

# Projektni zadatak

Programiranje mobilnih sistema i uređaja, 2019/2020. godina

## Uvod

Vaš zadatak je da implementirate Android aplikaciju za rad sa elektronskom poštom - email aplikacija, po zahtevima specificiranim u ovom dokumentu.

## Funkcije sistema

Korisnički interfejs mora minimalno da se sastoji od: *SplashActivity*, *CreateEmailActivity*, *CreateFolderActivity*, *EmailsActivity*, *EmailActivity*, *FoldersActivity*, *FolderActivity*, *SettingsActivity*, *ProfileActivity*, *ContactsActivity*, *CreateContactActivity* i *ContactActivity*. Ove aktivnosti pružaju korisnicima sledeće funkcionalnosti:

- *SplashActivity* uvodi korisnika u aplikaciju tako što mu prikazuje logo aplikacije, ali i proverava da li je uređaj povezan na Wi-Fi ili mobilni internet. U slučaju da je korisnik povezan, nakon 5 sekundi biva prebačen na *LoginActivity*. Ako korisnik nije povezan na Wi-Fi ili mobilni internet, prikazuje se *Toast* ili *Snack* poruka da uređaj nije povezan na internet.
- *EmailsActivity* pruža korisniku uvid u sve *email*-ove za odabrani korisnički nalog u vidu liste. Svaki element liste treba da sadrži: datum prispeća poruke, kratak sadržaj poruke, predmet poruke i naziv pošiljaoca poruke, tagove koje ta poruka sadrži obojene različitim bojama. Ova aktivnost sadrži *Toolbar* sa akcijom za kreiranje nove *email* poruke. Klikom na ovo dugme, korisnik biva prebačen na *CreateEmailActivity*. *Toolbar* takođe sadrži i opciju *filter* gde korisnik može da unese tekst pretrage. U pretragu su uključeni: naslov poruke, sadržaj poruke i tagovi koje poruka sadrži, To, From, CC, BCC. Ova aktivnost sadrži i *NavigationDrawer* ili *NavigationView* iz koje korisnik može da izabere opciju pregleda profila - *ProfileActivity*, izmenu trenutno aktivnog naloga, podešenja aplikacije - *SettingsActivity* i pregled svih foldera za aktivan nalog - *FoldersActivity*. Klikom na neku od poruka iz liste, otvara se *EmailActivity* gde korisnik može da vidi detalje poruke.
- *CreateEmailActivity* pruža korisniku mogućnost da kreira novu *email* poruku. Korisnik treba da unese potrebne informacije (specificirane modelom aplikacije) nakon čega može da pošalje poruku ili da odustane od slanja. Ako korisnik odustane od slanja, poruka se briše. Ova aktivnost sadrži *Toolbar* sa dva dugmeta *send* i *cancel*. Aktivnost sadrži i opciju *back* gde se korisnik vraća na listu svih poruka, a nedovršena poruka biva automatski sačuvana u folder *Drafts* ako je korisnik popunio bar jedno polje.
- *EmailActivity* prikazuje korisniku sadržaj poruke koju je izabrao. Aktivnost sadrži *Toolbar* koji ima opcije *delete* gde korisnik može da obriše poruku, *reply* da odgovori na istu, *reply to all* odgovor na poruku ali svim učesnicima i *forward* gde korisnik može da prosledi sadržaj poruke na neku drugu adresu. Aktivnost sadrži opciju *back* gde se korisnik vraća na aktivnost koja prikazuje sve poruke, ali opciju da se sačuva prilog (save attachment).
- *FoldersActivity* prikazuje korisniku listu svih aktivnih foldera za aktivan nalog u vidu liste. Svaki element liste treba da sadrži naslov foldera, i trenutni broj poruka u tom folderu. Klikom na bilo koji folder otvara se aktivnosti *FolderActivity* koja prikazuje sadržaj tog foldera i osnovne informacije o tom folderu. Ova aktivnost sadrži *Toolbar* koji ima akciju *new* koja otvara *CreateFolderActivity* i korisnik može da kreira novi folder. Aktivnost takođe sadrži *NavigationView* ili *NavigationDrawer* gde korisnik može da pređe na druge delove aplikacije. Izborom jednog od foldera iz liste, korisniku se prikazuju sve poruke iz tog foldera.
- *FolderActivity* prikazuje sadržaj foldera u vidu liste poruka. Aktivnost sadrži i *Toolbar* sa opcijom *edit* čijim izborom korisnik može da izmeni podatke o folderu.
- *CreateFolderActivity* aktivnost omogućava korisniku da kreira novi folder sa svojstvima definisanim u modelu. Aktivnost sadrži *Toolbar* sa opcijom *save* koja čuva podatke o novom folderu i opciju *cancel*

tako da korisnik može da odustane od kreiranja foldera. Takođe sadrži i *back* dugme kojim se korisnik vraća na aktivnost koja prikazuje listu svih foldera.

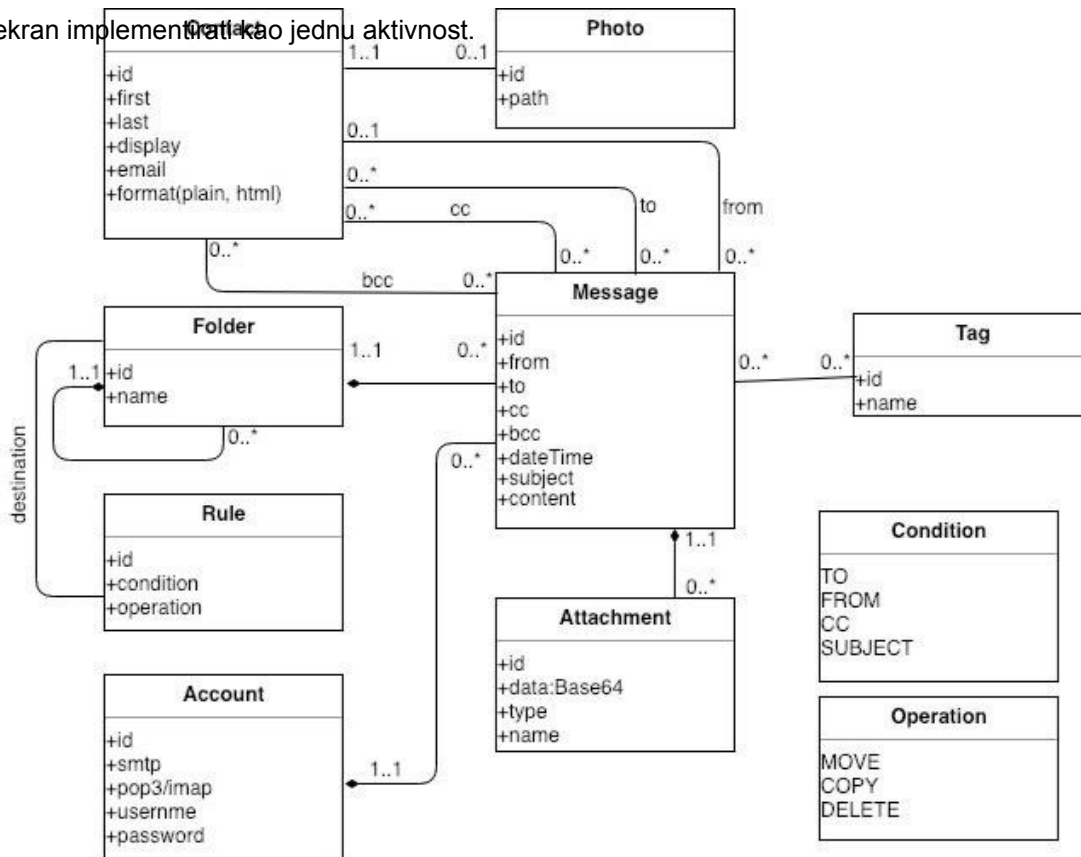
- *ProfileActivity* pruža korisniku informacije o svom profilu definisane modelom. Ova aktivnost sadrži *Toolbar* sa opcijom *logout* na koju korisnik može da se odjavi sa sistema. Izborom ove opcije korisnik biva prebačen na *LoginActivity*. Sadrži *NavigationDrawer* ili *NavigationView* gde korisnik može da pređe na druge delove aplikacije. Ova aktivnost omogućava korisniku da specificira pravila pod ko-jim će prispele poruke odlaziti u određeni folder.
- *SettingsActivity* omogućava korisniku osnovna podešenja aplikacije. Korisnik može da izabere vreme osvežavanja novih poruka izborom unapred specificiranog intervala. Sadrži opciju sortiranja poruka po datumu (rastuće/opadajuće). Aktivnost sadrži *Toolbar* i *back* button čime se korisnik vraća na prethodnu aktivnost.
- *ContactsActivity* prikazuje listu svih kontakata za prijavljenog korisnika. Kontakti se prikazuju u vidu liste, a svaki kontakt u listi treba da ima sliku i naziv. Klikom na bilo koji kontakt u listi otvara se nova aktivnost *ContactActivity*, gde korisnik može da vidi detalje ili izmeni podatke o kontaktu. Aktivnost sadrži *Toolbar* sa opcijom *new* nakon čega se otvara *CreateContactActivity*. Aktivnost sadrži i *NavigationDrawer* ili *NavigationView* gde korisnik može da pređe na druge delove aplikacije.
- *ContactActivity* prikazuje informacije o pojedinačnom kontaktu. Aktivnost sadrži *Toolbar* sa opcijom *save* čijim klikom se radi izmena podataka o kontaktu, i opciju *back* čime korisnik biva vraćen na prethodnu aktivnost.
- *CreateContactActivity* korisniku pruža mogućnost da doda novi kontakt popunjavajući podatke specificirane modelom. Aktivnost sadrži *Toolbar* sa opcijom *save* i *cancel* kojom korisnik čuva, odnosno odustaje od dodavanja novog kontakta i vraća se na prethodnu aktivnost.

## Model podataka

Model podataka je dat sledećim *UML* dijagramom. Na osnovu modela potrebno je kreirati Java klase, koje je potrebno strukturirati u odgovarajuće pakete.

## Struktura sistema

Svaki ekran implementira kao jednu aktivnost.



Strukturiranim podacima i fotografijama pristupati koristeći *ContentProvider* koji podatke skladišti u:

- A. lokalno (tako što se strukturirani podaci skladište u *SQLite* bazi podataka, a fotografije na eksternom skladištu podataka)
- B. na mreži (tako što se podacima pristupa pomoću *RESTful* web servisa koje ćete implementirati iz Osnova softverskih arhitektura)

Podešavanja aplikacije čuvati u *SharedPreferences*.

Za izbor fotografija koristiti *Photos/Camera* aplikaciju.

## Napomene

- Možete iskoristiti bilo koju biblioteku koja će vam ubrzati posao, ali morate znati da objasnite zašto i kako je koristite!
- Studenti koji odluče da aplikaciju ne povezuju sa *RESTful* web servisima, moraju da implementiraju i odgovorajuće *ContentProvider*-e. Nije dovoljno samo podacima pristupati koristeći *SQLite* ili *ORMLight* mapper!