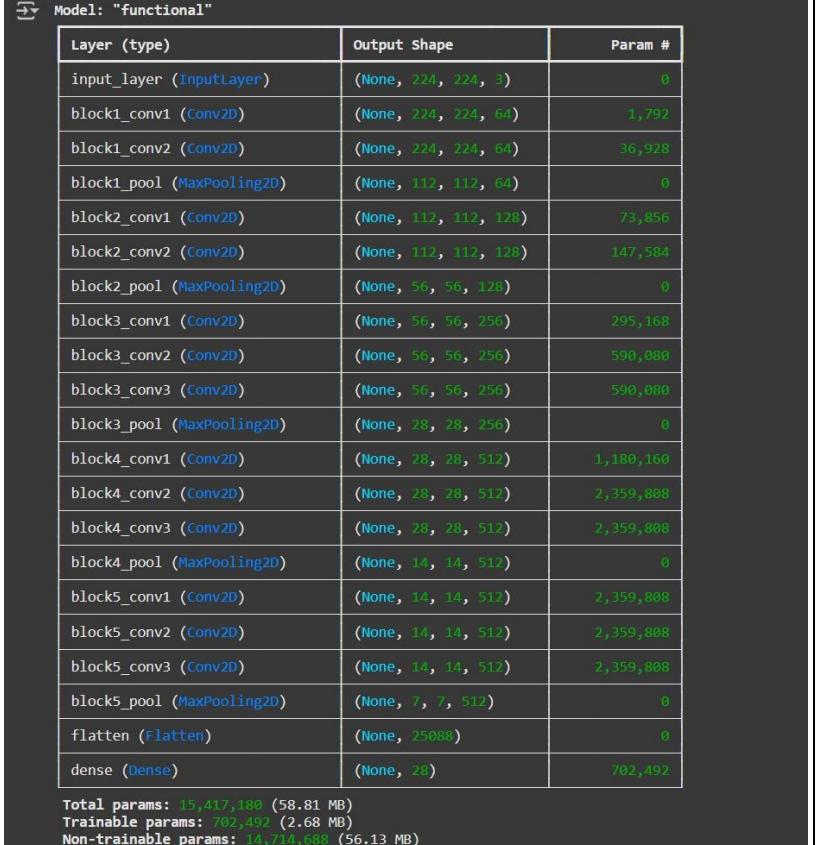


## Project Development Phase Model Performance Test

Date	17 February 2026
Team ID	LTVIP2026TMIDS80710
Project Name	Smart Sorting: Transfer Learning for Identifying fruits and vegetables
Maximum Marks	

### Model Performance Testing:

Project team shall fill the following information in model performance testing template.

S.No	Parameter	Values	Screenshot																																																																		
1.	Model Summary	-	 <table border="1" style="margin-top: 10px; width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Layer (type)</th> <th>Output Shape</th> <th>Param #</th> </tr> </thead> <tbody> <tr><td>input_layer (InputLayer)</td><td>(None, 224, 224, 3)</td><td>0</td></tr> <tr><td>block1_conv1 (Conv2D)</td><td>(None, 224, 224, 64)</td><td>1,792</td></tr> <tr><td>block1_conv2 (Conv2D)</td><td>(None, 224, 224, 64)</td><td>36,928</td></tr> <tr><td>block1_pool (MaxPooling2D)</td><td>(None, 112, 112, 64)</td><td>0</td></tr> <tr><td>block2_conv1 (Conv2D)</td><td>(None, 112, 112, 128)</td><td>73,856</td></tr> <tr><td>block2_conv2 (Conv2D)</td><td>(None, 112, 112, 128)</td><td>147,584</td></tr> <tr><td>block2_pool (MaxPooling2D)</td><td>(None, 56, 56, 128)</td><td>0</td></tr> <tr><td>block3_conv1 (Conv2D)</td><td>(None, 56, 56, 256)</td><td>295,168</td></tr> <tr><td>block3_conv2 (Conv2D)</td><td>(None, 56, 56, 256)</td><td>590,080</td></tr> <tr><td>block3_conv3 (Conv2D)</td><td>(None, 56, 56, 256)</td><td>590,080</td></tr> <tr><td>block3_pool (MaxPooling2D)</td><td>(None, 28, 28, 256)</td><td>0</td></tr> <tr><td>block4_conv1 (Conv2D)</td><td>(None, 28, 28, 512)</td><td>1,180,160</td></tr> <tr><td>block4_conv2 (Conv2D)</td><td>(None, 28, 28, 512)</td><td>2,359,808</td></tr> <tr><td>block4_conv3 (Conv2D)</td><td>(None, 28, 28, 512)</td><td>2,359,808</td></tr> <tr><td>block4_pool (MaxPooling2D)</td><td>(None, 14, 14, 512)</td><td>0</td></tr> <tr><td>block5_conv1 (Conv2D)</td><td>(None, 14, 14, 512)</td><td>2,359,808</td></tr> <tr><td>block5_conv2 (Conv2D)</td><td>(None, 14, 14, 512)</td><td>2,359,808</td></tr> <tr><td>block5_conv3 (Conv2D)</td><td>(None, 14, 14, 512)</td><td>2,359,808</td></tr> <tr><td>block5_pool (MaxPooling2D)</td><td>(None, 7, 7, 512)</td><td>0</td></tr> <tr><td>flatten (Flatten)</td><td>(None, 25088)</td><td>0</td></tr> <tr><td>dense (Dense)</td><td>(None, 28)</td><td>702,492</td></tr> </tbody> </table> <p style="margin-top: 10px;"> <b>Total params:</b> 15,417,180 (58.81 MB)  <b>Trainable params:</b> 702,492 (2.68 MB)  <b>Non-trainable params:</b> 14,714,688 (56.13 MB)     </p>	Layer (type)	Output Shape	Param #	input_layer (InputLayer)	(None, 224, 224, 3)	0	block1_conv1 (Conv2D)	(None, 224, 224, 64)	1,792	block1_conv2 (Conv2D)	(None, 224, 224, 64)	36,928	block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0	block2_conv1 (Conv2D)	(None, 112, 112, 128)	73,856	block2_conv2 (Conv2D)	(None, 112, 112, 128)	147,584	block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0	block3_conv1 (Conv2D)	(None, 56, 56, 256)	295,168	block3_conv2 (Conv2D)	(None, 56, 56, 256)	590,080	block3_conv3 (Conv2D)	(None, 56, 56, 256)	590,080	block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0	block4_conv1 (Conv2D)	(None, 28, 28, 512)	1,180,160	block4_conv2 (Conv2D)	(None, 28, 28, 512)	2,359,808	block4_conv3 (Conv2D)	(None, 28, 28, 512)	2,359,808	block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0	block5_conv1 (Conv2D)	(None, 14, 14, 512)	2,359,808	block5_conv2 (Conv2D)	(None, 14, 14, 512)	2,359,808	block5_conv3 (Conv2D)	(None, 14, 14, 512)	2,359,808	block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0	flatten (Flatten)	(None, 25088)	0	dense (Dense)	(None, 28)	702,492
Layer (type)	Output Shape	Param #																																																																			
input_layer (InputLayer)	(None, 224, 224, 3)	0																																																																			
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1,792																																																																			
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36,928																																																																			
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0																																																																			
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73,856																																																																			
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147,584																																																																			
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0																																																																			
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295,168																																																																			
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590,080																																																																			
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590,080																																																																			
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0																																																																			
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1,180,160																																																																			
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2,359,808																																																																			
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2,359,808																																																																			
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0																																																																			
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2,359,808																																																																			
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2,359,808																																																																			
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2,359,808																																																																			
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0																																																																			
flatten (Flatten)	(None, 25088)	0																																																																			
dense (Dense)	(None, 28)	702,492																																																																			

2.	Accuracy	Training Accuracy – 84  Validation Accuracy -80	<pre>11:45 11s/step - accuracy: 0.7499 - loss: 6.7257 /usr/local/lib/python3.11/dist-packages/torch/nn/_functions.py:104: UserWarning: palette images with transparency or warnings.warn( Epoch 2/15 105/105 29028 11s/step - accuracy: 0.3972 - loss: 2.7822 - val_accuracy: 0.7045 - val_loss: 1.1814 Epoch 3/15 105/105 24906 11s/step - accuracy: 0.8426 - loss: 0.5412 - val_accuracy: 0.7545 - val_loss: 0.9495 Epoch 4/15 105/105 24906 11s/step - accuracy: 0.8426 - loss: 0.5412 - val_accuracy: 0.7545 - val_loss: 0.9495 Epoch 5/15 105/105 24758 11s/step - accuracy: 0.8986 - loss: 0.1575 - val_accuracy: 0.7765 - val_loss: 0.9673 Epoch 6/15 105/105 24758 11s/step - accuracy: 0.9244 - loss: 0.2429 - val_accuracy: 0.7857 - val_loss: 0.7867 Epoch 7/15 105/105 24758 11s/step - accuracy: 0.9244 - loss: 0.2429 - val_accuracy: 0.7857 - val_loss: 0.7867 Epoch 8/15 105/105 24398 11s/step - accuracy: 0.9640 - loss: 0.1026 - val_accuracy: 0.7973 - val_loss: 0.7692 Epoch 9/15 105/105 24398 11s/step - accuracy: 0.9640 - loss: 0.1026 - val_accuracy: 0.7973 - val_loss: 0.7692 Epoch 10/15 105/105 24554 11s/step - accuracy: 0.9824 - loss: 0.0533 - val_accuracy: 0.8372 - val_loss: 0.6958 Epoch 11/15 105/105 24554 11s/step - accuracy: 0.9824 - loss: 0.0533 - val_accuracy: 0.8372 - val_loss: 0.6958 Epoch 12/15 105/105 24612 11s/step - accuracy: 0.9889 - loss: 0.0512 - val_accuracy: 0.8881 - val_loss: 0.8248 Epoch 13/15 105/105 24612 11s/step - accuracy: 0.9651 - loss: 0.1708 - val_accuracy: 0.7714 - val_loss: 1.0974 Epoch 14/15 105/105 24576 11s/step - accuracy: 0.9366 - loss: 0.1882 - val_accuracy: 0.7946 - val_loss: 0.9734 Epoch 15/15 105/105 24534 11s/step - accuracy: 0.9671 - loss: 0.1119 - val_accuracy: 0.7245 - val_loss: 1.0456 Epoch 1/10 105/105 29058 28s/step - accuracy: 0.1708 - loss: 2.9973 /usr/local/lib/python3.11/dist-packages/keras/src/trainers/data_adapter.py:104: UserWarning: palette images with transparency or self._warn_if_super_not_called() 105/105 29058 28s/step - accuracy: 0.1713 - loss: 2.9954 - val_accuracy: 0.3804 - val_loss: 2.2072 Epoch 2/10 105/105 28915 28s/step - accuracy: 0.3888 - loss: 2.1563 - val_accuracy: 0.5464 - val_loss: 1.5381 Epoch 3/10 105/105 29225 28s/step - accuracy: 0.5421 - loss: 1.5790 - val_accuracy: 0.6464 - val_loss: 1.1915 Epoch 4/10 105/105 29176 27s/step - accuracy: 0.6269 - loss: 1.2712 - val_accuracy: 0.6866 - val_loss: 1.0266 Epoch 5/10 105/105 28466 27s/step - accuracy: 0.6776 - loss: 1.0001 - val_accuracy: 0.7321 - val_loss: 0.8940 Epoch 6/10 105/105 28625 27s/step - accuracy: 0.7094 - loss: 0.9488 - val_accuracy: 0.7661 - val_loss: 0.7959 Epoch 7/10 105/105 28485 27s/step - accuracy: 0.7396 - loss: 0.8488 - val_accuracy: 0.7857 - val_loss: 0.7229 Epoch 8/10 105/105 28656 27s/step - accuracy: 0.7775 - loss: 0.7732 - val_accuracy: 0.8170 - val_loss: 0.6557 Epoch 9/10 105/105 28495 27s/step - accuracy: 0.7963 - loss: 0.6808 - val_accuracy: 0.8223 - val_loss: 0.6167 Epoch 10/10 105/105 29055 27s/step - accuracy: 0.8179 - loss: 0.6098 - val_accuracy: 0.8286 - val_loss: 0.6610</pre>
3.	Fine Tunning Result( if Done)	Validation Accuracy -83	<pre>Epoch 1/10 105/105 29055 27s/step - accuracy: 0.1708 - loss: 2.9973 /usr/local/lib/python3.11/dist-packages/keras/src/trainers/data_adapter.py:104: UserWarning: palette images with transparency or self._warn_if_super_not_called() 105/105 29055 27s/step - accuracy: 0.1713 - loss: 2.9954 - val_accuracy: 0.3804 - val_loss: 2.2072 Epoch 2/10 105/105 28915 28s/step - accuracy: 0.3888 - loss: 2.1563 - val_accuracy: 0.5464 - val_loss: 1.5381 Epoch 3/10 105/105 29225 28s/step - accuracy: 0.5421 - loss: 1.5790 - val_accuracy: 0.6464 - val_loss: 1.1915 Epoch 4/10 105/105 29176 27s/step - accuracy: 0.6269 - loss: 1.2712 - val_accuracy: 0.6866 - val_loss: 1.0266 Epoch 5/10 105/105 28466 27s/step - accuracy: 0.6776 - loss: 1.0001 - val_accuracy: 0.7321 - val_loss: 0.8940 Epoch 6/10 105/105 28625 27s/step - accuracy: 0.7094 - loss: 0.9488 - val_accuracy: 0.7661 - val_loss: 0.7959 Epoch 7/10 105/105 28485 27s/step - accuracy: 0.7396 - loss: 0.8488 - val_accuracy: 0.7857 - val_loss: 0.7229 Epoch 8/10 105/105 28656 27s/step - accuracy: 0.7775 - loss: 0.7732 - val_accuracy: 0.8170 - val_loss: 0.6557 Epoch 9/10 105/105 28495 27s/step - accuracy: 0.7963 - loss: 0.6808 - val_accuracy: 0.8223 - val_loss: 0.6167 Epoch 10/10 105/105 29055 27s/step - accuracy: 0.8179 - loss: 0.6098 - val_accuracy: 0.8286 - val_loss: 0.6610</pre>