

Hoewel ik geen technisch achtergrond heb, heb ik gekeken naar verschillende technieken waarmee de applicatie gerealiseerd kan worden. Ik ben in gesprek gegaan met developers om een beter beeld te krijgen van de mogelijkheden. Ik heb over meerdere technieken gelezen en 2 onderzocht.

Hybride applicatie

Een hybride app combineert 2 technieken. Het gedraagt zich als een traditionele (native) app, maar is daadwerkelijk een webapplicatie. Het voordeel van een hybride applicatie is dat er niet voor elk platform (ios/android) apart ontwikkeld dient te worden. Dit scheelt in het ontwikkelen, testen en onderhouden van de applicatie waardoor de time-to-market lager ligt en tevens de kosten gedrukt worden.

De applicatie kan gebouwd worden met een open source framework genaamd Ionic 6. Ionic staat bekend om het voordeel dat er 1 codebase is voor alle platformen (crossplatform), dus zowel het bouwen als onderhouden van de applicatie wordt hiermee vergemakkelijkt. Tevens heeft het grote voordelen voor de performance (100% lighthouse PWA performance score). Ook werkt het naadloos samen met de meest populaire front end frameworks als Angular, VueJS en ReactJS.

De backend van de applicatie kan opgevangen worden door Google Firebase. Dit is een development software waarmee servers en databases beheerd kunnen worden. Tevens verloopt de security (login, authenticatie, CRUD acties, etc.) via deze software. Hierdoor is het niet nodig om eigen servers te hosten. Ook bevat het tools voor tracking, analytics, reports, bijhouden en oplossen van crashes.

Low Code (mendix/OutSystems/Microsoft Power Apps)

Tijdens mijn onderzoek heb ik opgemerkt dat lowcode tools steeds vaker worden gebruikt. Het voordeel van lowcode is dat het snel geleerd en er razendsnel ontwikkeld kan worden, waardoor de time-to-market en kosten in hoge mate gedrukt worden. Het ontnemt vele zorgen waar je als ontwikkelaar niet aan hoeft te denken. Servers en databases gaan via het platform en de security wordt door het platform opgevangen. DTAP omgevingen en versiebeheer van de software zit standaard ingebakken. Als de architectuur goed is opgezet, kan je er probleem met verschillende teams aan werken. Dit allemaal zorgt er voor dat low code ontzettend snel en flexibel is. Zoals hierboven vermeld, geldt ook voor deze lowcode platformen dat er 1 codebase is, waaruit hybride applicaties gegenereerd kunnen worden voor iOS en Android. Er is ook geen afhankelijkheid van andere systemen. Alle logica en tools zitten in het platform, wat direct ook het nadeel is. Je applicatie is afhankelijk van dat ene platform en de licentiekosten ervan.