a. Im Seitentitel soll die Anzahl der im Warenkorb befindlichen Artikel angezeigt werden. **WICHTIG: Es soll auf eine passende Variable zugegriffen werden und nicht einfach hard-coded eine „0“ verwendet werden.**
- b. Anzeige einer Liste aller Produkte
- c. Für jedes Produkt:
 - i. Ein Produkticon. Hier kann einfach das Icon „sap-icon://product“ verwendet werden.
 - ii. Produkttitel
 - iii. Produktfarbe. Falls das Produkt keine Farben besitzt, soll das UI5 Element nicht sichtbar sein.
 - iv. Produktbeschreibung. Die Produktbeschreibung soll sich ausklappen lassen.

Floradise Straußkonfigurator in Git einchecken

1. Loggt euch in dem euch zugewiesenen Git Repository auf <https://git.octavia.de> mit einem Webbrowser ein.
2. Generiert ein Access Token
 - a. Klickt auf euren Account → Preferences:



- b. Klickt auf „Access Tokens“. Wählt einen beliebigen „Token name“ und wählt als Scope „write_repository“



- c. Klickt auf „Create personal access token“

WICHTIG: Speichert dieses Token an einem sicheren Ort. Wenn die Seite geschlossen wird, gibt es keine Möglichkeit mehr dieses Token anzuzeigen.

3. Folgt den Anweisungen für
 - a. Git Global setup
 - b. Push an existing folder.



Command line instructions

You can also upload existing files from your computer using the instructions below.

Git global setup

```
git config --global user.name " "
git config --global user.email " "

```

Create a new repository

```
git init
git add .
git commit -m "Initial commit"

```

Push an existing folder

```
cd existing_folder
git init --initial-branch=main
git remote add origin https://git.octavia.de/uni-kassel/bpe-ws-22-23/<matr.>.git
git add .
git commit -m "Initial commit"
git push -u origin main

```

eure matrikelnummer

4. Fügt den Tag „aufgabe-7-1“ zu dem Abgabecommit hinzu und pushed diesen:

- a. `> git tag aufgabe-7-1`
- b. `> git push origin --tags`
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://git.octavia.de/uni-kassel/bpe-ws-22-23/<matrikelnummer>.git
* [new tag] aufgabe-7-1 -> aufgabe-7-1

Wichtig: Wir kontrollieren diesen Tag. Falls er falsch gesetzt ist (z.B. weil ihr danach noch Commits gemacht habt), werden die Änderungen nicht berücksichtigt. Falls der Tag nicht gesetzt ist, gibt es Punktabzug und wir kontrollieren HEAD.

Stellt sicher, dass sich die Anwendung korrekt starten lässt und die Daten wie in der Vorlesung gezeigt darstellt.

Abgabeformat: Checkt eure Abgaben im OctaVIA GIT ein und benutzt den Tag „Aufgabe-7“!

Deadline:

Montag, 12.12.2022, 14:00 Uhr

Donnerstag, 15.12.2022, 14:00 Uhr