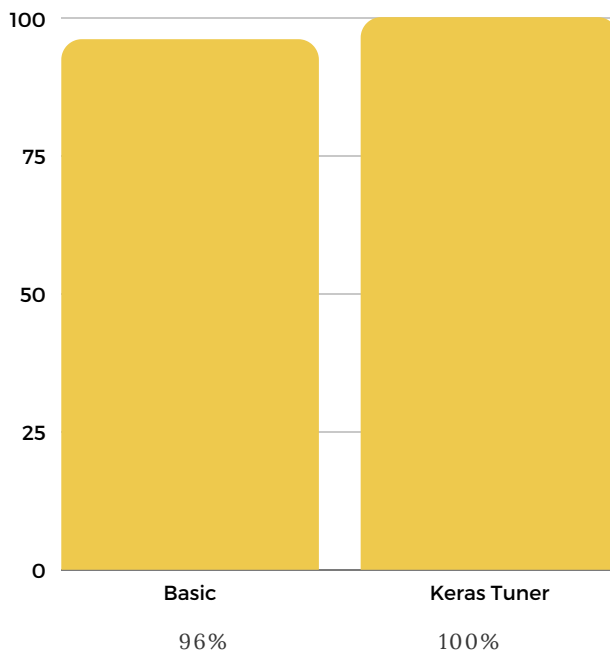


Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 62, 62, 64)	1792
max_pooling2d (MaxPooling2D)	(None, 31, 31, 64)	0
conv2d_1 (Conv2D)	(None, 29, 29, 128)	73856
max_pooling2d_1 (MaxPooling2D)	(None, 14, 14, 128)	0
dropout (Dropout)	(None, 14, 14, 128)	0
conv2d_2 (Conv2D)	(None, 12, 12, 64)	73792
max_pooling2d_2 (MaxPooling2D)	(None, 6, 6, 64)	0
flatten (Flatten)	(None, 2304)	0
dropout_1 (Dropout)	(None, 2304)	0
dense (Dense)	(None, 7)	16135

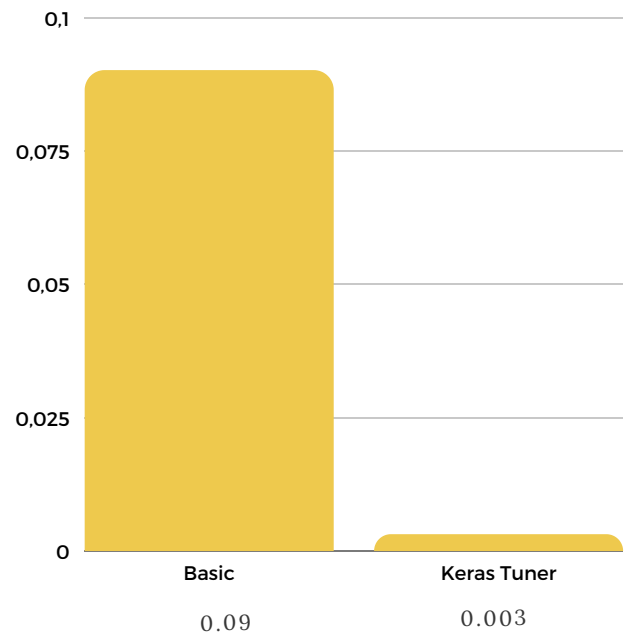
Total params: 165,575
 Trainable params: 165,575
 Non-trainable params: 0

BASIC MODEL**ACCURACY**

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 62, 62, 16)	448
max_pooling2d (MaxPooling2D)	(None, 31, 31, 16)	0
conv2d_1 (Conv2D)	(None, 29, 29, 32)	4640
max_pooling2d_1 (MaxPooling2D)	(None, 14, 14, 32)	0
dropout (Dropout)	(None, 14, 14, 32)	0
conv2d_2 (Conv2D)	(None, 12, 12, 64)	18496
max_pooling2d_2 (MaxPooling2D)	(None, 6, 6, 64)	0
flatten (Flatten)	(None, 2304)	0
dropout_1 (Dropout)	(None, 2304)	0
dense (Dense)	(None, 7)	16135

Total params: 39,719
 Trainable params: 39,719
 Non-trainable params: 0

KERAS TUNER MODEL**LOSS**

With the data obtained, we can observe that thanks to using Keras Tuner we considerably improve the results of our classifier, at the cost of spending time doing the grid search to get the optimal parameters.