**Implementation Process Documentation:**
**Challenges Encountered and Solutions:**
*Dataset Labelling Issue:* The first challenge I experienced was selecting dataset, because the dataset I initially worked on I think that was ASVSpoof 2021 they by mistake in the TAR file didn't add the labels file, what they did add was a file that contains file names.
*Second challenge:* I was tasked with 3 models , I had to make sure all 3 models offer something different something unique , what models to use was not that big of a job
*Third challenge :* The third challenge was overfitting, in the pursuit of highly accurate first model, I made a deep model which I corrected afterwards after finding that the validation class is highly oversampled by real data, which I quickly overcame by sing SMOTE and focal loss.
*False Alarms***:** A tqdm progress bar glitch initially caused concern, showing 56% completion despite the model performing well. This emphasised the importance of not relying solely on progress indicators and thoroughly evaluating model performance.

**Feature Extraction Research:** So the research in feature extraction had many levels, according to my pursuit of 3 approached should be unique the feature selection also had to be unique, I decided I will make1st model super accurate , so the features I used was MFCC which is Mel coefficient which is widely used in audio but the second feature I used in collaboration with this was pitch, not many people use pitch but pitch was a really good idea as it gave me high accuracy .
*Second feature*: The second feature I used was for my model that I wanted to tweak intelligently from basic to super good and the feature I used for this is spectral centroid, because according to my research spectral centroid is very effective and easily distinguished.
*Third set* So the third feature I used had to be the fastest to process and according to my research that was ZCR.

**Model Selection:** I divided my model selection in 3 models,
1) *Supposed to be super accurate*: I used CONV2D with BiLSTM which is supposed to be super accurate and is.
2) *Intelligent:* The second model I used was supposed to be underdog for a task with best feature fed to it, so I used logistic regression as it is binary classification.
3) *Lightening fast* : The third model I used is supposed to be lite and fast so I used Lightweight sequential MLP and fed it the fastest processing feature ZCR.

**Assumptions:**
*The assumptions I made were less but I made one which is that , my model is best and I wont be needing to use synthetic sampling, which I did use training models.*
**Analysis:**

**Model Selection Rationale:**
*Why I used Model 1 : CONV2D with BiLSTM To capture Long range temporal dependencies, I could've used DTW too but I had to map it to some other model.*
*Why I used model 2 :* So for model 2 I decided that model should be the best for binary classification since the feature amusing here is best. So there is nothing best than logistic regression for binary classification.( I read a whole research paper where there used cooccurent matrix with K means as intelligent tweaking but unfortunately this dataset was not showing good results for that.)
*Why I used model 3 :* This model needed to be lite so I used a super simple sequential model
**Technical Explanation:**

1. **CNN-BiLSTM Model (MFCC + Pitch)**

*Features Used:*

MFCCs (Mel-Frequency Cepstral Coefficients): Encodes timbral/textural characteristics (e.g., voice quality, instrument type).

Pitch: Captures fundamental frequency (melody/harmony information).
*How It Works:*
*2D Convolutional Layers:Treat MFCCs and pitch as a "time-frequency image" (timesteps × 1×1 channels).Kernels of shape (5,1) scan temporal patterns (e.g., pitch transitions, MFCC dynamics).*

*Bidirectional LSTM*:Processes the CNN-extracted features in both forward/backward directions to model temporal context (e.g., rising pitch → question intonation).
Why This Combo?
CNN detects local acoustic patterns, while BiLSTM understands longer-term dependencies (e.g., a pitch curve over time).
Use Case: Best for capturing voice characteristics and intonation patterns.

**2. Logistic Regression (Spectral Centroid)**

Feature Used: Spectral Centroid: Measures the "brightness" of sound (higher centroid = more high-frequency energy).
How It Works:

Simple Linear Classifier: Learns a decision boundary like if centroid > threshold → class 1.
Interpretability:
Coefficients directly show how spectral centroid affects predictions (e.g., brighter sounds → more likely class 1).
Why This Model?

Spectral centroid is a single-value feature per frame → No need for complex models.
Provides a computationally cheap baseline to compare against neural networks.
Use Case: Detecting sound brightness differences (e.g., speech vs. cymbals).

3. **MLP (Zero-Crossing Rate - ZCR)**

Feature Used:
ZCR: Counts how often the audio waveform crosses zero (higher ZCR ≈ more noise/percussive sounds).

How It Works:
*Multilayer Perceptron (MLP):*
A simple feedforward neural network with hidden layers.
Learns nonlinear relationships like if ZCR > X AND ZCR < Y → class 0.
*Advantage Over Logistic Regression*:
Can capture complex thresholds (e.g., mid-range ZCR = class 1, extremes = class 0).
Why This Model?

ZCR is nonlinear in its discriminative power → MLP handles this better than logistic regression.
Use Case:

Distinguishing voiced sounds (low ZCR) vs. unvoiced/noise (high ZCR).

**Performance Results:**
*Model1 :* Accuracy 99.96%, Loss 0.0082 , Val_accuracy 99.88%

*Model 2 :* 70% (can be tuned further)

Model 3: *83% accuracy*

**Strengths and Weaknesses:**
**Strengths:** Efficient processing suitable for devices with limited resources, potential for real-time analysis.
**Weaknesses:** this whole program may have very little weakness, but individual model may have some because each is created for a unique purpose
**Future Improvements:**
This is a dataset which we can easily be drag to 100% accuracy with right computational power.
**Reflection:**
**Most Significant Challenges:**
Overcoming the initial dataset labelling issue highlighted the critical importance of data integrity in machine learning projects.
Balancing model complexity with hardware constraints required careful consideration of architecture choices.
**Real-World Performance Expectations:**
The model has high accuracy and is curated for different unique purpose and I think the models serve their purpose, the fast one is fast, the super accurate is super accurate, the intelligent one is intelligent.
**Additional Data or Resources for Improvement:**
The more the merrier applies to additional data and resources ask. But specifically resources, higher computational power is required if one has to reach 100% accuracy without false positives and false negatives, and higher feature engineering, but this is possible with the help of LLMs but the ask for computational power remains
**Production Deployment Approach:**
Implement a containerised solution for easy deployment and scaling.
Develop a robust monitoring system to track model performance and detect concept drift.
Establish a pipeline for continuous model updating and retraining as new data becomes available.