**Handout: Electron.js**

**1. Was ist Electron?**

Electron ist ein Open-Source-Framework, das es Entwicklern ermöglicht, plattformübergreifende Desktop-Anwendungen mit Webtechnologien wie HTML, CSS und JavaScript zu erstellen.

Die Code-Editoren "Visual Studio Code" und "Atom" wurden mit Electron entwickelt.

**2. Wofür ist es gut?**

Electron ermöglicht es, eine einzige Codebasis für Windows, macOS und Linux zu verwenden, anstatt separate Anwendungen für jede Plattform zu programmieren.

Ein Unternehmen kann eine interne Verwaltungssoftware für alle Betriebssysteme bereitstellen, ohne mehrere Versionen pflegen zu müssen.

**3. Wo wird es benutzt?**

Electron wird in vielen bekannten Anwendungen verwendet.

**Beispiele:**

* **Discord** (VoIP- und Messaging-App)
* **Slack** (Teamkommunikation)
* **Trello** (Projektmanagement)
* **Skype** (Videotelefonie)

**4. Wann wird es gebraucht und wofür?**

Electron eignet sich besonders, wenn Webentwickler ihre Fähigkeiten auf Desktop-Apps ausweiten möchten oder wenn eine Anwendung mit vielen Webtechnologien erstellt werden soll.

**Beispiel:** Ein Startup möchte eine App entwickeln, die sowohl als Web-App als auch als Desktop-App funktioniert.

**5. Was macht es?**

Electron kombiniert die Rendering-Funktionen von Chromium (dem Chrome-Browser) mit der Node.js-Laufzeit, um native Betriebssystemfunktionen zu nutzen.

**6. Wie erstellt man Dateien?**

Eine Electron-App besteht aus zwei Hauptprozessen:

* **Main-Prozess** (steuert das Verhalten der App)
* **Renderer-Prozess** (zeigt die Benutzeroberfläche an)

**Schritt-für-Schritt Beispiel:**

1. **Electron installieren:** npm install -g electron
2. **Projektordner erstellen und initialisieren:** mkdir meine-app && cd meine-app && npm init -y
3. **Hauptdatei erstellen:** touch main.js
4. **Ein einfaches Fenster mit Electron:**

const { app, BrowserWindow } = require('electron');

let mainWindow;

app.whenReady().then(() => {

 mainWindow = new BrowserWindow({ width: 800, height: 600 });

 mainWindow.loadURL('https://www.example.com');

});

1. **App starten:** npx electron --no-sandbox

**7. Besonderheiten von Electron**

* **Cross-Plattform:** Eine Codebasis für mehrere Betriebssysteme.
* **Einfache Webtechnologien:** HTML, CSS, JavaScript.
* **Erweiterbar:** Viele Plugins und Module verfügbar.
* **Ressourcenverbrauch:** Electron-Apps können speicherintensiv sein.

Ein Entwickler kann native Funktionen wie Datei-Uploads, Benachrichtigungen oder Systemzugriffe einfach mit Node.js-Modulen integrieren.

**8. Beispiele aus der Präsentation**

**Main Prozess**



**Beispiel – Netzwerkstatus**







**Renderer Prozess**





Electron ist ein mächtiges Tool zur Entwicklung von Desktop-Apps mit Webtechnologien. Es ist einfach zu erlernen und ermöglicht schnelle, plattformübergreifende Entwicklung.