

•**Descripció de l'aplicació:**

Amb aquesta aplicació podràs fer Backups de les teves playlists de Spotify a partir de la teva ID d'usuari podràs accedir i veure les teves playlists(publiques) un cop dins seleccionaràs si vols fer una backups o retornar-ne una a partir d'un fitxer de text.

•**A qui va dirigida:**

L'aplicació va dirigida a tot el públic que utilitza Spotify(joves-gent gran).

•**Tipus de dispositius:**

L'aplicació està dirigida per Smartphones i Desktops.

•**Requisits:**

- 1.-El sistema ha de ser fàcil d'utilitzar
- 2.-La interfície té que ser simple.
- 3.-El sistema ha de ser multilingüe.
- 4.-El sistema ha de registrar bé les dades de les Backups (Local).
- 5.-El sistema ha de tenir en compte al restaurar les Backups (Format de fitxer que es passi sigui el accepta l'aplicació).
- 6.-El sistema ha de tenir en compte si l'Id de l'usuari és correcta.
- 7.-Que la interfície s'adapti bé amb la mida del dispositiu.
- 8.-L'aplicació ha de ser lleugera.
- 9.-L'aplicació ha de tenir un rendiment ràpid a l'hora de fer/restaurar les Backups.
- 10.-L'aplicació no ha de consumir molts de recursos.

•**Llistat de proves per tal de testar el correcte funcionament de l'aplicació:**

1.Proves Internes:

1.1.-Prova de Caixa negra – A l'hora d'iniciar sessió comprovar que la Id pública de l'usuari existeix realment en cas contrari no deixi accedir a la següent pantalla.

1.2.-Prova de Caixa negra – A l'hora de llegir les Backups que el programa que hi ha escollit (es pot canviar a configuració) estigui instal·lat en el dispositiu ja sigui Desktop o Smartphone.

1.3.-Revisions – (walktroughs) Diferents revisors revisaran l'aplicació fent Backups i restaurant-les en cas de que hi hagi algun error serà comunicat i en cas de que no pugui tenir una resolució es plantejarà l'opció de notificaró com un tipus d'error a l'usuari.

2.Proves Externes:

2.1.-Se li donarà l'aplicació un grup de persones que tinguin Spotify sense cap tipus d'instrucció de com funciona l'aplicació per saber si la saben utilitzar i així poder comprovar que l'aplicació es intuïtiva.

2.2.-Es reunirà a un grup de persones (gent gran) i se'ls i explicarà l'aplicació se'ls i deixarà un temps perquè l'utilitzin i donin un feedback quin es la part de la interfície que han trobat menys intuïtiva o que en cas de que no haguessin tingut instruccions no haguessin sapigut fer.

2.3.-Per acabar es reunirà un grup de persones i se'ls i donarà a cada meitat un tipus de tasca concreta en l'aplicació (Tasca 1 – Fer una backups d'alguna de les teves playlists) i (Tasca 2 – Restaurar una Playlists a partir d'una backup que se'ls i serà distribuïda).

Testos a JUNIT5:

He fet 6 testos en total 2 a cada funció que he trobat que necessitava ser comprovada ja que sinó el programa aniria malament. Tot seguit al veure que els testos em donaven errors he anat modificant el codi fins que l'aplicació passes tots els testos:

·Testos 1 i 2 (Comproven la funció **llegirFitxer**)«S'utilitza al restaurar una Backup»:

```
public class SeconSceneControllerTest {

    //TESTOS PER LLEGIR FITXERS
    no usages
    @Test
    public void test1() throws IOException {
        SecondSceneController secondSceneController=new SecondSceneController();
        String[] resultat =secondSceneController.llegirFitxer(new File( pathname: "C:/d.json"));
        Assertions.assertEquals( expected: null,resultat);
    }

    no usages
    @Test
    public void test2() throws IOException {
        SecondSceneController secondSceneController=new SecondSceneController();
        String[] resultat =secondSceneController.llegirFitxer(new File( pathname: "C:\\Users\\iferr\\OneDrive\\Escritori\\nuevo.txt"));
        Assertions.assertNotEquals( unexpected: null,resultat);
    }
}
```

·Testos 3 i 4 (Comproven la funció **escriureFitxer**)«S'utilitza al fer una Backup»:

```
no usages
@Test
public void test2() throws IOException {
    SecondSceneController secondSceneController=new SecondSceneController();
    String[] resultat =secondSceneController.llegirFitxer(new File( pathname: "C:\\Users\\iferr\\OneDrive\\Escritori\\nuevo.txt"));
    Assertions.assertNotEquals( unexpected: null,resultat);
}

//TESTOS PER ESCRIURE FITXERS
no usages
@Test
public void test3() throws IOException {
    SecondSceneController secondSceneController=new SecondSceneController();
    boolean resultatEsperat=false;
    Boolean resultat =secondSceneController.escriureFitxer(new File( pathname: "C:/backupMusiques.json"),new HashMap<String, String>());
    Assertions.assertEquals(resultatEsperat,resultat);
}
}
```

·Testos 5 i 6 (Comproven la funció **remplenaUsuari**) «S'utilitza al iniciar sessió a la conta»:

```
//TESTOS PER COMPROBAR SI EL ID USUARI A ESTAT ENTRAT
no usages
@Test
public void test5(){
    MainController mainController=new MainController();
    usuari usuariEsperat=null;
    usuari usuariResultat =mainController.remplenaUsuari( id: null);
    Assertions.assertEquals(usuariEsperat,usuariResultat);
}

no usages
@Test
public void test6(){
    MainController mainController=new MainController();
    usuari usuariNoEsperat=null;
    usuari usuariResultat =mainController.remplenaUsuari( id: "qt0gu1lmwo1t703dsojfkf839i");
    Assertions.assertNotEquals(usuariNoEsperat,usuariResultat);
}
}
```

·Resultats dels testos passats:

SeconSceneControllerTest			Tests passed: 6 of 6 tests – 1 sec 681 ms
✓ SeconSceneControllerTest	1 sec 681 ms	"C:\Program Files\Java\jdk-17.0.4.1\bin\java.exe" ...	
✓ test1	803 ms	Process finished with exit code 0	
✓ test2	3 ms		
✓ test3	1 ms		
✓ test4	4 ms		
✓ test5	4 ms		
✓ test6	866 ms		