Proiect MDS

Membrii (241):

Masgras Vlad

Simionescu Cosmin

Descriere

Aplicatia dezvoltata este destinata telefoanelor Android si vrea sa vina in sprijinul calatorilor cu trenul.

Ca functionalitate de baza, utilizatorul poate cauta rute de tren in functie de orasele de plecare si de destinatie si isi poate selecta una din rute in functie de orar. De asemenea, un utilizator poate sa vada toate biletele rezervate de pe contul pe care este logat.

Exista si o parte la care are acces administratorul, de unde se pot introduce trenuri cu rutele asociate, in baza de date.

User stories

**US: Abilitatea sa ma inregistrez cu un email personal**

Ca utilizator, imi doresc sa imi pot crea un cont de pe care sa interactionez cu aplicatia

**US:Abilitatea sa ma loghez ca utilizator**

Ca utilizator, imi doresc sa ma pot loga, eventual de pe conturi diferite.

**US:Abilitatea sa ma loghez ca administrator**

Ca administrator, imi doresc sa am un tip de cont special, de pe care sa configurez baza de date

**US:Abilitatea de a adauga trenuri in baza de date**

Ca administrator, imi doresc sa pot adauga trenuri in baza de date, in functie de ruta si program.

**US:Abilitatea de a adauga vagoane unui tren.**

Ca administrator, imi doresc sa pot adauga un numar de vagoane unui tren, unde fiecare vagon sa aiba un numar de locuri asociat.

**US:Abilitatea de a cauta curse de tren.**

Ca utilizator, imi doresc ca introducand o statie de plecare si una de destinatie sa mi se afiseze toate trenurile care efectueaza ruta respectiva si programul asociat lor.

**US:Abilitatea de a rezerva bilete**

Ca utilizator, imi doresc sa pot rezerva un bilet sau chiar mai multe pe o ruta aleasa de mine.

**US:Abilitatea de a vedea bilete rezervate de mine**

Ca utilizator, imi doresc sa vad toate biletele pe care le-am rezervat de-a lungul timpului.

**US:Atentionare la introducerea datelor de login si inregistrare.**

Ca utilizator, imi doresc sa fiu atentionat daca am facut vreo greseala la introducerea datelor(parola prea scurta).

**US:Posibilitatea de a ma deloga ca sa intru pe un alt cont**

Ca utilizator, imi doresc sa ma pot deloga de pe contul curent.

Bug reporting

Bug 1:

Indiferent ca ma logam ca utilizator sau ca admin, aplicatia imi afisa meniul de admin.

Bug 2:

Cand incercam sa ocup ultimul loc dintr-un vagon, aplicatia imi ocupa acest loc + primul loc din urmatorul vagon.

Bug 3:

Atunci cand rezerv un bilet, desi operatia iese cu succes si imi este rezervat biletul corect, aplicatia imi afiseaza un mesaj cum ca nu ar mai fi locuri.

Diagrame UML

Baza de date

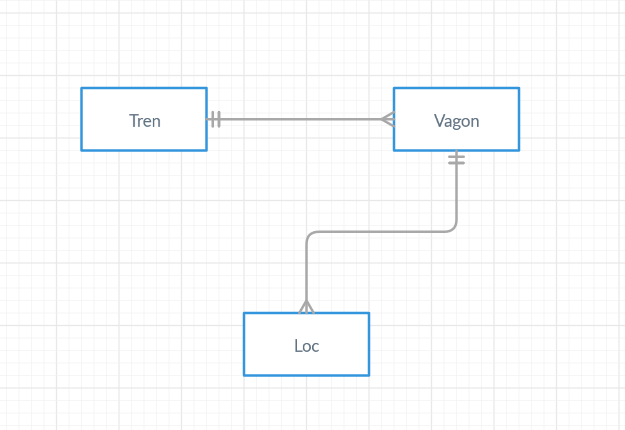
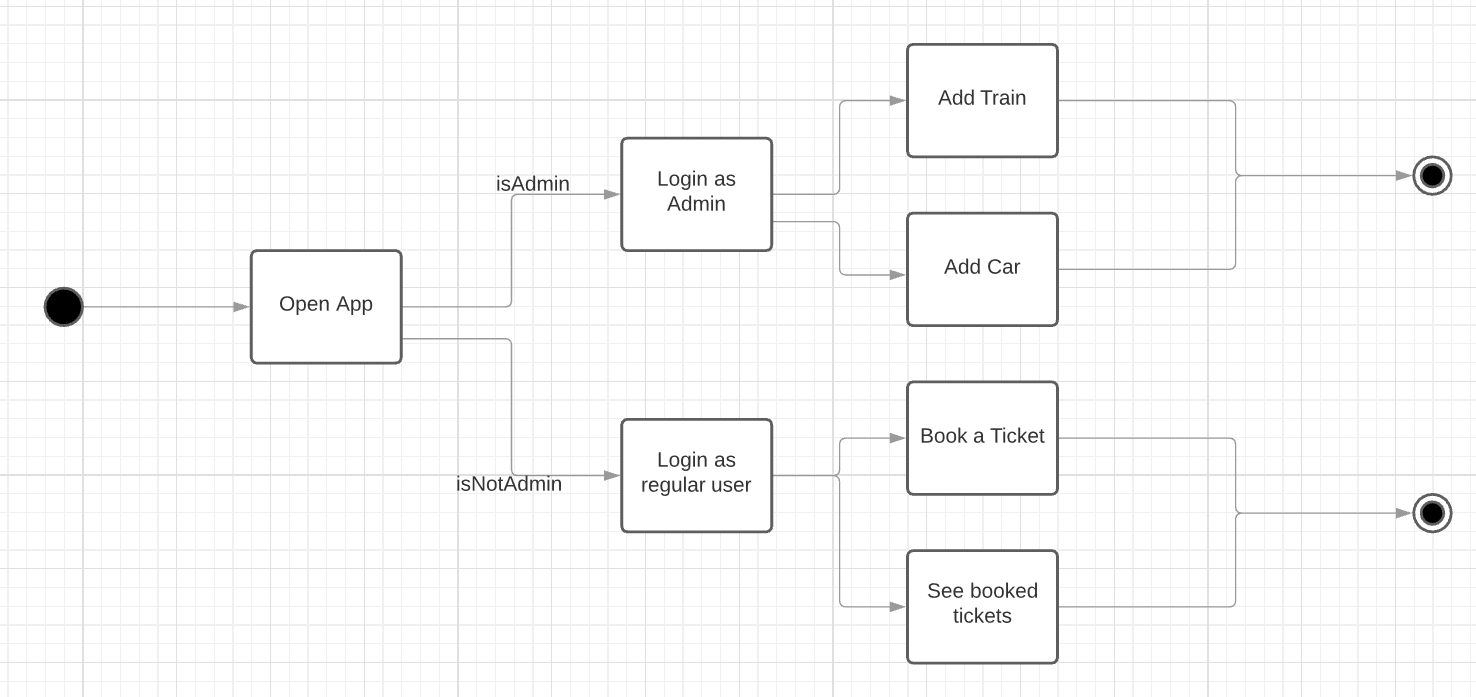


Diagrama Stari



Source control

Github: <https://github.com/VladMasgras/proiect-mds>

Teste

@Test  
public void testLaunchTextFields(){  
 View email = loginActivity.findViewById(R.id.*text\_input\_email\_login*);  
 View pass = loginActivity.findViewById(R.id.*text\_input\_password\_login*);  
  
 *assertNotNull*(email);  
 *assertNotNull*(pass);  
}  
  
@Test  
public void testLogin(){  
 View email = loginActivity.findViewById(R.id.*text\_input\_email\_login*);  
 View pass = loginActivity.findViewById(R.id.*text\_input\_password\_login*);  
  
 String mailInput = "andrei@gmail.com";  
 String passInput = "123456";  
  
 Espresso.*onView*(*withId*(R.id.*email\_login*)).perform(*typeText*(mailInput));  
 Espresso.*closeSoftKeyboard*();  
 Espresso.*onView*(*withId*(R.id.*pass\_login*)).perform(*typeText*(passInput));  
 Espresso.*closeSoftKeyboard*();  
  
 Espresso.*onView*(*withId*(R.id.*email\_sign\_in\_button*)).perform(*click*());  
 Activity meniuAdmin = *getInstrumentation*().waitForMonitorWithTimeout(monitorAdmin, 5000);  
  
 *assertNotNull*(meniuAdmin);  
  
}

@Test  
public void testLaunchAddVagoane(){  
 *assertNotNull*(meniuAdmin.findViewById(R.id.*admin\_adauga\_vagon*));  
 *onView*(*withId*(R.id.*admin\_adauga\_vagon*)).perform(*click*());  
  
 Activity addVagon = *getInstrumentation*().waitForMonitorWithTimeout(monitorVag, 5000);  
  
 *assertNotNull*(addVagon);  
 addVagon.finish();  
  
}  
  
@Test  
public void testLaunchAddTren(){  
 *assertNotNull*(meniuAdmin.findViewById(R.id.*admin\_adauga\_tren*));  
 *onView*(*withId*(R.id.*admin\_adauga\_tren*)).perform(*click*());  
  
 Activity addTren = *getInstrumentation*().waitForMonitorWithTimeout(monitorTren, 5000);  
  
 *assertNotNull*(addTren);  
 addTren.finish();  
}

@Test  
public void testTren(){  
 Tren tren = new Tren(1234, 200, "Constanta", "18:00", "Bucuresti","20:00");  
 *assertEquals*(1234, tren.getSerie());  
 *assertEquals*(200, tren.getDistanta());  
 *assertEquals*("Constanta", tren.getStatiePlecare());  
 *assertEquals*("18:00", tren.getOraPlecare());  
 *assertEquals*("Bucuresti", tren.getStatieSosire());  
 *assertEquals*("20:00", tren.getOraSosire());  
}

@Test  
public void testList(){  
  
 ListView list = vBileteActivity.findViewById(R.id.*listviewBilete*);  
 *assertNotNull*(list);  
}

Build tool

Am folosit Gradle, care e inclus in Android Studio

Continut build.gradle pentru proiect

// Top-level build file where you can add configuration options common to all sub-projects/modules.  
  
buildscript {  
   
 repositories {  
 google()  
 jcenter()  
   
 }  
 dependencies {  
 classpath 'com.android.tools.build:gradle:3.6.3'  
 classpath 'com.google.gms:google-services:4.3.3'  
 classpath ("org.junit.platform:junit-platform-gradle-plugin:1.0.0-M4")  
  
 // NOTE: Do not place your application dependencies here; they belong  
 // in the individual module build.gradle files  
 }  
}  
  
allprojects {  
 repositories {  
 google()  
 jcenter()  
   
 }  
}  
  
task clean(type: Delete) {  
 delete rootProject.buildDir  
}

Continut build.gradle pentru modulul principal

apply plugin: 'com.android.application'  
apply plugin: 'com.google.gms.google-services'  
  
android **{** compileSdkVersion 29  
 buildToolsVersion "29.0.3"  
  
 defaultConfig **{** applicationId "com.example.proiect"  
 minSdkVersion 21  
 targetSdkVersion 29  
 versionCode 1  
 versionName "1.0"  
  
 testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"  
 **}** buildTypes **{** release **{** minifyEnabled false  
 proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'  
 **}  
 }  
  
}**dependencies **{** implementation fileTree(dir: 'libs', include: ['\*.jar'])  
  
 implementation 'androidx.appcompat:appcompat:1.1.0'  
 implementation 'androidx.constraintlayout:constraintlayout:1.1.3'  
 implementation 'com.google.firebase:firebase-database:16.0.4'  
 implementation 'androidx.appcompat:appcompat:1.1.0'  
 implementation 'androidx.constraintlayout:constraintlayout:1.1.3'  
 implementation 'com.google.firebase:firebase-auth:16.0.5'  
 testImplementation 'junit:junit:4.12'  
 androidTestImplementation 'androidx.test.ext:junit:1.1.1'  
 androidTestImplementation 'androidx.test.espresso:espresso-core:3.2.0'  
 implementation 'com.google.android.material:material:1.1.0'  
 androidTestImplementation 'androidx.test:rules:1.2.0'  
 androidTestImplementation 'androidx.test:runner:1.2.0'  
  
 androidTestImplementation 'com.android.support:support-annotations:28.0.0'  
  
 androidTestImplementation 'com.android.support.test:runner:1.0.2'  
  
 androidTestImplementation 'com.android.support.test:rules:1.0.2'  
  
  
**}**