

Technische Dokumentation Friendly

„Entwicklung mobiler Anwendungen“

Abgabedatum: 31.01.22

Team: Giresse-Landry Djeufack-Tsapfack
Dominik Staemmler
Donald Tagne

Inhaltsverzeichnis

Was ist Friendly?	1
Komponenten.....	1
Activities	1
SplashScreen.....	1
MainActivity	1
SingupScreen	2
Interesse	3
IdentityVerificationScreen.....	3
Forget_pass	3
HomeScreen	3
Profilscreen.....	4
Privacy	4
Hobbysanpassung.....	4
Fragments.....	5
HomeFragment.....	5
ActivityFragment	5
FriendsFragment	5
ChatFragment.....	6

Was ist Friendly?

Friendly ist eine App die es einem ermöglicht sich mit Menschen zu treffen, die ähnliche Interessen haben wie man selbst. Sie soll dem User ermöglichen neue Freunde, die in der Nähe wohnen, kennenzulernen. Hierfür kann der User seine eigenen Interessen in seinem Profil hinterlegen. Basierend auf den eigenen Angaben werden Vorschläge für potenzielle Freunde und Aktivitäten gemacht. Des Weiteren ist es möglich nach Aktivitäten und Usern zu suchen.

Komponenten

Es folgen die wichtigsten Komponenten der App, diese wurden in die Kategorien Activities, Fragments und Resources unterteilt. Es werden ihre Pfade angegeben und kurz ihre Funktionen/Aufgaben beschrieben.

Activities

SplashScreen

Activity: `app/src/main/java/com/example/myfirstapplication/SplashScreen.kt`
Layout: `app/src/main/res/layout/activity_splash_screen.xml`

Diese Activity ist die Erste die man nach dem Starten der App sieht. Sie zeigt das Logo sowie den Namen der App an. Des Weiteren wird eine ProgressBar angezeigt. „SplashScreen“ ruft die Activity „MainActivity“ auf.

MainActivity

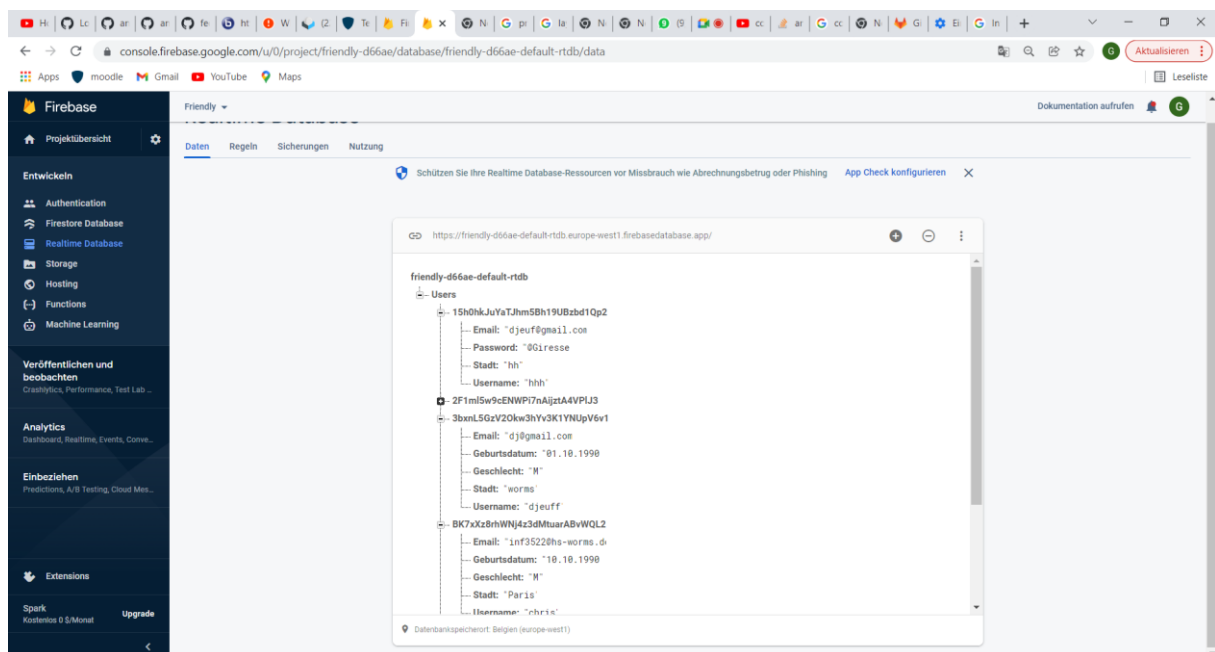
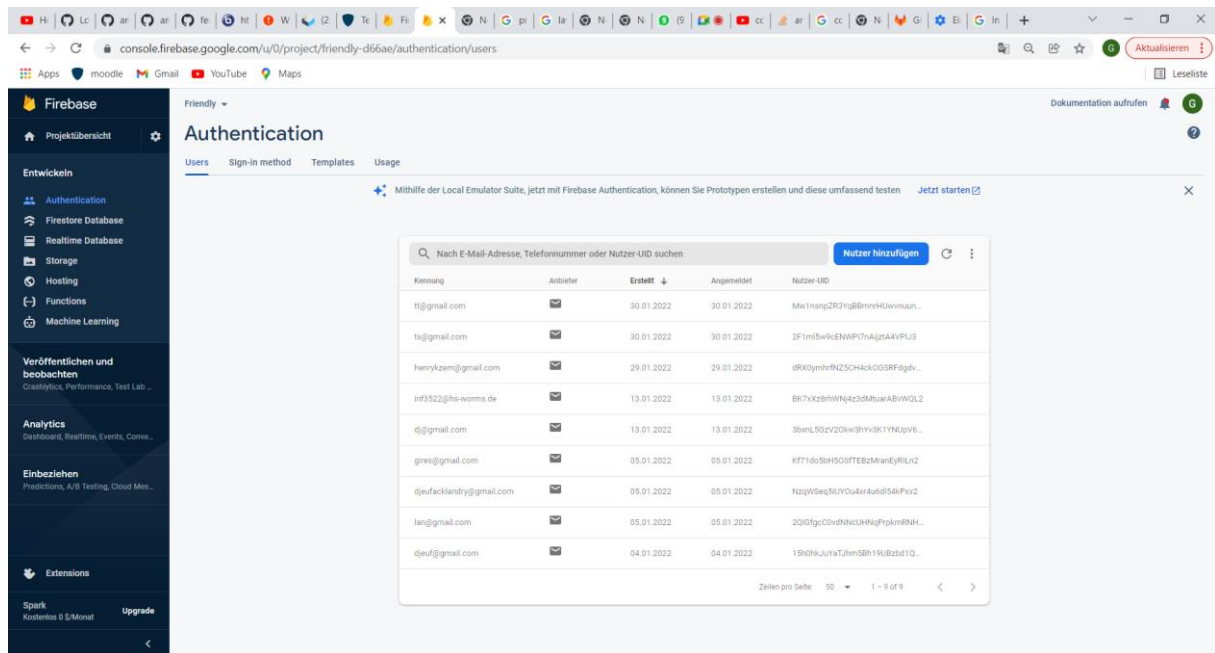
Activity: `app/src/main/java/com/example/myfirstapplication/MainActivity.kt`
Layout: `app/src/main/res/layout/activity_main.xml`

MainActivity stellt die Anmeldungsseite der App dar, hier kann der User seine Anmeldungsdaten wie E-Mail und Passwort eingeben um sich so zu authentifizieren. Ebenso kann er sich auf die Registrierungsseite weiterleiten lassen (SingupScreen). Falls er sein Passwort vergessen hat kann er die Activity „forget_pass“ aufrufen.

SingupScreen

Activity: `app/src/main/java/com/example/myfirstapplication/SingUpScreen.kt`
Layout: `app/src/main/res/layout/activity_sing_up_screen.xml`

SingupScreen stellt die Registrierungsseite der App dar. Der User kann hier seine Daten zur Registrierung eingeben wie Username, E-Mail, Passwort etc. Und diese dann mit dem Button „Weiter“ bestätigen, dadurch werden auch die E-Mail und Passwort authentifiziert. Die Daten werden dann an die Firebase-Realtime Database weitergeleitet. Der Button ruft dann die Activity „Interesse“ auf.



Interesse

Activity: `app/src/main/java/com/example/myfirstapplication/Interesse.kt`
Layout: `app/src/main/res/layout/activity_interesse.xml`

Mit der Activity „Interesse“ kann der User seine Interessen auswählen. Die Interessen sind in Hauptthemen unterteilt, die man in Form von Buttons auswählen kann. Mit den Buttons gelangt man dann auf Dialogfenster, in denen man mit Checkboxes seinen Interessen auswählen kann. Mit dem Button „Speichern“ gelangt man zur Activity „IdentityVerificationScreen“.

IdentityVerificationScreen

Activity: `app/src/main/java/com/example/myfirstapplication/IdentityVerificationScreen.kt`
Layout: `app/src/main/res/layout/activity_identity_verification_screen.xml`

Mit der Activity „IdentityVerificationScreen“ kann der User ein Bild von sich aufnehmen um seine Identität zu bestätigen. Es wird zuerst die Berechtigung, Bilder und Videos aufzunehmen, abgefragt. Nun hat man die Möglichkeit mit dem oberen Button seine Systemkamera zu öffnen. Nachdem man ein Bild aufgenommen hat wird dieses in einem ImageView angezeigt. Falls man dies überspringen will kann man mit dem Button „Weiter“ die Activity „MainActivity“ aufrufen.

Forget_pass

Activity: `app/src/main/java/com/example/myfirstapplication/forget_pass.kt`
Layout: `app/src/main/res/layout/activity_forget_pass.xml`

Mit der Activity „forget_pass“ kann der User seine E-Mail eingeben, falls er sein Passwort vergessen hat und mit dem Button „Weiter“ seine Eingabe bestätigen. Dem User wird dann eine E-Mail zum Zurücksetzen des Passwortes geschickt.

HomeScreen

Activity: `app/src/main/java/com/example/myfirstapplication/HomeScreen.kt`
Layout: `app/src/main/res/layout/activity_home_screen.xml`

Die Activity „HomeScreen“ besitzt eine BottomNavigation mit der man auf vier verschiedenen Fragments zugreifen kann „HomeFragment“, „ActivityFragment“, „FriendsFragment“ und „ChatFragment“.

Profilscreen

Activity: `app/src/main/java/com/example/myfirstapplication/Profilscreen.kt`
Layout: `app/src/main/res/layout/activity_profilscreen.xml`

Die Activity „Profilscreen“ stellt drei Buttons zur Verfügung die die Activities „activityprofil“, „Privacy“ und „hobbysanpassung“ aufrufen. Des Weiteren gibt es noch einen Button der den User abmeldet und einen weiteren Button mit dem der User sein Konto löschen kann(noch nicht implementiert). Dieser Button öffnet ein Dialogfenster wo der User den Grund der Löschung angeben kann. Die Activity besitzt eine Toolbar mit einem ImageButton der den User zurück zum Fragment „HomeFragment“ leitet.

Activityprofil

Activity: `app/src/main/java/com/example/myfirstapplication/activityprofil.kt`
Layout: `app/src/main/res/layout/activity_activityprofil.xml`

Die Activity „activityprofil“ zeigt die Daten des Users an, wie zum Beispiel E-Mail, Geschlecht, Geburtsdatum an. Die Daten werden von der Firebase Realtime Database entnommen. Mit einem FloatingActionButton kann man sich zur Activitiy „IdentityVerificationScreen“ weiterleiten lassen, um sein Profilbild zu aktualisieren.

Privacy

Activity: `app/src/main/java/com/example/myfirstapplication/Privacy.kt`
Layout: `app/src/main/res/layout/activity_privacy.xml`

Die Activity „Privacy“ ermöglicht es dem User Anpassungen machen, wer bestimmte Inhalte des eigenen Profils einsehen kann(noch nicht implementiert). Es stehen fünf Buttons zur Verfügung die jeweils ein Dialogfenster, mit den Einzel-Auswahlmöglichkeiten „None“, „Niemand“, „Freunde“, „Jeder“, aufrufen.

Hobbysanpassung

Activity: `app/src/main/java/com/example/myfirstapplication/hobbysanpassung.kt`
Layout: `app/src/main/res/layout/activity_hobbysanpassung.xml`

Mit der Activity „hobbysanpassung“ kann der User seine Interessen anpassen. Die Interessen sind in Hauptthemen unterteilt, die man in Form von ImageButtons auswählen kann. Mit den Buttons gelangt man dann auf Dialogfenster, in denen man mit Checkboxes seinen Interessen auswählen kann. Gespeichert werden diese mit dem „Speichern“-Button, so gelangt man auch zurück zur Activity „Profilscreen“.

Fragments

HomeFragment

Fragment: `app/src/main/java/com/example/myfirstapplication/fragments/HomeFragment.kt`
Layout: `app/src/main/res/layout/fragment_home.xml`

Das Fragment „HomeFragment“ besitzt ein `ImageView` in der Mitte, das nach Drücken eine `Ripple-Animation` startet und User in der Nähe anzeigen soll(nicht implementiert). Unten ist ein `Slider` platziert mit dem man den Radius der Suche einstellen kann(nicht implementiert). Oben ist eine `Toolbar` platziert, mit einem `ImageButton` der die Activity „Profilscreen“ aufruft.

ActivityFragment

Fragment: `app/src/main/java/com/example/myfirstapplication/fragments/ActivityFragment.kt`
Layout: `app/src/main/res/layout/fragment_activity.xml`
Adapter: `app/src/main/java/com/example/myfirstapplication/fragments/Activity1Adapter.kt`
 `app/src/main/java/com/example/myfirstapplication/fragments/Activity2Adapter.kt`

Das Fragment „ActivityFragment“ besitzt einen `Searchview` und zwei horizontale `RecyclerViews`, die jeweils eine Liste von `CardViews` enthalten. Die dazu erforderlichen Adapter liegen in dem Ordner „fragments“. Sie sind verantwortlich für das Verlinken der Daten mit dem `RecyclerView`. Der User soll hier nach Aktivitäten suchen können und es sollen Vorschläge für den User gemacht werden(noch nicht implementiert).

FriendsFragment

Fragment: `app/src/main/java/com/example/myfirstapplication/fragments/FriendsFragment.kt`
Layout: `app/src/main/res/layout/fragment_friends.xml`

Noch keine Funktion implementiert.

ChatFragment

Fragment: `app/src/main/java/com/example/myfirstapplication/fragments/ChatFragment.kt`
Layout: `app/src/main/res/layout/fragment_chat.xml`
Adapter: `app/src/main/java/com/example/myfirstapplication/fragments/ChatAdapter.kt`

„ChatFragment“ besitzt einen Searchview mit dem man einen RecyclerView durchsuchen kann(noch nicht implementiert). Der RecyclerView ist eine Liste von Usern, mit denen man bereits geschrieben hat (noch nicht implementiert). Der erforderliche Adapter liegt im Ordner „fragments“. Es werden hier zwei Sting-Datentypen mit dem RecyclerView verlinkt.