

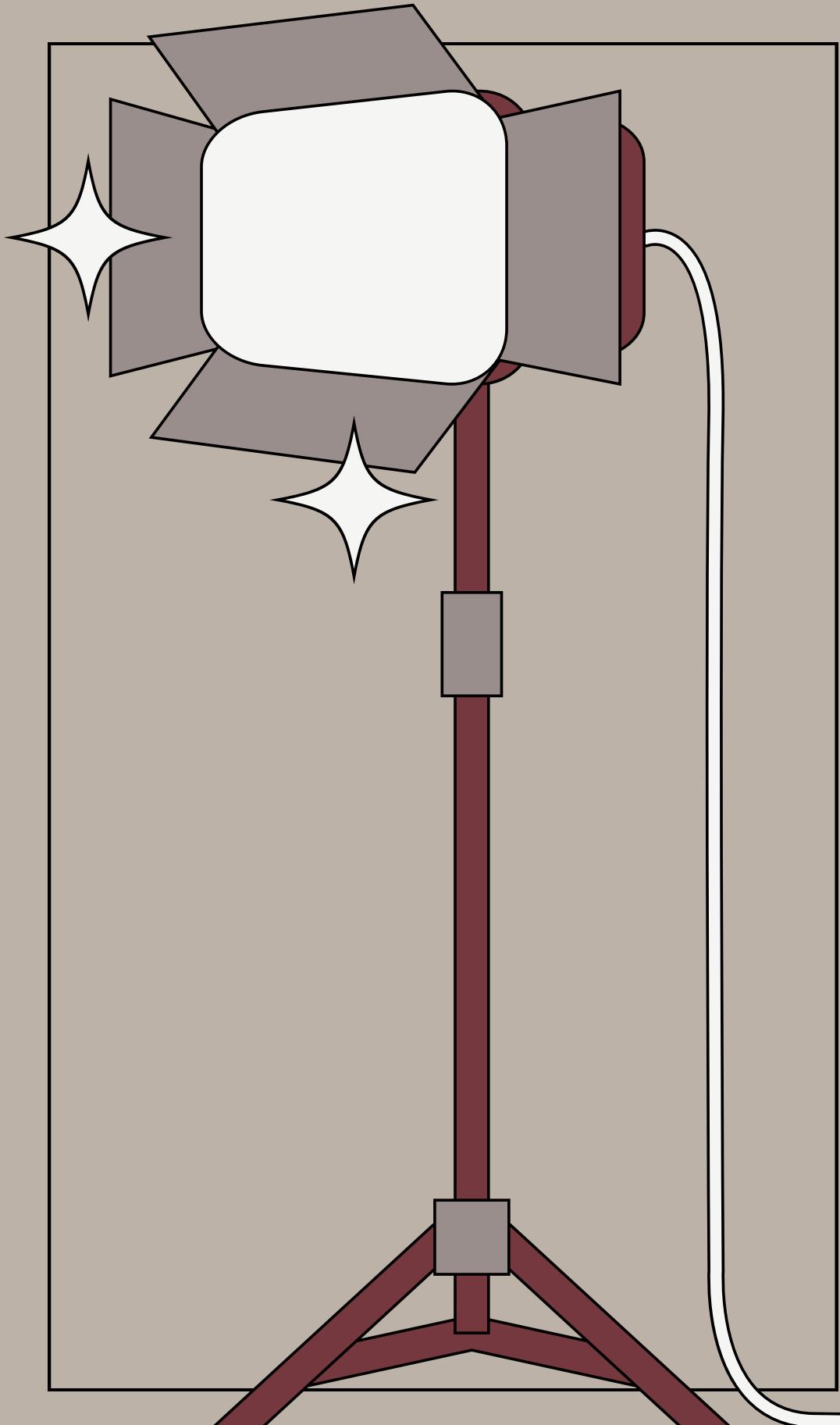


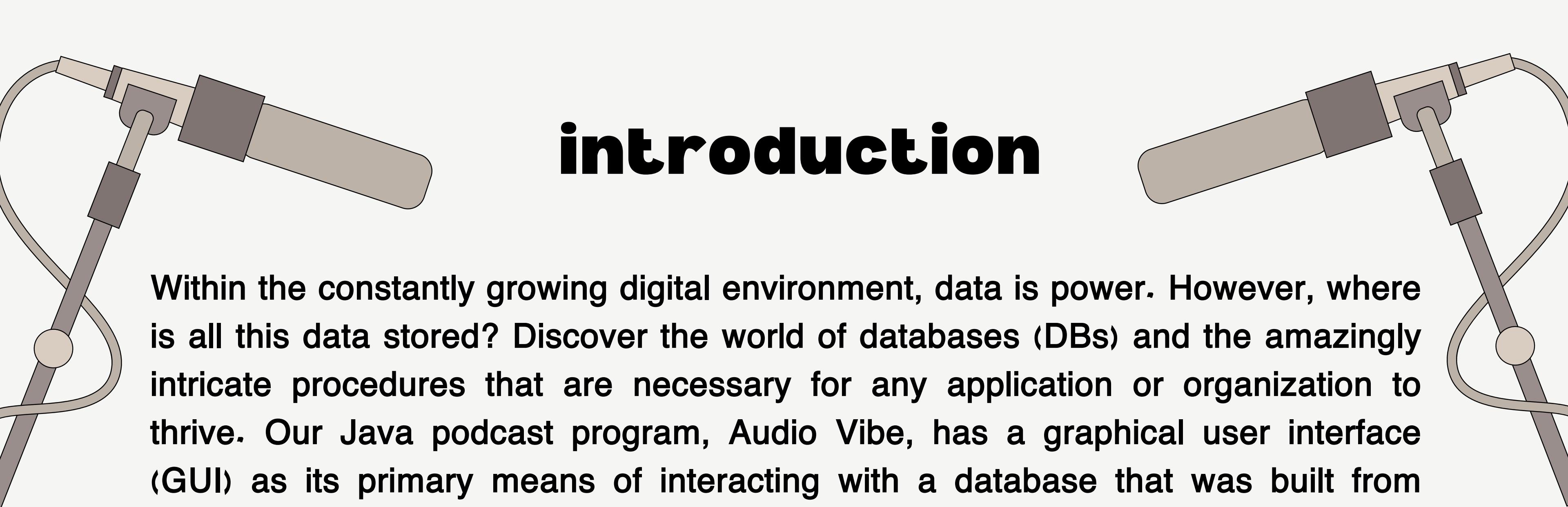
# **project OOP2& Database**

## AudioVibe

# Group members:

- 1 **Lojain Hafez Alibrahim**
- 2 **Shatha Ahmed Alanazi**
- 3 **Noura Saad Alqahtani**
- 4 **Shahad Mohammed Alowfi**
- 5 **Rahaf Adnan Alshammari**
- 6 **Amal Ali Alqahtani**

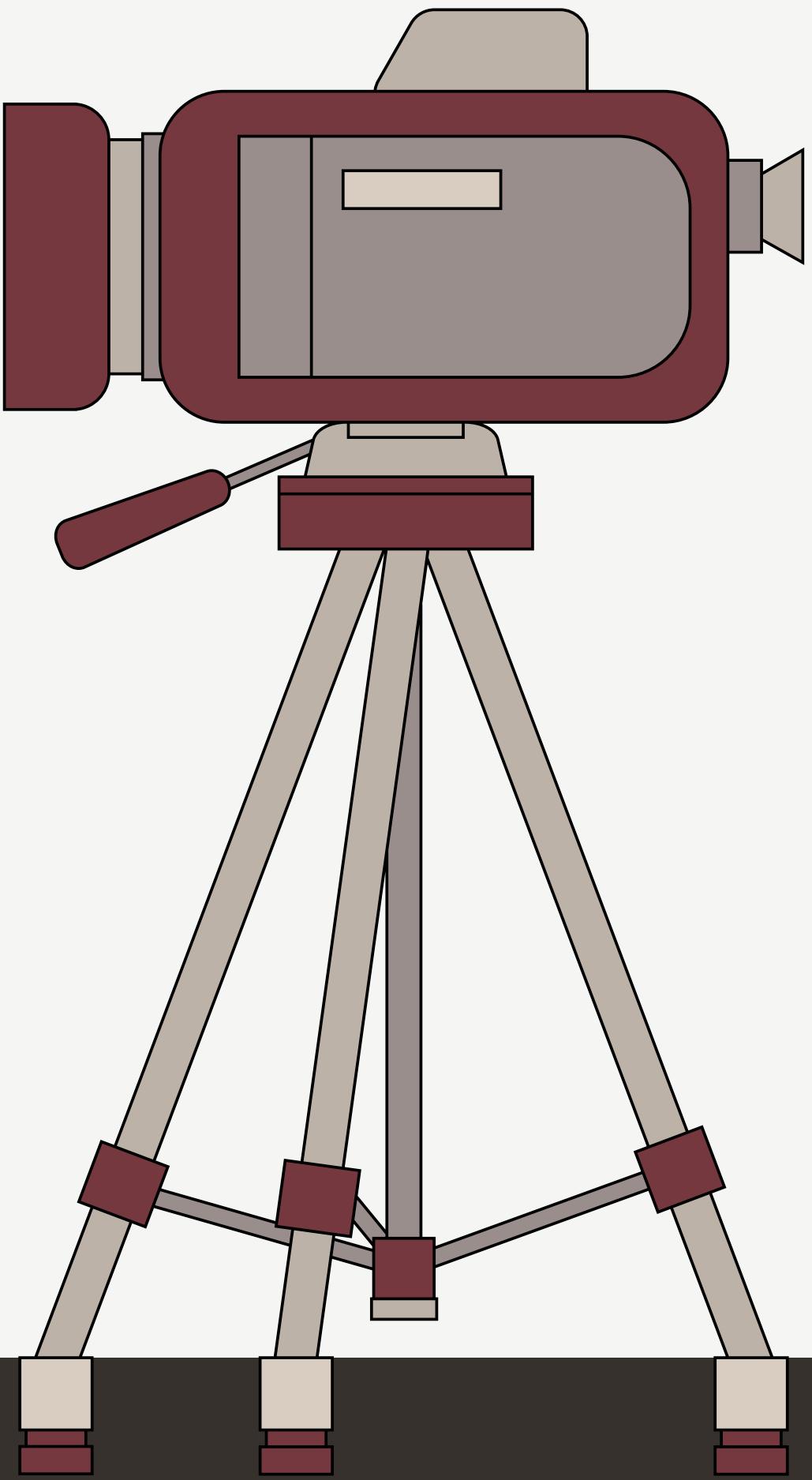




# introduction

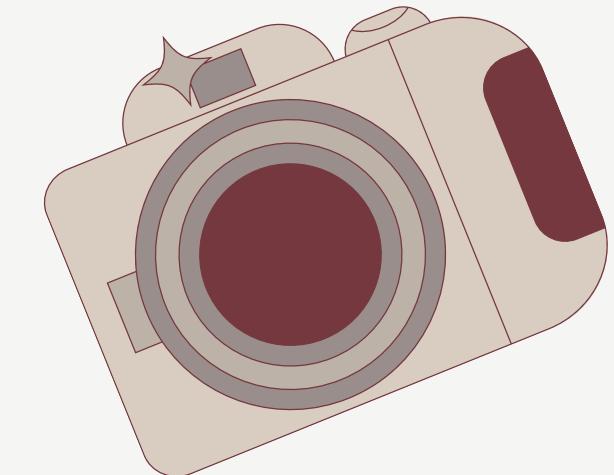
Within the constantly growing digital environment, data is power. However, where is all this data stored? Discover the world of databases (DBs) and the amazingly intricate procedures that are necessary for any application or organization to thrive. Our Java podcast program, Audio Vibe, has a graphical user interface (GUI) as its primary means of interacting with a database that was built from scratch. Furthermore, the elegance of Java GUI lies in its capacity to breathe life into thousands of lines of code and convert them into intuitive user interfaces. This is Java GUI's magic: it lets programmers combine visual components with code to create apps that are both engaging and functional. In addition, picture a world in which interactive windows, menus, and buttons clearly represent complicated features. Regardless of their level of technical expertise, users can easily traverse these interfaces and complete tasks with a single click or drag.

# Database



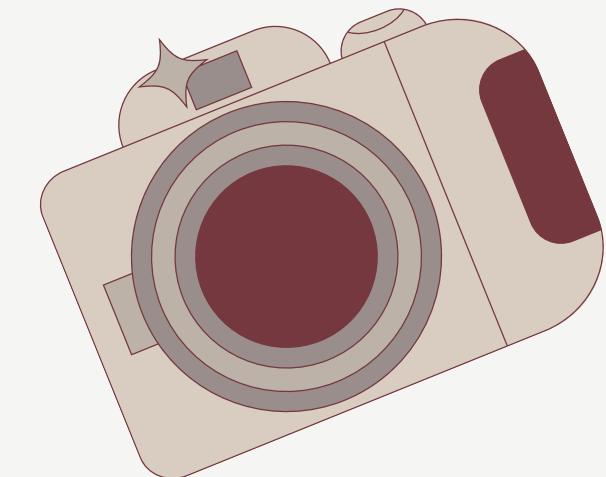
# The requirements:

- 1 Different types of users.**
- 2 Each entity type should have unique data types.**
- 3 Strong and weak entity types.**
- 4 Relationships of all types.**
- 5 Relationships with total and partial participation.**
- 6 Entity types' attributes should be from different types.**

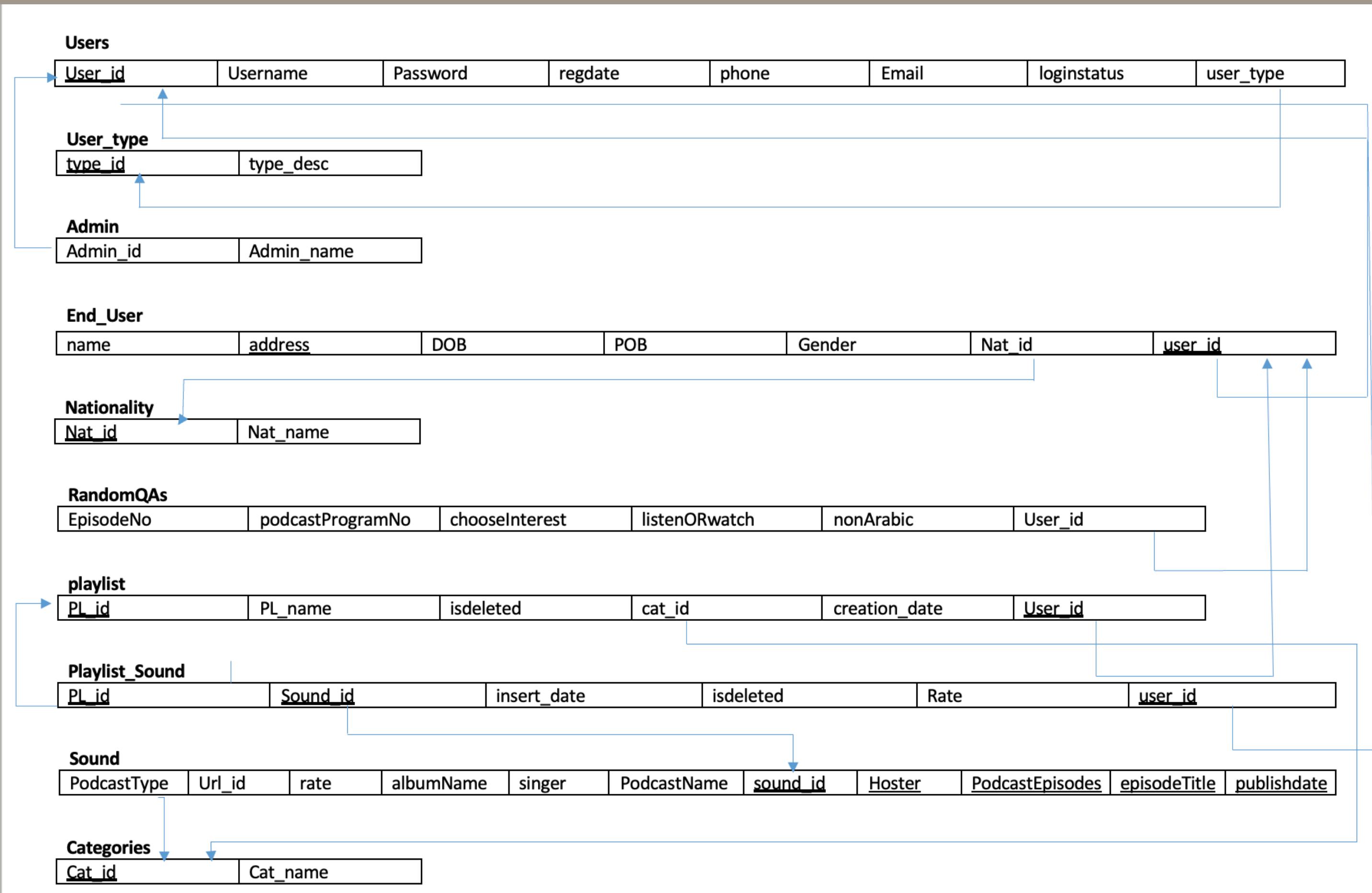


# The requirements:

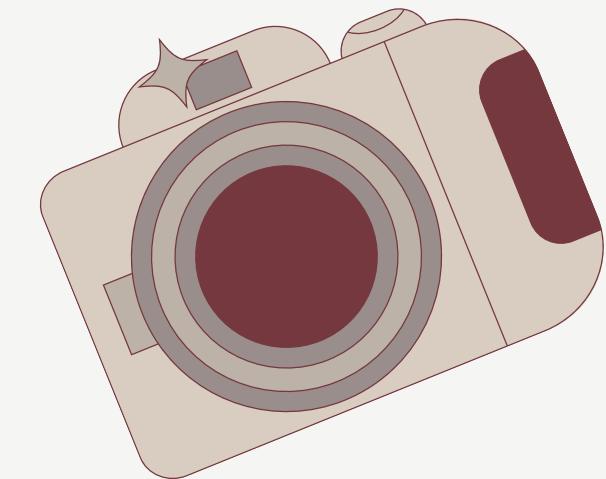
- 7 **Use rules to map your ER into tables.**
- 8 **Insert tuples.**
- 9 **Delete tuples.**
- 10 **Update tables.**
- 11 **View user's profile.**
- 12 **Run at least 15 advanced queries.**



# Mapping:



# The requirements:



## 1 **Different types of users.**

- Admin
- User

## 2 **Each entity type should have unique data types.**

- Int
- Char
- Date

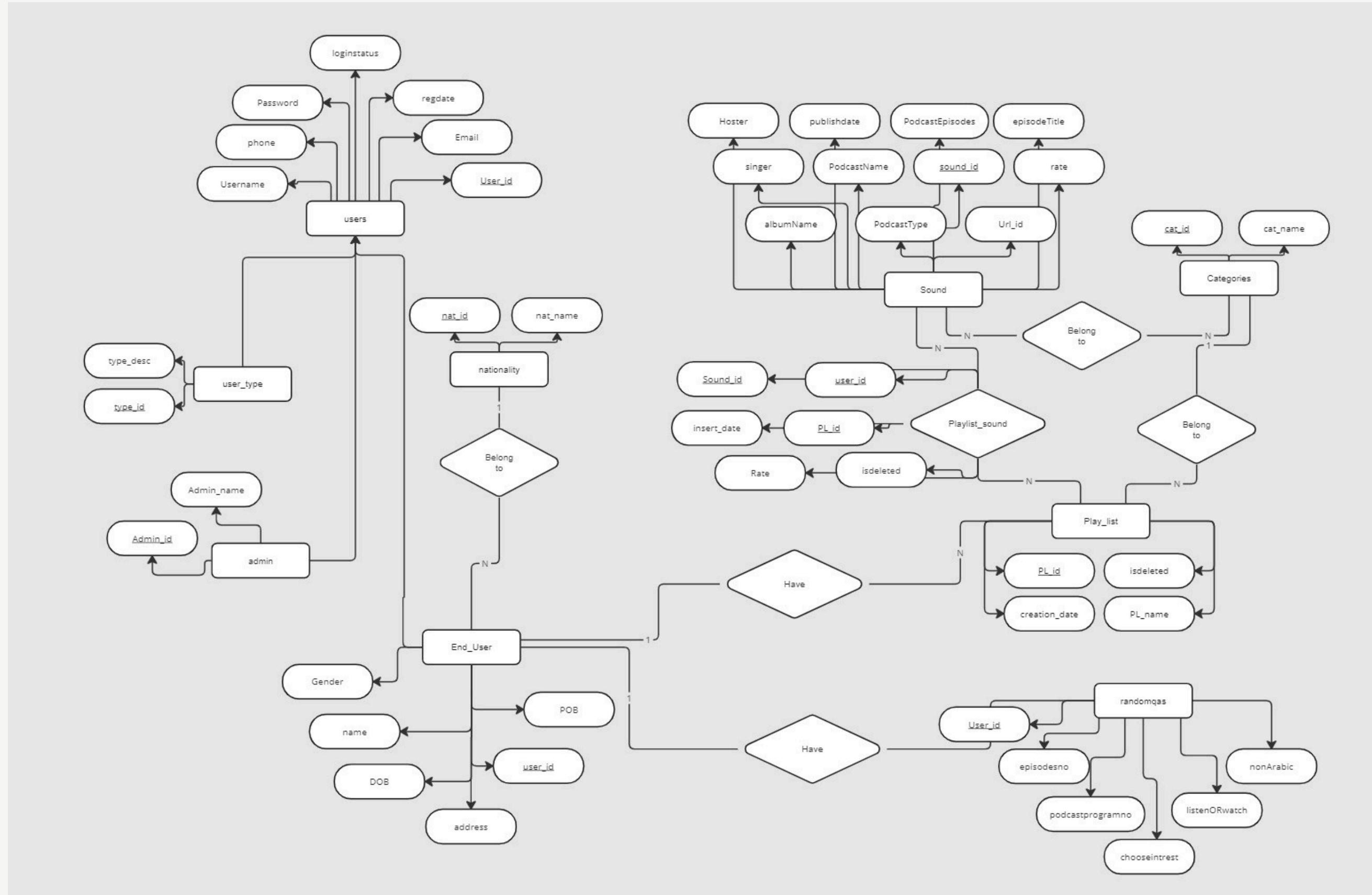
# The requirements:

3 **Strong and weak entity types.**

Strong entity.

4 **types of Relationships**

1-M relationships.  
N-M relationships.



# The requirements:

5

## **Relationships with total and partial participation.**

### **RELATIONSHIPS WITH TOTAL AND PARTIAL PARTICIPATION:**

#### **Total:**

- user\_type with users.
- users with end\_user and admin.
- end\_user with users and nationality.
- playlist with categories and end\_user.
- randomqas with end\_user.
- sound with categories.
- playlist\_sound with sound, playlist, users.

#### **Partial:**

- admin with users.

# The requirements:

- 6 Entity types' attributes should be from different types.

## MULTI-VALUE COLUMN:

Playlist\_sound Table.

## COMPOSITE ATTRIBUTE:

In the sound table, 'publishdate' could potentially be composite if it's broken down into day, month, year components.

**But it is not recommended to implement.**

## COMPUTED COLUMN:

None

# Detailed Tables

```
CREATE TABLE `user_type` (
  `type_id` int NOT NULL,
  `type_desc` varchar(100) NOT NULL,
  PRIMARY KEY (`type_id`)
);

CREATE TABLE `users` (
  `User_id` int NOT NULL,
  `Username` varchar(100) NOT NULL,
  `Password` varchar(100) NOT NULL,
  `regdate` varchar(45) DEFAULT NULL,
  `phone` varchar(45) DEFAULT NULL,
  `Email` varchar(100) DEFAULT NULL,
  `loginstatus` int NOT NULL,
  `user_type` int NOT NULL,
  PRIMARY KEY (`User_id`),
  CONSTRAINT `user_types` FOREIGN KEY (`user_type`) REFERENCES `user_type` (`type_id`)
);

CREATE TABLE `categories` (
  `cat_id` int NOT NULL AUTO_INCREMENT,
  `cat_name` varchar(100) NOT NULL,
  PRIMARY KEY (`cat_id`)
);
```

```
CREATE TABLE `end_user` (
  `name` varchar(100) NOT NULL,
  `address` varchar(100) DEFAULT NULL,
  `DOB` varchar(45) DEFAULT NULL,
  `POB` varchar(100) DEFAULT NULL,
  `Gender` char(1) DEFAULT NULL,
  `Nat_id` int NOT NULL,
  `user_id` int NOT NULL,
  PRIMARY KEY (`user_id`),
  CONSTRAINT `nat_user` FOREIGN KEY (`Nat_id`) REFERENCES `nationality` (`nat_id`),
  CONSTRAINT `user_Users` FOREIGN KEY (`user_id`) REFERENCES `users` (`User_id`)
);

CREATE TABLE `randomqas` (
  `episodesno` int NOT NULL,
  `podcastprogramno` int DEFAULT NULL,
  `chooseintrest` varchar(100) DEFAULT NULL,
  `listenORwatch` varchar(100) DEFAULT NULL,
  `nonArabic` varchar(100) DEFAULT NULL,
  `User_id` int NOT NULL,
  PRIMARY KEY (`User_id`),
  CONSTRAINT `user_id_QAs` FOREIGN KEY (`User_id`) REFERENCES `end_user` (`user_id`)
);
```

# Detailed Tables

```
CREATE TABLE `sound` (
  `PodcastType` int NOT NULL,
  `Url_id` varchar(200) DEFAULT NULL,
  `rate` int DEFAULT NULL,
  `albumName` varchar(100) DEFAULT NULL,
  `singer` varchar(100) DEFAULT NULL,
  `PodcastName` varchar(100) DEFAULT NULL,
  `sound_id` int NOT NULL,
  `Hoster` varchar(100) DEFAULT NULL,
  `PodcastEpisodes` int DEFAULT NULL,
  `episodeTitle` varchar(100) DEFAULT NULL,
  `publishdate` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`sound_id`),
  CONSTRAINT `soundcat` FOREIGN KEY (`PodcastType`) REFERENCES `categories` (`cat_id`)
);
CREATE TABLE `nationality` (
  `nat_id` int NOT NULL AUTO_INCREMENT,
  `nat_name` varchar(100) NOT NULL,
  PRIMARY KEY (`nat_id`)
);
```

```
CREATE TABLE `admin` (
  `Admin_id` int NOT NULL,
  `Admin_name` varchar(100) NOT NULL,
  PRIMARY KEY (`Admin_id`),
  CONSTRAINT `Admin_users` FOREIGN KEY (`Admin_id`) REFERENCES `users` (`User_id`)
);
```

```
CREATE TABLE `playlist` (
  `PL_id` int NOT NULL,
  `PL_name` varchar(100) NOT NULL,
  `isdeleted` int NOT NULL DEFAULT '0',
  `cat_id` int NOT NULL,
  `creation_date` varchar(45) NOT NULL,
  `user_id` int NOT NULL,
  PRIMARY KEY (`PL_id`, `user_id`),
  CONSTRAINT `pl_cat` FOREIGN KEY (`cat_id`) REFERENCES `categories` (`cat_id`),
  CONSTRAINT `pl_users` FOREIGN KEY (`user_id`) REFERENCES `end_user` (`user_id`)
);
CREATE TABLE `playlist_sound` (
  `PL_id` int NOT NULL,
  `sound_id` int NOT NULL,
  `insert_date` varchar(45) NOT NULL,
  `isdeleted` int DEFAULT '0',
  `Rate` int DEFAULT NULL,
  `user_id` int NOT NULL,
  PRIMARY KEY (`PL_id`, `sound_id`, `user_id`),
  CONSTRAINT `pl_sound` FOREIGN KEY (`PL_id`) REFERENCES `playlist` (`PL_id`),
  CONSTRAINT `sound_pl` FOREIGN KEY (`sound_id`) REFERENCES `sound` (`sound_id`),
  CONSTRAINT `user_plylst` FOREIGN KEY (`user_id`) REFERENCES `users` (`User_id`)
);
```

# Tables

User_id	Username	Password	regdate	phone	Email	loginstatus	user_type
1	shatha	123	1/1/2024	0096765	shataa@gmail.com	0	1
2	lojain	876	6/1/2022	0096530	Loj2@gmail.com	0	1
3	Noura	789	18/3/2022	0096653	noura112@gmail.com	0	2
4	shahad	479	28/5/2022	0096007	Shahad21@gmail.com	0	2
5	Amal	983	10/7/2023	0096987	Amal.5@gmail.com	0	2
6	Reema	783	8/3/2024	0096955	Reema45@gmail.com	0	2
7	Rahaf	003	2/1/2022	0096667	Rahaff22@gmail.com	0	2
8	Reem	1111	23/4/2023	0096674	reem@gmail.com	0	2
9	Shikah	1260	2/8/2023	009694	Shikah@gmail.com	0	2
10	jumanah	1911	2/4/2021	0096008	jojo984@gmail.com	0	2
11	sondos	9484	5/3/2023	0096986	soso998@gmail.com	0	2
12	rfeef	23500	20/11/20...	0096238	fofo34@gmail.com	0	2
13	jood	2311	24/12/20...	0096932	jood334@gmail.com	0	2
14	Laila	6841	11/1/2021	0096655	laila32@gmail.com	0	2
15	Reemas	5511	7/6/2022	009669...	Reemas@gmail.com	0	2

type_id	type_desc
1	admin
2	user

Admin_id	Admin_name
1	shatha ahmed
2	lojain

# Tables

name	address	DOB	POB	Gender	Nat_id	user_id
Noura shahad	Saudi Arabia Kuwait	18/03/2003 28/05/2003	Khobar Al Jahra	F F	1 20	3 4
Amal	Saudi Arabia	10/07/2003	Dhahran	F	1	5
Reema	Qatar	01/04/1996	Doha	F	3	6
Rahaf	UAE	01/11/1998	Dubai	F	2	7
Reem	Saudi Arabia	02/08/1999	Riyad	F	1	8
Shikah	Jordan	22/11/1990	beirut	F	5	9
jumanah	Kuwait	01/04/1996	Ali sabah alsalem	F	20	10
sondos	Qatar	13/9/2004	alrayan	F	3	11
rfeef	UAE	24/10/1994	abu dhabi	F	2	12
jood	Armenia	14/01/2006	yerevan	F	13	13
Laila	Austria	24/8/1999	vienna	F	19	14
Reemas	Australia	14/06/2000	canberra	F	18	15

nat_id	nat_name
1	Saudi Arabia
2	United Arab Emirates
3	Qater
4	Palestine
5	Jordan
6	Afghanistan
7	Angola
8	Anguilla
9	Albania
10	Andorra
11	Netherlands Antilles
12	Argentina
13	Armenia
14	American Samoa
15	Antarctica
16	French Southern ter...
17	Antigua and Barbuda
18	Australia
19	Austria
20	Kuwait

# Tables

episodesno	podcastprogram...	chooseintre...	listenORwatch	nonArabic	User_id
3	8	food	watch	yes	3
4	4	comedy	listen	No	4
5	5	Comedy	listen	yes	5
2	3	Stories	listen	No	6
2	9	Stories	listen	yes	7
10	10	Food	watch	yes	8
1	5	History	listen	yes	9
7	20	song	listen	yes	10
3	9	Comedy	listen	no	11
1	1	movie	watch	no	12
2	8	Stories	listen	yes	13
5	7	Self improv...	listen	yes	14
8	9	Educational	watch	yes	15

PL_id	PL_name	isdeleted	cat_id	creation_da...	user_id
1001	Sh-list	0	2	2023-07-02	4
1009	JoodList	0	3	2022-04-24	13
1011	ShStories	0	3	2023-07-01	4
2001	Amallist	0	1	2023-02-12	5
2004	S-List	0	4	2023-03-02	11
3001	ReemList	0	4	2023-07-02	8
3002	RafeefList	0	3	2020-04-12	12
4001	Therapy	0	2	2023-10-09	7
4005	Laila-List	0	5	2022-03-18	14
5001	RList	0	3	2023-01-01	6
6001	NNN	0	3	2023-03-18	3
7001	improve...	0	8	2023-07-06	6
8001	ShikahL	0	6	2023-07-03	9
9001	ShList	0	1	2023-06-28	4
10001	JojoList	0	7	2023-07-04	10

# Tables

PodcastType	Url_id	rate	albumName	singer	PodcastName	sound_id	Hoster	PodcastEpisod...	episodeTitle	publishdate
3	https://podcasts.apple.com(sa/podcast/%D9%8...	5	NULL	NULL	Manbet	1101	Sowt	123	S4 E9	21-06-2022
5	https://podcasts.apple.com(sa/podcast/foodcast/...	5	NULL	NULL	FoodCast	1102	Amer Sherif	20	FoodCast9	14-01-2024
3	https://podcasts.apple.com(sa/podcast/%D9%8...	4	NULL	NULL	Manbet	1103	Sowt	123	S6 E8	07-03-2024
5	https://podcasts.apple.com(sa/podcast/foodcast/...	3	NULL	NULL	FoodCast	1106	Amer Sherif	20	FoodCast12	04-02-2024
5	https://podcasts.apple.com(sa/podcast/foodcast/...	4	NULL	NULL	FoodCast	1107	Amer Sherif	20	FoodCast8	07-01-2024
3	https://podcasts.apple.com(sa/podcast/%D8%A...	4	NULL	NULL	Jinaya	1108	MohamedAlbeez	30	Manzl mubark	27-03-2023
3	https://podcasts.apple.com(sa/podcast/%D8%A...	4	NULL	NULL	Jinaya	1109	MohamedAlbeez	30	Haqibah al ahlam	10-04-2023
3	https://podcasts.apple.com(sa/podcast/%D8%A...	4	NULL	NULL	Jinaya	1110	MohamedAlbeez	30	Khatem sulaiman	22-05-2023
3	https://podcasts.apple.com(sa/podcast/%D9%8...	4	NULL	NULL	Manbet	1111	Sowt	123	S4 E11	26-07-2022
7	https://podcasts.apple.com/us/podcast/kristen-st...	5	NULL	NULL	Smartless	1112	JasonBateman&SeanHayes&WillArnett	20	Kirsten Stewart	01-04-2024
7	https://podcasts.apple.com/us/podcast/jake-gyll...	5	NULL	NULL	Smartless	1113	JasonBateman&SeanHayes&WillArnett	20	Jake Gyllenhaal	25-03-2024
7	https://podcasts.apple.com/us/podcast/smartles...	5	NULL	NULL	Smartless	1114	JasonBateman&SeanHayes&WillArnett	20	John Oliver	18-03-2024
7	https://podcasts.apple.com/us/podcast/smartles...	5	NULL	NULL	Smartless	1115	JasonBateman&SeanHayes&WillArnett	20	Amy Schumer	11-03-2024
3	https://podcasts.apple.com(sa/podcast/%D9%8...	4	NULL	NULL	Manbet	1116	Sowt	123	S6 E10	21-03-2024
3	https://podcasts.apple.com(sa/podcast/%D9%8...	4	NULL	NULL	Manbet	1117	Sowt	123	S5 E8	10-07-2024
5	https://podcasts.apple.com(sa/podcast/foodcast/...	4	NULL	NULL	FoodCast	1119	Amer Sherif	20	FoodCast13	11-02-2024
5	https://podcasts.apple.com(sa/podcast/foodcast/...	4	NULL	NULL	FoodCast	1120	Amer Sherif	20	FoodCast11	28-01-2024
7	https://podcasts.apple.com/us/podcast/smartles...	5	NULL	NULL	Smartless	1123	JasonBateman&SeanHayes&WillArnett	20	Josh Brolin	04-03-2024
7	https://podcasts.apple.com(sa/podcast/27-%D8...	4	NULL	NULL	YsierKheir	1124	AbdullahAlhussain	10	#27	20-03-2024
7	https://podcasts.apple.com(sa/podcast/23-%D9...	5	NULL	NULL	YsierKheir	1125	AbdullahAlhussain	10	#23	14-02-2024
3	https://podcasts.apple.com(sa/podcast/%D8%A...	5	NULL	NULL	Jinaya	1126	MohamedAlbeez	30	Wahsh al daherh	29-05-2023
8	https://podcasts.apple.com/us/podcast/how-to-b...	5	NULL	NULL	How to keep...	1127	The Atlantic	7	How to be self-a...	05-10-2021
8	https://podcasts.apple.com/us/podcast/how-to-k...	4	NULL	NULL	How to keep...	1128	The Atlantic	7	How to know yo...	12-10-2021
8	https://podcasts.apple.com/us/podcast/how-not-...	5	NULL	NULL	How to keep...	1129	The Atlantic	7	How not to be y...	19-10-2021
8	https://podcasts.apple.com/us/podcast/how-to-...	5	NULL	NULL	How to keep...	1130	The Atlantic	7	How to know th...	26-10-2021
8	https://podcasts.apple.com/us/podcast/how-to-...	5	NULL	NULL	How to keep...	1131	The Atlantic	7	How to find the...	02-11-2021
8	https://podcasts.apple.com/us/podcast/how-to-li...	5	NULL	NULL	How to keep...	1132	The Atlantic	7	How too live w...	09-11-2021
8	https://podcasts.apple.com/us/podcast/how-to-i...	5	NULL	NULL	How to keep...	1133	The Atlantic	7	How to identify...	23-11-2021
1	https://podcasts.apple.com(sa/podcast/lana-del-...	4	SONGS	Lana	UNRELEASED	1134	NULL	4	NULL	19-08-2022
1	https://podcasts.apple.com(sa/podcast/taylor-sw...	4	Taylor	corboo	Taylor Swift...	1135	NULL	10	NULL	01-01-2022
4	https://podcasts.apple.com(sa/podcast/ielts-pod...	5	IELTS	NULL	IELTS Podcast	1136	Ben Worthington	20	NULL	20-02-2020
4	https://podcasts.apple.com(sa/podcast/voa-lear...	4	Learning E...	NULL	VOA Learnin...	1137	Loie	20	NULL	20-02-2021
6	https://podcasts.apple.com(sa/podcast/islamic-h...	5	History	NULL	Islamic History	1138	Mohammed	93	NULL	01-03-2020
6	https://podcasts.apple.com(sa/podcast/the-rest-i...	3	HistoryR	NULL	The Rest Is...	1139	Tom,Dominic	611	NULL	12-02-2020
7	https://podcasts.apple.com(sa/podcast/gold-min...	5	ComedyP...	NULL	Gold Minds	1140	Kevin	140	NULL	20-06-2021
7	https://podcasts.apple.com(sa/podcast/the-daily...)	4	DailyPodcast	NULL	The Daily Sh...	1141	Jon	2000	NULL	16-04-2021
4	https://podcasts.apple.com(sa/podcast/english-li...	4	English Pr...	Con...	Listening Time	1142	Conner	154	What I Learned...	07-05-2024

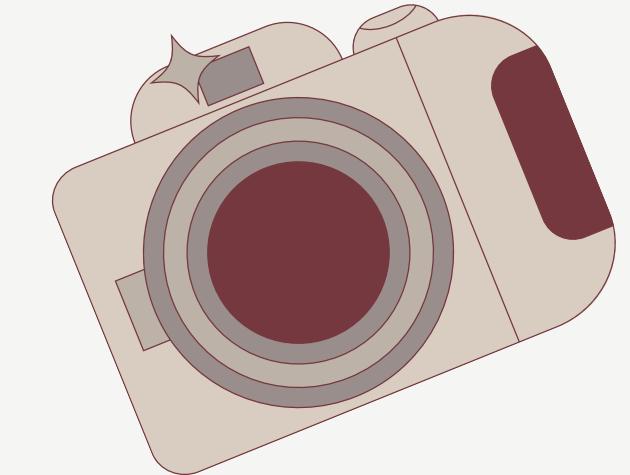
# Tables

PL_id	sound_id	insert_date	isdeleted	Rate	user_id
1001	1116	21-03-2024	0	4	4
1001	1117	15-02-2024	0	5	4
1001	1119	02-09-2023	0	4	4
1001	1120	11-10-2023	0	4	4
2001	1103	01-07-2023	0	5	5
2001	1106	13-02-2024	0	4	5
2001	1107	30-01-2024	0	4	5
2001	1108	15-01-2024	0	5	5
2001	1109	05-02-2024	0	3	5
2001	1110	10-01-2024	0	4	5
4001	1111	02-04-2024	0	5	7
4001	1112	26-03-2024	0	5	7
4001	1113	19-03-2024	0	5	7
4001	1114	12-03-2024	0	5	7
4001	1115	05-03-2024	0	5	7
5001	1101	10-04-2024	0	4	6
5001	1102	05-04-2024	0	4	6
6001	1123	22-05-2023	0	4	3
6001	1124	10-04-2023	0	4	3
6001	1125	27-03-2023	0	4	3

cat_id	cat_name
1	song
2	movie
3	Story
4	Educational
5	Food
6	History
7	Comedy
8	Self improvement

# The requirements:

## 8 Insert tuples.

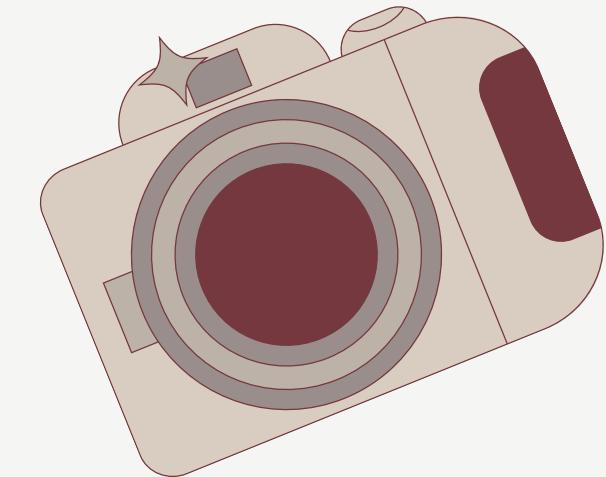


```
INSERT INTO `users` VALUES
(1,'shatha','123','2024-01-01','0096765','shataa@gmail.com',0,1),
(2,'lojain','876','2022-01-06','0096530','Loj2@gmail.com',0,1),
(3,'Noura','789','2022-03-18','0096653','noura112@gmail.com',0,2),
(4,'shahad','479','2022-05-28','0096007','Shahad21@gmail.com',0,2),
(5,'Amal','983','2023-07-10','0096987','Amal.5@gmail.com',0,2),
(6,'Reema','783','2024-03-08','0096955','Reema45@gmail.com',0,2),
(7,'Rahaf','003','2022-01-02','0096667','Rahaff22@gmail.com',0,2),
(8,'Reem','1111','2023-04-23','0096674','reem@gmail.com',0,2),
(9,'Shikah','1260','2023-08-02','009694','Shikah@gmail.com',0,2),
(10,'jumanah','1911','2021-04-02','0096008','jojo984@gmail.com',0,2),
(11,'sondos','9484','2023-03-05','0096986','soso998@gmail.com',0,2),
(12,'rfeef','23500','2022-11-20','0096238','fofo34@gmail.com',0,2),
(13,'jood','2311','2022-12-24','0096932','jood334@gmail.com',0,2),
(14,'Laila','6841','2021-01-11','0096655','laila32@gmail.com',0,2),
(15,'Reemas','5511','2022-06-07','00966986','Reemas@gmail.com',0,2);

INSERT INTO `nationality` VALUES
(1,'Saudi Arabia'),
(2,'United Arab Emirates'),
(3,'Qater'),
(4,'Palestine'),
(5,'Jordan'),(6,'Afghanistan'),(7,'Angola'),(8,'Anguilla'),(9,'Albania'),
(10,'Andorra'),(11,'Netherlands Antilles'),(12,'Argentina'),
(13,'Armenia'),(14,'American Samoa'),(15,'Antarctica'),(16,'French Southern territories'),
(17,'Antigua and Barbuda'),(18,'Australia'), (19,'Austria'),(20,'Kuwait');
```

# The requirements:

- 9 Delete tuples.



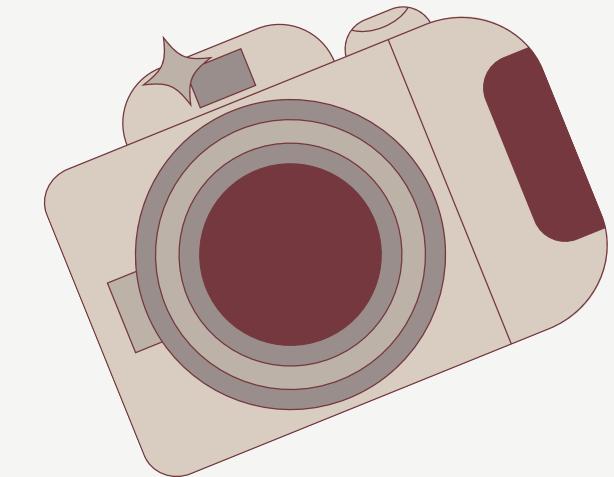
```
319    DELETE FROM playlist_sound  
320    WHERE PL_id = '1001';
```

# The requirements:

10

**Update tuples.**

```
UPDATE playlist  
SET PL_name = 'Favorites'  
WHERE PL_id = 6001 AND user_id = 3;
```



# The requirements:

11

## Advanced Queries.

```
302 • SELECT user_type, COUNT(*)
```

```
303     AS total_users
```

```
304     FROM users GROUP BY user_type;
```

```
307 • SELECT * FROM users
```

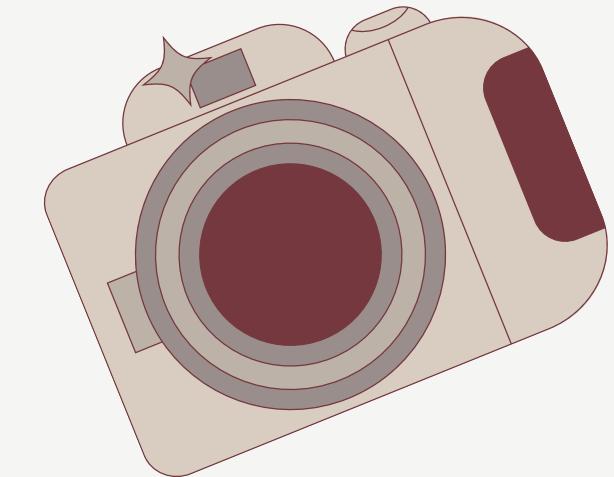
```
308     ORDER BY regdate DESC;
```

```
311 • SELECT * FROM playlist
```

```
312     WHERE cat_id = 1;
```

```
315 • SELECT * FROM randomqas
```

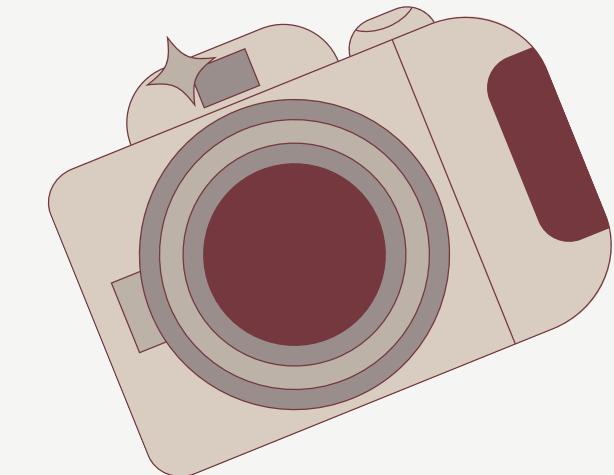
```
316     WHERE listenORwatch = 'Listen';
```



# The requirements:

11

## Advanced Queries.



```
319 • SELECT * FROM randomqas
```

```
320 WHERE User_id = 3;
```

```
323 • SELECT * FROM sound
```

```
324 WHERE PodcastType = 4;
```

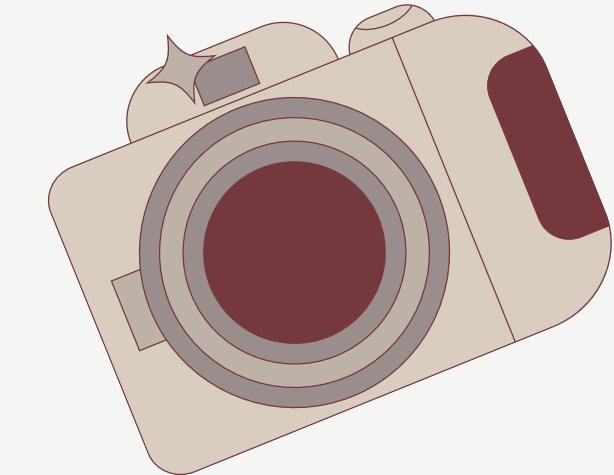
```
327 • SELECT * FROM sound
```

```
328 WHERE publishdate = '2022-01-01';
```

# The requirements:

11

## Advanced Queries.



```
334 • SELECT * FROM end_user  
335 WHERE DOB BETWEEN '2000-01-01' AND '2005-12-31';
```

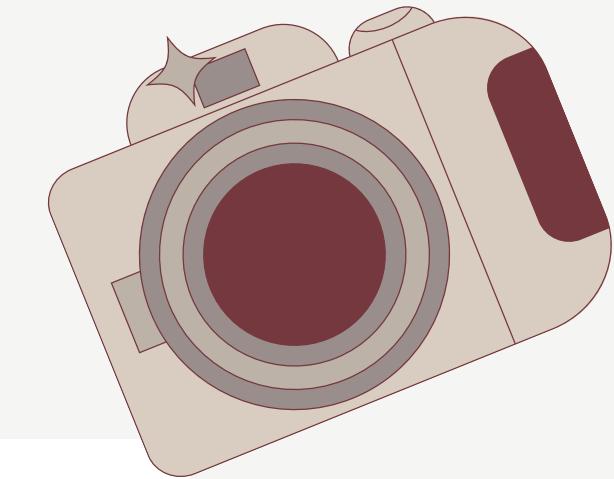
```
338 • SELECT users.User_id, users.Username,  
339 COALESCE(p.total_playlists, 0)  
340 AS total_playlists  
341 FROM users  
342 LEFT JOIN (  
343     SELECT user_id, COUNT(*) AS total_playlists  
344     FROM playlist  
345     GROUP BY user_id  
346 ) AS p ON users.User_id = p.user_id;
```

```
349 • select * FROM sound  
350 WHERE Hoster LIKE '%MohamedAlbeez%';
```

# The requirements:

11

## Advanced Queries.



353 • UPDATE playlist

354      SET PL\_name = 'Favorites'

355      WHERE PL\_id = 6001 AND user\_id = 3;

360 • CREATE INDEX idx\_user\_id ON users (User\_id);

364 • SELECT User\_id, Username, Email, NULL AS address, NULL AS DOB  
365      FROM users

366      UNION

367      SELECT user\_id, NULL AS Username, NULL AS Email, address, DOB

368      FROM end\_user;

# The requirements:

11

## Advanced Queries.

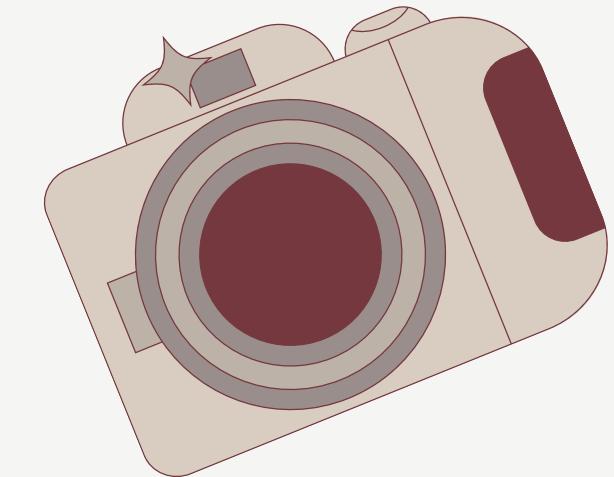
```
SELECT User_id FROM users
```

```
INTERSECT
```

```
SELECT user_id FROM end_user;
```

```
383 • SELECT s.PodcastName, AVG(ps.Rate) AS avg_rate  
384   FROM sound s  
385   JOIN playlist_sound ps ON s.sound_id = ps.sound_id  
386   GROUP BY s.PodcastName  
387   HAVING AVG(ps.Rate) > 4.5;
```

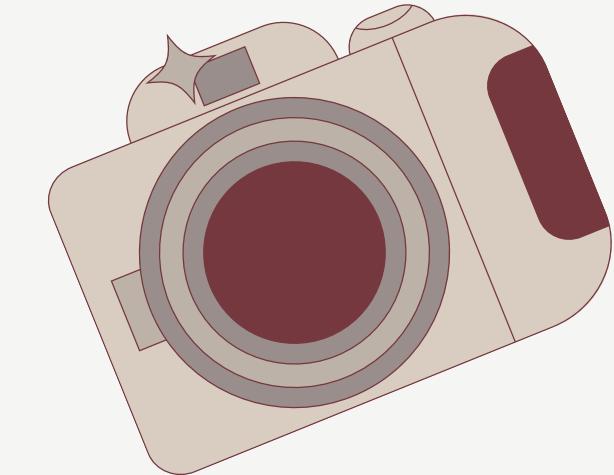
```
391 • SELECT *  
392   FROM categories  
393   WHERE NOT cat_id = 1;
```



# The requirements:

11

## Advanced Queries.



```
396 • SELECT DISTINCT Username
```

```
397   FROM users;
```

```
400 • SELECT *
```

```
401   FROM sound
```

```
402   WHERE PodcastType IN (1, 2, 3);
```

```
405 • SELECT
```

```
406       COUNT(sound_id) AS total_sounds,
```

```
407       AVG(rate) AS average_rate
```

```
408   FROM sound;
```

```
411 • SELECT *
```

```
412   FROM sound
```

```
413   WHERE albumName IS NULL;
```

# The requirements:

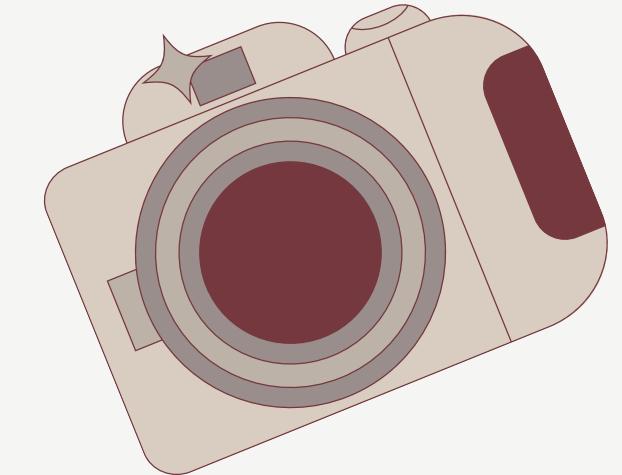
11

## Advanced Queries.

```
416 •   SELECT *
417     FROM end_user
418     WHERE user_id > ANY (SELECT user_id FROM users);
```

```
CREATE TRIGGER
update_last_login
BEFORE UPDATE ON users
FOR EACH ROW
SET new.regdate = NOW();
```

```
SHOW TRIGGERS LIKE 'update_last_login';
```



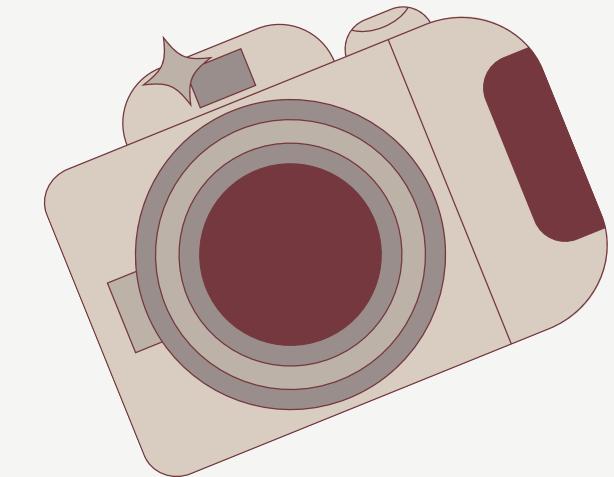
# The requirements:

11

## Advanced Queries.

```
415 •   SELECT
416     Username,
417     regdate,
418     CASE
419         WHEN regdate BETWEEN '2023-01-01' AND '2023-12-31' THEN '2023 User'
420         WHEN regdate BETWEEN '2022-01-01' AND '2022-12-31' THEN '2022 User'
421         ELSE 'Other'
422     END AS registration_year
423
424 FROM users;
```

```
426 •   SELECT *
427     FROM sound
428     WHERE PodcastType IN (
429         SELECT cat_id
430         FROM categories
431         WHERE cat_name = 'Food'
432     )
433     OR PodcastType IN (
434         SELECT cat_id
435         FROM categories
436         WHERE cat_name = 'song'
437     );
```

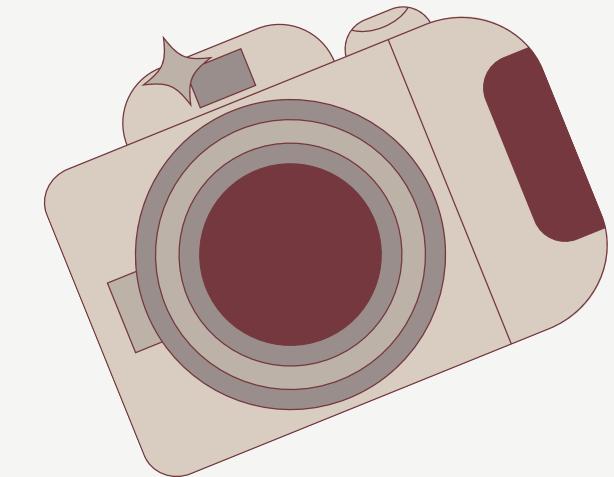


# The requirements:

11

## Advanced Queries.

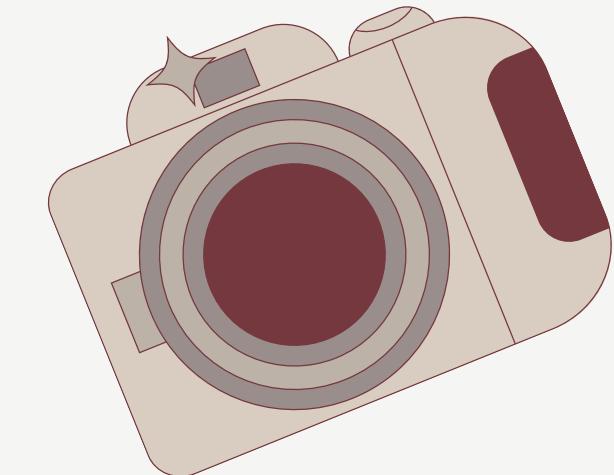
```
440      DELIMITER $$  
441 •   CREATE PROCEDURE GetAllUsers()  
442     BEGIN  
443       SELECT * FROM users;  
444     END$$  
445      DELIMITER ;  
446  
447 •   CALL GetAllUsers();  
450      DELIMITER $$  
451 •   CREATE FUNCTION CountUserPlaylists(userId INT)  
452     RETURNS INT DETERMINISTIC  
453     BEGIN  
454       DECLARE playlistCount INT;  
455  
456       SELECT COUNT(*) INTO playlistCount  
457       FROM playlist  
458       WHERE User_id = userId;  
459       RETURN playlistCount;  
460     END$$  
461  
462      DELIMITER ;  
464 •   SELECT CountUserPlaylists(5);
```



# The requirements:

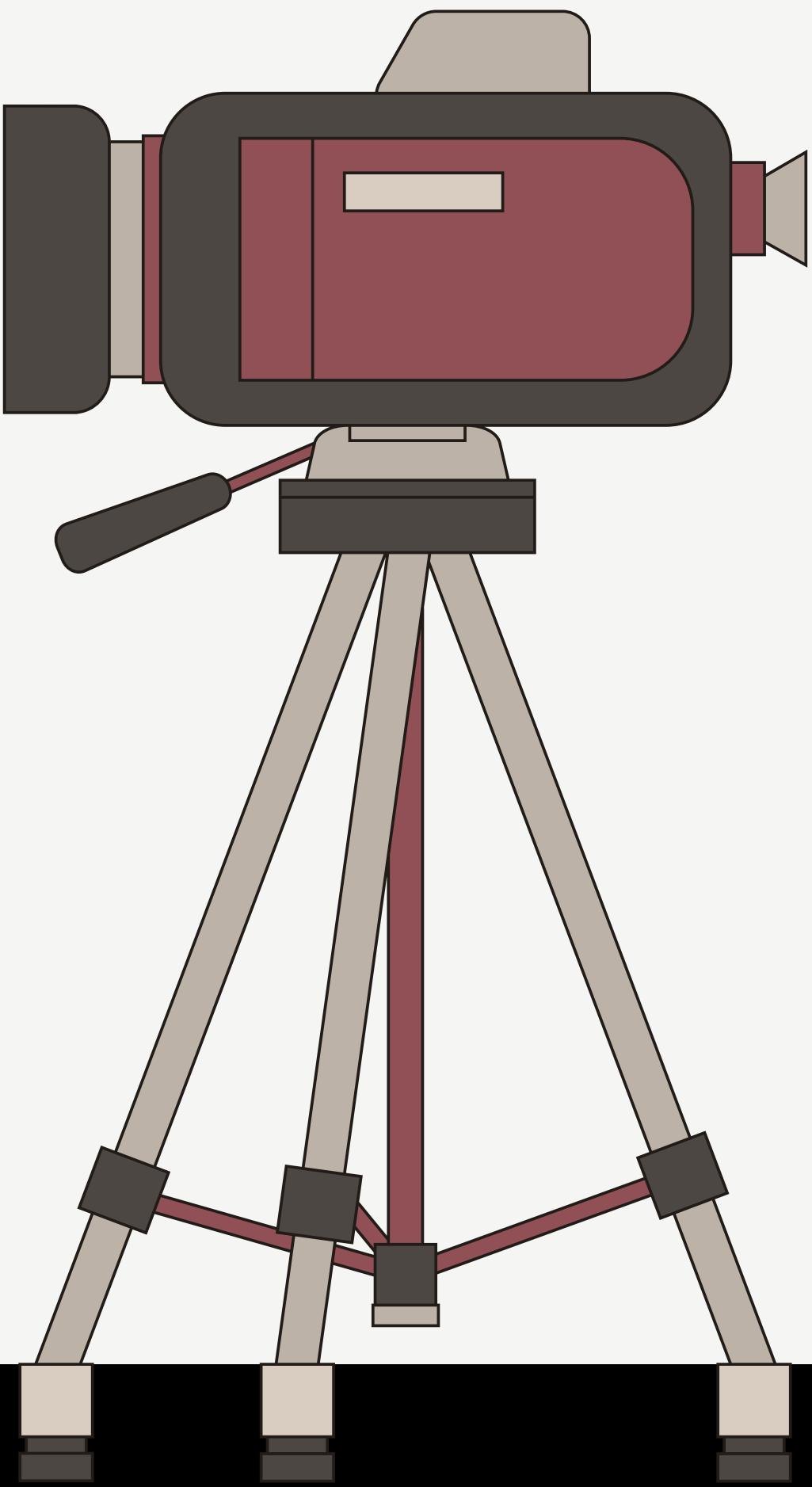
12

## View user's profile.



User_id	Username	Password	regdate	phone	Email	loginstatus	user_type
1	shatha	123	1/1/2024	0096765	shataa@gmail.com	0	1
2	lojain	876	6/1/2022	0096530	Loj2@gmail.com	0	1
3	Noura	789	18/3/2022	0096653	noura112@gmail.com	0	2
4	shahad	479	28/5/2022	0096007	Shahad21@gmail.com	0	2
5	Amal	983	10/7/2023	0096987	Amal.5@gmail.com	0	2
6	Reema	783	8/3/2024	0096955	Reema45@gmail.com	0	2
7	Rahaf	003	2/1/2022	0096667	Rahaff22@gmail.com	0	2
8	Reem	1111	23/4/2023	0096674	reem@gmail.com	0	2
9	Shikah	1260	2/8/2023	009694	Shikah@gmail.com	0	2
10	jumanah	1911	2/4/2021	0096008	jojo984@gmail.com	0	2
11	sondos	9484	5/3/2023	0096986	soso998@gmail.com	0	2
12	rfeef	23500	20/11/20...	0096238	fofo34@gmail.com	0	2
13	jood	2311	24/12/20...	0096932	jood334@gmail.com	0	2
14	Laila	6841	11/1/2021	0096655	laila32@gmail.com	0	2
15	Reemas	5511	7/6/2022	009669...	Reemas@gmail.com	0	2

# OOP2



# The requirements:

- 1 Classes Management**
- 2 Professional Appearance GUI**
- 3 Data Access Tools and Techniques**
- 4 Data Export/ Import**
- 5 Error Handling**
- 6 Working Demo, Evaluation, Delivery , & Justification**



# The requirements:

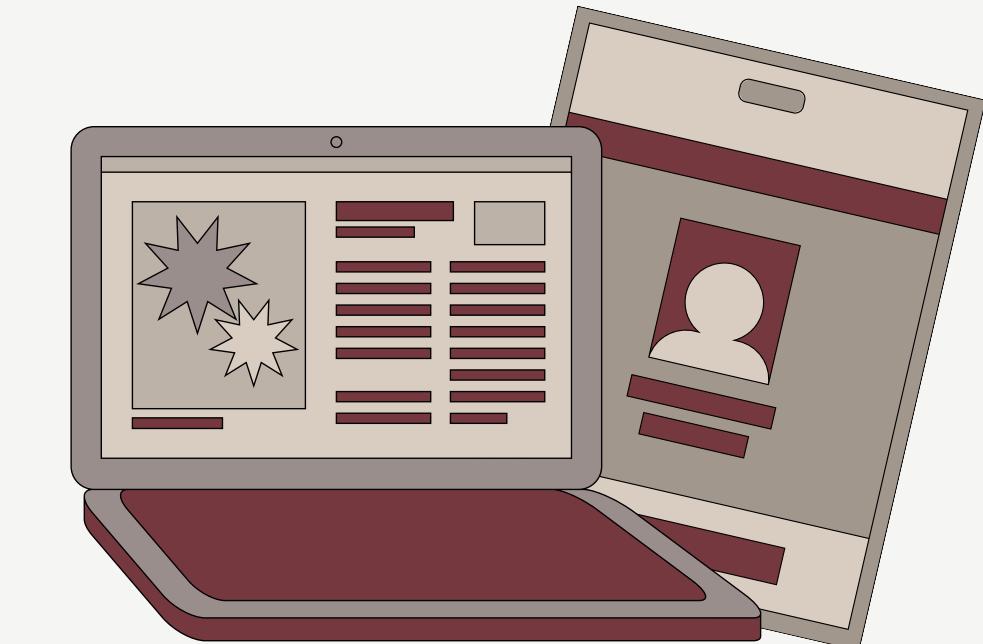
## 1 Classes Management

### SUPERCLASSES

- users
- Categories
- My\_dbCon
- Nationality
- PlayList
- PlyLstSound
- RandomQAs
- Sounds
- User\_type
- podcast

### SUBCLASSES

- End\_user
- admin



The other classes are extending **javax.swing.JFrame** which is a class in Java's Swing framework, used for creating and managing a windowed GUI (Graphical User Interface) application. It provides methods and components to create a window with various functionalities such as menus, buttons, text fields, etc. `JFrame` serves as the main container for building GUI applications in Java. It allows developers to design and customize the appearance and behavior of their application's window.

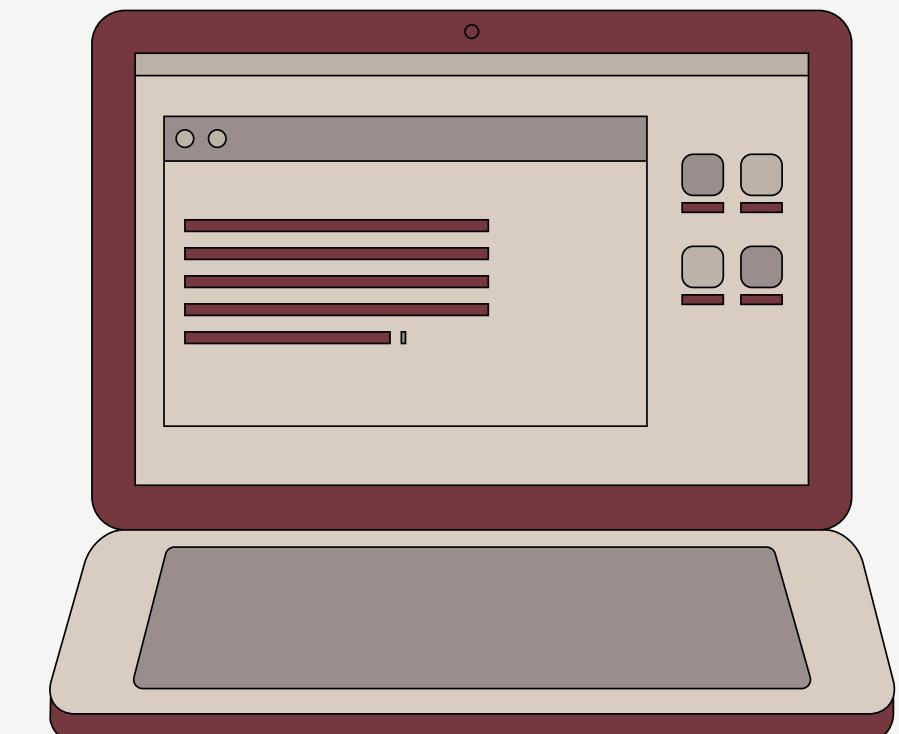
# The requirements:

2

## Professional Appearance GUI

To enhance the appeal of our podcast, we've employed a diverse range of GUI components, classes. These include the incorporation of images, adjustments to font size and style, and customization of background and foreground colors. **Among these components are:**

- JTextField, TextField
- JPasswordField
- JPanel
- ButtonGroup
- JDateChooser
- JRadioButton
- JButton
- JLabel
- JScrollPane
- JTextArea
- JTable
- JComboBox

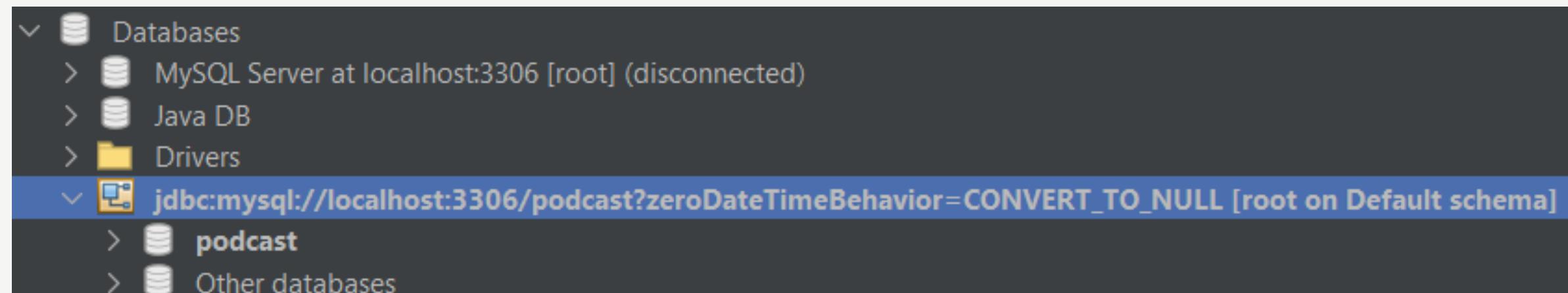
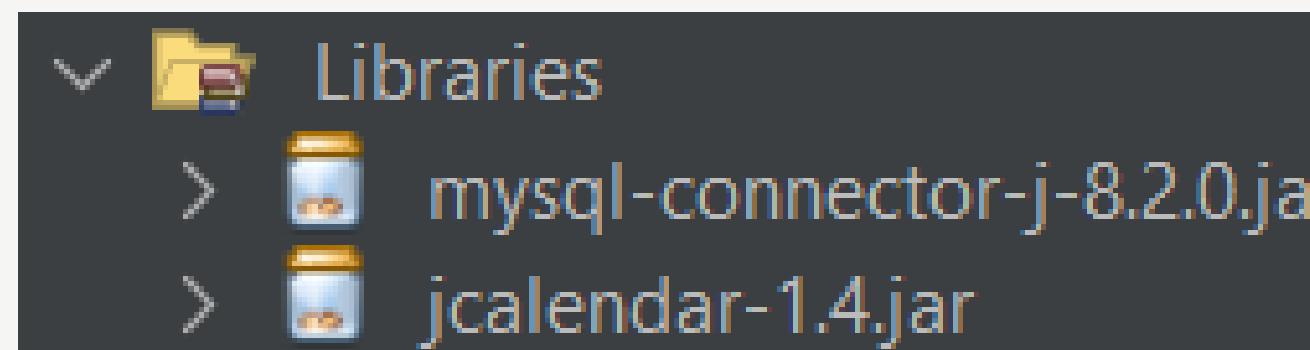


# The requirements:

3

## Implementing correct DB connections

We've seamlessly linked our project to our MySQL Workbench 8.0 CE database by incorporating the necessary connector jars into our Java platform. By importing the appropriate interfaces and classes, we established a connection between the application and the database. This connection allows us to retrieve, manipulate (such as adding, deleting, or updating), and display data from the database. Finally, we ensure a successful closure of the database connection once our tasks are completed.



# The requirements:

## 4 Data Export/ Import

### USED IN WELCOME INTERFACE:

The program will access a file, retrieve its contents, and display them on the welcome interface as a greeting message for the users.

```
private void formWindowActivated(java.awt.event.WindowEvent evt) {  
  
    try  
    {  
        String filename="Welcome.txt";  
        StringBuilder content = new StringBuilder();  
        BufferedReader reader = new BufferedReader(new FileReader(fileName: filename));  
        String line;  
        while ((line = reader.readLine()) != null) {  
            content.append(str:line).append(str: "\n");  
        }  
        reader.close();  
  
        txtArea.setText(t: content.toString().trim());  
    }  
    catch (IOException e) {  
        e.printStackTrace();  
    }  
}
```

# The requirements:

## 5 Error Handling

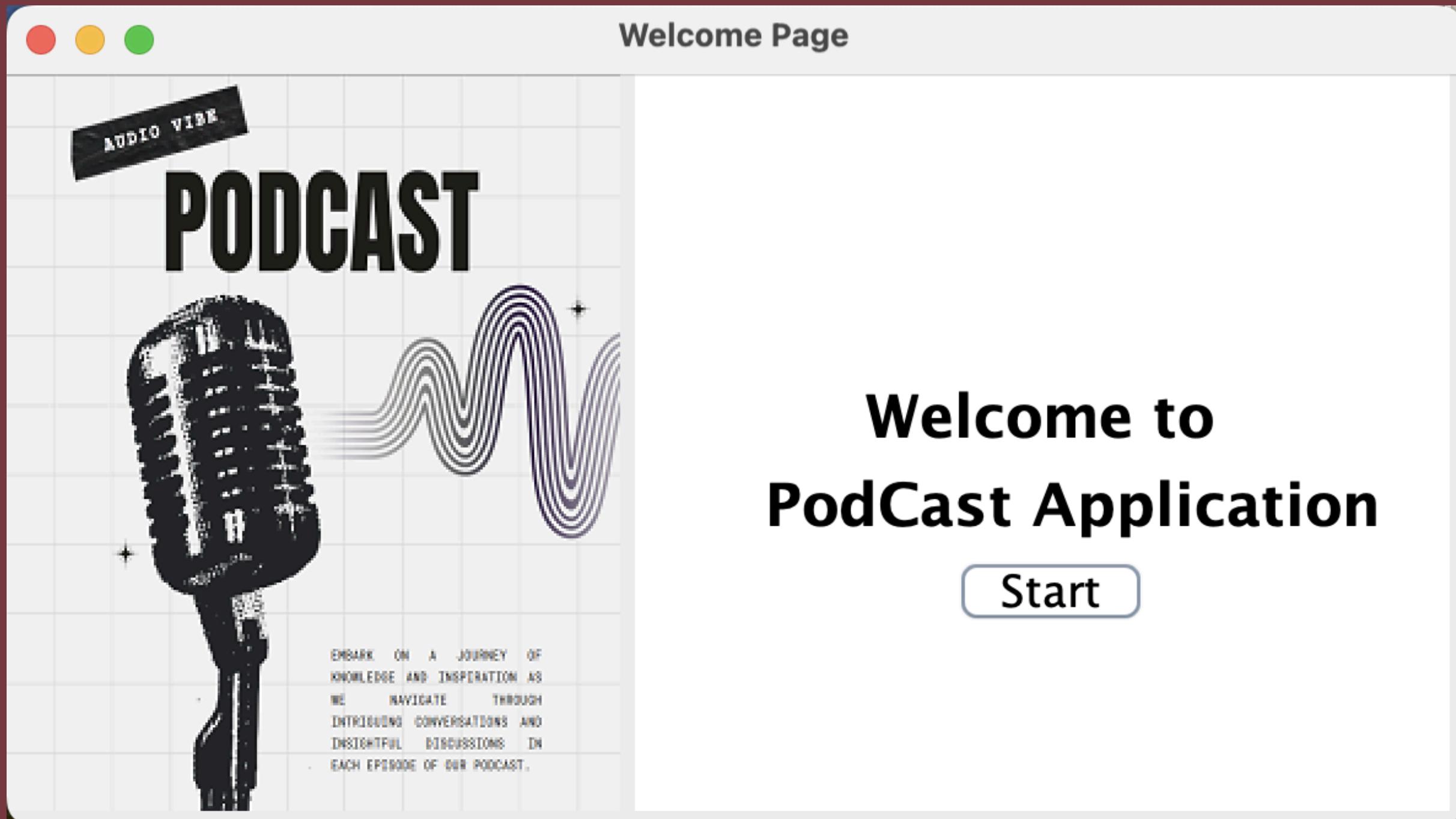
### CHECKED EXCEPTIONS:

- ClassNotFoundException
- SQLException
- InstantiationException
- IllegalAccessException
- UnsupportedLookAndFeelException
- IOException

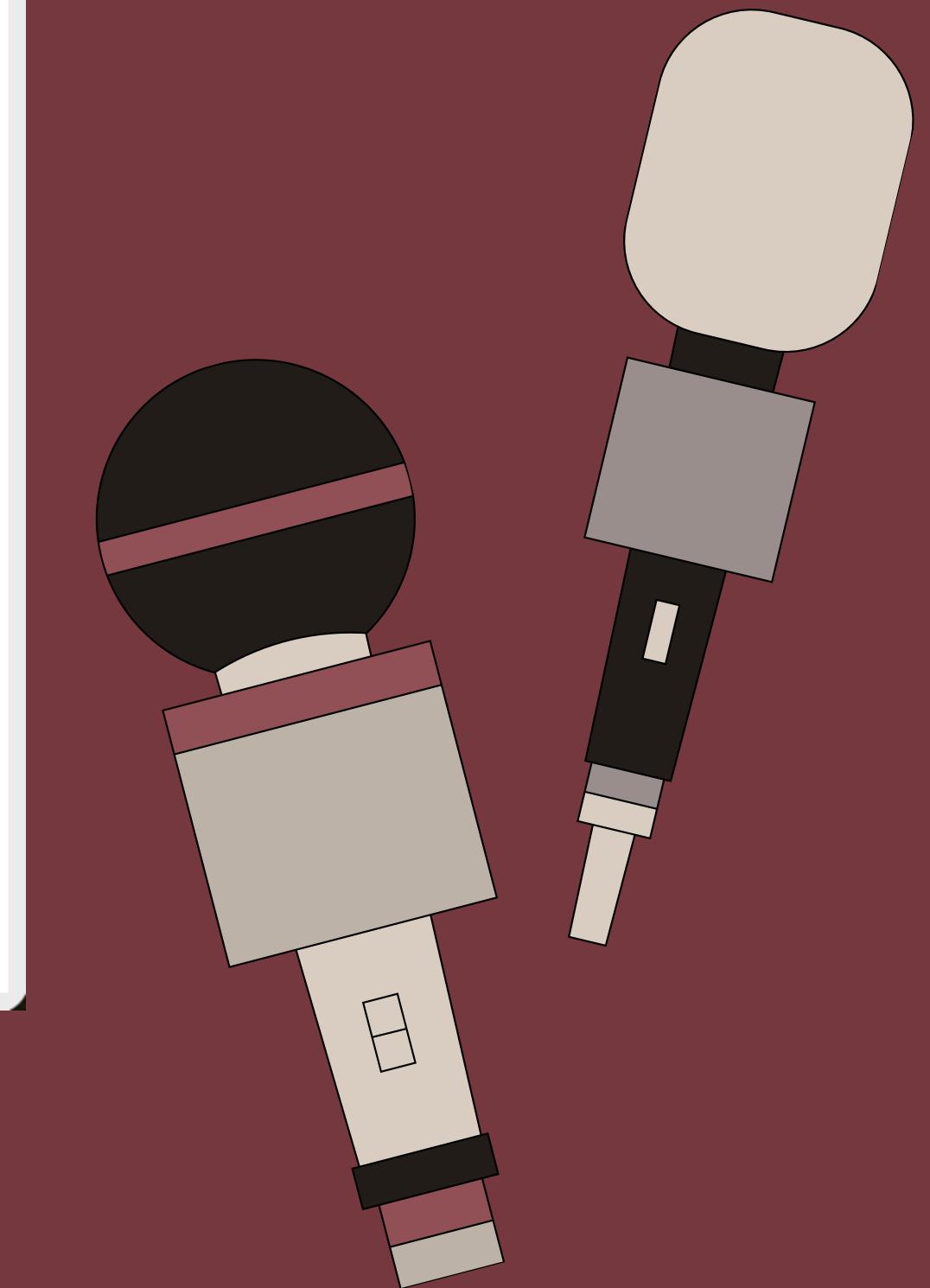
### UNCHECKED EXCEPTIONS:

- UncheckedcheckboxException

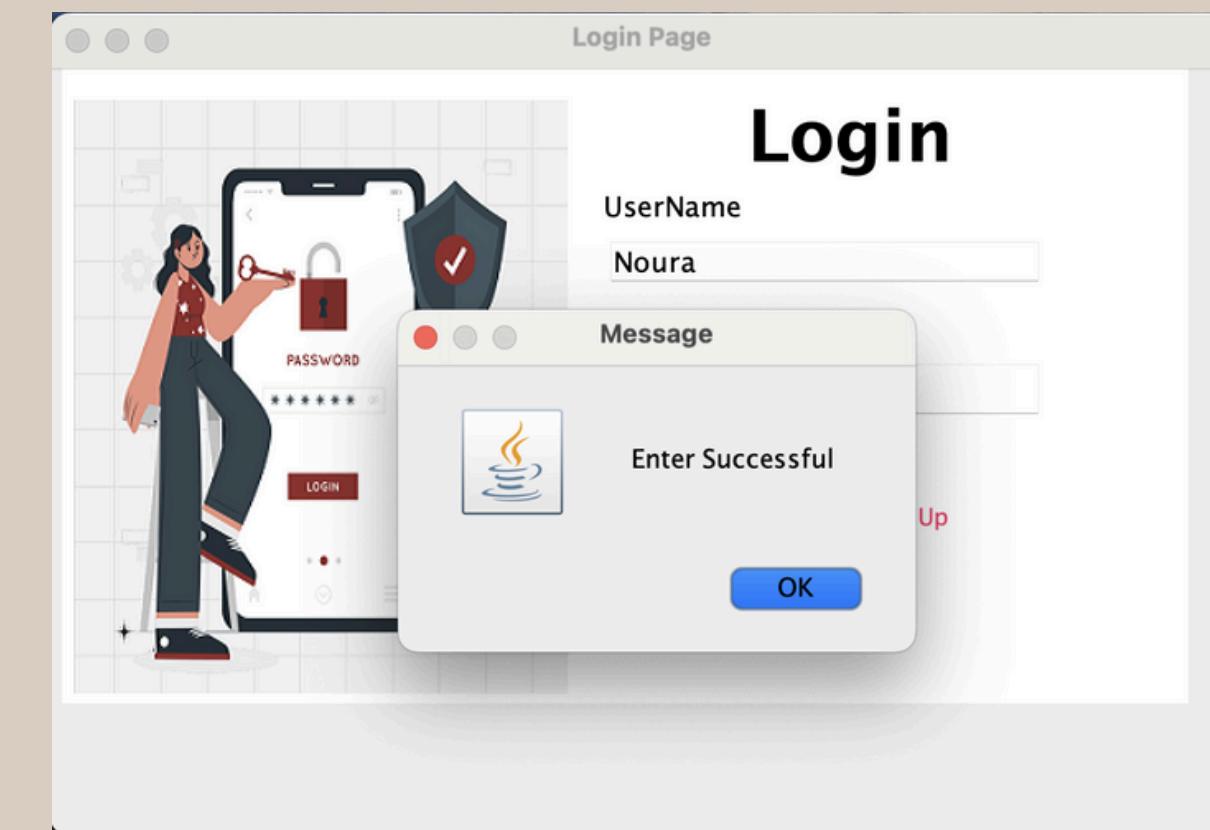
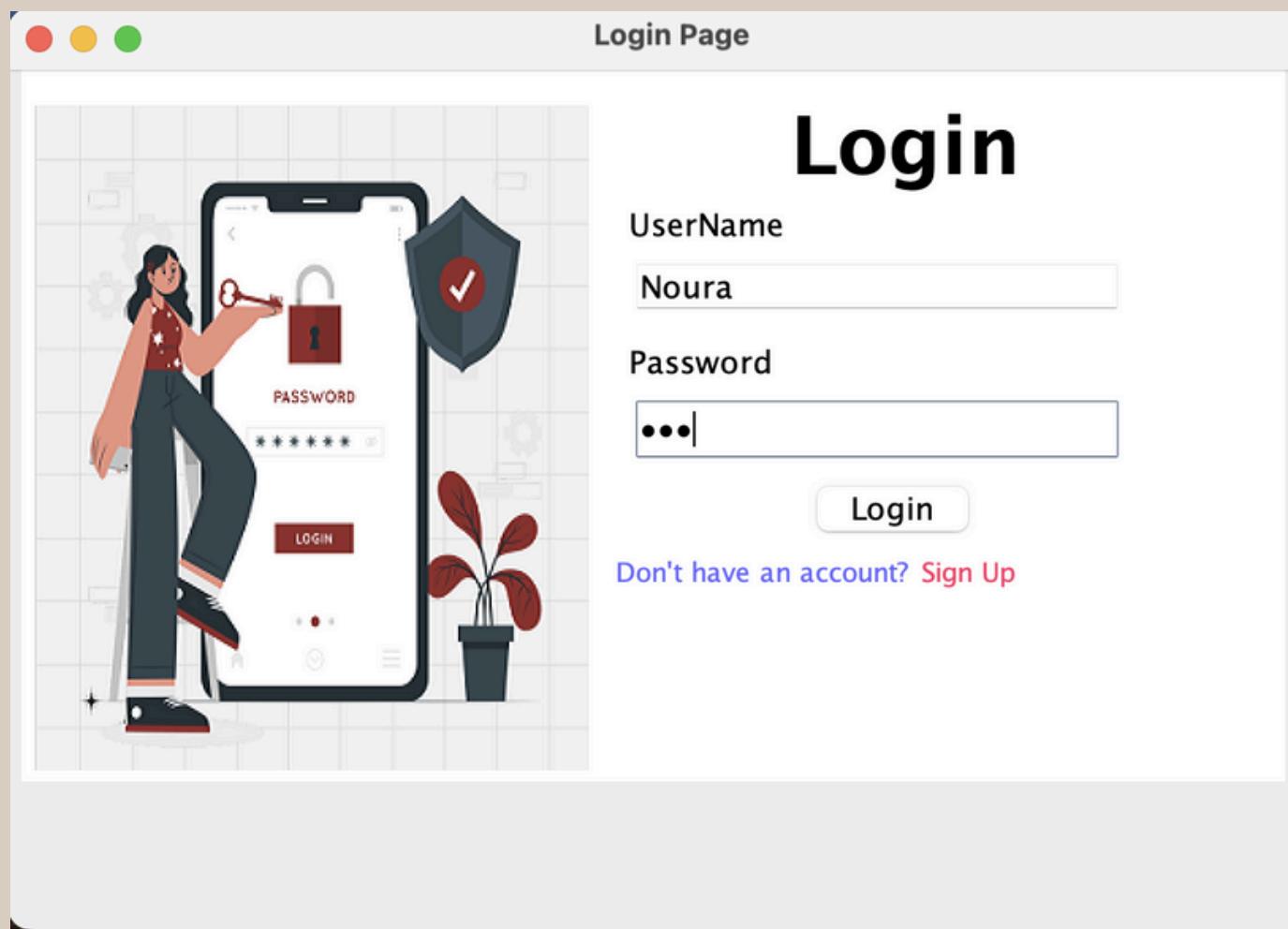
# interfaces



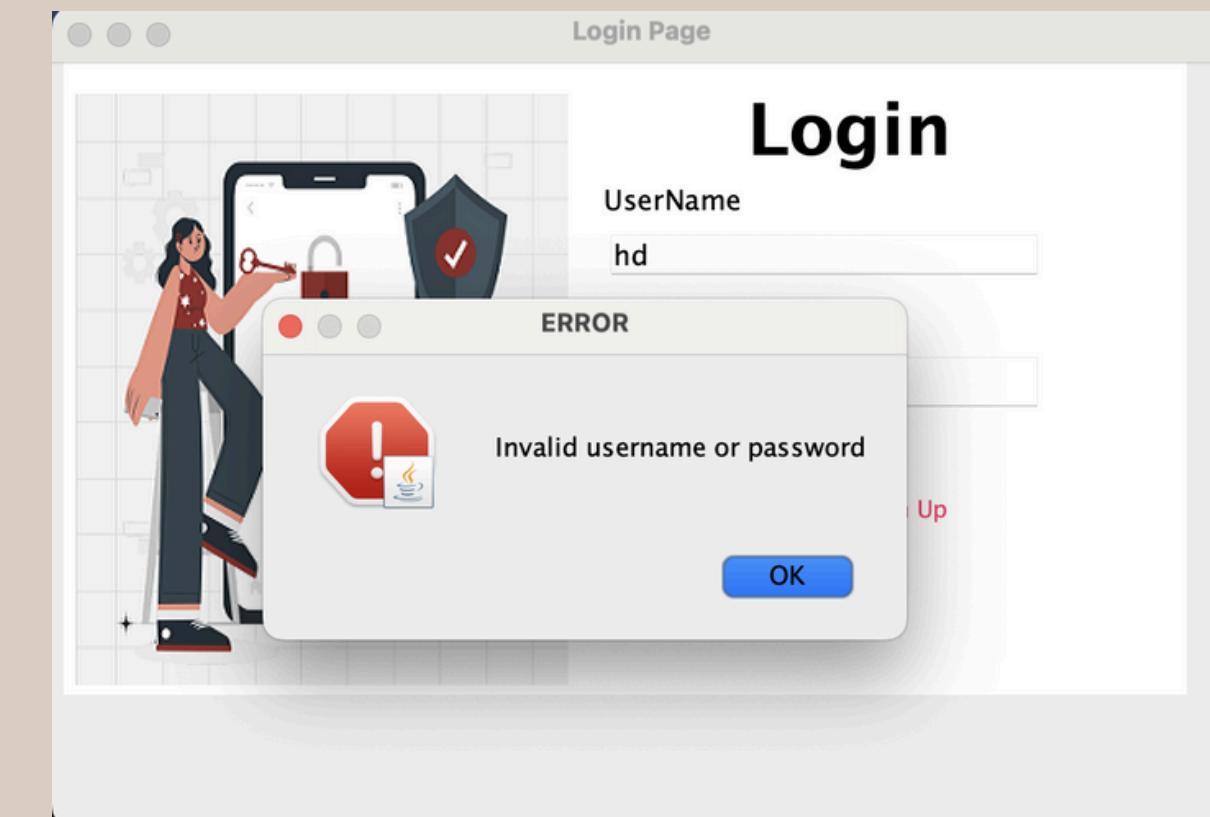
- Welcome Interface: The initial interface featuring the file open option as the greeting message.



# interfaces



- Login Interface: This secondary interface enables both administrators and end users to access and utilize their respective privileges. If the user is not registered in the database, an error message will be displayed. Otherwise, a success message will confirm their login.



# interfaces

**Sign Up**

User Id

UserName: lolo

Password: \*\*\*

Full Name: lolo mohameed

Email: lolo@gamil.com

Phone: 099686

Place of Birth: khobar

Date of Birth: 17/05/2000

Address: Saudi Arabia

Gender: Male

Nationality: 1

register Date: 12-2-2022

User ID: 17

How many podcast episodes have you watched during the...

3

How many podcast program have you watched:

2

Do you listen to non Arabic podcasts:

yes

Would you rather listen or watch the episodes:

listen

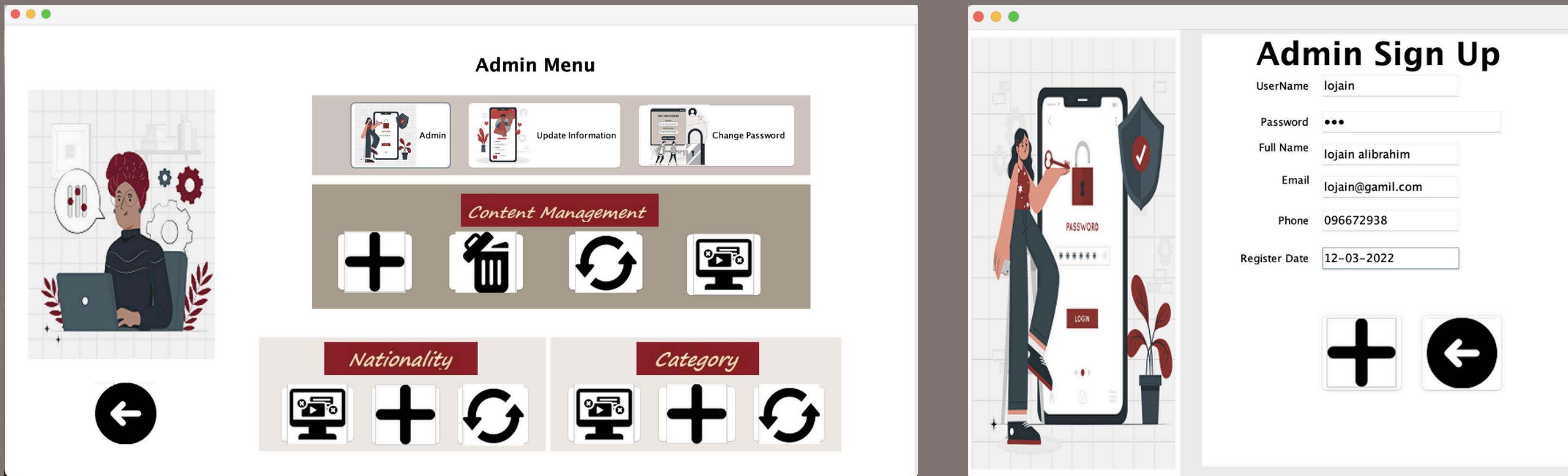
Choose your interests:

Fashion

Submit

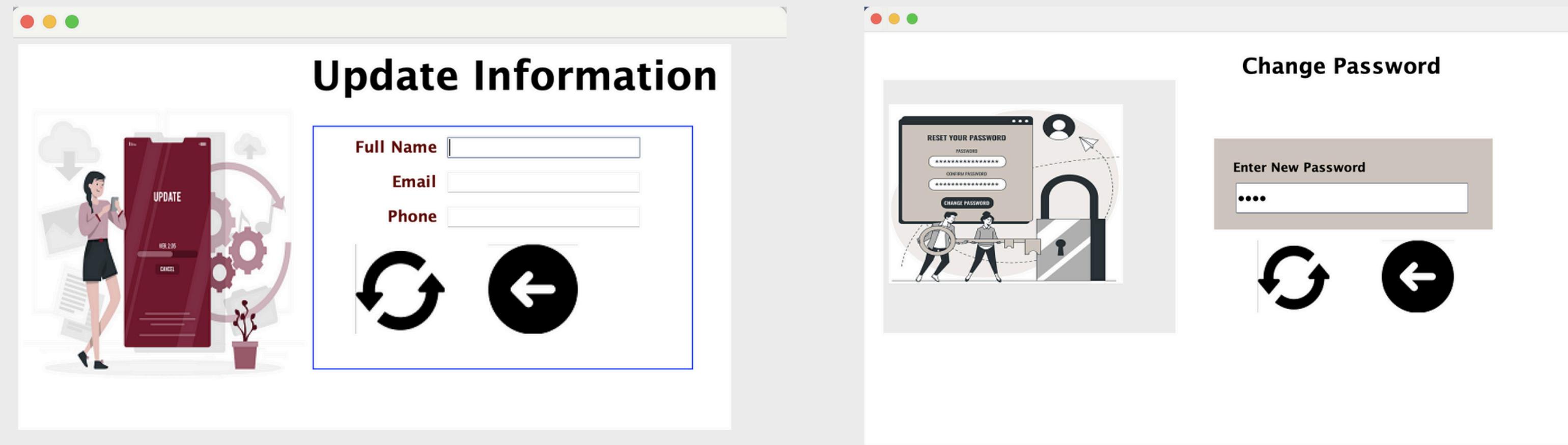
- Sign Up and QA Interfaces: The Sign-Up interface allows new end users to register. After submitting their information, the QA (Questions and Answers) interface is displayed to gather preferences from the user.

# interfaces



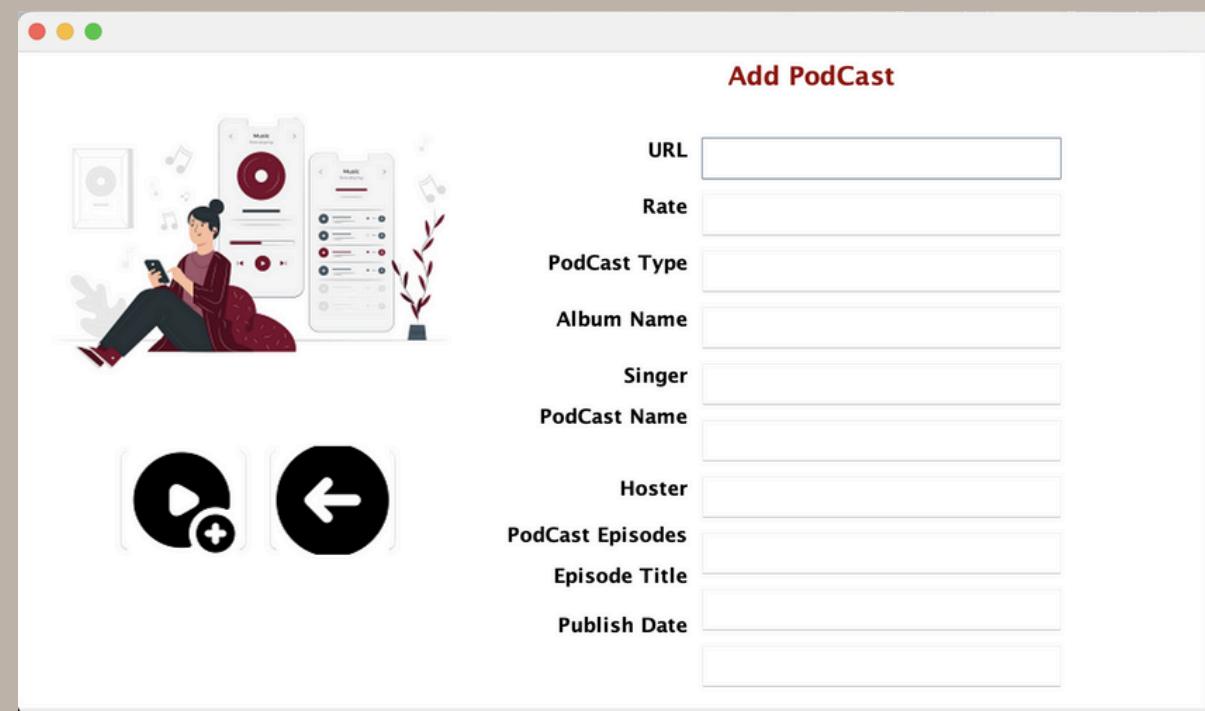
- Admin Interface: This interface facilitates various administrative tasks such as registering new administrators, updating existing admin information or passwords, as well as managing content, nationalities, and categories through options to add, delete, update, or display.
- Admin Sign Up Interface: This interface is specifically designed for registering new administrators into the database.

# interfaces

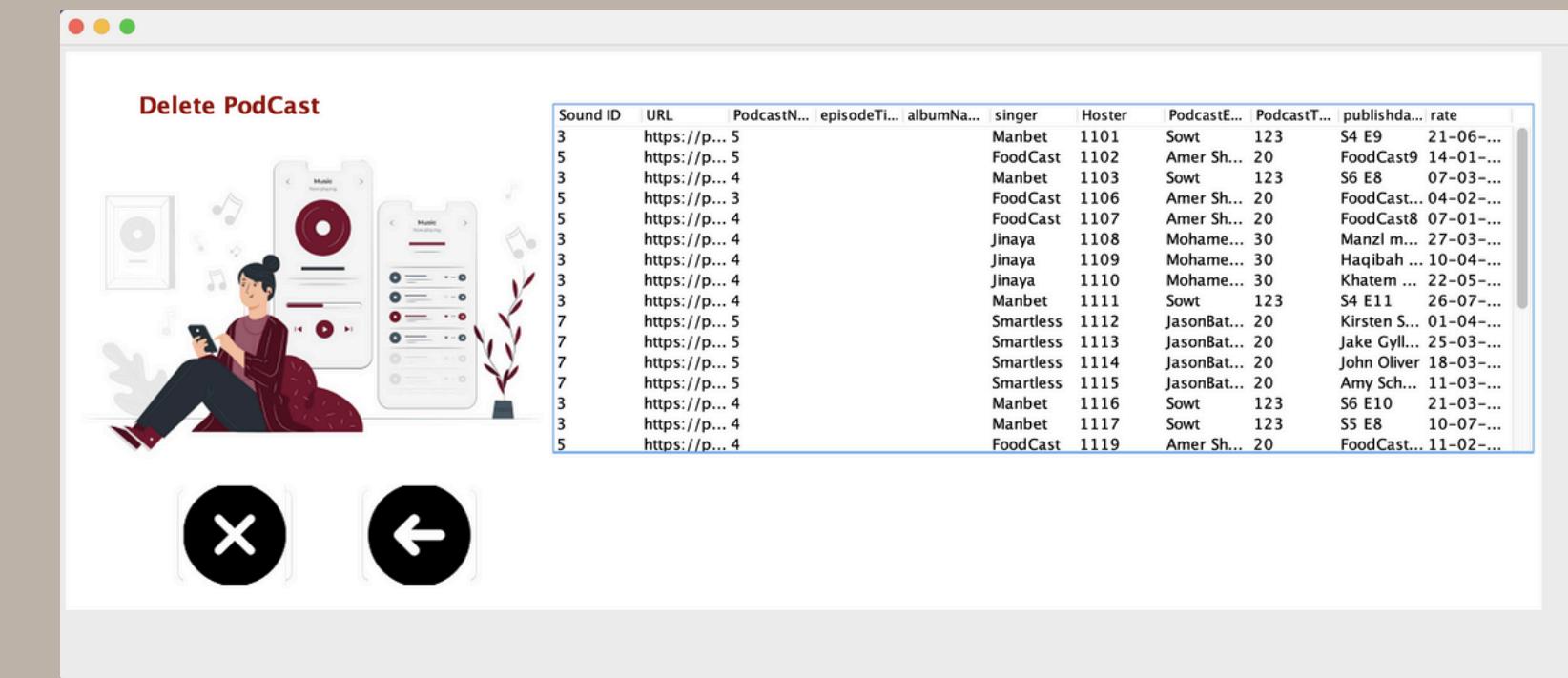


- Update Information Interface: This interface is dedicated to modifying the personal information of administrators.
- Change Password Interface: This interface is accessible both in the admin menu and for end users, allowing them to change their password and update the database with the new one.

# interfaces

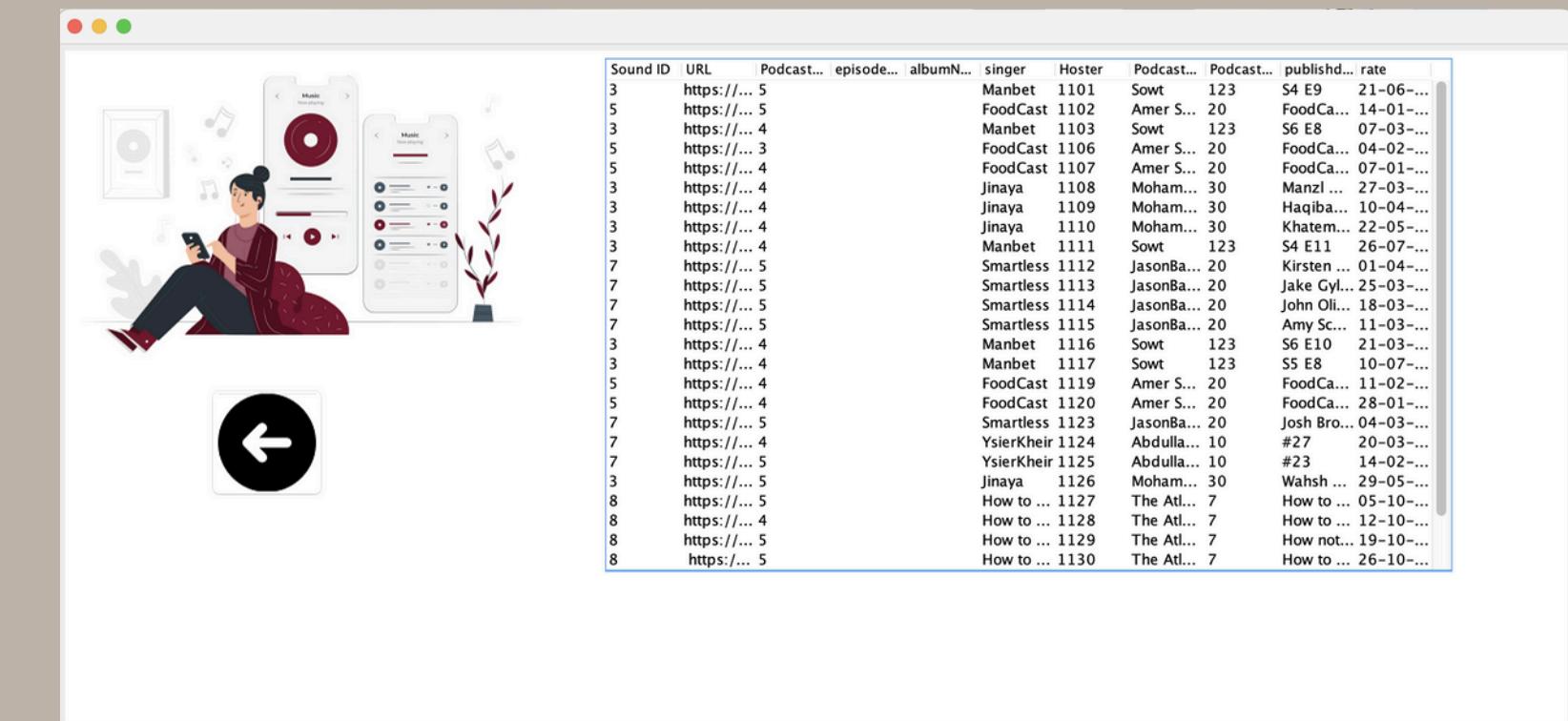


The Add Podcast interface is a form window titled "Add Podcast". It features a background illustration of a person sitting on a cushion, surrounded by various devices like a speaker, a smartphone, and a laptop displaying a music player interface. The form contains fields for URL, Rate, PodCast Type, Album Name, Singer, PodCast Name, Hoster, PodCast Episodes, Episode Title, and Publish Date. Below the form are two large black circular buttons with white icons: one with a play button and a plus sign, and another with a back arrow.



The Delete Podcast interface is a form window titled "Delete Podcast". It has the same background illustration as the Add Podcast interface. The form includes a "X" button and a back arrow button. To the right of the form is a table titled "Delete PodCast" showing a list of sounds from the database. The columns include Sound ID, URL, PodcastN..., episodeTi..., albumNa..., singer, Hoster, PodcastE..., PodcastT..., publishda..., and rate. The data in the table is identical to the one in the Display Sounds interface.

- Add Podcast, Delete Podcast, and Display Sounds Interfaces: These interfaces enables administrators to add new podcasts to the database, delete an existing one, or view the current sounds that have been added to the database for end users.



The Display Sounds interface is a form window titled "Delete Podcast" (though it functions as a display). It features the same background illustration and includes a back arrow button. To the right of the form is a table titled "Delete PodCast" showing a list of sounds from the database. The columns include Sound ID, URL, PodcastN..., episodeTi..., albumNa..., singer, Hoster, PodcastE..., PodcastT..., publishda..., and rate. The data in the table is identical to the one in the Delete Podcast interface.

Sound ID	URL	PodcastN...	episodeTi...	albumNa...	singer	Hoster	PodcastE...	PodcastT...	publishda...	rate
3	https://p... 5				Manbet	1101	Sowt	123	\$4 E9	21-06...
5	https://p... 5				FoodCast	1102	Amer Sh...	20	FoodCast9	14-01...
3	https://p... 4				Manbet	1103	Sowt	123	\$6 E8	07-03...
5	https://p... 3				FoodCast	1106	Amer Sh...	20	FoodCast8	04-02...
5	https://p... 4				FoodCast	1107	Amer Sh...	20	FoodCast8	07-01...
3	https://p... 4				Jinaya	1108	Mohame...	30	Manzl m...	27-03...
3	https://p... 4				Jinaya	1109	Mohame...	30	Haqibah...	10-04...
3	https://p... 4				Jinaya	1110	Mohame...	30	Khatem ...	22-05...
3	https://p... 4				Manbet	1111	Sowt	123	\$4 E11	26-07...
7	https://p... 5				Smartless	1112	JasonBat...	20	Kirsten S...	01-04...
7	https://p... 5				Smartless	1113	JasonBat...	20	Jake Gyll...	25-03...
7	https://p... 5				Smartless	1114	JasonBat...	20	John Oliver	18-03...
7	https://p... 5				Smartless	1115	JasonBat...	20	Amy Sch...	11-03...
3	https://p... 4				Manbet	1116	Sowt	123	\$6 E10	21-03...
3	https://p... 4				Manbet	1117	Sowt	123	\$5 E8	10-07...
5	https://p... 4				FoodCast	1119	Amer S...	20	FoodCa...	11-02...
7	https://p... 5				FoodCast	1120	Amer S...	20	FoodCa...	28-01...
7	https://p... 5				Smartless	1123	JasonBa...	20	Josh Bro...	04-03...
7	https://p... 4				YsierKheir	1124	Abdulla...	10	#27	20-03...
7	https://p... 5				YsierKheir	1125	Abdulla...	10	#23	14-02...
3	https://p... 5				Jinaya	1126	Moham...	30	Wahsh ...	29-05...
8	https://p... 5				How to ...	1127	The Atl...	7	How to ...	05-10...
8	https://p... 4				How to ...	1128	The Atl...	7	How to ...	12-10...
8	https://p... 5				How to ...	1129	The Atl...	7	How not...	19-10...
8	https://p... 4				How to ...	1130	The Atl...	7	How to ...	26-10...

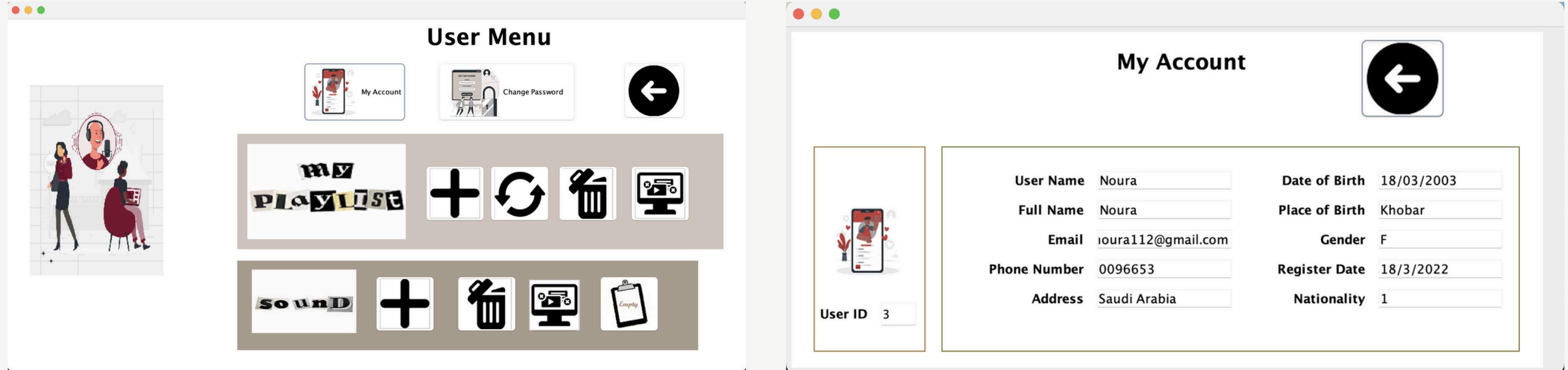
# interfaces

Category ID	Category Name
1	song
2	movie
3	Story
4	Educational
5	Food
6	History
7	Comedy
8	Self improvement

- Nationality and Category Interfaces:  
These interfaces provide functionality to display, add, or update nationality and category values within the database.

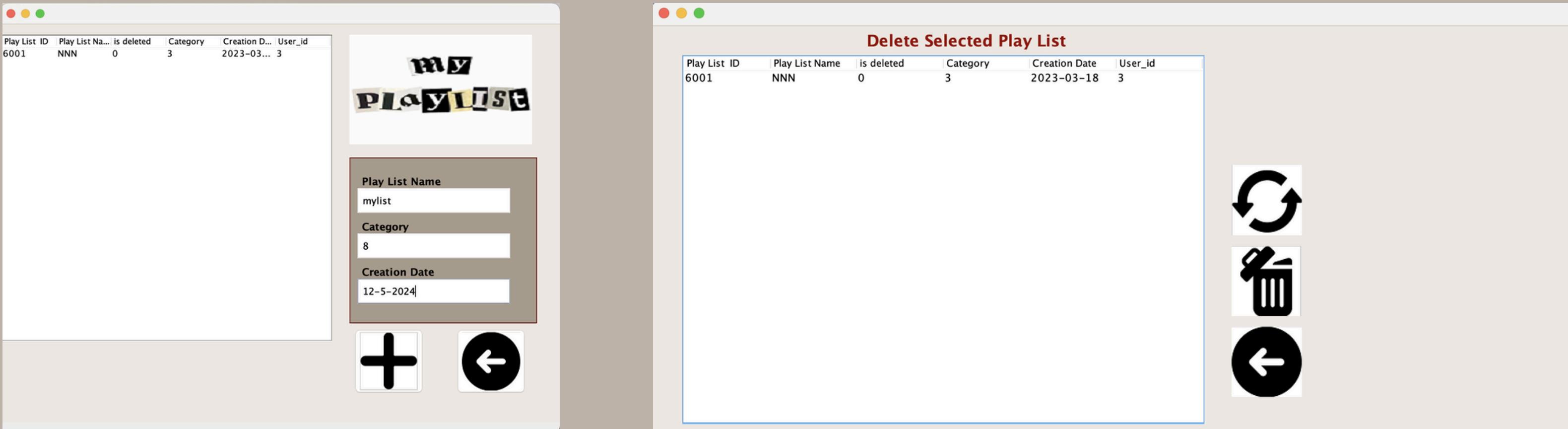
Nationality ID	Nationality Name
1	Saudi Arabia
2	United Arab Emirates
3	Qater
4	Palestine
5	Jordan
6	Afghanistan
7	Angola
8	Anguilla
9	Albania
10	Andorra
11	Netherlands Antilles
12	Argentina
13	Armenia
14	American Samoa
15	Antarctica
16	French Southern territories
17	Antigua and Barbuda
18	Australia
19	Austria
20	Kuwait

# interfaces

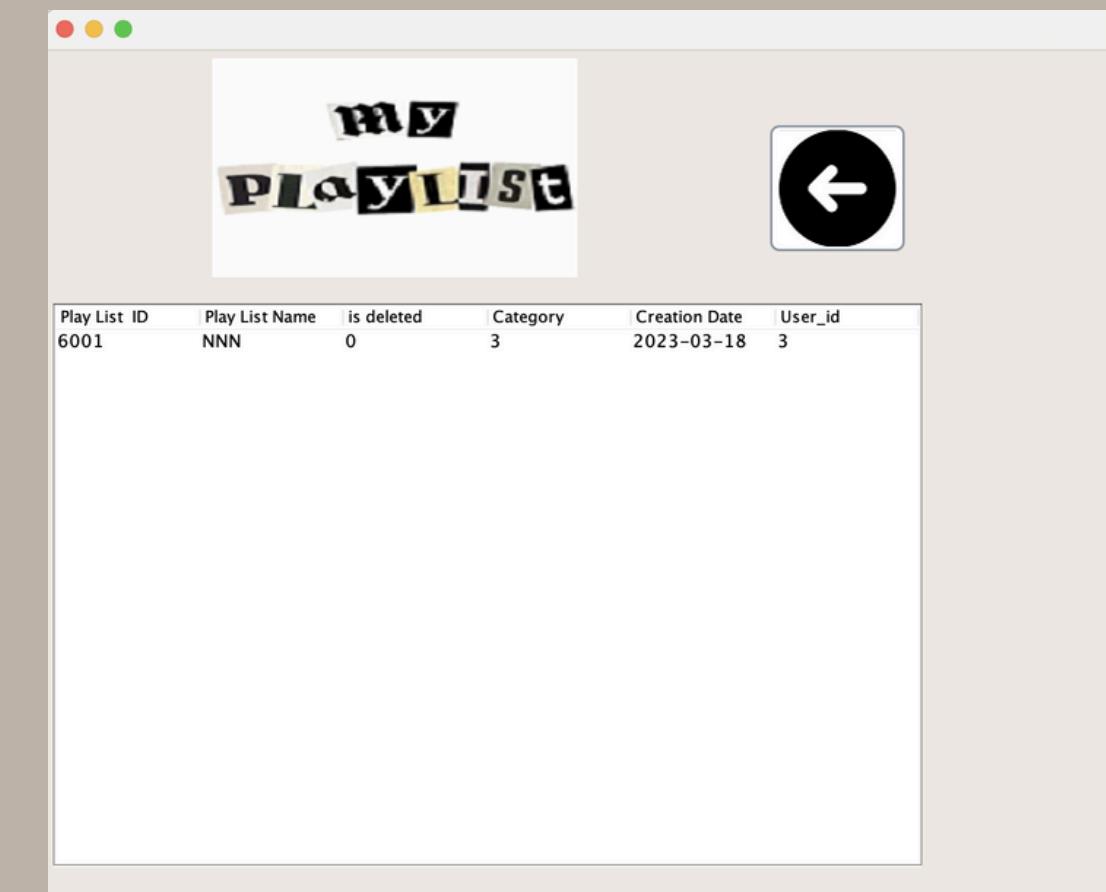


- User Menu Interface: This interface is dedicated to end users and includes options such as 'My Account', 'Change Password', as well as functionalities for adding, updating, deleting, and displaying their own playlists and sounds.
- My Account Interface: This interface is designed to retrieve the current user's personal information from the database.

# interfaces



- Playlist Interfaces: These interfaces enable users to add, update, delete, and display playlists they've created from the available sounds.



# interfaces

The image displays two side-by-side screenshots of a software application interface.

**Left Screenshot:** This screenshot shows a "Play List" window and a "sound List" window.

- Play List Window:** Contains a table with three columns: Play List ID, Play List Name, and Category. One row is visible: Play List ID 6001, Play List Name NNN, and Category 0.
- sound List Window:** Contains a table with five columns: Play List..., Sound ID, Insert D..., Is Deleted, rate, and User ID. Three rows are visible:
  - Play List ID 6001, Sound ID 1123, Insert Date 22-05-0, rate 4, User ID 3
  - Play List ID 6001, Sound ID 1124, Insert Date 10-04-0, rate 4, User ID 3
  - Play List ID 6001, Sound ID 1125, Insert Date 27-03-0, rate 4, User ID 3

**Right Screenshot:** This screenshot shows a "Play List" window and a "sound List" window.

- Play List Window:** Contains a table with three columns: Play List ID, Play List Name, and Category. One row is visible: Play List ID 6001, Play List Name NNN, and Category 0.
- sound List Window:** Contains a table with five columns: Play List ID, Sound ID, Insert Date, Is Deleted, rate, and User ID. The table is currently empty.

Both windows have standard OS X-style title bars and close buttons. The "sound List" windows contain input fields for "Sound ID" and "Insert Date" with placeholder text "Search sound..." and "Search insert date..." respectively, and a large "+" button at the bottom left.

- Sound Interfaces: These interfaces facilitate actions such as adding, deleting, displaying, or emptying the list of sounds.