# Technologieentscheidung: Flask und ProstgresSQL

Wir haben uns nach ausführlichen Überlegen und Abwägen aus folgenden Gründen für Flask entschieden:

* Dash von Plotly funktioniert über Flask. Eine Anbindung an Django wäre komplizierter und eine wirklich so interaktive Alternative ist uns nicht bekannt. Selbstverständlich könnten wir mit Hilfe von JavaScript, Python etc. eine ähnlich interaktive Plattform aufbauen, doch das wäre fehleranfälliger, schlechter wartbar, schlechter testbar und es ist fragwürdig, ob es wirklich so gut funktionieren würde wie Dash.
* Flask fokussiert vom Hause aus Erweiterbarkeit und Dokumentation. Durch die einfachere Handhabung als mit Django wird uns dies leichter fallen. Es wäre daher auch unproblematisch PostgresSql oder eine andere gängige Datenbank anzuschließen.
* Man kann bestehende Bibliotheken ohne bestimmte Komponenten einfach integrieren, wodurch die Flask Anwendung simpel und klein bleibt.
* Recherchen ergaben, dass Flask einfach an Server wie Heroku oder Gunicorn anzuschließen ist.
* Django wäre für diesen Zweck zu groß. Natürlich kann man alles mit Django umsetzen, man kann sogar über Tricks Dash darüber laufen lassen, jedoch wäre es für diese Anwendung zu komplex, wenn man es auch mit Flask lösen kann. Bei Django muss man auf einiges mehr achten, beispielsweise, dass CSS Code in einen Static Ordner muss oder dass man die installierten Apps immer aktualisiert. Wie bereits erwähnt, wäre es möglich, aber nicht so schön.

Für uns sind die Hauptgründe, dass wir uns für Flask entschieden haben die bessere Wartbarkeit, Testbarkeit und dass es minimal ist, aber trotzdem erweiterbar. Wegen Dash brauchen wir für die Graphen an sich kein JavaScript mehr wirklich extra, wo durch, wie erwähnt, Fehlerquellen vermieden werden und der Code übersichtlicher gehalten wird.

Als Datenbank haben wir PostgresSQL gewählt, weil wir damit eine bessere Interoperabilität sicherstellen können, als mit MariaDB oder MySql, bei denen man die sämtliche Daten aufbereiten müsste. Selbstverständlich ist uns bekannt, dass PostgresSQL etwas langsamer ist, als MariaDB oder MySQL, was jedoch oftmals an der Implementierung liegt. Dazu ist PostgresSQL auf jeden Fall Open Source, wird stets weiterentwickelt von der Community, ist Plattformunabhängig und hat Security- Features. Dazu wird diese Datenbank wie MySQL hauptsächlich für dynamische Webseiten eingesetzt, weshalb es sich erst Recht für dieses Projekt anbieten sollte die bereits vorhandene Datenbank zu nehmen.