



Leitfaden zur Veröffentlichung von Projektergebnissen

- Bereitstellung von Code auf GitHub -

Stand: 20.12.2021

Verfasser: Andreas Richter

1. Einleitung

In museum4punkt0 entstehen digitale Angebote für neue Arten des Lernens, Erlebens und Partizipierens im Museum, die auch anderen Museen in Deutschland zu Gute kommen sollen. Aus diesem Grund werden die Anwendungen soweit möglich quellcode-offen für eine flexible Nachnutzung bereitgestellt. Wir nutzen GitHub, um auch längerfristig eine freie und öffentliche Zugänglichkeit der Codes zu gewährleisten. Das hier vorliegende Dokument soll den Teilprojekten bei der Bereitstellung der Quellcodes und der Quelldateien (im folgenden Code genannt) auf GitHub helfen.

2. Bereitstellung digitaler Anwendungen

Ziel ist es, digitale Anwendungen, die im Rahmen von museum4punkt0 entwickelt werden, an zentraler Stelle online frei zugänglich und nachnutzbar bereitzustellen. Zu diesem Zweck greifen wir auf den Online-Dienst GitHub zurück, der zu den weltweit größten Plattformen für die Verteilung von Open Source Software zählt.¹ Für die Bereitstellung und gemeinsame Veröffentlichung von digitalen Anwendungen wurde auf GitHub ein gemeinsamer Publikationsraum, das Organisationskonto museum4punkt0, eingerichtet (<https://github.com/museum4punkt0>). In diesem Gemeinschaftskonto wird für jede Anwendung ein eigenes Repositorium erstellt, in dem die Codes hochgeladen, bereitgestellt und veröffentlicht werden.

Die Veröffentlichung von digitalen Anwendungen, die in den einzelnen Teilprojekten von museum4punkt0 entwickelt werden, ist Teil eines Workflows, der verbundweit die Bereitstellung der Codes und Quelldateien koordiniert und die Rahmenbedingungen für deren Nachnutzung schafft. Team Z koordiniert die Bereitstellung und sowie einen Code Review: Der Code Review wird von Dritten ausgeführt und überprüft, ob alle formalen Anforderungen für eine Nachnutzung des bereitgestellten Codes sowie der dazugehörigen technischen Dokumentation erfüllt sind. Die im Folgenden beschriebenen Empfehlungen und Anleitungen sollen bei der Bereitstellung der Codes helfen und auf die formalen Voraussetzungen für eine Nachnutzung der Codes hinweisen. Bei Detailfragen stimmen Sie sich bitte mit Team Z (Andreas Richter) ab.

2.1. Bereitstellung auf GitHub

Folgende Tipps sollen Ihnen in Vorbereitung auf die Bereitstellung von Code und den Code Review helfen.

Klärung des Datenvolumens

Die Nutzung von GitHub für die Ablage und Distribution von Code unterliegt bestimmten Beschränkungen. Da aufgrund der zeitlich befristeten Förderungsdauer von museum4punkt0 dem Verbund ausschließlich die kostenfreie Nutzung von GitHub zur Disposition steht, sind für die Bereitstellung zwei Einschränkungen zu beachten:

¹ In Einzelfällen werden digitale Anwendungen auch auf Plattformen wie App Store, Google Play, Steam, o.ä.. bereitgestellt, oder der Code wird lokal abgelegt und auf Anfrage bereitgestellt. Auf diese Sonderform der Bereitstellung muss dann in der Publikation hingewiesen werden.

- 1) Das Gesamtvolumen eines Repositoriums überschreitet nicht die Größe von 1 GB und
- 2) Einzeldateien sollten nicht größer als 50 MB sein.

Der Upload von Dateien, die größer als 100 MB sind, wird blockiert. Jedoch können größere Dateien mit einem Gesamtvolumen bis zu 1GB beim Upload zusätzlich in ein Large File Storage (LFS) ausgelagert werden. Sollte eine der beiden Bedingungen nicht erfüllt werden können, wenden Sie sich bitte an Team Z, um für den Einzelfall eine geeignete Alternative für die Bereitstellung der Codes/Quelldateien zu verabreden.

Klärung der Lizenzierung/Wahrung der Rechte Dritter/Umgang mit Assets

Alle von museum4punkt0 geförderten Software-Entwicklungen sind so weit möglich unter eine Open Source Lizenz zu stellen; mögliche Lizenztypen sind: GNU GPL (General Public License), MIT License, BSD-Lizenz, CPL (Common Public License), EUPL (European Public License) oder vergleichbare. Falls lizenzpflichtige Teilkomponenten im Gesamtwerk einbezogen wurden, ist der durch museum4punkt0 finanzierte Code als Open Source ohne die lizenzpflichtigen Komponenten bereitzustellen und in der Dokumentation darauf entsprechend hinzuweisen. Zusätzlich sollte in der Dokumentation aufgezeigt werden, wie und unter welchen Voraussetzungen der bereitgestellte Code nachgenutzt werden kann.

Sollte ein komplettes Produkt aus rechtlichen Gründen nicht als Open Source zu veröffentlichen sein, sollte alternativ zur Bereitstellung auf GitHub dennoch ein konkretes Nachnutzungsangebot unterbreitet werden.

Die Dokumentation zum Code

Ziel der Dokumentation ist es, die digitale Anwendung (einschließlich ihrer Schnittstellen) in geeigneter Weise verständlich und nachvollziehbar auf Deutsch oder Englisch zu beschreiben. Die Dokumentation soll andere Museen und ihre Dienstleister in die Lage versetzen, die Struktur des Systems, den Ablauf der Datenverarbeitung, die Funktionen der einzelnen Einheiten und die Bedeutung von verwendeten Daten zu verstehen.

Kommentierung im Quellcode

Die code-interne Dokumentation soll durch Annotationen im Quelltext eine Übersicht über die Struktur und den Aufbau des Codes sowie eine Nachvollziehbarkeit des Datenflusses und der Verarbeitung für Dritte gewährleisten. Darüber hinaus können Stellen im Code, die auf Assets verweisen, die für eine Nachnutzung nicht von Bedeutung sind oder aufgrund lizenzrelevanter Beschränkungen von der Veröffentlichung ausgeschlossen sind, auskommentiert und mit einem entsprechenden Kommentar und/oder einem Link zum Asset oder Rechteinhaber versehen werden. Der Einsatz von Platzhaltern ist auch möglich. Die Dateien von Assets, zu denen Sie nicht das Recht zur Weitergabe haben, müssen aus den Ablageordnern zum Code gelöscht werden.

Die Readme-Datei

Die Readme-Datei ist der Teil des Repositoriums, der beim Öffnen des Repositoriums den ersten Einstieg und einen Überblick über alle wesentlichen Informationen bietet.

Der formale Aufbau der Readme-Datei ist Teil des Code Reviews und sollte mindestens folgende Abschnitte in der angegebenen Reihenfolge enthalten:

1. Name der Anwendung

Die Bezeichnung entspricht dem Titel der Anwendung auf der Website/Publikation. Bitte achten Sie auf eine einheitliche Terminologie.

Inhaltsverzeichnis (optional)

Ermöglichen Sie den Leser*innen eine Übersicht und einen Schnelleinstieg in die folgenden Kapitel der Readme-Datei. Hinweise zur Erstellung eines Inhaltsverzeichnisses finden Sie am Ende des Kapitels.

2. Kurzbeschreibung

Beschreiben Sie in mindestens drei Sätzen, worum es in der hinterlegten Anwendung geht, z. B.: „Tromp L'Oeil“ ist eine prototypische mobile Augmented-Reality-App für die Sammlungsstücke der Gemäldegalerie der Staatlichen Museen zu Berlin. Mit dieser Augmented-Reality-Anwendung werden Potentiale untersucht, die AR-Technologie zur Erforschung, Darstellung, Erzählung und Vermittlung von kulturellen Inhalten bietet. [...]

Entstehungskontext & Förderhinweis

Textbaustein:

Diese [Art der Webanwendung angeben, z.B. Website, Progressive-Web-App] ist entstanden im Verbundprojekt museum4punkt0 – Digitale Strategien für das Museum der Zukunft, Teilprojekt [Teilprojektname einfügen gemäß Verbund-Website, Kurzform möglich]. Das Projekt museum4punkt0 wird gefördert durch die Beauftragte der Bundesregierung für Kultur und Medien aufgrund eines Beschlusses des Deutschen Bundestages. Weitere Informationen: www.museum4punkt0.de

Falls benötigt – Englische Variante:

This [digital application] is part of the project museum4punkt0 - Digital Strategies for the Museum of the Future, sub-project [englischer Teilprojektname einfügen, gemäß museum4punkt0-Website]. The project museum4punkt0 is funded by the Federal Government Commissioner for Culture and the

Media in accordance with a resolution issued by the German Bundestag (Parliament of the Federal Republic of Germany). Further information: www.museum4punkt0.de

Zusätzlich muss im Rahmen der Projektförderung aus dem Rettungs- und Sofortprogramm „Neustart Kultur“ das entsprechende Logo eingebunden werden. Daher bietet sich an, gleich beide Logos von BKM am Ende des Förderhinweises zu platzieren.

Die beiden Logos sind auf GitHub verfügbar unter:

https://github.com/museum4punkt0/images/blob/2c46af6cb625a2560f39b01ecb8c4c360733811c/BKM_Fz_2017_Web_de.gif

https://github.com/museum4punkt0/media_storage/blob/a35eedb36e5b502e90cd76d669a6b337002b230a/BKM_Neustart_Kultur_Wortmarke_pos_RGB_RZ_web.jpg

Dort finden Sie auch einen Code-Snippet, mit dem die beiden Logos aus der Readme heraus verlinkt werden können.

3. Installation

Geben Sie hier Informationen zur Installation und der dazu verwendeten/benötigten Hard- & Software.

4. Benutzung

Beschreiben Sie die Benutzung der Anwendung nach der Installation. Fügen Sie bestenfalls Screenshots ein, um die Handhabung zu illustrieren. Auch Anforderungen an Geräte / Ausstattung bzw. Systemvoraussetzungen für den Betrieb sollten hier genannt sein. Falls Ihre Anwendung über ein Backend verfügt, sollte Ihnen ein Benutzerhandbuch für die Backend-Nutzung vorliegen. Stellen Sie dieses bitte auch zur Verfügung, indem Sie es (als PDF/A-3) im GitHub-Repositorium hochladen und in der Readme-Datei darauf verlinken. Liegt zudem ein Betriebskonzept vor, sollten Sie dieses von hier aus per Link zugänglich machen.

5. Beteiligung (optional)

Geben Sie hier an, ob und in welcher Art man sich an Ihrer Programmierung beteiligen kann. Detailinformationen werden in einer separaten Datei angelegt. Weitere Hinweise hierzu finden Sie unter: <https://help.github.com/en/articles/setting-guidelines-for-repository-contributors>

6. Credits

Auftraggeber*innen/Rechteinhaber*innen (Name Ihrer Institution), Urheber*innen (Namen der beteiligten Entwickler*innen oder Firma).

Die Autor*innen des Codes bzw. zugehörige Agenturen / Firmen / Institutionen können hier verlinkt sein.

7. Lizenz

Rechteangabe nach dem Schema: Copyright © Jahr, Institution / Urheber*innen.

Geben Sie an, unter welcher Lizenz der Quellcode / die Anwendung steht und verlinken Sie von hier zur Lizenz-Datei, die ebenfalls im Repositorium abgelegt sein sollte. Der Lizenztext kann hier auch komplett wiedergegeben werden.

Bitte listen Sie hier auch gegebenenfalls wiederverwendete Programmteile Dritter mit deren Lizenzen auf.

Sonstiges

Um die Lesbarkeit der Readme-Datei zu verbessern, nutzen Sie für die Formatierung des Textes bitte folgende Markdown-Elemente.

Legen Sie die o.g. Abschnitte in Ihrer Readme-Datei mit Überschriften an. Formatieren Sie den Titel als Hauptüberschrift mit einem #(+Leerzeichen) und alle weiteren untergeordneten Überschriften mit zwei oder drei #(+Leerzeichen).

Erstellen Sie zu Beginn der Readme-Datei ein Inhaltsverzeichnis mit Sprungmarken zu den einzelnen Abschnitten. Bspw.:

```
## Inhaltsverzeichnis

* [Kurzbeschreibung](#Kurzbeschreibung)

* [Förderhinweis](#Förderhinweis)

* [Installation](#Installation)

* [Benutzung](#Benutzung)

* [Credits](#Credits)

* [Lizenz](#Lizenz)
```

Weitere Hinweise zur Formatierung des Textes finden Sie hier:

<https://guides.github.com/features/mastering-markdown/>

Weitere Hilfestellungen zur Anfertigung einer Readme-Datei und zur Codedokumentation finden Sie hier: <https://guides.github.com/features/wikis/#creating-a-readme>

Zusätzliche Angaben

„About“: Geben Sie im Abschnitt „About“ eine kurze, prägnante Auskunft zum Inhalt des Repositoriums.

„Topics“: Taggen Sie das Repository mit Schlagworten, um die Auffindbarkeit Ihres Beitrags zu verbessern.

Die Übertragung des Codes

Auf Anfrage durch das TP wird von Team Z für die Bereitstellung des Codes ein Repository im Gemeinschaftskonto auf GitHub angelegt.

Die Bereitstellung des Codes in einem Repository auf GitHub kann auf unterschiedlichem Wege erfolgen. In den meisten Fällen werden die Entwickler*innen selbst den Code, gemeinsam mit einer Readme- und Lizenzdatei, in ein Repository übertragen. Projektpartner, die den Code auf GitHub selbst bereitstellen, können bei Bedarf eine von Team Z erstellte Anleitung, die detailliert die einzelnen Schritte zur Übertragung des Codes beschreibt, zurückgreifen.

Bei Fragen wenden Sie sich bitte an Andreas Richter (a.richter@smb.spk-berlin.de).