|  |
| --- |
| Photo displaying partial image of two pie charts on a canvas-textured page |
| Design Document SL4  Chat Applicatie |
| |  |  |  | | --- | --- | --- | | Mohammed El Malki (student) | 10/12/23 | Software Language 4 | |

**Inhoudsopgave**

[**Inleiding** 2](#_Toc153121799)

[**Opdracht beschrijving** 2](#_Toc153121800)

[**Gekozen Framework** 2](#_Toc153121801)

[**Diagrammen project** 2](#_Toc153121802)

[**Testplan** 3](#_Toc153121803)

## **Inleiding**

In dit Design Document zal er uitgebreid uitgelegd worden wat het project precies inhoudt en hoe dit tot stand is gekomen. Dit zal gedaan worden aan de hand van verschillende soorten deelonderwerpen. Hierin zal ik het hebben over de opdracht omschrijving, wat er gerealiseerd zal worden, de gekozen framework, relevante diagrammen en testaanpak.

## **Opdracht beschrijving**

Het doel van dit flutter project is het creëren van een chatapplicatie met Firebase-integratie. Binnen dit project worden verschillende soorten functies geïmplementeerd om een interactieve chat app te maken. Dit omvat het gebruik maken van Realtime database van Firebase en het gebruik maken van Firebase Authenticatie voor telefoonnummerverificaties.  
  
**Functional requirements:**

1. Het inloggen doormidden van telefoonnummer
2. Gebruikersprofielen aanmaken
3. Chatinterface
4. Notificaties verzenden
5. Het versturen van verschillende soorten bijlagen
6. Emoji ondersteuning
7. Het bewerken/verwijderen van berichten

**Non-Functional requirements:**

1. Functioneel op Web, Windows, Android en IOS
2. Responsieve op alles platformen
3. Veilige gegevensoverdracht
4. Prestaties(**moet soepel werken**)
5. Aanpasbaar voor gebruikers

Binnen deze applicatie zullen de volgende acceptatiecriteria ’s worden gerealiseerd :

1. **Het inloggen doormidden van telefoonnummer**  
   Binnen de applicatie krijgt de gebruiker de mogelijkheid om een account te maken door zijn/haar telefoonnummer in te voeren. Zodra de telefoonnummer is ingevuld, wordt automatisch een verificatienotificatie verstuurd naar dat nummer. De gebruiker moet deze notificatie verifiëren om het account te kunnen maken.  
     
   Nadat het telefoonnummer is geverifieerd krijgt de gebruiker de mogelijkheid om een profiel foto up te loaden, Zijn/haar naam in te voeren en een biografie toe te voegen. Deze persoonlijke informatie helpt andere gebruikers de persoon beter te leren kennen.
2. **Gebruikersprofielen aanmaken**De gebruikers krijgt de mogelijkheid om zijn/haar gebruiksprofiel te kunnen bewerken met behulp van Firebase Realtime Database om persoonlijke informatie op te kunnen slaan, denk hierbij aan: Een profielfoto, naam, status en biografie.
3. **Chatinterface**Doormiddel van een realtime database zoals Firebase Realtime database krijg je de mogelijkheid om snel en consistent meldingen naar elkaar te kunnen versturen. Verder zal het ook tekstberichten ondersteunen en bijvoegende Multi media inhoud.
4. **Notificaties verzenden**Op het moment dat een gebruiker een nieuwe bericht krijgt van een andere gebruiker, word er automatisch een melding naar de ontvanger verstuurd. Dit is bedoeld om de gebruiker op de hoogte te stellen van nieuwe berichten, zelfs wanneer de app in de achtergrond draait.
5. **Het versturen van verschillende soorten bijlagen**De gebruiker krijgt de mogelijk om verschillende soorten bijlages te kunnen versturen. Dit bedraagt het versturen en zien van afbeeldingen en video’s, het luisteren en maken van audioclips, het opnemen van documenten en het verzenden van gifs
6. **Emoji ondersteuning**Tijdens een chatgesprek met een ander gebruiker heeft de gebruiker de mogelijkheid om een Emoji te kunnen versturen. Dit betekent dat de gebruiker in staat is om eenvoudig emoji’s in te voegen en weer te geven in de chatinterface, ook zullen de emoji’s correct weergegeven worden bij de desbetreffende platform.
7. **Het bewerken/verwijderen van berichten**De gebruiker heeft binnen de chatinterface de mogelijkheid om berichten te bewerken en te verwijderen. Deze functie stelt de gebruiker in staat om een chatbericht te bewerken als er een fout is opgetreden. Daarnaast kunnen gebruikers een chatbericht verwijderen als die per ongeluk is verzonden of om een andere redenen.

**API Beschrijving**Voor het ontwikkelen van de chatapplicatie heb ik gekozen voor integratie met Firebase, een platform dat verschillende diensten beidt, waaronder realtime databases en telefoonnummerverificatie. Ik zal Firebase Realtime database gebruiken voor backend-data zoals chatberichten, telefoonnummerverificatie en persoonsinformatie. Deze NoSQL-databases zal de gegevens opslaan in een JSON-formaat, waardoor een er een gestructureerde benadering is van gegevens.

**Endpoints**:

**/api/users**: Hier worden gebruiker gegevens opgeslagen, inclusief informatie zoals profielfoto, profielnaam, status en biografie.  
**/api/messages:** In deze endpoint worden berichten opgeslagen en opgehaald. Hier zullen berichten worden opgeslagen in de database en wijzigingen worden direct naar alle verbonden client doorgestuurd.

**/api/verifyPhoneNumber**: In deze endpoint initieert het proces van telefoonnummerverificatie. Een verificatiecode wordt naar het opgegeven telefoonnummer gestuurd en na een succesvol verificatie word de gebruiker geautoriseerd.  
  
Doormiddel van deze API-endpoint, kunnen kan er gemakkelijk gecommuniceerd worden met de Firebase Realtime Database, communiceren met andere gebruikers via chatberichten, telefoonnummerverificatie, het verzenden en ontvangen van gegevens en persoonsinformatie binnen de applicatie.

## **Gekozen Framework**

In de keuzen van mijn framework voor het ontwikkelen van de **Chat Applicatie,** heb ik gekeken welke vereisten nodig zijn om het project kunnen realiseren en welke functionaliteiten ik wil implementeren. Daaruit ben ik tot de conclusie gekomen dat het gebruiken van Flutter, in combinatie met Dart de meest geschikte keuze is. Deze beslissing heb ik genomen om zo een veelzijdige applicatie te kunnen realiseren wat functioneert op verschillende platformen, waaronder Web, Windows, Android en IOS.

Flutter is een open course framework, ontwikkelt door Google, dat ontworpen is voor het bouwen van hoogwaardige, native gecompileerde en Multi-platformapplicaties vanuit één codebase. Het framewrok maakt gebruikt van de Dart-programmeertaal en introduceert het concept van widgets. Widgets kunnen gezien worden als bouwstenen voor het bouwen van gebruikersinterface. Ze vormen de fundamentele elementen waarmee verschillende interactie en visuele componenten zoals knoppen en lay-outs kunnen worden samengeteld. Verder is Flutter platformonafhankelijk wat betekent dat dezelfde codebase kan worden gebruikt voor de ontwikkeling van applicaties op verschillende platforms wat mogelijk gemaakt wordt door de Dart-compiler.  
  
Binnen het gebruik maken van Flutter zitten ook voor en nadelen dat bettrekking hebben tot het gekozen project. Denk hierbij aan :

**Voordelen** (Slava, 2019) **1. Integratie in bestaande applicaties**Flutter kan gemakkelijk geïntegreerd worden in bestaande applicaties, waardoor een programmeurs dubbel werk kunnen besparen.  
 **2. Stateful Hot Reload**Door de Stateful Hot Reload krijgen programmeurs de mogelijkheid om wijzigingen in real-time te ervaren tijden het coderen. Dit stelt ze instaat om gemakkelijk aan de applicatie kunnen werken zonder telkens opnieuw te hoeven opstarten. Hierdoor verloopt het ontwikkelingsproces efficiënter waardoor programmeurs direct feedback kunnen krijgen op de effecten van hun code.  **3.Kwaliteit en snelheid**Het programmeren van een applicatie voor IOS en Android met flutter gaat ongeveer 1,7 keer sneller dan met andere toolkit. Het implementeren van features kost minder moeite, doormiddel van de Dart-compiler. **4.Efficientie en tijdsbesparing**Flutter is relatief nieuw framework, waardoor het toegankelijk is voor nieuwe gebruikers en gemakkelijk te leren is. Bovendien leidt het gebruik van flutter tot tijdsbesparing, met een daling van vijftig procent in de tijd die nodig is voor een Quality Assurance, gegevens compiler gaat snel en ongeveer zeventig procent van de broncode wordt gedeeld en wordt hergebruikt.  
 **Nadelen  
1.Beperkte toegang tot hardware-features**Sommige hardware-features, zoals een camera kunnen alleen worden aangeroepen in de native code, wat voor verschillende soorten uitdagingen met zich mee kan brengen en wat soms voor dubbel werk kan opleveren. **2.Grote bestanden**Helaas hebben apps van Flutter-ontwikkelaars een groot formaat, wat kan leiden tot een langere downloadtijd voor gebruikers, dit komt omdat flutter-apps vaak worden geleverd ,met hun eigen runtime en een set van basisbibliotheken, wat kan resulteren in een grote bestandsgrote. Dit kan een probleem zijn voor apparaten met weinig opslag of voor gebruikers met lage downloadsnelheden. **3. Gebrek aan ondersteuning van derde partijen**Het gebrek aan libraries van derde partijen kan beperkingen hebben ten opzichten van andere tools. Dit komt omdat Flutter nog een vrij nieuwe framework is vergeleken met andere frameworks zoals React Native.  
 **4.Leercurve**Voor ontwikkelaar die nog niet bekend zijn met Dart-programmeer taal kan er een leercurve zijn bij het beginnen met flutter dit komt, omdat de programmeertaal nog nieuw is denk hierbij aan de syntax en het gedrag van **Dart**. Verder is **Dart** single-threaded met behulp van een event loop voor aysnchronische operaties wat nog wennen kan zijn voor ontwikkelaars die er niet gewend aan zijn.

**Aansluiting framework**Bij het ontwerpen van een chatapplicatieis Flutter een uitstekende keuze omdat flutter een krachtige framework is, waarbij je de mogelijkheid hebt om gemakkelijk cross-platform te ontwikkelen. Verder biedt het ook een rijke set van aanpasbare widgets en ontwerpmogelijkheden, wat voor mooie gebruikersinterfaces kan creëren. Verder is flutter ook goed geoptimaliseerd, waardoor berichten direct weergegeven en verstuurd kunnen worden.

## **Diagrammen project**

-Het Design Document bevat wireframes van de applicatie  
-Het Design Document bevat een flowdiagram van de applicatie  
A logo of a chat

Description automatically generated  
A screenshot of a login

Description automatically generatedA screenshot of a login

Description automatically generatedA screenshot of a phone

Description automatically generatedA screenshot of a phone

Description automatically generatedA screenshot of a chat

Description automatically generatedA screenshot of a chat

Description automatically generated   
Pc/Desktop

## **Testplan**

-Het Design Document bevat een testplan

## **Bronnen**

In dit hoofdstuk zullen alle bronnen neergezet worden die gebruikt zijn tijdens het realiseren van dit project.

# Bibliografie

Slava, T. (2019, February 22). *De voordelen en nadelen van Flutter*. Retrieved from moqod: https://moqod.com/nl/blog/de-voordelen-en-nadelen-van-flutter/