

Kerntaak 3

Optimaliseer een product

Tim Bouma

10-05-2021

# Inleiding

**Project**

Flutter en Firebase applicatie. Compleet met database van Firebase Firestore en een authenticatie systeem met Firebase Authenticatie. De applicatie is ontwikkeld met het framework van google Flutter.

Het idee om een applicatie om bouwen is ontstaan uit plan om eerst kennis van ontwikkelen op te bouwen. Als er genoeg ervaring is ontwikkeld kunnen we doorgaan naar uitgebreide project die ook echt gebruikt kunnen worden.

Hiervoor heb ik als barista gewerkt bij de CoffeeCompany. Bij de coffeecompany ben ik veel bezig geweest met het zetten en branden van koffie. In de applicatie kunnen gebruiker inloggen en registreren met Firebase authenticatie. Om daarna een koffie voorkeur te noteren. Zo weet de stagiaire van elke werknemer hoe hij zijn koffie wilt. Want het halen van koffie is een van mijn werk functies.

Het project is succesvol als er na twee weken een werkende applicatie is neergezet. Als het lukt om deze applicatie opzetten via de voorgeschreven Flutter ontwikkelmethodes krijg je een goed beeld van hoe je een cross-platform applicatie moet ontwikkelen. Tijdens het ontwikkelen zelf word stagiair uitgedaagd om zich te verbeteren. Het kan dan ook voorkomen dat er extra functie worden ontwikkeld die buiten de scope van het PvE vallen.

**Verslag**

In dit verslag worden exact drie werkprocessen beschreven.

Het eerste werkproces (B1-K3-W1) is het optimaliseren van het product. Voor dit werkprocess stel ik een acceptatietest op voor het ontwikkelde product. Deze test laat ik uitvoeren door een of meerdere gebruikers en begeleidt ze hierbij. Foto’s zullen een extra informatie geven over het verloop van de acceptatietest. De feedback van de gebruikers wordt geanalyseerd en bepaald welke punten we gaan verwerken. Verwerking worden doorgevoerd en beschreven. Hierbij laat ik een stuk code zien met toelichting. Daarnaast worden deze verandering verwerkt in de documentatie van de applicatie.

Werkproces (B1-K3-W2) is het opleveren van een product. Hierbij demonstreer ik het product aan de werkgever. Het demonstreren gaat aan de hand van het programma van eisen. Daarna vraag ik om goed goedkeuring van het product via mail. Deze mail word verwerk in een verslag over het verloop van het opleverprocess..

Werkproces (B1-K3-W3) is het evalueren van het opgeleverde product met een evaluatieverslag. Hierin evalueer ik het product en opleverprocess met de betrokken en bepaal wat er wel of niet goed is gegaan. Daarnaast analyseer ik ook wat er verbeterd kan worden in volgende projecten. Deze verbetering worden hierin beschreven.

# Werkproces B1-K3-W1

# Optimaliseer een product

## Acceptatietest

**Test 1 Inloggen**

Rol : Gebruiker

Functie : Inloggen

Beginpunt :

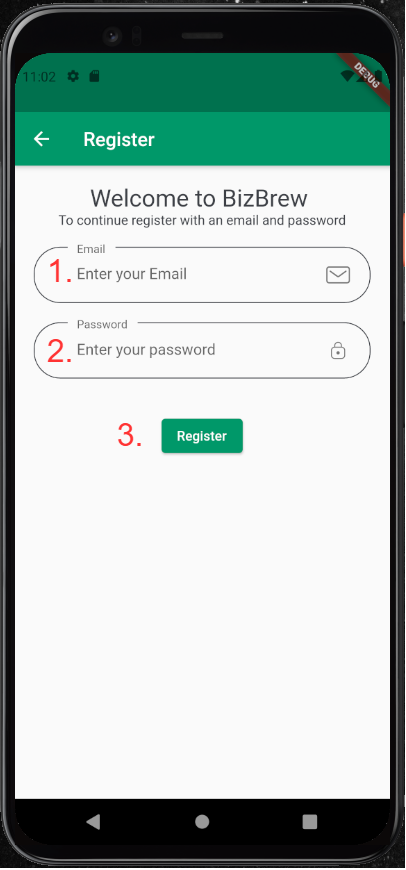
Eindpunt :

Acties:

1. Vul e-mail in.
2. Vul password in.
3. Druk op inlogknop.

Resultaat : Gebruiker is Ingelogd en is aangekomen op het

home scherm



**Test 2 Registreren**

Rol : Gebruiker

Functie : Registreren

Beginpunt :

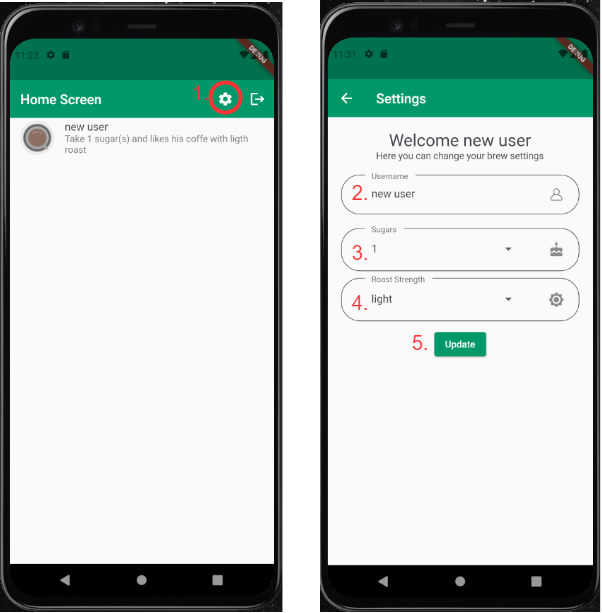
Eindpunt :

Acties:

1. Vul e-mail in
2. Vul password in.
3. Druk op registratieknop

Resultaat : Gebruiker is geregistreerd in de database met gegevens(gebruikersnaam,

suikers en sterkt). Gebruiker is Ingelogd en aangekomen op het homescherm.

**Test 3 Settings updaten van gebruiker**

Rol : Gebruiker

Functie : Updaten van gebruikersgegevens

Beginpunt :

Eindpunt :

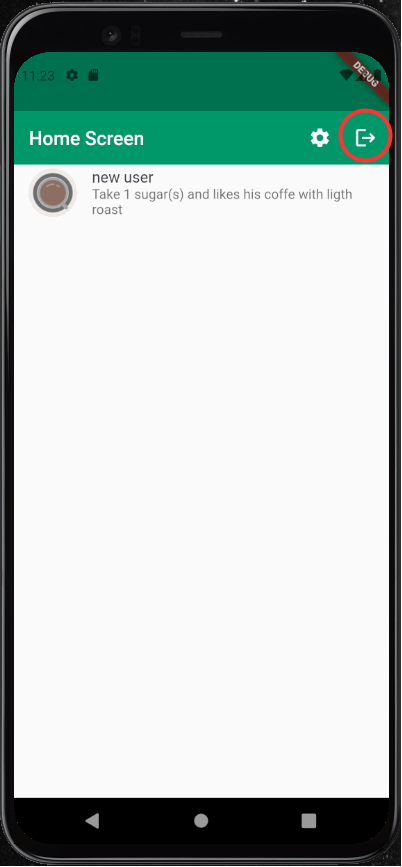
Acties :

1. Klikt op settings wheel.
2. Aanpassen gebruikersnaam
3. Aanpassen aantal suiker
4. Aanpassen roast sterkte.
5. Klik op de update knop.

Resultaat : Gebruiker settings zijn aangepast

in de database en worden correct

weergegeven in de applicatie.



**Test 4 Uitloggen**

Rol : Gebruiker

Functie : Uitloggen

Beginpunt : home scherm

Eindpunt : welcome scherm met de keuze om weer te

registreren of inloggen

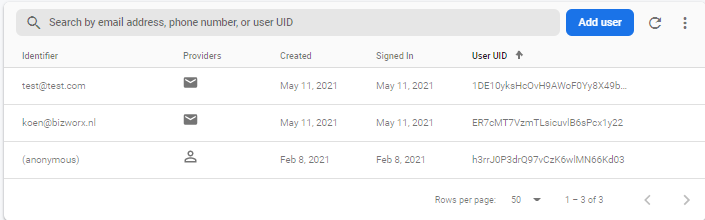
Actie : Klikt op uitloggen.

Resultaat : Gebruiker is ingelogd.

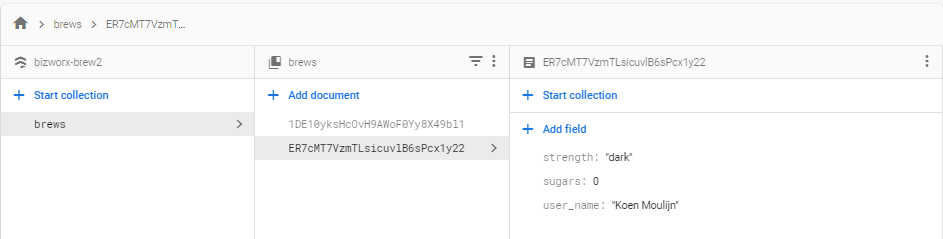
## Uitvoeren Acceptatietest

De acceptatietest begint met korte en duidelijke inleiding welke functionaliteiten de testen gaan voorkomen. Daarbij vraag ik aan hem of hij kan letten op punten die verbeterd kunnen worden. Verder geeft ik niet te veel informatie weg hoe hij de test moet uitvoeren. Tijdens het programmeren is er namelijk een happy path (happy flow) ontstaan. Dit is een scenario die geen Foutmelding geeft om dat bij het elke test dezelfde parameters zijn gebruikt. Door de testpersoon niet verwachte parameters te gebruiken komen er nieuwe foutmeldingen naar boven. Deze foutmeldingen kunnen later worden geanalyseerd.

Een opvallende moment was dat de gebruiker op het welkomscherm probeert via de tekst te navigeren. De tekst geef aan om naar registreren te gaan als je nieuw bent. Als je al een account hebt ga dan naar inloggen. Voor de gebruiker was dit alleen niet duidelijk.Verder gaat het registreren volgens begin en eind scenario. Gebruiker is ingelogd en gegevens zijn opgeslagen.



Het aanpassen van de gebruikers settings is niet helemaal duidelijk. Gebruiker ziet niet direct de settings knop. De gebruiker probeer eerst door op verschillende onderdelen te klikken naar het settings scherm te navigeren. Zonder succes. Vervolgens klikt de gebruiker op de settings knop die hem brengt bij het settings scherm. Gebruiker past zijn gegevens aan en word zonder foutmelding opgeslagen in de database en aangepaste op het homescherm.



Gebruiker sign out en belandt weer op het homescherm. Nu moet de gebruiker weer inloggen. De gebruikere weet nog van het registren hoe hij moet navigeren. Gebruiker beland op het inlogscherm. Hier vult hij zijn gegevens van het registreren in. Bij het invoeren wordt er alleen een foutief wachtwoord ingevuld. Vervolgens wordt de gebruiker teruggestuurd naar het homescherm. Gebruiker navigeert terug naar sign in en logt met de correcter credentials in en eindigt op het homescherm. Het inloggen gaat volgens begin en eind scenario. Dit scenario moet welk opnieuw worden uitgevoerd als er verkeerd gegevens worden ingevoerd.

Resultaten van de acceptatietest

|  |  |  |  |
| --- | --- | --- | --- |
| **Functie** | **Acties** | **Feedback** | **Goed** |
| Navigeren | Navigeren naar signin of register scherm. | Via de tekst kunnen navigeren. | -/+ |
| Inloggen | Inloggen in applicatie | Gebruiker verwacht een foutmelding bij verkeerd invoeren van e-mail of wachtwoord. | - |
| Register | Registreren voor de applicatie | Gebruiker verwacht een foutmelding bij verkeerd invoeren van e-mail of wachtwoord.  Ook wil de gebruiker de gebruiker gelijk na het registreren zijn username en koffie settings aanpassen. | - |
| Update Settings | Het veranderen van username, suikers en roast. | Verloopt goed als er net van de happy flow wordt afgeweken.  Eventuele foutmeldingen geven bij het verkeerd invoeren van gegevens. | + |
| Sign Out | UItloggen in de applicatie | Verloopt goed en intuïtief | + |

## Analyse acceptatietest

*De feedback die ik ga verwerken uit de acceptatietest.*

**Registreren**

Het registreren gaat zoals verwacht. De gebruiker beland op het homescherm. Op het homescherm zijn de verschillende koffie specificaties van de gebruikers te zien. De gebruiker was alleen verward welke gegevens van hem zijn. Een oplossing hiervoor is het toevoegen van een veld voor de gebruikersnaam. Zo na het registreren gelijk duidelijk welke gegevens bij hem horen.

Het zou ook mooi zijn als de gebruiker gelijk na het registreren zijn koffie voorkeur gelijk kan aanpassen. Het toevoegen van deze functie is makkelijk. De gebruiker hoeft namelijk alleen na het registreren naar op settings schermen te belanden.

**Inloggen**

Voor de gebruiker is het belangrijk om te weten of er iets fout is gegaan tijdens het inloggen. Nu is het geval dat de gebruiker terug naar het welkom scherm wordt gestuurd. Als aanpassing stel ik het toevoegen van een foutmelding onderaan het inlog form voor. De foutmelding wordt weergegeven met een snackbar bericht. Omdat dit vaker voorkomt in mobiel applicaties.

**Settings**

Feedback die ik heb gekregen voor de settings scherm is het valideren van de invulvelden. Als er niks in de velden wordt ingevuld is er de kans dat er de gegevens in de database worden opgeslagen als *null*. Deze *null* gegevens zorgen vooreen foutmelding op het homescherm. Waardoor de applicatie crasht. Het is daarom van groot belang dat het valideren van de velden wordt toegevoegd. Het toevoegen moet voorkomen dat deze foutmelding opnieuwen plaats vind.

**Navigeren**

De gebruiker vindt het navigeren op het beginscherm onduidelijk. Voorstel van de gebruiker was het kunnen navigeren via tekst knoppen. De tekst knoppen moeten de gebruiker navigeren naar het inlog en registratiescherm. Het toevoegen van deze functionaliteiten gaat de gebruiker tijd besparen.

Als alle aanpassingen zijn verwerk en toegepast in de code van de applicatie, voldoet de applicatie aan alle eisen van het PvE. Het slagen van alle acceptatietesten betekent ook dat de wrapper van de applicatie werkt. De wrapper is het onderdeel dat luistert naar de veranderingen in de authenticatie. De wrapper bepaald aan de hand van de authenticatie status(niet of wel ingelogd) of de gebruiker het beginscherm of het home scherm te zien krijgt. Zie afbeeldingen acceptatietest.

*Feedback die ik niet ga verwerken uit de acceptatietest.*

**Settings**

De gebruiker had even moeite met het vinden van de settings knop. Feedback van de gebruiker was om gebruikersgegevens ook klikbaar te maken. Dit ga ik alleen niet aanpassen in de code omdat dit voor mee verwarring gaat leiden bij andere gebruikers. Dit maakt de settings knop button voor nu de beste oplossing. Het is altijd nog mogelijk om in een nieuwe versie van de applicatie deze feature te veranderen.

Feedback op de style van de applicatie worden niet meegenomen in de aanpassingen. Dit project was voornamelijk bedoeld om verschillende functionaliteiten te verwerken en niet om een design en graphics te ontwerpen.

## Aanpassingen na acceptatietest

Na de feedback analyse is besloten dat er een aantal onderdelen worden aangepast en verbeterd. In het weergeven van de aanpassingen wordt GitHub gebruikt. Door het uitvoeren van een Github Commit voor en na de acceptatietest, worden de aanpassingen aan de code duidelijk zichtbaar. Zeker als je in een team werkt kan Github erg handig. Daarnaast is github wel gebruikt voor het bijhouden van de versiebeheer en delen van de source code met de opdrachtgever.

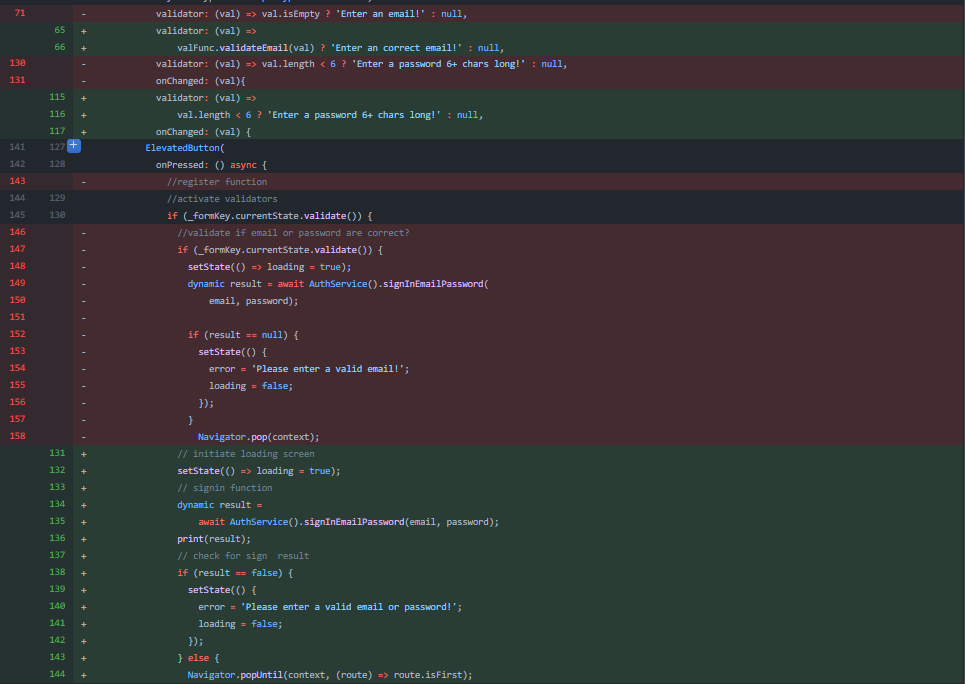
**Github Project**

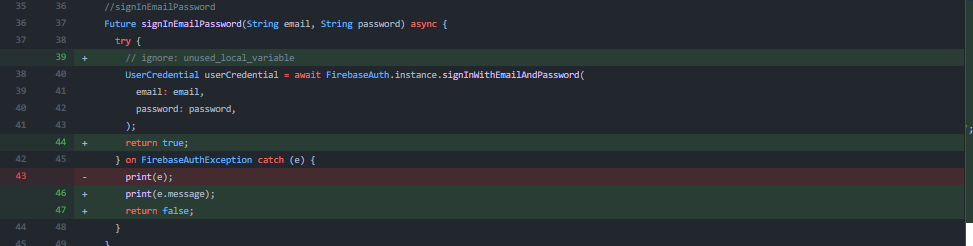
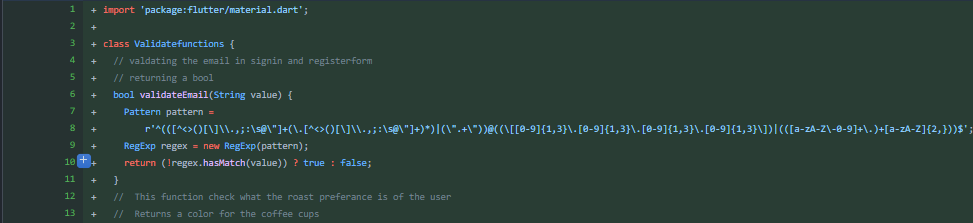
<https://github.com/Baqiwaqi/BizBrew>

**Github Commit changes** <https://github.com/Baqiwaqi/BizBrew/commit/6f79da1cefe6554484f6b098148cc38ca3de29de>

**Inloggen**

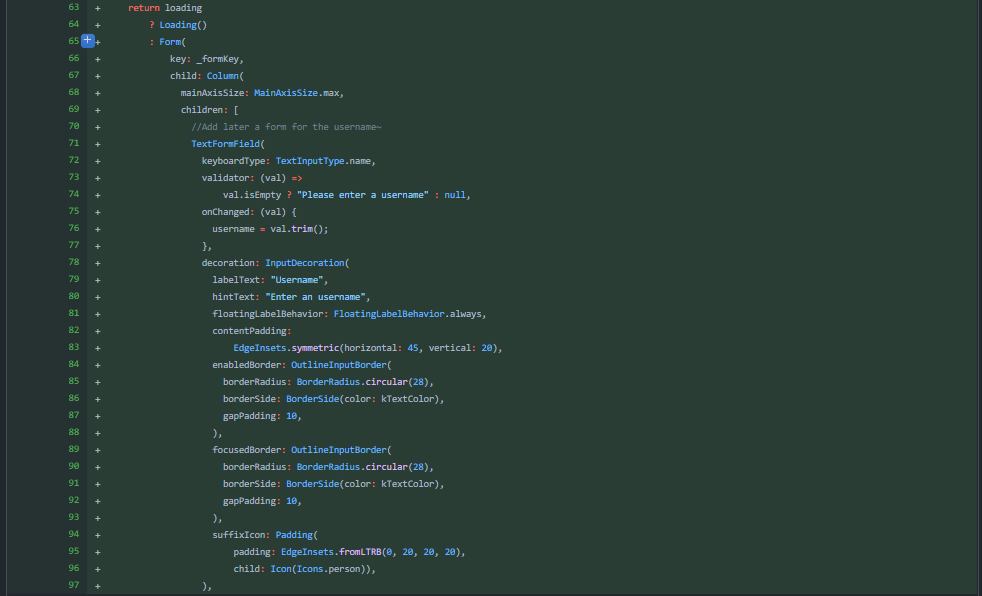
Het inloggen van de applicatie is voorzien van een formvalidator en een foutmelding voor de gebruiker. Bij het valideren zijn twee oplossingen toegepast. Als eerste is er in elke textformfield een validator toegevoegd. Als tweede oplossing heb ik een functie geschreven die controleert of de gebruiker een geldig e-mailadres gebruikt. Zeker voor het hergebruiken van code is dit handig omdat bij het registreren ook moet worden gecontroleerd op e-mailadres.



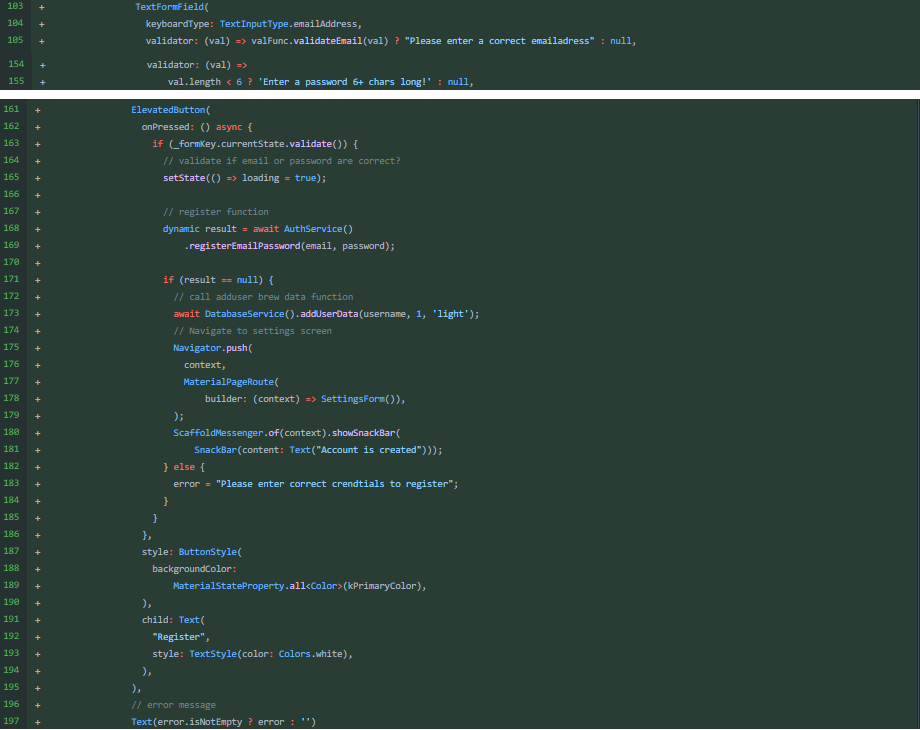


**Registreren**

Het registratie form is uitgebreid met een username textfield form. De gebruikersnaam is de naam die zichtbaar wordt weergegeven op het home scherm van de applicatie. Daarnaast zijn alle velden voorzien van een validator om te voorkomen dat er verkeerde gegevens worden opgeslagen in de database.

****

Na het registreren wordt de gebruiker direct doorverwezen met een navigator naar het settings scherm om daar zijn gebruikersnaam en koffie voorkeur direct aan te passen. Na het aanpassen van zijn gegevens wordt de gebruiker weer terug genavigeerd naar het home scherm. Dit proces wordt geregeld door de wrapper. Die eerder in dit document beschreven staat.

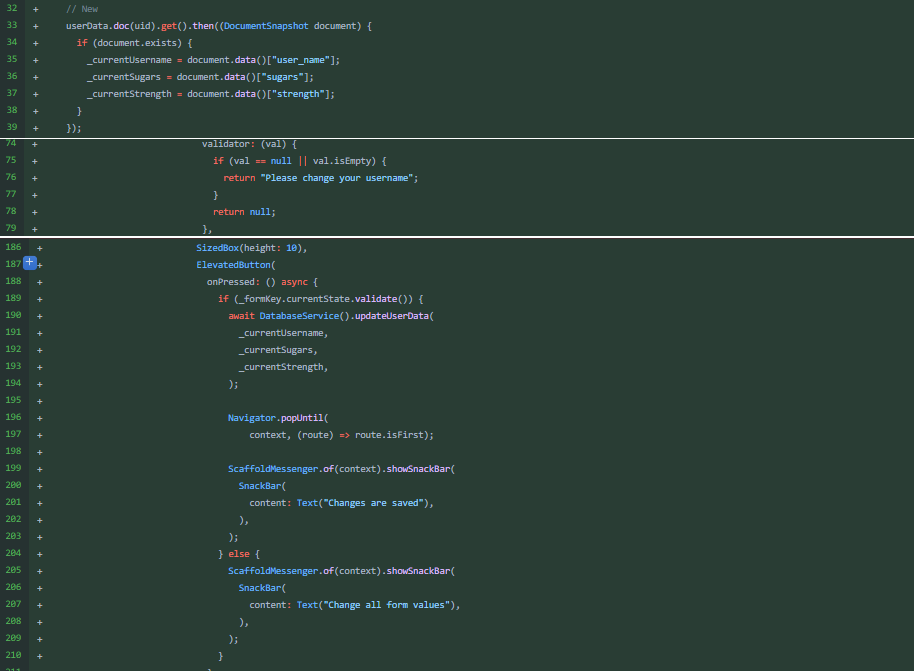
****

**Settings**

De aanpassingen in de code moeten voorkomen dat *null* gegevens worden opgeslagen in de database. Hiervoor heb ik eerst onderzoek gedaan naar hoe ik dit kan voorkomen.

Als eerste oplossing heb veld validator toegevoegd.Deze validator controleert of het veld leeg is (.isEmpty). Deze oplossing gaf helaas niet het gewenste resultaat. Gegevens konden nog steeds worden opgeslagen als *null* waarden.

De tweede oplossing ben ik tegengekomen in de Flutter Fire documentatie. De oplossing verwijst eerste gegevens naar properties. Vervolgens worden de properties automatisch ingevuld en weergegeven in de velden. Hierdoor kunnen er ook nooit meer *null* gegevens worden opgeslagen omdat er altijd een gegeven is ingevuld. De eerste oplossing zorgt er wel voor dat een geen velden leeg kunnen worden opgeslagen.

****

**Clean Code**

Flutter command zoals *flutter analyze* kan ook helpen bij het opschonen van je code. De functie geeft alle unused en foutmeldingen in de code terug in regels. Zo kan je makkelijk achterhalen waar fouten zitten en oplossen. Zeker als je code in productie genomen gaat worden, is het van belang dat er geen onnodig imports en informatie beschikbaar is voor de gebruiker. De meldingen die heb gekregen van *flutter analyze* heb vervolgens weggewerkt tot er geen melding meer te zien was.

## Documentatie van applicatie aanpassen

In het *Programma van Eisen*  heb ik twee stukken toegevoegd. Analyse en conclusie. Analyse is geeft aan hoe ik de acceptatietest heb verwerkt. Welke aanpassing ik heb gedaan aan de code. De conclusie is een reflectie op het project en iteratie van het project. Welke fouten ik in het volgende project kan vermijden.

**Analyse**

Tijdens het programmeren van de Flutter applicatie zijn er meerdere iteraties van de applicatie gemaakt. Bij elke iteratie van de applicatie is gelet op de architectuur van de code en folderstructuur. Er is ook gebruik gemaakt van de laatste versie van alle packages. Hierdoor is code een paar keer opnieuw geschreven om aan de laatste specificatie te voldoen. BloC (Business Logic Components) is design method die is gebruikt bij het ontwikkelen van de applicatie. Deze methode wordt gebruikt om om production quality applicaties te ontwikkelen. Waarbij het managen van de staat van de applicatie op elke moment kan worden bepaald.

Het valideren van gegevens voordat ze opgeslagen worden in de database was een van de functies die ik meerdere keren was vergeten. Zowel bij het registreren als het inloggen kwamen er foutmeldingen naar boven die het gebruiker onduidelijk waren. Bij het inloggen werd de gebruiker terug genavigeerd naar het home scherm na verkeerd invoeren van een e-mail of wachtwoord. Tijdens registreren was dit ook een probleem. Daarnaast was het voor de gebruiker onduidelijk welk account hij had aangemaakt. De oplossing die hiervoor is gebruikt was het toevoegen van een gebruikersnaam. Ook werd de gebruiker gelijk na het registreren doorverwezen naar het settings scherm waar hij zijn koffie gegevens kan aanpassen. In het settings scherm bleek achteraf veel fouten te zitten. Het was namelijk mogelijk dat er null gegevens werden opgeslagen in de database. Dit probleem is verholpen door het alleen overschrijven van de gegevens die gewijzigd zijn.

**Conclusie**

Door veel tijd te stoppen in het verbeteren van code en het onderzoeken van verschillende ontwikkelmethoden zoals BloC is er een goed basis opgebouwd voor het ontwikkelen van Flutter applicaties. Daarnaast is zijn werkproces zoals Git ook behandeld. Een proces dat bij nieuwe projecten van vast gaat komen. Fundamentals van Firebase zijn ook behandeld. Hierdoor is het mogelijk om weer snel een nieuwe firebase project opzetten. Tijdens het ontwikkelen zijn er natuurlijk ook fouten gemaakt. Deze fouten zie ik als ik ervaring voor nieuwe projecten die helpen bij het versnellen en verbeteren van het ontwikkelen.

De foutmeldingen die het meest zijn voorgekomen zijn veroorzaakt door het testen van maar een workflow. Door het testen van maar een workflow kwamen maar een bepaald aantal fouten omhoog. Pas nadat de acceptatietest werd uitgevoerd door een gebruiker die niet aan het project had gewerkt kwamen er nieuwe foutmeldingen naar boven. Dat komt doordat de gebruiker afweek van de eerder geteste scenario’s. In het vervolg is het belangrijk dat een aantal scenario’s worden uitgeschreven voordat er code wordt geschreven. Deze oplossing zal in de toekomst het aantal foutmeldingen die in de analyse staan beschreven voorkomen.

# Werkproces B1-K3-W2

# Levert het product op

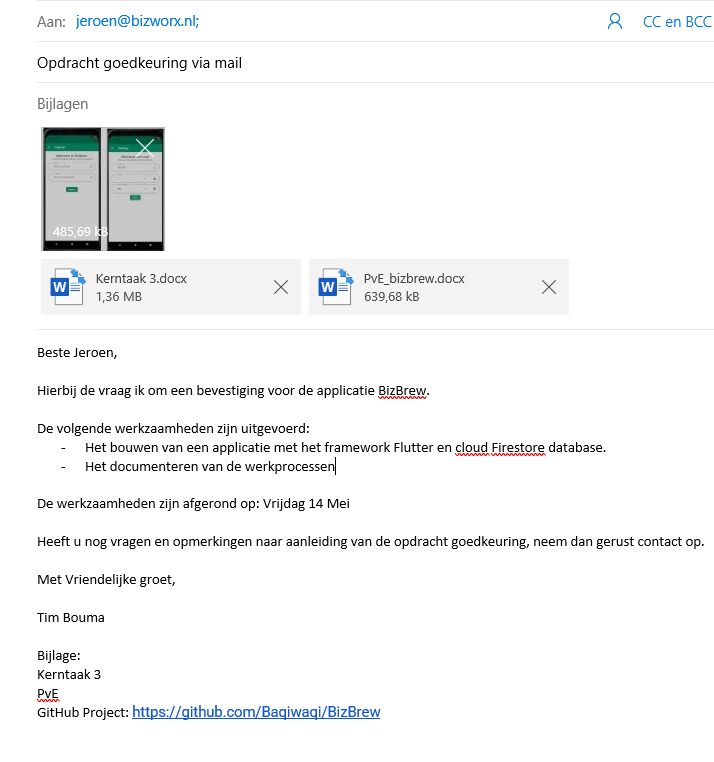
#### De kandidaat demonstreert het product aan de hand van het programma van eisen, en vraagt om goedkeuring van de opdrachtgever.

Verschillende iteraties zijn gedemonstreerd aan de op de opdrachtgever. Tijdens het ontwikkelen van de applicatie heb ik functionaliteiten van de applicatie laten zien aan de opdrachtgever. Hierop heb ik feedback gekregen wat er moet worden verbeterd. Uiteindelijk was de applicatie klaar om voor de acceptatie. De acceptatietest is uitgevoerd door een collega. Resultaten uit de applicatie test zijn geanalyseerd, verwerkt en toegepast. Dit proces staat beschreven in werkproces B1-K3-W1. Aan de hand een verslag en afbeeldingen wordt dit gedemonstreerd.

Functionele Eisen die in het PvE staan beschreven worden overtuigend aangetoont dat ze zijn verwerkt in de productieversie van de applicatie. Dit is gedaan door met een demonstratie nadat de aanpassingen van de acceptatietest waren verwerkt. Vervolgens is de code in Git van Main branch naar de Master branch gecommit. De Master branch is ook wel de productie branch. De code die in de Master branch staat is getest en aangepast en klaar voor productie gebruik.

Een mail naar de opdrachtgever voor goedkeuring (Zie Afbeelding) is gestuurd. In de inhoud van mail wordt staat dat het project is afgerond en klaar is voor gebruik. In de bijlage van mail is documentatie van de applicatie toegevoegd. Naast de documentatie is er een ook afbeelding van de applicatie toegevoegd.

De demonstratie van het product heb ik tegelijkertijd gecombineerd met het examen spreken voor Nederlands. Tijdens het examen heb ik dit project gebruikt voor de presentatie. Deze zelfde presentatie heb ik in een vereenvoudigde vorm gegeven voor de opdrachtgever. Deze presentatie is opgenomen als bevestiging dat dit onderdeel is uitgevoerd.



*Verwerken Reactie opdrachtgever*

De documentatie geeft volgens de opdrachtgever duidelijk aan dat de applicatie werkt en voldoet aan de requirement die zijn beschreven in het plan van aanpak. De afbeeldingen van de acceptatietest geven de werking van de afgesproken functionaliteiten goed weer. De analyse was soms cryptisch en zinnen waren opgebouwd uit nederlands en Engels. Hiervoor is de documentatie nog een keer doorgelezen en waar nodig bijgewerkt.

# Werkproces B1-K3-W3

# Evalueer het opgeleverde product

#### De kandidaat evalueert het product met de betrokkenen (wat is goed gegaan, verbeterpunten) en legt dit vast in een evaluatieverslag.

Het doel was dat de stagiaire kennis en werkervaring gaat op te bouwen voor het ontwikkelen van applicaties. Dit is voor de stagiair gelijk ook toegewezen als stageopdracht. Het resultaat waren meerder iteratie van een koffie applicatie. Het effect hiervan was de er een kennis op verschillende vlakken van ontwikkelen van software is behandeld. Daarnaast zijn de werkproces uitgewerkt en gedocumenteerd in de vorm van verschillende verslagen.

De kennis die ik had rondom het ontwikkelen van software was minimaal. Daarom was het idee om gelijk te beginnen met ontwikkelen. Door gelijk te beginnen met programmeren bouw je snel veel kennis op. Ondanks corona was het mogelijk op kantoor te werken. De opdrachtgever en stagebegeleider heeft ondanks zijn eigen werk veel tijd over voor het begeleiden van het project. Hierdoor had ik een nauwe samenwerking waarin hij mij uitdaagt in het verbeteren van mijn werk. Door deze samenwerking ben je meer bewust van het project en heb je het gevoel dat het ook een doel heeft.

De samenwerking verliep goed. Het was voornamelijk dat ik een functionaliteit aan de hand van de code demonstreer. Feedback kreeg op de code en een opdracht om de applicatie verder uit te werken. Wat wel beter kan is dat ik meer vragen stel. Soms was het geval dat ik vastliep en lang op een regel code bleef hangen. Dit is zonde van de tijd want vaak als je iemand anders even naar je code laat kijken zien die gelijk het probleem.

Tijdens het coderen werd de focus gelegd op het schrijven van schaalbare, veilige en herbruikbare code. Wat ik heb behandeld van Flutter en software architectuur is nog maar het topje van de ijsberg. Software architectuur is een onderdeel waar ik me graag verder in wil verdiepen. Het begrijpen van concept en fundamentele structuren en de daarvan onderlinge relaties en eigenschappen.

De stageopdracht bestaat uit het documenteren van verschillende werkprocessen. Het uitwerken van de verschillende werkprocess heeft erg geholpen bij het vormgeven van het project. Door zorgvuldig een documentatie bij te houden kan er gericht aan de software gewerkt worden. Zo is het duidelijk voor mijn stagebegeleider waar er aan wordt gewerkt en kan die begeleiding geven. Handelingen waar fouten in zijn gemaakt zijn beschreven in de documentatie. Om te voorkomen dat soortgelijk fouten in een volgend project ontstaan.

De versiebeheer beheer werd gedaan met Git. Waarbij de code wordt verstuurd naar GitHub om vervolgens gecontroleerd te worden. In theorie is werken met git workflows handig als je in teams werkt. Werk alleen aan een project ben je niet zo snel geneigd om gebruik te maken van een git workflow. Een volgende keer zou er in de samenwerking met de opdrachtgever afspraken worden gemaakt voor het gebruik van een Git workflow.

Dit project kende 1 risico: dat er niet genoeg functionaliteit voor een stageopdracht in de applicatie zou zitten. Het project heeft daarom ook verschillende iteraties aangenomen. Om verwarring te voorkomen is ook met mijn mentor gesproken of dit project zou voldoen aan de stage opdracht. Dit heeft goed gewerkt. Alleen voor een volgende keer moeten er eerst afspraken komen of het project voldoende is voordat het ontwikkelproces begint.

Er was kritiek vanuit de stageplek dat het Roc weinig informatie heeft geleverd over een stageopdracht. Door een gebrek aan informatie zijn we maar gaan ontwikkelen aan kleine projectjes. Het gevolg hiervan was dat er wel te laat is begonnen aan het uitwerken van de stage opdracht. Dit is vervelend want we zijn nu bezig zijn aan een project dat we stop moeten zetten om aan de stage opdracht te werken. Als verbeterpunt zal ik mezelf meegeven om eerder de meegeleverd docoumenten van uit school te lezen. Dan kan er eerst aan de opdrachten worden gewerkt en is er daarna tijd om aan leuk projecten te werken.

Verbeteringen om een volgend project succesvoller te maken is het eerst documenteren van een project. Dit geeft de programmeur en de opdrachtgever beide duidelijkheid. Deze documentatie hoeft niet zo uitgebreid te zijn als het schrijven van een : PvE, project plan, functioneel en technisch Ontwerp. Voor de stageopdracht is dit wel gebeurt. Dit heeft veel tijd gekost. Tijd die we beter kunnen stoppen in ontwikkelen van het programma.

Het gebruiken van Overview waarbij de het probleem , de succescriteria en de scope met requirement worden omschreven kan voldoende zijn voor een volgend project. Daarnaast is het ook handig om gebruik te maken van een kanban bord. Dit geeft overzicht van wat er nog moet gebeuren en wat is afgerond.

#### De kandidaat laat zijn evaluatieverslag accorderen.

#### 

(graag een mail van je begeleider met akkoord op je evaluatieverslag)