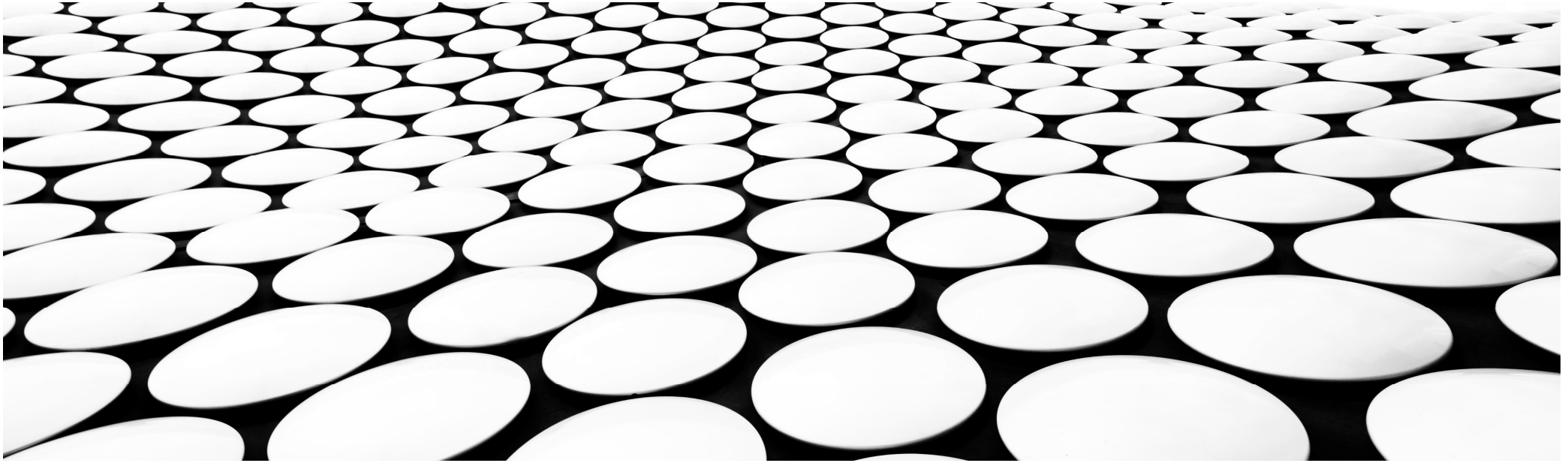
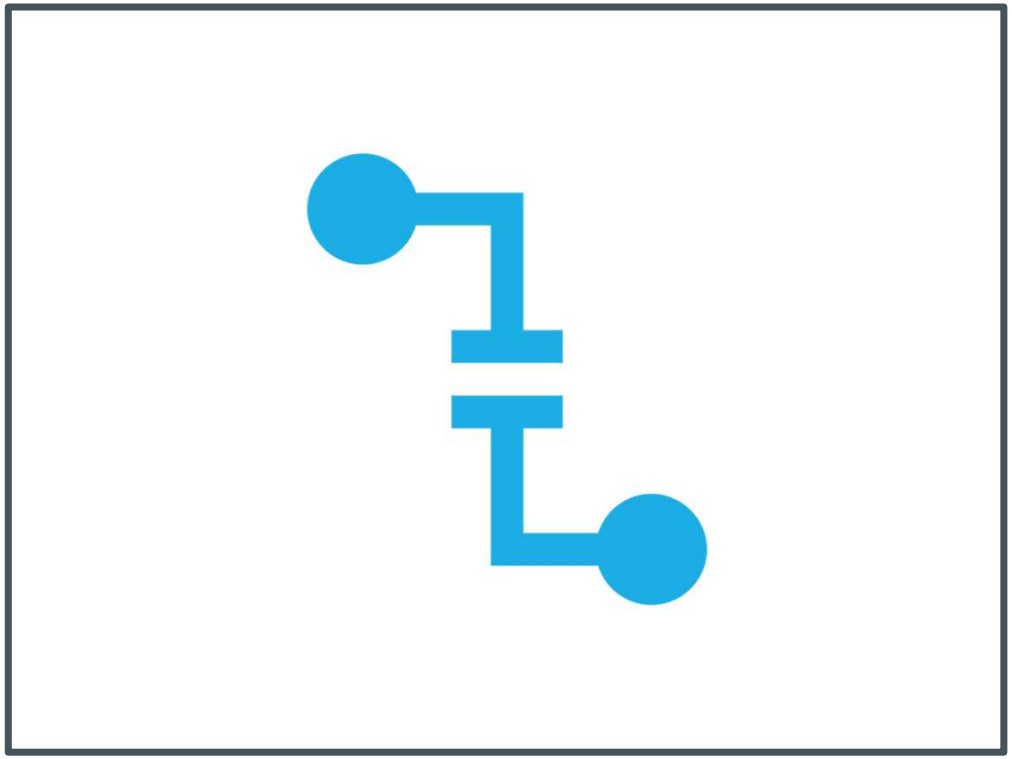
INTERNSHIP PRESENTATION

VIBHOR SENGAR



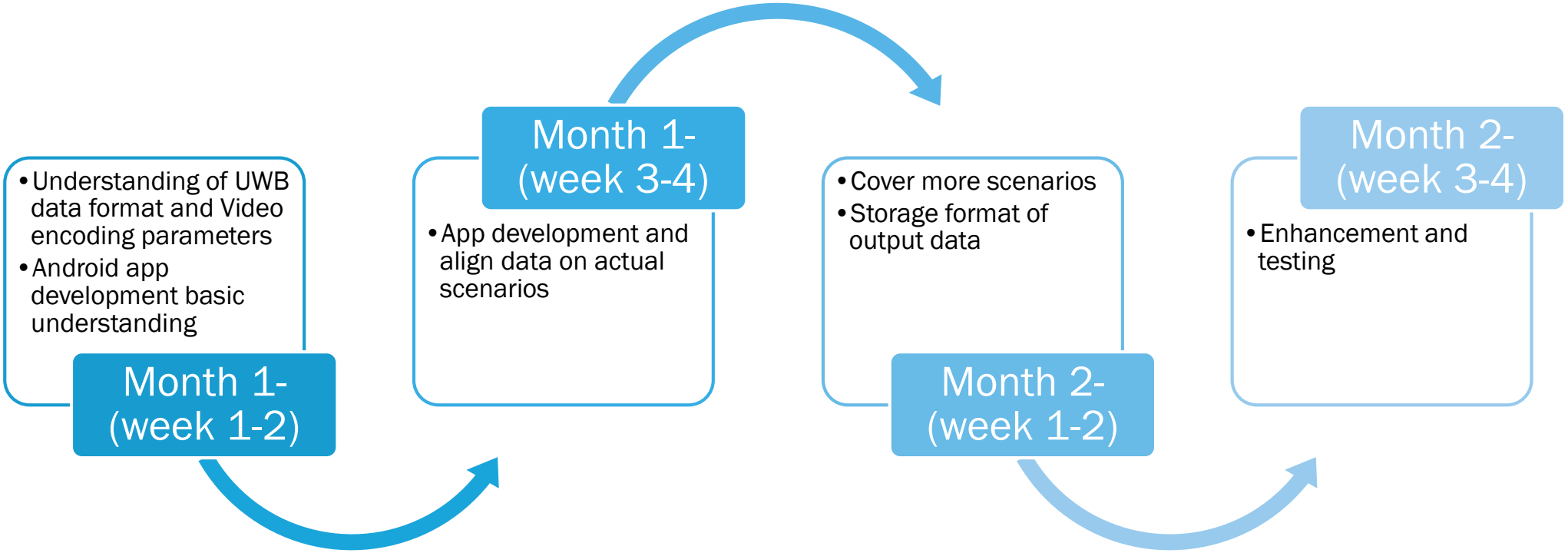
PROBLEM STATEMENT

UWB DATA ALIGNMENT WITH UNDERLYING ENVIRONMENT AND DATA COLLECTION



- UWB radar data is reflected impulse temporal data obtained from real objects in the environment.
- The alignment and preparation of timestamp-based mapped data to the actual environment are crucial in this project.
- The main objective is to develop an app that records video and captures UWB data simultaneously, aligning it with the recorded video.
- The video-recording parameters and UWB data temporal parameters may vary.
- The app aims to synchronize the UWB data and video to ensure accurate mapping of the data to the corresponding video frames.

PROJECT PLAN



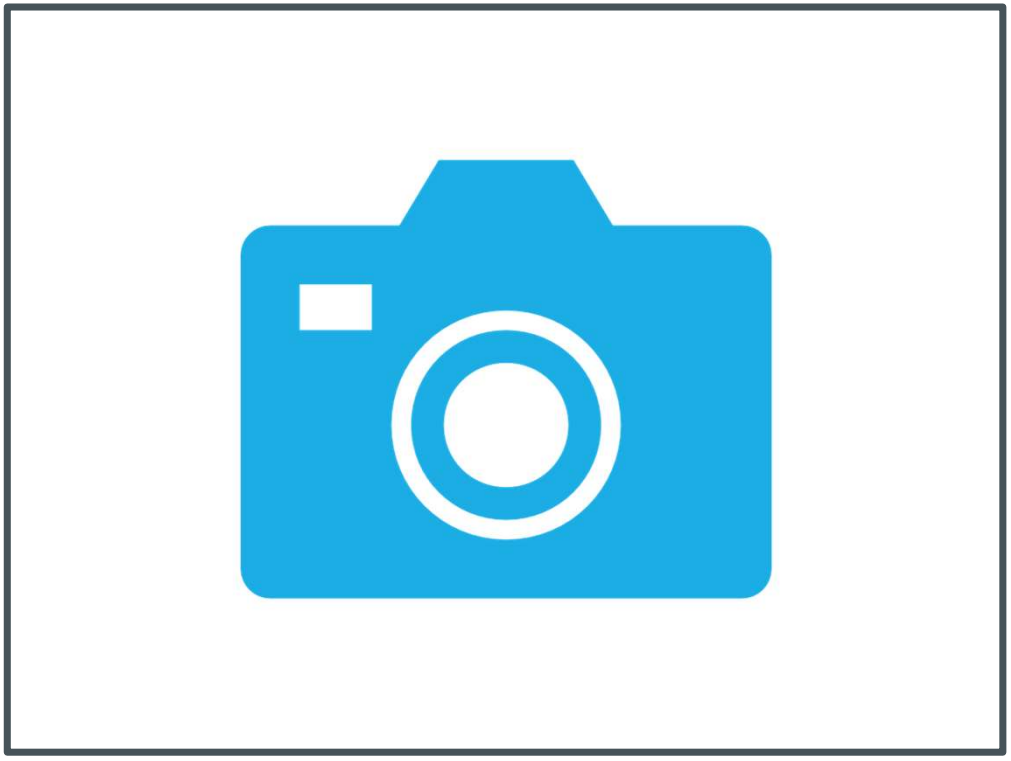
APPROACH

Android data alignment app

Proper timestamp alignments

Prepare storage format for different scenarios data

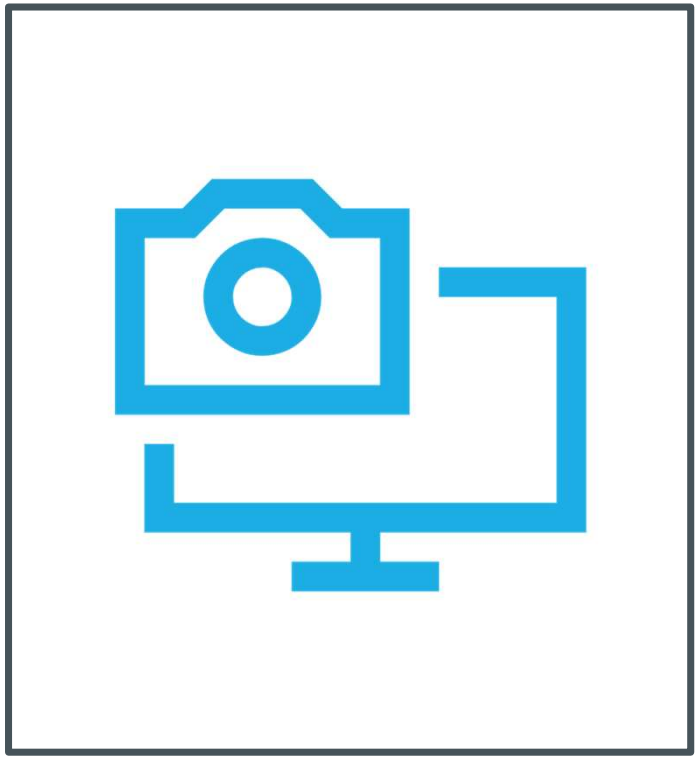
VIDEO RECORDER APP



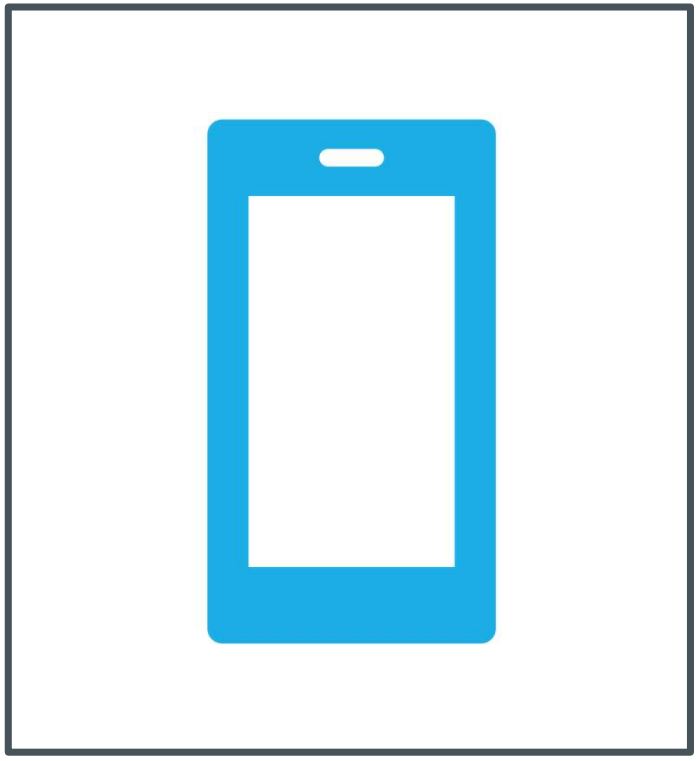
- TextureView, CameraManager, Imagereader and MediaRecorder were used to create a video recorder app.
- TextureView displays the live camera preview on the app's UI.
- CameraManager provides access to the device's camera(s) for capturing video frames.
- Imagereader captures frames from the preview or recorded surface.
- MediaRecorder handles the recording of those frames to create a video file.
- By utilizing these components, I developed a functional and user-friendly video recorder app.

TEXTUREVIEW

- TextureView is an Android view class that efficiently displays video streams or OpenGL ES rendering, making it ideal for showing the device's camera output.
- It enables the integration of live camera previews directly into our app's user interface.
- With TextureView, we have fine-grained control over the camera preview's positioning and scaling within your app's layout.
- Additionally, TextureView is compatible with OpenGL ES, allowing you to incorporate custom graphics or effects seamlessly.
- Overall, TextureView offers a powerful solution for displaying camera output in real-time, making it a valuable tool for creating engaging camera-based applications.



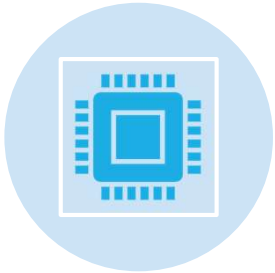
CAMERAMANAGER



- The CameraManager class is an Android API that provides access to the device's camera(s) and their functionalities.
- It acts as an interface between the app and the camera hardware, allowing developers to interact with the camera features programmatically.
- By using the CameraManager, developers can open and close the camera, configure camera parameters (such as resolution and focus), capture still images or video frames, and perform various camera-related operations.
- The CameraManager library simplifies the process of managing the camera functionality within an app, providing developers with the necessary tools to create camera applications with ease.

IMAGEREADER

The ImageReader class is an Android API that provides an efficient way to capture and process individual frames or images from various sources, including the camera.



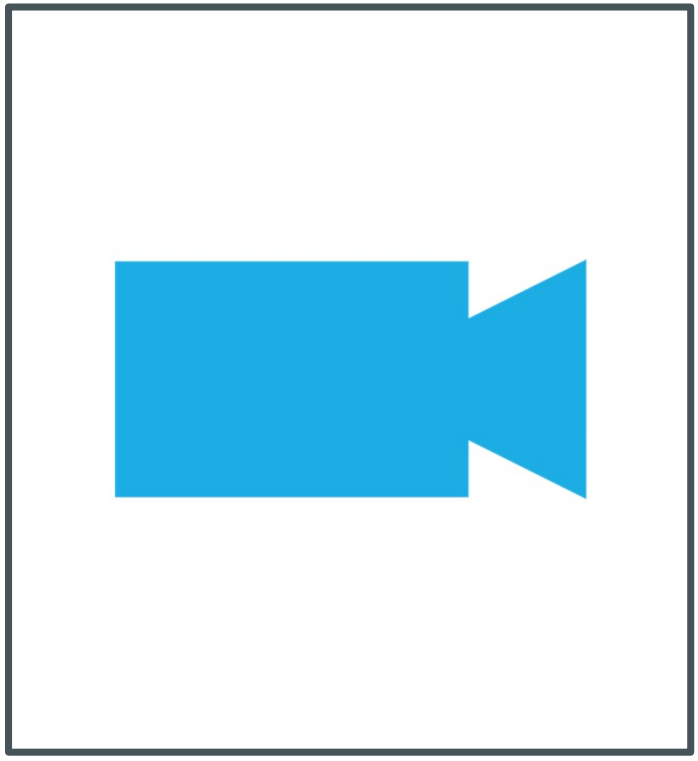
It allows developers to access and manipulate raw image data, making it useful for tasks such as computer vision, image processing, and real-time analysis.

With ImageReader, you can set the desired image format, size, and the number of images to be processed. It provides a callback mechanism to retrieve each captured image, enabling developers to perform custom operations on the images.

The ImageReader library simplifies the process of capturing and working with individual frames, offering flexibility and control in image-based applications.

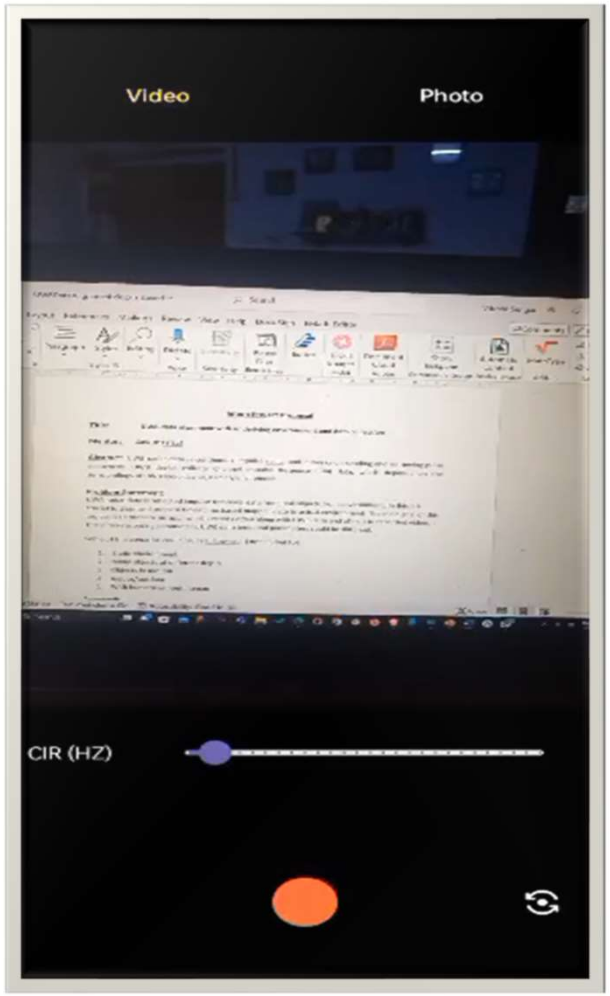
MEDIARECORDER

- The MediaRecorder class is an Android API that provides an interface for recording audio and video.
- It offers a convenient way to capture multimedia content from various sources, such as the device's camera or microphone.
- MediaRecorder allows developers to configure recording parameters, including audio and video sources, output format, encoding settings, and output file location.
- It handles the encoding and storage of captured audio and video data into a desired format, such as MP4.
- The MediaRecorder library simplifies the process of recording audio and video, providing developers with an efficient and straightforward solution for incorporating multimedia recording capabilities into their applications.



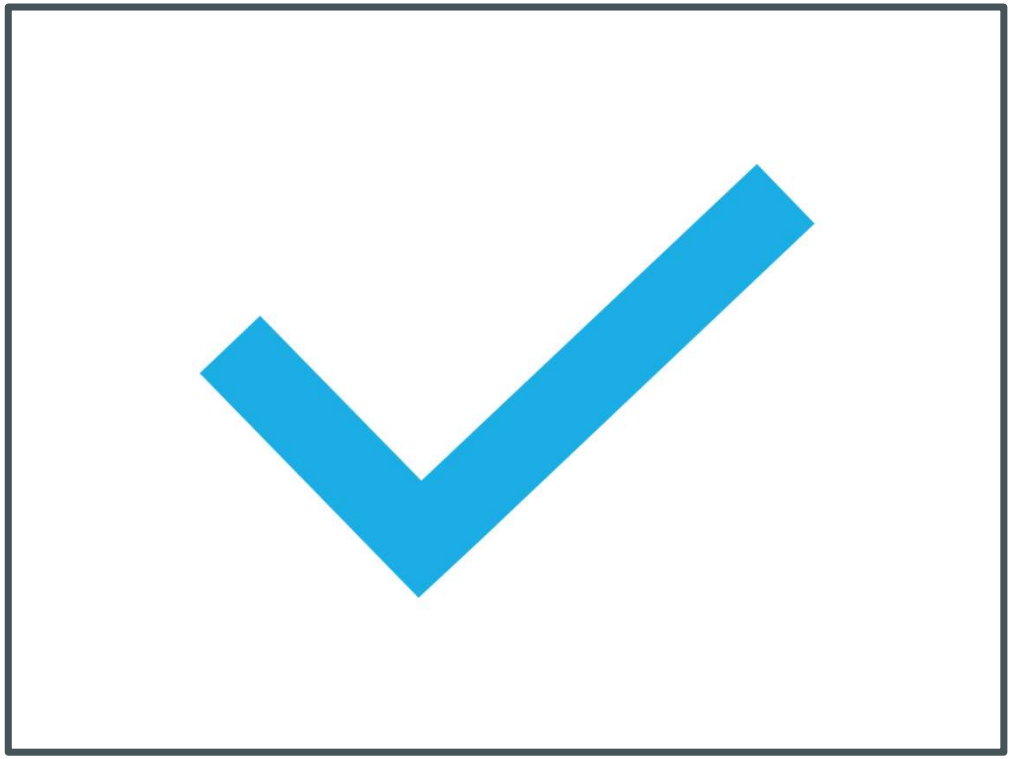
ALIGNMENT AND STORAGE

- The app allows users to save the recorded videos to the "Movies/Camera2VideoImage" directory.
- Additionally, it performs real-time timestamp alignment with UWB (Ultra-Wideband) CIR (Channel Impulse Response) data timestamps.
- The alignment process ensures that the recorded video is synchronized with the corresponding UWB CIR data.
- Furthermore, the aligned timestamps are stored in Excel format and saved to the "Download/Timestamps" directory.
- This allows for easy access and analysis of the synchronized data.
- By combining video recording, real-time timestamp alignment, and Excel storage, the app provides a comprehensive solution for capturing synchronized video and UWB CIR data, facilitating further analysis and processing.



DEMO VIDEO

REFERENCES



- An overview of UWB standards
- Camera API
- Camera 2 API
- TextureView
- Cameramanager
- Imagereader
- Mediarecorder

THANK YOU

At last, I would like to thank my RM (Reporting Manager) **Mr. Ankur Trisal** for guiding me throughout my Internship.