

</> 1 Software fertigstellen

Stellt eure Software fertig. Konzentriert euch dabei auf die Tasks, die zum Spielen unbedingt notwendig sind. Schaut auch in eure Issues und behebt die Fehler, die beim Spielen Probleme verursachen.

Wenn ihr am Ende noch Zeit haben solltet, könnt ihr noch weitere Tasks und Issues angehen.

</> 2 Quellcode für die Veröffentlichung vorbereiten

Softwareprojekte müssen gepflegt und instand gehalten werden. Dafür ist es wichtig, dass man auch nach Wochen noch versteht, was die Software eigentlich tut. **Kommentiert daher euren Code möglichst aussagekräftig.** Als Faustregel sollte mindestens jede Funktion kommentiert werden.

Für zukünftige Anwender ist es wichtig zu wissen, wie sie die Software verwenden können. **Ergänzt daher eure README-Datei** um einen Abschnitt zu Installation bzw. Ausführung. In den meisten Fällen sollte dafür `npm install` und anschließend `npm start` ausreichen.

Für zukünftige Entwickler ist es wichtig zu wissen, was der Plan für die Erstellung eurer Software war. **Falls ihr UML-Diagramme o.ä. habt, fügt sie eurem Repo im docs-Ordner hinzu.** Überlegt euch außerdem, ob ihr den Quellcode eurer Software für die Bearbeitung durch andere freigeben möchtet. Dabei hilft die Vergabe einer Lizenz.



Open Source

Bei Software kann man sehr vereinfacht zwischen *Open Source* und *Proprietärer Software* unterscheiden. Bei Open Source Software geht es darum, dass die Software inkl. des verwendeten Quellcodes zur Verfügung gestellt wird und so jeder die Möglichkeit hat, an dem Projekt mitzuarbeiten und es weiterzuentwickeln. Proprietäre Software darf in der Regel nicht beliebig angepasst oder weiterverwendet werden.

Die *Open Source Initiative* hat zehn Kriterien aufgestellt, die Software erfüllen muss, um Open Source zu sein:

1. **Freie Weitergabe**
Niemanden darf daran gehindert werden, die Software zu verkaufen oder diese zusammen mit anderer Software weiterzugeben.
2. **Verfügbarer Quelltext**
Der Quelltext muss in einer menschenlesbaren Form vorliegen.
3. **Abgeleitete Arbeiten**
Änderungen und abgeleitete Werke müssen unter derselben Lizenz verbreitet werden können.
4. **Integrität des Autoren-Quelltexts**
Software, die aus geändertem Quellcode erstellt wurde, muss verteilt werden dürfen.
5. **Keine Diskriminierung von Personen oder Gruppen**
Einzelne Personen oder Personengruppen darf die Nutzung der Software nicht verweigert werden.
6. **Keine Nutzungseinschränkungen**
Der Verwendungszweck der Software darf nicht eingeschränkt werden.
7. **Lizenzerteilung**
Die verbundenen Rechte müssen für alle gelten, an die Software weitergegeben wird.
8. **Produktneutralität**
Die mit dem Programm verbundenen Rechte dürfen nicht davon abhängen, ob die Software Teil eines bestimmten Softwarepakets ist,
9. **Keine Softwareeinschränkungen**
Die Lizenz darf keine Einschränkungen für andere Software, die mit der lizenzierten Software verteilt wird, machen.
10. **Technologieneutralität**
Die Lizenz darf nicht auf einer einzelnen Technologie beruhen.



3 Eine Lizenz vergeben

Wenn ihr eure Software veröffentlichen wollt, solltet ihr eine entsprechende Lizenz wählen. Falls ihr euch für eine Open-Source-Lizenz entscheidet, findet ihr unter choosealicense.com eine Entscheidungshilfe für eine passende Lizenz. Im Video seht ihr, wie ihr eurem Repo eine Lizenz hinzufügen und eure eigenen Repos veröffentlichen könnt.



4 Launch vorbereiten

Bei der Abschlussveranstaltung präsentiert und demonstriert jede Gruppe ihre Software. Bereitet dafür euren Launch vor. Ihr solltet dabei die folgenden Fragen beantworten:

- ☐ Wie heißt eure Software?
- ☐ Welche Idee steckt hinter eurem Produkt?
- ☐ Was macht euer Projekt besonders?
- ☐ Was waren die größten Herausforderungen?
- ☐ Wie hat die Entwicklung geklappt?



5 Feedback

Das Science Camp Softwareentwicklung neigt sich dem Ende zu. Wie hat dir das Science Camp gefallen? Was können wir besser machen? Fülle den Fragebogen aus und gib uns damit wichtiges Feedback.

