

Host: Herzlich willkommen zu unserer heutigen Podcast-Folge auf unserem virtuellen Studio. Ich bin Anna und mit mir ist heute Finn. Finn, wie geht es dir?

Guest: Hallo Anna, mir geht es wunderbar, danke! Ich freue mich darauf, mit dir heute ein wenig über die Schichtenarchitektur in der Softwareentwicklung zu sprechen, besonders im Kontext von React Native.

Host: Das klingt spannend! Beginnen wir vielleicht mit der Frage: Was genau ist Schichtenarchitektur?

Guest: Gerne! Die Schichtenarchitektur ist im Grunde eine Methode in der Softwareentwicklung, bei der eine Anwendung in verschiedene logische Schichten unterteilt wird. Jede dieser Schichten hat eine klar definierte Aufgabe, was die gesamte Software übersichtlicher, flexibler und einfacher wartbar macht. Ein gutes Beispiel, das in unserem Material erwähnt wird, ist eine Wetter-App.

Host: Ah, das macht Sinn. Also könnte man sagen, dass jede Schicht wie ein eigenes kleines Puzzlestück ist, das zum Gesamtbild der Software beiträgt?

Guest: Genau, das ist eine treffende Analogie! Man kann sich zum Beispiel vorstellen, dass eine Schicht dafür verantwortlich ist, Wetterdaten anzuzeigen, während eine andere Schicht diese Daten verarbeitet, die eventuell von einem externen Wetterdienst stammen.

Host: Interessant! Welche Schichten sind denn am wichtigsten?

Guest: Es gibt zwei Hauptschichten, die wir betrachten sollten: die Präsentationsschicht und die Fachschicht. Die Präsentationsschicht befasst sich mit der Benutzeroberfläche und der Interaktion mit dem Benutzer. Bei React Native würden das Komponenten wie `<View>`, `<Text>`, und `<Button>` sein.

Host: Also ist die Präsentationsschicht sozusagen wie der Kellner in einem Restaurant, der mit den Gästen interagiert?

Guest: Absolut, das ist eine perfekte Analogie! Und dann gibt es noch die Fachschicht, die die

Geschäftslogik der Anwendung verarbeitet. Sie ist vergleichbar mit der Küche eines Restaurants, die die Bestellungen vorbereitet.

****Host**:** Das ist ein sehr anschauliches Bild. Warum ist es denn so wichtig, diese Schichten zu trennen?

****Guest**:** Es gibt mehrere Vorteile: Zum einen ist die Software übersichtlicher und Änderungen können leichter durchgeführt werden, da sie oft nur eine der Schichten betreffen. Zudem können Schichten in anderen Projekten wiederverwendet werden, was gerade bei komplexen Anwendungen wie einer Wetter-App sehr nützlich ist.

****Host**:** Und wie genau sieht das in der Praxis mit React Native aus, zum Beispiel bei einer solchen Wetter-App?

****Guest**:** In der Präsentationsschicht würden wir die Temperatur und vielleicht ein Wetter-Icon darstellen, die Fachschicht hingegen würde die Wetterdaten von einer externen API abrufen und verarbeiten, um beispielsweise Durchschnittstemperaturen zu berechnen.

****Host**:** Perfekt, das klingt alles sehr logisch. Was für eine Übung würdest du jemandem empfehlen, der diese Konzepte besser verstehen möchte?

****Guest**:** Eine gute Übung wäre, an eine App zu denken, die man regelmäßig nutzt, wie etwa WhatsApp oder Spotify. Man könnte dann überlegen, welche React Native-Komponenten zur Präsentationsschicht dieser Apps gehören und welche Aufgaben die Fachschicht übernimmt.

****Host**:** Das ist eine tolle Idee! Finn, vielen Dank für diese faszinierenden Einblicke in die Schichtenarchitektur und die Anwendung in React Native. Hast du noch abschließende Gedanken?

****Guest**:** Nur noch kurz, dass die Schichtenarchitektur wirklich ein fundamentales Konzept in der modernen Softwareentwicklung ist, das vor allem in Frameworks wie React Native hilft, komplexe Anwendungen sauber und effizient zu strukturieren.

****Host**:** Das war wirklich aufschlussreich. Danke, dass du heute hier warst, Finn. Und danke auch an

unsere Hörer, dass Sie eingeschaltet haben. Bis zum nächsten Mal!