**Mobile Computing: Final Project Report**

**Team 0MBPS :** Bhavesh Khatri, Mohd. Huzaifa, Sitaram, Prince Yadav

**Motivation**

Nowadays most of the digital content we consume is via smartphones. When some friends in a group want to binge some common media content, then they will have to play the concerned media file separately on each of the smartphones which might not be synchronous on all the devices.

Or when some people in a group are vibing on a song or want to have a small party with some music, they will require a speaker, as the volume of a smartphone is very low for a party like celebration and also, it is very hard to manually synchronize the playback of song on multiple smartphone devices.

The motivation is to develop an interface which can play media files synchronously on multiple devices which can help people watch movies/webshows in a group, synchronously on their smartphones. It can also eliminate the use of the small speaker if we have enough smartphones which can multiply the volume of the song being played.

**Problem statement**

Develop an android application that can play media files on multiple mobile phones synchronously.The host can control the playback(play/pause/forward) of the media files on the connected devices.

**Related work**

[**AmpMe**](https://www.ampme.com/) is a mobile application that enables audio synchronization across various mobile devices for a better experience.

[**SyncPlay**](https://syncplay.pl/) is a computer application that enables video synchronization across various computers. But it is not compatible with mobile devices such as smartphones.

**Features proposed**

1. Stable connection among multiple devices using wifi-connection.
2. Synchronous playback of audio files among the connected devices.
3. Ability to play and pause, interact with seekbar in Exoplayer, from the host device.
4. Sharing and playback of the video media files among the connected devices.

**Features implemented**

1. Wi-Fi connection among multiple devices.
2. Synchronized audio playback on multiple devices using socket server and client.
3. Play/Pause and progress bar implementation in the media player.
4. Interacting UI with toasts and hints for easy navigation.

**Challenges faced**

1. The main challenge was to synchronize the playback of the media files on the linked devices. To accomplish this, we used socket programming for the communication among the devices during play and pause operations on the host device. But this communication also takes some time in milliseconds which can introduce some delay. To get rid of that, we added some offset delay of T seconds based on the time difference between the devices to bring all the devices to a common timeframe to start the media player.
2. We initially used WiFi Direct for the connection among the devices, but not all the smartphone devices have the support of WiFi Direct. So, we shifted to utilizing WiFi connection for the connections.
3. We had to switch to the ExoPlayer from MediaPlayer because when we extended our app to multiple devices, we faced some issue in the synchronization as there was some delay in the playback in the devices, though the logic of the code was fine. We got improved results after switching to exoplayer. Exoplayer is more robust than mediaplayer and provides some amazing features and customizations.

**Future work**.

1. Making it energy efficient
2. Provide 3D sound effects.
3. Extend this application to other media files,eg. video files.
4. The idea can also be extended to play the live streaming media on multiple devices.

**Individual Contribution:**

| Bhavesh Khatri | 1. Creating Wifi direct connections 2. Creating Socket connections 3. Managing Music transfer and sending sync\_messages through socket for synchronization 4. Making it work for multiple devices 5. Switching from mediaPlayer to Exoplayer |
| --- | --- |
| Mohd. Huzaifa | 1. Creating Wifi direct connections 2. Creating Socket connections 3. Managing Music transfer and sending sync\_messages through socket for synchronization 4. Making it work for multiple devices 5. Switching from mediaPlayer to Exoplayer |
| Prince Yadav | 1. UI design + related switching/flow of activities 2. Selection of media files for transfer 3. Receiving sync\_messages and managing mediaPlayer accordingly. 4. Switching from wifi-direct to wifi (or same network) 5. Synchronization support, when user slides the progress bar to any point or (forward/backward) |
| Sitaram | 1. UI design + related switching/flow of activities 2. Selection of media files for transfer 3. Receiving sync\_messages and managing mediaPlayer accordingly 4. Switching from wifi-direct to wifi (or same network) 5. Synchronization support, when user slides the progress bar to any point or (forward/backward) |