

Dokumentacja

aplikacja mobilna

FootballAPP – aplikacja dla drużyny piłkarskiej

1. Cele projektu

Celem projektu było stworzenie aplikacji, która umożliwia dodawania użytkowników, zawodników do bazy danych oraz wykonanie funkcji do liczenia statystyk. Dzięki tej funkcji, użytkownik ma możliwość liczenia zdobytych punktów oraz popełnionych fauli przez drużyny.

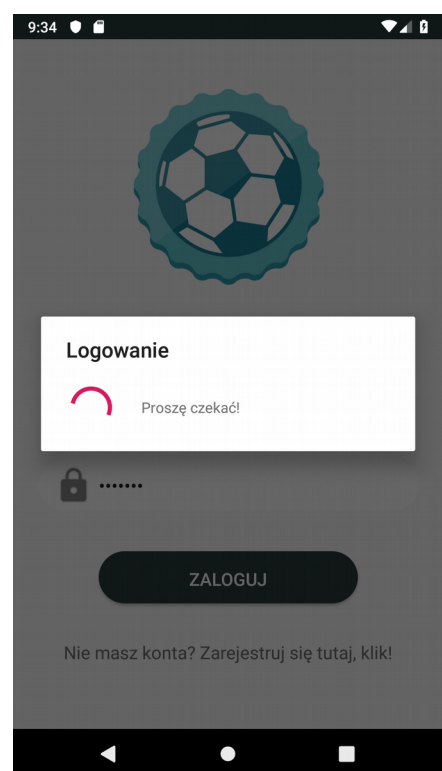
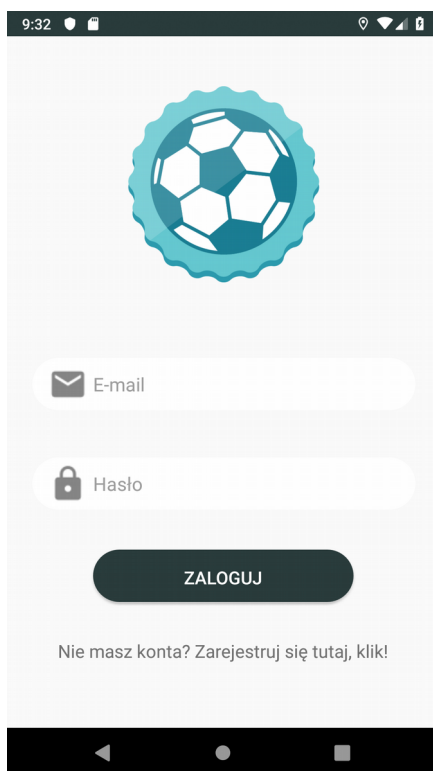
2. Wykorzystane technologie

Aplikację została utworzona w środowisku programistycznym Android Studio. Logiczna część aplikacji napisana została w języku Java, natomiast część graficzna została zaimplementowana plikach XML. Jako bazy danych skorzystałem z Firebase.

3. Opis projektu

Aplikacja składa się z 5 activities, 1 klasy oraz 7 plików layout.

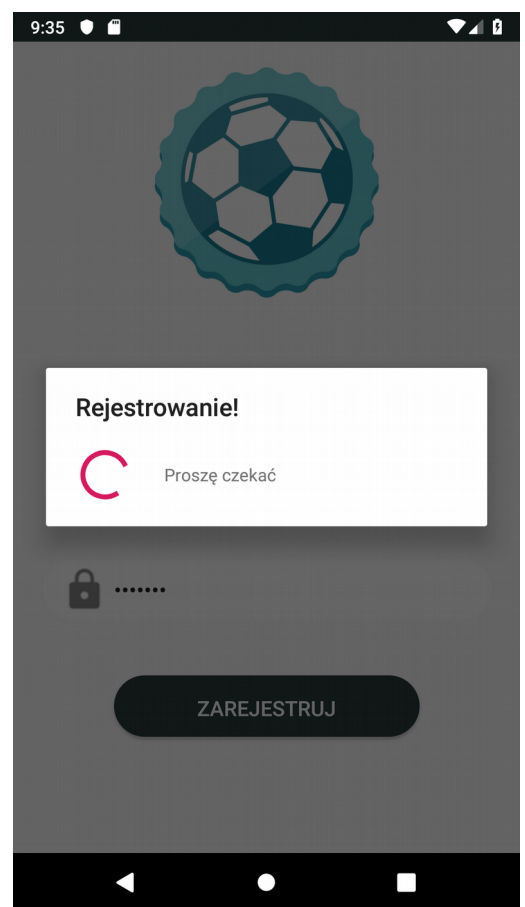
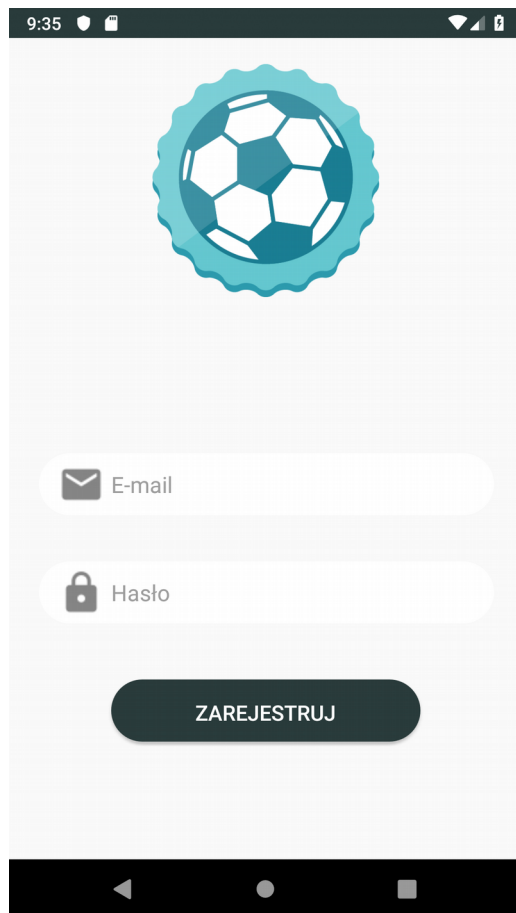
Ekran Startowy (LoginActivity)



Na sam początek, użytkownika wita ekran logowania. Po wpisaniu odpowiednich danych, przycisk zaloguj przenosi nas do głównego ekranu (MainActivity)

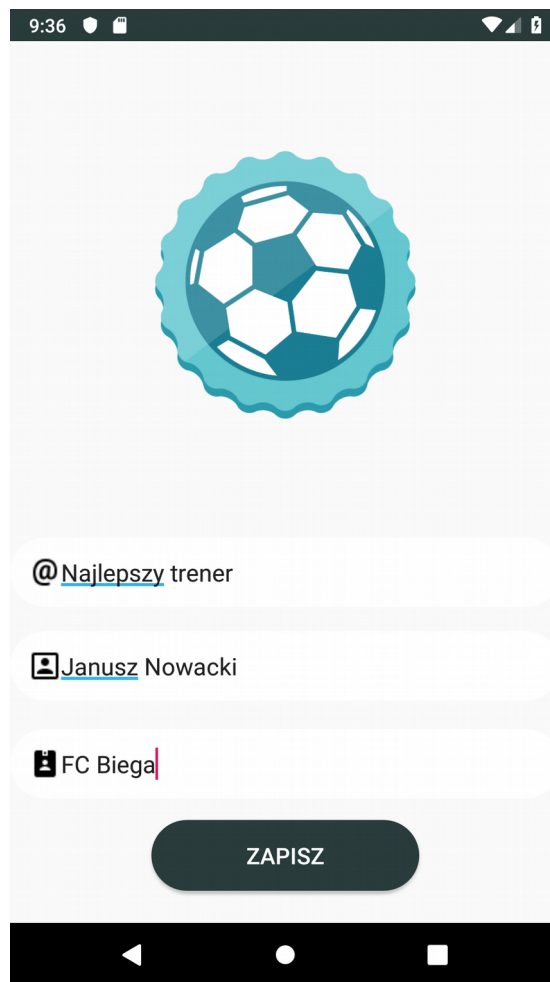
Napis „Nie masz konta?... , klik!” służy do wyświetlenia ekranu odpowiadającego za rejestrację do bazy danych (RegisterActivity)

Ekran logowania (RegisterActivitiy)




Po wpisaniu danych i wciśnięciu przycisku „ZAREJESTRUJ”, dane wysyłane są do bazy FireBase a aplikacja przenosi użytkownika do MainActivity


Ekran profilowy (SetupActivity)




9:36



@Najlepszy trener

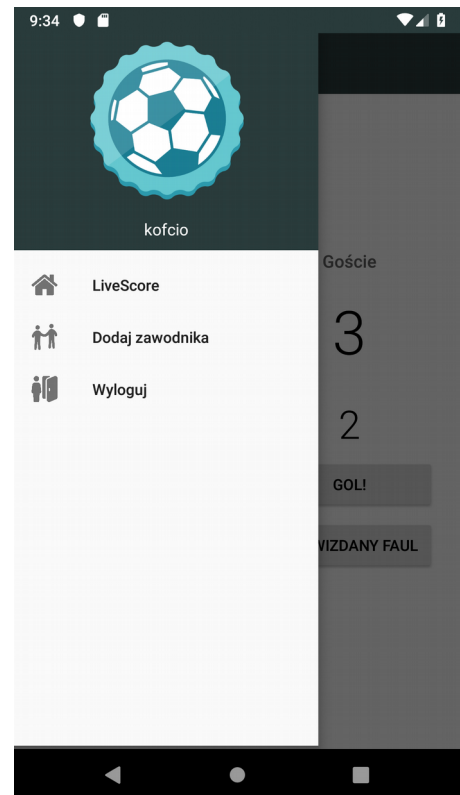
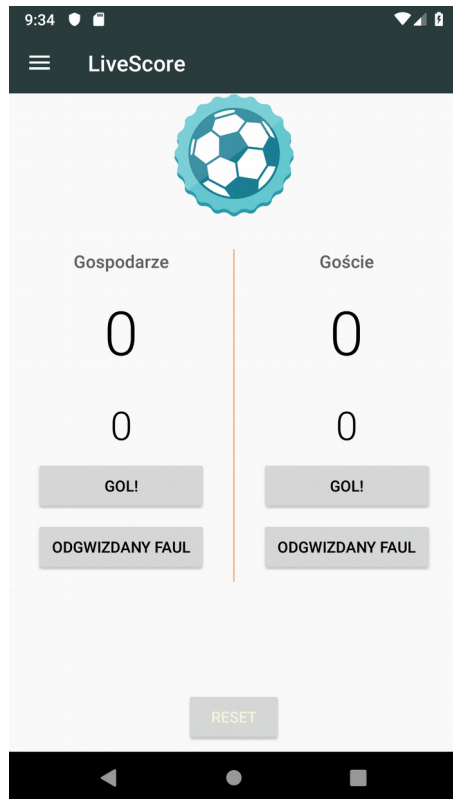
 Janusz Nowacki

 FC Biega

ZAPISZ

Ekran profilowy oferuje wpisanie do bazy danych pseudonim, imię i nazwisko oraz ulubioną drużynę użytkownika. Po wciśnięciu przycisku „ZAPISZ”, aplikacja przenosi użytkownika do Ekranu głównego.

Ekran główny (MainActivity)

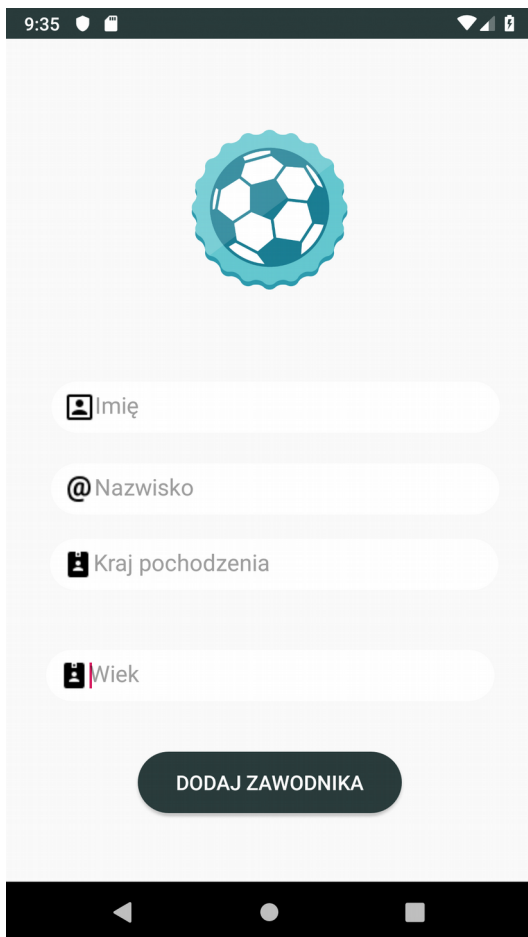


Na ekranie głównym znajduje się funkcja zliczenia statystyk zatytułowana „LiveScore”.


Przyciskiem reset, usuwamy wcześniej wpisane statystki z ekranu.


Ekran główny oferuje przycisk „hamburger”, dzięki któremu wysuwa się z lewej strony ekranu menu. W wysuwanym menu mamy do wyboru dwie funkcje. Funkcja dodania użytkownika do bazy danych oraz przycisk wyloguj, który kończy sesję użytkownika i przenosi go do ekranu logowania.


Ekran dodawania zawodników (PlayerAddActivity)





9:35



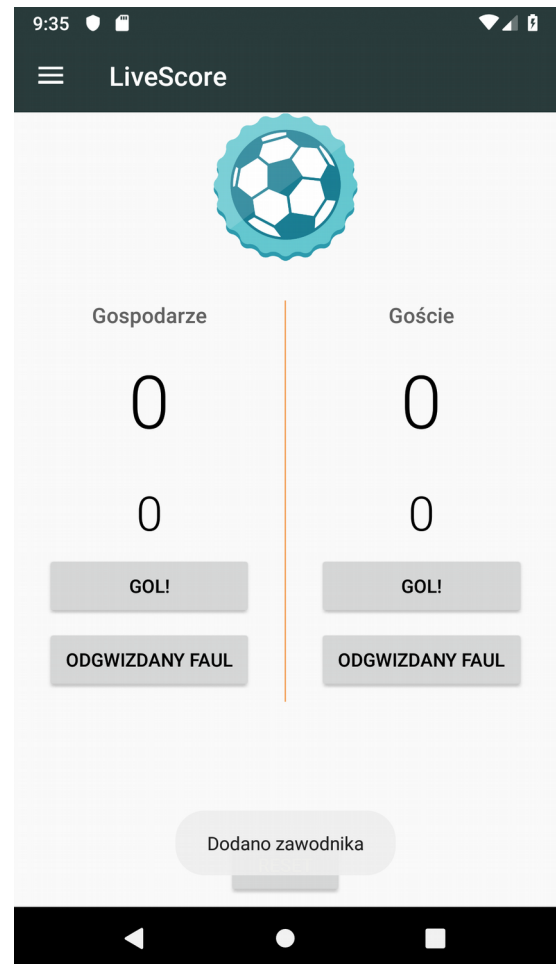
 Imię

 @Nazwisko


 Kraj pochodzenia


 Wiek

DODAJ ZAWODNIKA



9:35

 LiveScore



Gospodarze	Goście
0	0
0	0
GOL!	GOL!
ODGWIZDANY FAUL	ODGWIZDANY FAUL

Dodano zawodnika

Ekran dodawania zawodników umożliwia dodanie zawodnika do bazy danych. Użytkownik zobligowany jest do wpisania podstawowych danych zawodnika tj. Imię, Nazwisko, Kraj pochodzenia oraz wiek. Przyciskiem „DODAJ ZAWODNIKA” potwierdza wpisane dane a aplikacja przenosi go do ekranu głównego. Na ekranie wyświetla się potwierdzenie dodania.

```

private void CreateNewAccount()
{
    String email = UserMail.getText().toString();
    String password = UserPassword.getText().toString();
    if(TextUtils.isEmpty(email))
    {
        Toast.makeText(this, "Wpisz E-Mail!", Toast.LENGTH_SHORT).show();
    }
    else if(TextUtils.isEmpty(password))
    {
        Toast.makeText(this, "Wpisz hasło!", Toast.LENGTH_SHORT).show();
    }
    else
    {
        loadingBar.setTitle("Rejestrowanie!");
        loadingBar.setMessage("Proszę czekać");
        loadingBar.show();
        loadingBar.setCanceledOnTouchOutside(true);
        mAuth.createUserWithEmailAndPassword(email,password)
            .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                @Override
                public void onComplete(@NonNull Task<AuthResult> task)
                {
                    if(task.isSuccessful())
                    {
                        SendUserToSetupActivity();
                        Toast.makeText(RegisterActivity.this,
"Zarejestrowano!", Toast.LENGTH_SHORT).show();
                        loadingBar.dismiss();
                    }
                    else
                    {
                        String message = task.getException().getMessage();
                        Toast.makeText(RegisterActivity.this, "Błąd: " +
message, Toast.LENGTH_SHORT).show();
                        loadingBar.dismiss();
                    }
                }
            });
    }
}

```

Funkcja ta umożliwia stworzenie oraz rejestrację konta użytkownika. Funkcja „Toast” odpowiada za ostrzeżenie dla użytkownika, gdy ten nie wpisze e-maila bądź hasła. Wyskakuje okienku w dolnej części ekranu. Funkcja loadingBar służy do wyświetlenia ekranu ładowania w trakcie wysyłania danych do bazy. Metoda createUserWithEmailAndPassword wpisuje dane do bazy.

Jeżeli wszystko zostało wpisane poprawnie, na ekranie zostanie wyświetlony komunikat „Zarejestrowano” dzięki funkcji onComplete.

```

private void SendUserToSetupActivity()
{
    Intent setupIntent = new Intent(RegisterActivity.this, SetupActivity.class);
    setupIntent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK |
Intent.FLAG_ACTIVITY_CLEAR_TASK);
    startActivity(setupIntent);
    finish();
}

```

Funkcja, która służy do wysyłania użytkownika do kolejnych ekranów po skorzystaniu z danej funkcji.

```

protected void onStart() {
    super.onStart();
    FirebaseUser currentUser = mAuth.getCurrentUser();
    if(currentUser != null)
    {
        SendUserToMainActivity();
    }
}

```

Funkcja, odpowiadająca za wysłanie użytkownika do ekranu głównego gdy użytkownik był zalogowany przed uruchomieniem aplikacji.