Project Elective

Hitham

-Shachi Bhavsar (MT2016036)
-Gunjan Acharya (MT2016004)
-Sumedh Gupte (MT2016056)
-Meet Patel (MT2016102)

MOTIVATION FOR THE PROBLEM

Main motive is to create an interactive interface that helps the students of Hitham organization improve their music activities and collect appropriate feedback . They can have collection of songs on their device, so they can play them, and teacher can get feedback about their activities and can manage accordingly.

This is the project which is being deployed. We will get exposure to minute detailing of project development life cycle. We will get the chance to learn new technologies. Some practical scenario of understanding user requirements which one need to understand and design the application, this is a good addon.

THE ORIGINAL SCOPE

The original scope includes two different application:

1) Student Application (Mobile)
2) Teacher and Admin Application (Web)

Student Application

- It starts with a login activity which is used by student for login in to app and they can view a list of all playlists assigned to them, and for each playlist, there are songs which can be downloaded/played.
- There are multiple approaches to design the playlist-song hierarchy association, and current app seperates them into two screens. Different approaches can be worked out in future.
- When a song or a playlist's information is not available in app's local storage, it's icon will be downloaded during the login phase.
- The period after a login request is successful and the app actually changes the screen, is variable and depends on the amount of new content that is available w.r.t the previously stored content.
- Songs will be downloaded the first time you play them.

- The main component is music player. This should include play, pause and seek functionalities. It should display some information about the song it is playing.
- The app is for students who are suffering from autism which creates one necessary requirement of pictures and colors for songs. This information also need to accommodate and will be displayed with song.
- Another functionality is student activity. When they are playing song, they will listen certain portion of song only, and we also track the playing frequency. They are seeking to position in song. This all activity information is needed by teacher to analyze student progress and effect of songs on them.

Teacher and Admin Application

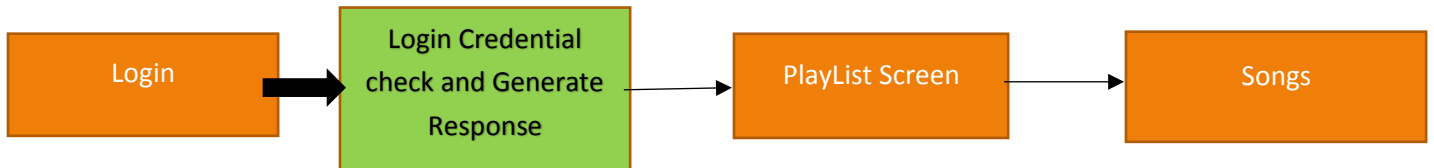There are two persons who manages the activities at different level.

Role of Admin:

- Admin will create student and teacher profiles.
- Admin is also responsible for adding songs into repository.
- Student information is recorded, and a username-password is generated by admin. Same is done for teacher also.
- Admin will assign student to teacher. One teacher can handle more than one student.
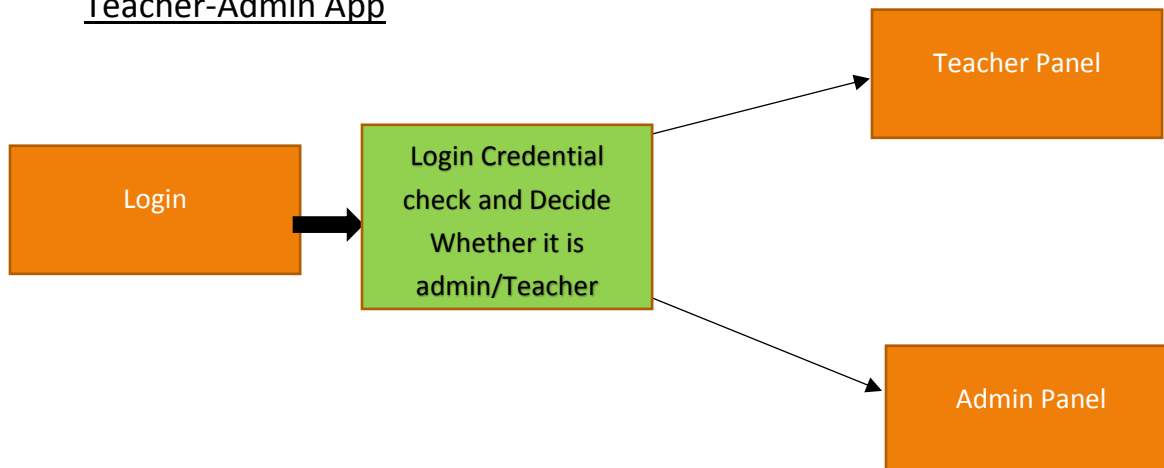
Role of Teacher:

- Teacher will create playlist and assign songs to it.
- Teacher can create recording for song. Picture and color are associated with every recording.
- Teacher will add recording to playlist.
- Teacher can also assign playlist to their students. She will get record of student activity.
- Teacher or admin can make changes in the thing they have done. Edit and delete option should be there in every functionality.

THE DESIGN APPROACH AND HIGH−LEVEL DESIGN

Student App

| Login | → | Login Credential check and Generate Response | → | PlayList Screen | → | Songs |

Teacher-Admin App

| Login | → | Login Credential check and Decide Whether it is admin/Teacher | → | Teacher Panel |
| | | | ↘ | Admin Panel |

- In the above diagram Green boxes show the functions done at the back end. Orange boxes are high level modules.

Detailing of above blocks:

Admin has three tabs:

1)Student – create/delete/edit student, reset password is in edit module

2)Teacher – create/delete/edit teacher, Assign Student, reset password is in edit module

3)Songs- create/delete/edit song details.

Teacher has four tabs:

1)playlist- create/delete/edit playlist

2)recording- create/delete/edit recording

3)student- assign playlist to song

4)Activity – view student activities and download in form of excel

Hierarchy of Song-recording-playlist:

- Song is the basic component which contains songurl, name, singer, taala, raga, composer details.
- For each song we can have different recordings. Recording has extra details color, picture associated with song. One song can have more then one recording. So that teacher can assign pictures and color according to them for their students.
- And in last you can add recordings to playlist.
- Playlists are assigned to students.
- One song can be assigned to different playlists.
- Teacher can assign any playlist to their allocated students only.

Other

- Teacher can see activities of their student only.
- Students can have more than one teachers assigned.
- Relationships in backend are shown in below database diagram.
- For delete operation in any table, no actual delete is happening. Table_status column is handling deletion operation. (status can be active/deactivated)

Below EER diagram shows the overall structure of back end database.

Database Design:
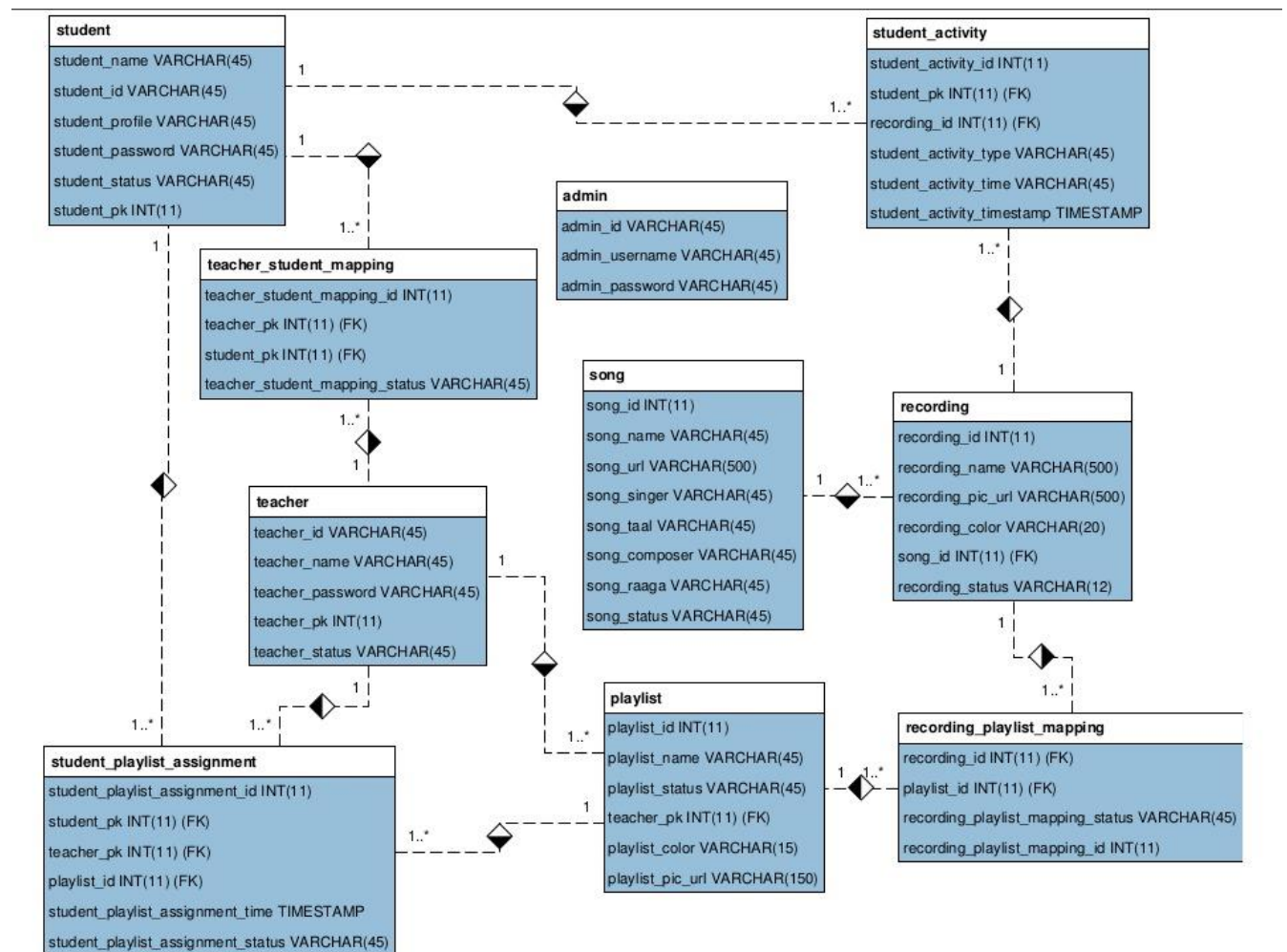
TOOLS/TECHNOLOGIES

Back End – Java, REST (Jersey)

IDE – Eclipse mars

Database – MySQL

Front End – HTML5, Bootstrap, JavaScript, jQuery

Apache Maven build automation tool

Maven Dependencies:

- mysql-connector-java – used to connect to MySQL database
- jersey-media-moxy – for JSON binding in Jersey
- org.json – to manipulate JSON object

Mobile App – React-Native

Packages of React-Native:

- AsyncStorage – for local storage
- RNFetchBlob – to download image and songs
- React-Native-Sound – for music player
- NetInfo – to check status of internet

WHAT WAS ACHIEVED AND UP TO WHAT EXTENT PLANNED SCOPE WAS COMPLETED

Planned scope is completed with most of functionalities. Some functionality need to improve as per the suggestions by actual users.

- Web application – It is working fine. Data (Song, images) are being stored in google drive. It is smoothly working. CRUD operations of student, teacher, song, playlist, recording is tested completely.
- Rest API - is developed for all functionalities of scope. Student Activity is generated in excel format. It can be analyzed easily.
- Security- For this stage we have implemented security handling for SQL injection attacks. Password encryption is done. It can be extended.
- Mobile Application- It is smoothly working for all basic functionalities. Audio player is designed in a way that it can handle all use cases of play, pause, seek. Information is going to server about these activities.

Main intention is to develop the functionalities that can be extended by next team easily is done successfully.

CHALLENGES FACED

- Design database hierarchy for song, recording and playlist.
- We need to be careful while taking small decisions. This need to be taken with a future version expectation and need also.
- Most often we faced challenges are related to react-native.
  1. Music player design. This should be simple yet incorporates all basic things. Find the appropriate music playing component in react is tough task.
  2. React native is evolving technology. So, there are many deprecated things from previous versions. It is used for both ios and android combine development.
  3. Understand Asynchronous storage design and to use it.
  4. Stack navigation of react-native library.

- Storage is done in google drive. It generates different URL while using sharable link and other options from mobile and desktop site. For making application not to fail in any case in need to consider all the cases.
- Security is also one of the challenges. Design password security using encryption.

FUTURE WORK AND OPEN ISSUES

IOS application – In future ios application need to be developed.

UI – This is the main thing that need to be done. React UI components need to be implemented carefully. Design of UI structure according to user feedback.

Emotions – Student emotions on song need to be recorded and on the teacher app side it should be shown.

Navigation – stack navigation is used in mobile application. This need to understand thoroughly.

Logout- Application logout is an issue. And while solving this, we need to be careful about android and ios application structure. For exiting the app, there are some solution need to be implement.

Security – more advance security can be implemented.

Parent app – It need to be designed. We think that this app will be like student app and will be having some parent related specific details. E.g feedback on songs.