

# Content-Verwertungsmodelle und ihre Umsetzung in mobilen Systemen

## Programmierprojekt

Vorlesung im Sommersemester an der Technischen Universität Ilmenau



von  
Privatdozent Dr.-Ing. habil. Jürgen Nützel  
Vorstand der  
deenovum Technologies AG  
jn (at) deenovum (dot) com

*Diese Folien und weitere Informationen unter:*  
[www.juergen-nuetzel.de/content\\_verwertungsmodelle\\_mobile\\_umsetzung.html](http://www.juergen-nuetzel.de/content_verwertungsmodelle_mobile_umsetzung.html)

# Aufbau der Veranstaltung

- Vorlesung (2 SWS)
  - Ich berichte über Sachen, die ich für wichtig halte
  - 2026 werde ich für die KI-Nutzung alles auf den Kopf stellen und neu machen.
- Programmierprojekt mit Übung (2 SWS)
  - Sie erstellen Ihre eigene App und stellen diese bei  Google play online, oder im Apple AppStore
- Ergibt 5 Leistungspunkte (LP)
- Alternative Prüfungsleistung
  - Schriftliche Klausur zur Vorlesung mit Note
  - Note wird durch Testat (vor der Klausur) auf Programmierprojekt freigeschaltet

# Anforderungen an die App

- Native oder hybride App, die über einen App-Store veröffentlicht werden kann.

**Keine Web-App!**

- App muss mit Content, der über das Internet geladen wird, arbeiten. Bzw. einen Cloud-Dienst nutzt.

**Keine reine Offline-App.**

- App muss die Nutzer-Mobilität über min. einen der Gerätesensoren verarbeiten.
- Der Quellcode muss abgegeben werden.
- Eine Beschreibung der App-Funktionalität muss mitgeliefert werden ...

# Beschreibung der App-Funktionalität

- Geben Sie eine umgangssprachliche kurze Funktionsbeschreibung Ihrer App an.
- Beschreiben Sie Ihre typischen App-Nutzer.  
Geben Sie min. eine Persona an:
  - [https://de.wikipedia.org/wiki/Persona\\_\(Mensch-Computer-Interaktion\)](https://de.wikipedia.org/wiki/Persona_(Mensch-Computer-Interaktion))
- Definieren Sie die wichtigsten User-Stories, die Ihre App umsetzen soll:
  - <https://de.wikipedia.org/wiki/User-Story>
- Beschreiben Sie Zielplattform (Betriebssystem), Endwicklungsumgebung und genutzte externe Systeme (wo kommt der Content her)
- Umfang: 1-2 DIN-A4-Seiten
- Termin (Dead-Line): 1. Juni (Per E-Mail als Word oder PDF)

# Ich fragte Claude zum Programmierprojekt

*“Ich möchte mit deiner Hilfe eine mobile App für Android und iOS entwickeln. Welches Framework mit welcher Programmiersprache ist aus deiner Kenntnis am besten geeignet, bzw. am besten von Dir unterstützt”*

## Empfohlen

### Flutter

Dart · Google

#### STÄRKEN

Native Performance Einheitliches UI  
Hot Reload Web & Desktop

#### SCHWÄCHEN

Dart lernen Große App-Größe

#### KI-UNTERSTÜTZUNG

Sehr gut 90%

## Sehr beliebt

### React Native

JavaScript / TypeScript · Meta

#### STÄRKEN

Riesiges Ökosystem JS-Kenntnisse nutzen  
Viele Bibliotheken

#### SCHWÄCHEN

JS-Bridge-Overhead Debug-Komplexität

#### KI-UNTERSTÜTZUNG

Exzellent 92%

## Wachsend

### Kotlin Multiplatform

Kotlin · JetBrains

#### STÄRKEN

Geteilte Logik Native UI möglich  
Kotlin-Qualität

#### SCHWÄCHEN

Noch unreif Steile Lernkurve

#### KI-UNTERSTÜTZUNG

Gut 72%

## Einsteigerfreundlich

### Expo (React Native)

JavaScript / TypeScript

#### STÄRKEN

Sehr schneller Start OTA-Updates  
Managed Workflow

#### SCHWÄCHEN

Weniger Kontrolle  
Native Module eingeschränkt

#### KI-UNTERSTÜTZUNG

Sehr gut 88%

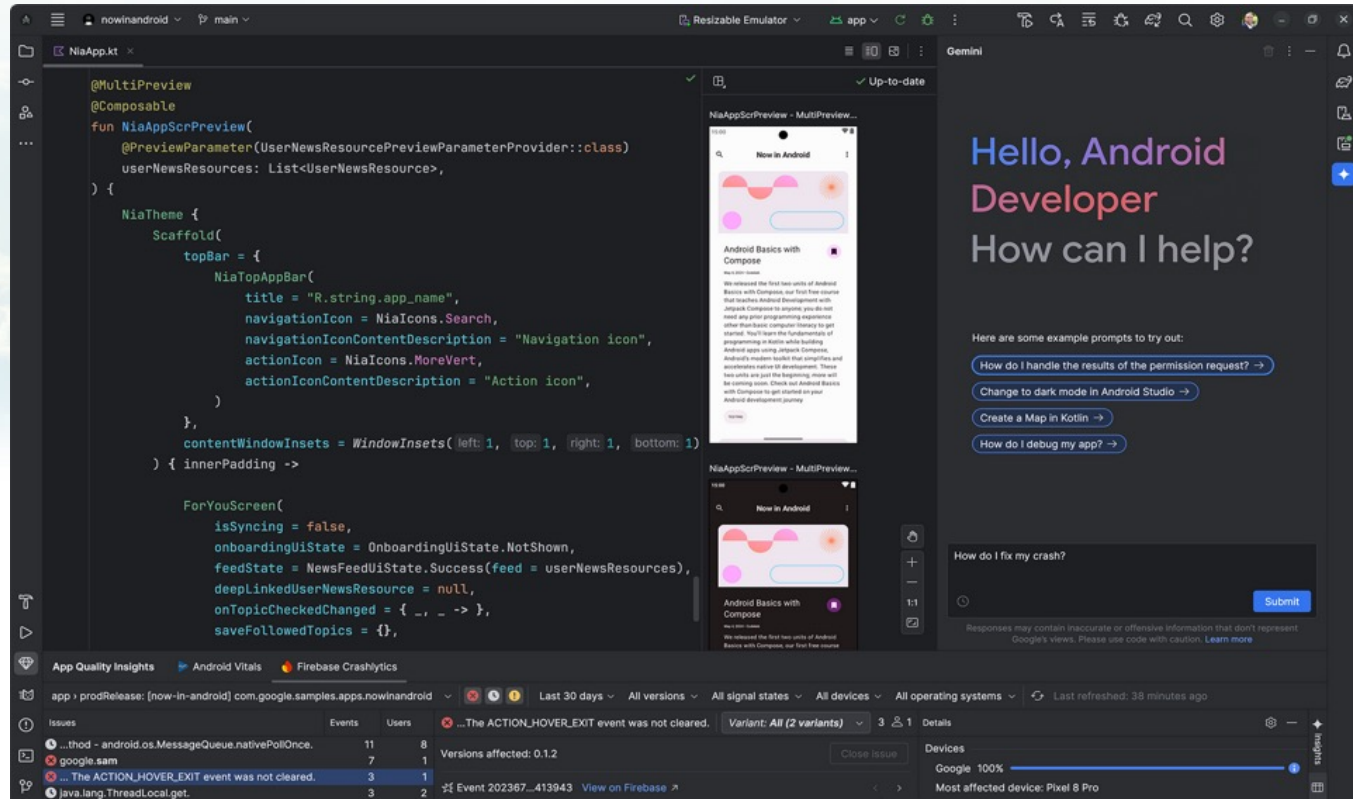
# Und ChatGPT 5.2 meint dazu:

- *Meine klare Empfehlung also:*
  - *Standardempfehlung: React Native + Expo + TypeScript*
  - *Wenn Design/UI absolut im Vordergrund steht: Flutter + Dart*
  - *Wenn native UI und langfristige Plattformtiefe am wichtigsten sind: Kotlin Multiplatform*
- *Wenn du mit mir gemeinsam starten willst, würde ich React Native + Expo + TypeScript nehmen. Als nächstes wäre sinnvoll, direkt die Grundarchitektur für eure App festzulegen: Login, Navigation, Backend-Anbindung, Datenmodell und Release-Setup für iOS/Android.*

# Programmierprojekt z.B. mit

- Android Studio <https://developer.android.com>

- Native Apps für Android mit Kotlin oder Java

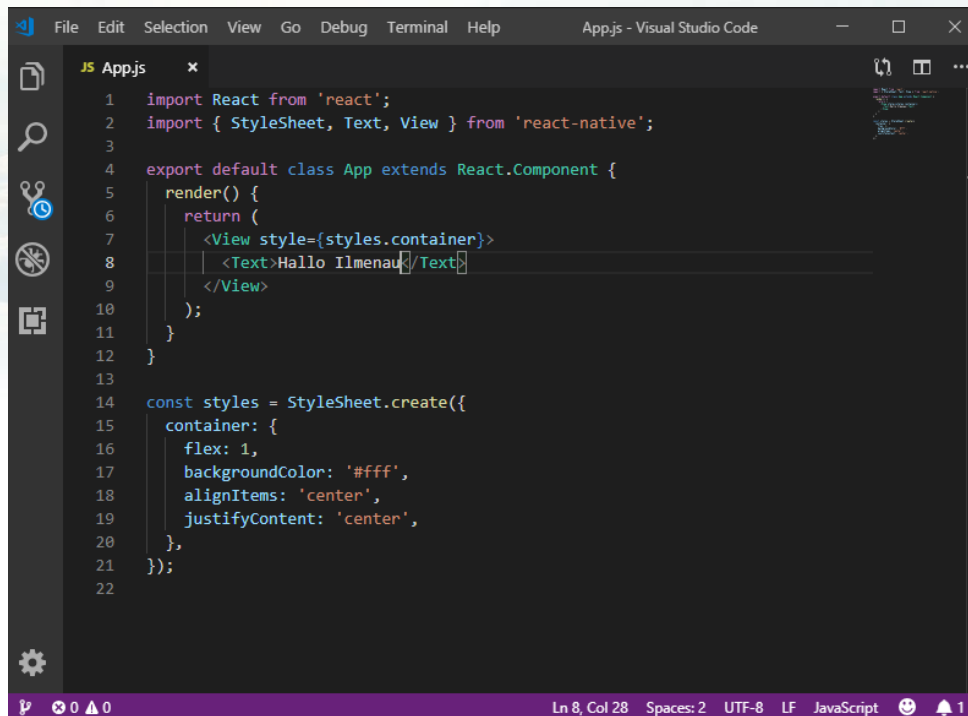


Android Studio Meerkat

# Programmierprojekt z.B. mit

- **React Native** <https://reactnative.dev/>

Native Apps für Android, iOS programmiert in JavaScript mit  
Visual Studio Code



```
1 import React from 'react';
2 import { StyleSheet, Text, View } from 'react-native';
3
4 export default class App extends React.Component {
5   render() {
6     return (
7       <View style={styles.container}>
8         <Text>Hallo Ilmenau</Text>
9       </View>
10     );
11   }
12 }
13
14 const styles = StyleSheet.create({
15   container: {
16     flex: 1,
17     backgroundColor: '#fff',
18     alignItems: 'center',
19     justifyContent: 'center',
20   },
21 });
22
```

**Einfaches Testen mit Expo**

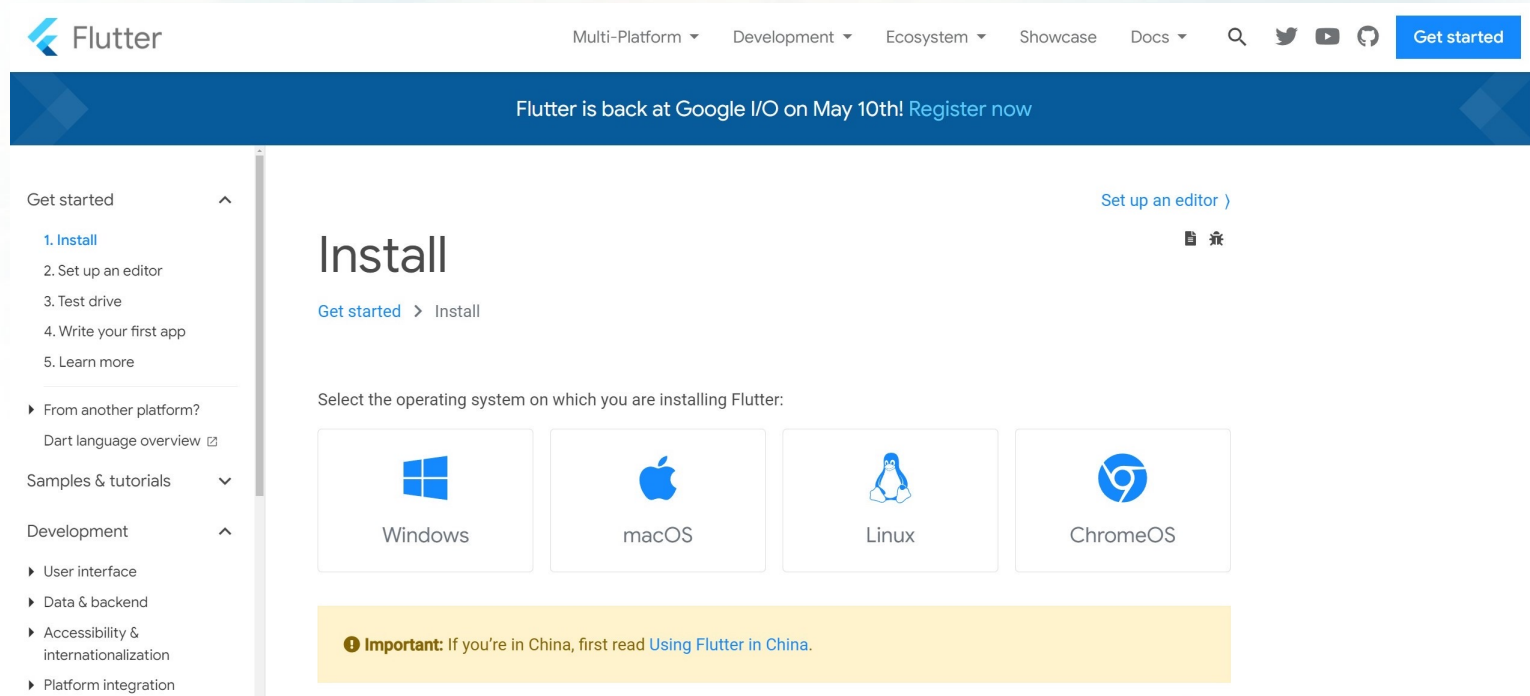
<https://expo.dev>

# Programmierprojekt z.B. mit

- Flutter: <https://flutter.dev/>

Native Apps für Android und iOS. Und weitere Plattformen.

Installation: <https://docs.flutter.dev/get-started/install>



# Server-Part des Programmierprojekts

- Oder Google Firebase

- <https://firebase.google.com/>

- Authentication

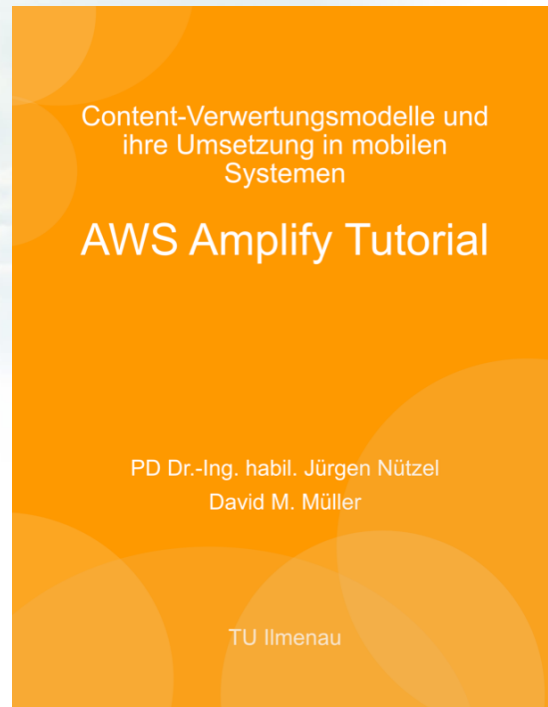
- <https://firebase.google.com/docs/auth>

- Database: Cloud Firestore or Realtime Database

- <https://firebase.google.com/docs/database/rtdb-vs-firestore>

# Flutter mit AWS Amplify

- Geschenke App in Flutter mit Cloud-Backend
- Nutzen von Amazons Cloud-Diensten durch Amplify
  - AWS Amplify: <https://docs.amplify.aws/>
  - Tutorial: [www.juergen-nuetzel.de/AWS\\_Amplify\\_Tutorial.pdf](http://www.juergen-nuetzel.de/AWS_Amplify_Tutorial.pdf)
  - Sample Code:  
[https://github.com/damamu888/gift\\_app-v.1](https://github.com/damamu888/gift_app-v.1)  
[https://github.com/damamu888/gift\\_app-v.2](https://github.com/damamu888/gift_app-v.2)  
[https://github.com/damamu888/gift\\_app-v.3](https://github.com/damamu888/gift_app-v.3)



# Weitere Infos und Kontakt



Privatdozent Dr.-Ing. habil. Jürgen Nützel

jn (at) deenovum (dot) com

[www.juergen-nuetzel.de](http://www.juergen-nuetzel.de)

[www.deenovum.com](http://www.deenovum.com)