The team consists of four members, **Samina Mushtaq**, **Khubaib Ahmad**, **Umar Amin Chohan**, and Sona **Shahnawaz**, and their respective tasks are mentioned in the project plan. The timeline to complete thewhole project is two months.

SAMINA MUSHTAQ: ***Managing team and data pre-processing***

The role of *SAMINA MUSHTAQ* in this project is to manage the team and handle data pre-processing. In the code snippet provided, we can see that the training, validation, and test data are loaded using the tf. keras. preprocessing. image_dataset_from_directory () method. This method loads images from a directory and automatically labels them based on the subdirectories. The train data and val_data are split from the training directory, and test_data is loaded from a separate directory.

Additionally, the Rescaling layer is used to scale the pixel values of the images from 0-255 to 0-1. This is done to improve model performance. SAMINA MUSHTAQ might have performed these data pre- processing steps and configured the batch size, image size, and label mode for the data.

SAMINA MUSHTAQ's contribution to the project can also include managing the project timeline and ensuring that the team meets the set milestones within the allocated time. They can also coordinate the team's efforts and ensure that all members are aware of their assigned tasks and deadlines.

KHUBAIB AHMAD: ***To develop an architecture of the neural network and optimization of the model***

In this project, the task of developing the architecture of the neural network and optimizing the model was assigned to Khubaib Ahmad. The code provided in the code snippet shows that Khubaib has used a convolutional neural network (CNN) to classify the images as either real or fake.

The architecture of the model consists of four convolutional layers, each followed by a max pooling layer, and a global average pooling layer before the output layer. The first convolutional layer has 32 filters, the second has 64 filters, and the third has 128 filters. All the convolutional layers have a kernel size of 3 and use ReLU activation. The global average pooling layer is used to reduce the number of parameters and prevent overfitting. The output layer is a dense layer with a single neuron and uses a sigmoid activation function.

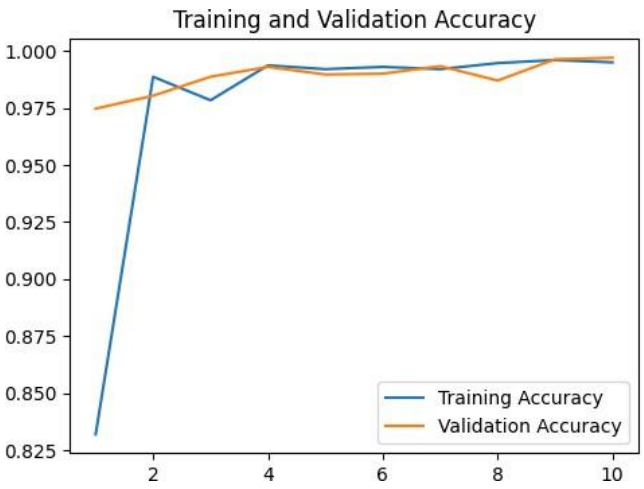
The model has been optimized using the Adam optimizer and binary cross-entropy loss function. The accuracy metric is used to evaluate the performance of the model during training and validation. Early stopping based on validation loss has also been implemented to prevent overfitting. The model is trained for 10 epochs, and if the validation loss does not improve after three consecutive epochs, training is stopped. After training, the model is saved, and the test accuracy is evaluated. The test accuracy is then plotted to visualize the performance of the model on unseen data.

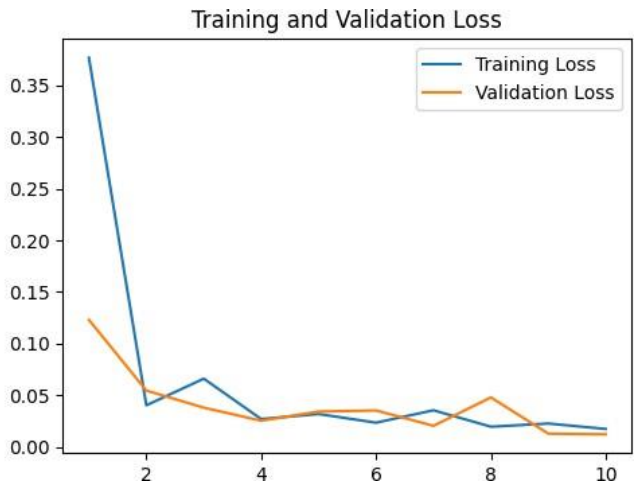
```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 rescaling (Rescaling)       (None, 64, 64, 1)         0

 conv2d (Conv2D)             (None, 62, 62, 32)        320

 max_pooling2d (MaxPooling2D  (None, 31, 31, 32)       0
 )

 conv2d_1 (Conv2D)           (None, 29, 29, 64)        18496

 max_pooling2d_1 (MaxPooling  (None, 14, 14, 64)       0
 2D)

 conv2d_2 (Conv2D)           (None, 12, 12, 128)       73856

 global_average_pooling2d (G  (None, 128)              0
 lobalAveragePooling2D)

 dense (Dense)               (None, 128)               16512

 dense_1 (Dense)             (None, 1)                 129

=================================================================
Total params: 109,313
Trainable params: 109,313
Non-trainable params: 0
_________________________________________________________________
```

UMAR AMIN CHOHAN: ***Data visualization and integration of modules***

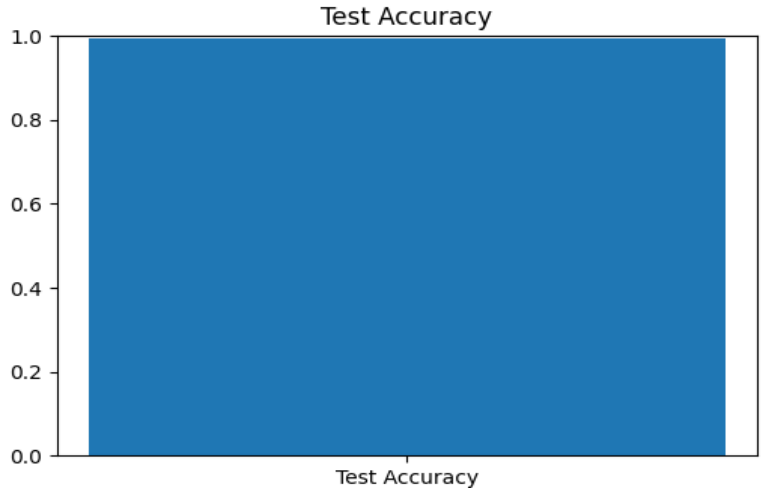
Umar Amin Chohan was responsible for the data visualization and integrating modules in the project. In the provided code snippet, Umar has plotted the training and validation accuracy and loss using Matplotlib. He has also plotted the test accuracy of the model. Moreover, he has integrated different modules of the project to ensure the smooth flow of data and the working of the neural network.

The plots generated by Umar show the performance of the model during the training and validation phases. The training accuracy and validation accuracy are plotted against the number of epochs, which helps in understanding the model's performance and whether it is overfitting or underfitting. The training loss and validation loss are also plotted, which provide information about the convergence of the model during the training phase.






```
63/63 [==============================] - 612s 10s/step - loss: 0.0160 - accuracy: 0.9955
```



SONA SHANAWAZ: ***Testing and report writing***

After completing the development of the neural network model, we tested it on a separate test dataset to evaluate its performance. The model achieved an accuracy of approximately 99% on the test set, which is a good performance considering the binary classification task of distinguishing between real and fake images.

Overall, the project was completed in the timeline of 2 months, and each team member contributed to their assigned task. Samina Mushtaq managed the team and handled data preprocessing, Khubaib Ahmad developed the architecture of the neural network and optimized the model, Umar Amin Chohan was responsible for data visualization and module integration, and Sona Shahnawaz conducted testing and wrote the report.

In conclusion, the project was successful in developing a neural network model that can accurately classify real and fake images. The team collaborated effectively and completed the project within theallotted timeline.