**Innerhalb dieser Projetarbeit:**

**WebGL** (JS Graphik Renderer/engine) -> wird nur indirekt genutz, man kommt als Programmierer so gut wie nicht damit in Kontakt

**Three.js** (nutzt WebGL) -> Bibliothek, die es Erlaubt die WebGL Funktionen effektiver zu benutzen

**Node.js** (Bibliotheken und der Liveserver) -> Laufzeitumgebung und LiveServer, Ausführung von JS ohne Browser auf Server oder Befehlszeile; kommt mit Paketmanager; verarbeitung großer datenmengen durch skalierbarkeit dank Asynchronem Verhalten -> für effiziente und Leistungsfähige Platformen -->Dank node.js keine Probleme mit CORS (siehe oben), nachteil: muss installiert sein, Laufzeitumgebung mit Compiler bzw. Transpier für react und typescript in JavaScript

**React.js** (unteranderem API für Three.js \ genutzt für bspw. Facebook ..) -> Bibliotheken und vereinfachtes entwckeln von Benutzeroberflächen, gut struckturierter Code, ist komponentenbaiert, HTML-Elemente im JS oder TS code, (auch für mobile apps), Wiederverwendbarkeit von Kompnenten, Drittanbieterpakete(Three.js), Fehler reduktion, Laufzeit aktualisierungen (angenehmer währed des Entwickelns)

**Typescript** [TS] Erweiterung von JavaScript, erleichtert Fehlersuche, vollständige unterstützung von JS, statische Typisierung und angenehmere Entwicklungserfahrung, stellt interfaces bereit(Vorraussetzungen einer Klasse an eigenschaften und methoden) ist eher gedacht zum überprüfen und regeln festlegen- taucht im js-code dann auch nicht mehr auf,