

Companion app

Realisatiedocument

Jana Vande Velde
Student Bachelor in de Toegepaste Informatica – Applicatieontwikkeling

Inhoudsopgave

| | |
|--------------------------------------|---|
| 1. INLEIDING | 3 |
| 2. ANALYSE | 4 |
| 3. ONTWIKKELING VAN DE COMPANION APP | 5 |
| BESLUIT | 6 |
| LITERATUURLIJST | 7 |
| BIJLAGEN | 8 |

1. Inleiding

Voor mijn stage bij Strobbo in Lommel werkte ik mee aan de ontwikkeling van een nieuwe mobiele applicatie: de Companion App. Strobbo biedt een platform aan waarmee werkgevers hun personeel kunnen inplannen, registreren en beheren. Tot voor kort verliep dit voornamelijk via een desktopapplicatie. In de praktijk bleek echter dat veel werkgevers behoefte hadden aan een meer flexibele oplossing die hen ook op de werkvloer ondersteunde.

De stageopdracht was om, samen met een andere stagiair, een eerste versie van deze mobiele Companion App te realiseren. In deze app zouden enkele essentiële functies uit de desktopversie beschikbaar gemaakt worden voor gebruik op smartphones. Dit document beschrijft het verloop van deze opdracht.

Eerst geef ik een overzicht van de analysefase: welke tools en technologieën werden gebruikt, waarom deze keuzes gemaakt zijn en hoe de voorbereiding verliep. Vervolgens bespreek ik in detail de ontwikkeling van de Companion App en de belangrijkste onderdelen waaraan ik zelf heb gewerkt. Tot slot rond ik af met een besluit waarin ik het project samenvat, de belangrijkste resultaten bespreek en enkele suggesties doe voor mogelijke uitbreidingen en verbeteringen van de app in de toekomst.

2. Analyse

Tijdens mijn stage heb ik gebruik gemaakt van verschillende tools en technologieën die grotendeels vooraf bepaald waren door het stagebedrijf. Toch vond ik het belangrijk om zelf ook te onderzoeken waarom deze keuzes gemaakt zijn. Hiervoor heb ik onder andere de Weighted Ranking Methode toegepast, waarmee ik alternatieven met elkaar vergeleek op basis van criteria zoals technische kennis binnen het team, online documentatie, performantie en leercurve.

Voor de ontwikkeling van de front-end van de mobiele applicatie werkte we met React Native in combinatie met TypeScript. Hoewel dit op voorhand vastlag, heb ik gekeken naar mogelijke alternatieven zoals Flutter, Kotlin (voor Android) en Swift (voor iOS). Omdat het belangrijk was dat de app beschikbaar zou zijn voor zowel Android- als iOS-gebruikers, vielen Kotlin en Swift al snel af vanwege hun platform gebonden karakter. De keuze kwam daardoor neer op React Native of Flutter. Na vergelijking bleek React Native het meest geschikt, vooral dankzij de grote community, de uitgebreide documentatie en de ervaring binnen het team met JavaScript. (Zie Bijlage 1 voor een vergelijking van de front-end technologieën)

Dan voor de back-end gebruikte ik Node.js. Ook hiervoor heb ik alternatieven overwogen, zoals Python met Flask of Django, Ruby on Rails en .NET. Toch had Node.js in dit geval de meeste voordelen. De asynchrone aard van Node.js maakte het uitermate geschikt voor een mobiele toepassing waarbij veel I/O-operaties plaatsvinden. Bovendien sloot het goed aan op de gekozen front-end technologie, aangezien beide gebaseerd zijn op JavaScript. Dit zorgde voor consistentie in de codebase en vereenvoudigde de samenwerking tussen front- en back-end componenten. (Zie Bijlage 2 voor een vergelijking van de back-end technologieën)

Voor de story maps en sprint presentaties gebruikte we Miro. In dit online whiteboard kun je eenvoudig story maps maken met post-its, werkvoortgang weergeven en presentaties maken. Hoewel er andere tools zijn zoals Trello, Jira of Lucidchart, bood Miro meer vrijheid en visuele duidelijkheid, wat goed aansloot bij de manier waarop er gewerkt en gecommuniceerd wordt binnen het Strobbo team. (Zie Bijlage 3 voor een vergelijking van de tools)

Hoewel de technische keuzes grotendeels vooraf vastlagen, vond ik het zinvol om ze kritisch te bekijken en te onderbouwen. Dit gaf me meer inzicht waarom we juist deze technologieën voor ons stageproject gebruiken.

3. Ontwikkeling van de Companion App

In dit hoofdstuk beschrijf ik het resultaat van mijn stageproject bij Strobbo: de functionaliteiten die ik heb ontwikkeld, de opzet van de toepassing en de principes die ik daarbij heb toegepast.

Tijdens mijn stage werkte ik samen met een andere stagiair aan de ontwikkeling van een nieuwe mobiele applicatie, genaamd de Companion App. Deze app is specifiek bedoeld voor werkgevers. Waar zij voorheen enkel via de desktopapplicatie werknemers konden inplannen, afwezigheden melden en in- en uitklokken, biedt de nieuwe mobiele toepassing de mogelijkheid om deze acties snel en eenvoudig via een smartphone uit te voeren. Aangezien werkgevers vaak zelf actief zijn op de werkvloer, was er behoefte aan een meer flexibele en toegankelijke oplossing. Hieruit is onze stageopdracht ontstaan.

Een van de onderdelen waar ik aan gewerkt heb, is de kloktijdenoverzichtspagina. Op deze pagina kan de werkgever zien welke werknemers die dag gepland zijn om te werken. Naast de geplande aanwezigheid wordt ook weergegeven of een werknemer reeds is ingeklokt, te laat is, of al te lang aan het werk is. De actuele status van elke werknemer wordt visueel aangeduid via duidelijke status labels zoals "Te laat", "Werkend", "Te lang bezig" en "Gepland". Ook worden de geplande start- en eindtijden samen met de effectieve inkloktijden getoond, zodat de werkgever een volledig beeld heeft van de actuele werksituatie. (Zie Bijlage 4 voor een visuele weergave van deze pagina)

Wanneer je op een kloktijd in deze lijst klikt, krijg je de mogelijkheid om een werknemer in of uit te klokken, of om de in of uit kloktijd te wijzigen. (Zie Bijlage 5 voor een visuele presentatie van hoe deze acties in de applicatie weergegeven worden)

Voor deze functionaliteit heb ik aanpassingen gedaan aan de back-end endpoints. Bij het inklokken wordt er een endpoint gebruikt om een nieuwe time-entry aan te maken. De eindtijd heb ik optioneel gemaakt, omdat bij het inklokken nog niet bekend is wanneer de werknemer zal uitklokken. Bovendien heb ik een 'workdayld' toegevoegd om de nieuwe time-entry te kunnen koppelen aan de bestaande. Hierdoor verdwijnt de oude entry zodra er een nieuwe wordt aangemaakt, zodat het overzicht actueel blijft.

Daarnaast ontwikkelde ik de contactenpagina. Op deze pagina krijgt de werkgever een volledig overzicht van alle werknemers binnen zijn bedrijf. Er is een mogelijkheid voorzien om werknemers te filteren op basis van werkvestiging, zoals bijvoorbeeld Lommel of Geel, en op werkgebied, zoals keuken of zaal. Bovendien kan er eenvoudig gezocht worden op naam, zodat de juiste werknemer snel teruggevonden kan worden. Wanneer de werkgever op de naam van een werknemer klikt, wordt hij automatisch doorgestuurd naar de contactendetailpagina. Op deze detailpagina is extra informatie over de werknemer beschikbaar, zoals het telefoonnummer, e-mailadres, pincode (voor de tikklok) en de toegekende werkvestiging(en) en werkgebied(en). Daarnaast zijn er op deze pagina actieknoppen voorzien, zodat de werkgever direct een werknemer kan bellen, een sms kan sturen of een e-mail kan opstellen. (Zie Bijlage 6 voor een visuele weergave van deze pagina)

Met de toevoeging van de Companion App wordt het voor werkgevers een stuk eenvoudiger om hun personeel te beheren, ongeacht waar zij zich bevinden.

BESLUIT

Tijdens mijn stage bij Strobbo heb ik een waardevolle bijdrage kunnen leveren aan de ontwikkeling van de Companion App, een mobiele toepassing die het voor werkgevers mogelijk maakt om hun personeel snel en eenvoudig te beheren. Dit project bood een praktische oplossing voor de behoefte aan flexibiliteit en toegankelijkheid die werkgevers op de werkvloer ervaren. Samen met mijn collega-stagiair hebben we een applicatie ontwikkeld die essentieel is voor het vereenvoudigen van processen zoals het in- en uitklokken van werknemers, het beheren van werktijden, en het beheren van contactinformatie.

De keuze voor de gebruikte technologieën zoals React Native, Node.js en Miro zijn zorgvuldig overwogen en onderbouwd, zowel door de vooraf bepaalde keuzes van het stagebedrijf als door de extra analyse die ik zelf heb uitgevoerd. Dit onderzoek naar de geschikte tools en technologieën heeft mijn inzicht in het project verder vergroot. De Weighted Ranking Methode bleek hierbij een nuttige manier om alternatieven te evalueren en de best passende keuzes te maken voor de ontwikkeling van zowel de front-end als de back-end van de app.

De belangrijkste functionaliteiten die ik heb ontwikkeld, zoals de kloktijdenoverzichtspagina en de contactendetailpagina, dragen bij aan een efficiënter personeelsbeheer. De implementatie van de kloktijdenfunctionaliteit, inclusief de mogelijkheid om werknemers in of uit te klokken, biedt een toegevoegde waarde die het dagelijkse werkproces voor werkgevers vereenvoudigt. Ook de contactenpagina maakt het eenvoudiger om werknemers snel terug te vinden en met hen te communiceren.

Bij de evaluatie van het project is duidelijk geworden dat de Companion App de gestelde doelstellingen heeft behaald: het aanbieden van een flexibele en gebruiksvriendelijke oplossing voor werkgevers. De app sluit goed aan bij de wensen van de gebruikers en biedt de benodigde functionaliteit in een eenvoudig te navigeren interface.

Voor de toekomst zie ik verschillende mogelijkheden voor verdere uitbreiding van de app. Zo kan er bijvoorbeeld gedacht worden aan de toevoeging van automatische planning van een andere werknemer wanneer iemand onverwacht ziek meldt. Verder zou de app geoptimaliseerd kunnen worden door het verbeteren van de gebruikerservaring en het toevoegen van extra aanpassingsmogelijkheden voor specifieke bedrijfsbehoeften.

Al met al was het een leerzame en waardevolle ervaring die mijn kennis van mobiele app-ontwikkeling en backend-integratie heeft vergroot, en ik kijk ernaar uit om mijn ervaring die ik hier heb opgedaan mee te nemen in mijn professionele carrière.

LITERATUURLIJST

Magalhães, F. M. (2020, januari 20). *I built the same app with Flutter, React Native and Ionic*. Medium. <https://medium.com/@fmmagalhaes/i-built-the-same-app-with-flutter-react-native-and-ionic-33ff8b358562>

TechMagic. (2023, juli 26). *Node.js vs .NET: What to choose for your next project*. TechMagic Blog. <https://www.techmagic.co/blog/node-js-vs-net-what-to-choose>

Miro Guide. (z.d.). *Miro*. Geraadpleegd op 3 mei 2025, van <https://miro.com/how-to-use-miro/>

Trello Guide. (z.d.). *Trello*. Geraadpleegd op 3 mei 2025, van <https://trello.com/guide>

Jira Software Documentation. (z.d.). *Atlassian*. Geraadpleegd op 3 mei 2025, van <https://support.atlassian.com/jira-software>

Lucidchart. (z.d.). *Lucidchart tour*. Geraadpleegd op 3 mei 2025, van <https://www.lucidchart.com/pages/tour>

Stagementors Strobbo. (2025). *Eindevaluatie van de applicatie*. Interne documentatie, Strobbo NV.

BIJLAGEN

Bijlage 1 – Weighted Ranking Matrix (Front-end)

Tabel met de scores en motivatie voor de front-end technologie keuze. (Verwezen naar in hoofdstuk 2, Analyse)

| Criteria | Gewicht (1-5) | React-Native | Flutter |
|-------------------------|---------------|--------------|------------------------|
| Community-ondersteuning | 5 | 5 | 4 |
| Documentatie | 4 | 5 | 4 |
| Performantie | 3 | 4 | 4 |
| Leercurve | 3 | 5 | 3 |
| Teamervaring | 4 | 5 | 3 |
| Toekomstbestendigheid | 2 | 4 | 4 |
| Totaalscore | | 100 | 77 <small>miro</small> |

Bijlage 2 – Weighted Ranking Matrix (Back-end)

Tabel met de scores en motivatie voor de back-end technologie keuze. (Verwezen naar in hoofdstuk 2, Analyse)

| Criteria | Gewicht (1-5) | React-Native | Flutter |
|-------------------------|---------------|--------------|------------------------|
| Community-ondersteuning | 5 | 5 | 4 |
| Documentatie | 4 | 5 | 4 |
| Performantie | 3 | 4 | 4 |
| Leercurve | 3 | 5 | 3 |
| Teamervaring | 4 | 5 | 3 |
| Toekomstbestendigheid | 2 | 4 | 4 |
| Totaalscore | | 100 | 77 <small>miro</small> |

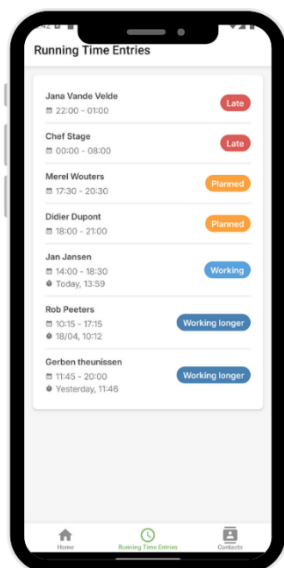
Bijlage 3 – Weighted Ranking Matrix (Tools)

Tabel met de scores en motivatie voor de tools keuze. (Verwezen naar in hoofdstuk 2, Analyse

| Criteria | Gewicht (1-5) | Miro | Trello | Jira | Lucidchart |
|-------------------------------------|---------------|------------|-----------|-----------|-------------------------------|
| Visuele duidelijkheid | 4 | 5 | 3 | 2 | 4 |
| Flexibiliteit in gebruik | 5 | 5 | 4 | 3 | 3 |
| Gemak voor story mapping | 4 | 5 | 3 | 3 | 2 |
| Geschikt voor presentaties | 3 | 4 | 2 | 2 | 3 |
| Samenwerking in teamcontext | 4 | 5 | 4 | 3 | 3 |
| Leercurve / gebruiksvriendelijkheid | 3 | 4 | 5 | 3 | 4 |
| Totaalscore | | 109 | 81 | 62 | 72 <small>miro</small> |

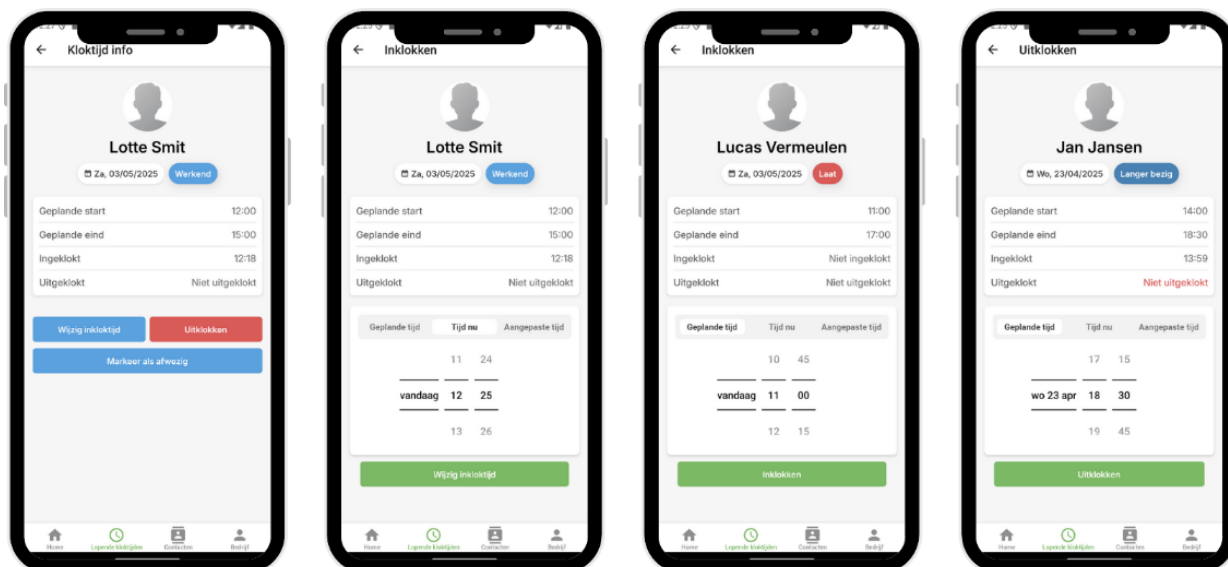
Bijlage 4 – Screenshot kloktijdenoverzichtspagina

Toont hoe geplande werktijden en actuele klokstatussen worden weergegeven. (Verwezen naar in hoofdstuk 3, Ontwikkeling van de Companion App)



Bijlage 5 – Screenshot in-/uitklofunctionaliteit

Visualisatie van de actieknoppen na klikken op een kloktijd. (Verwezen naar in hoofdstuk 3)



Bijlage 6 – Screenshot contactenpagina en detailpagina

Weergave van de zoek- en filterfunctionaliteiten en detailinformatie per werknemer. (Verwezen naar in hoofdstuk 3)

