Nutrify – Your Healthy Eating App

Algorithm

DRIVER FUNCTION ALGORITHM :

```
FUNCTION mainDriver()
   // Step 1: Capture Image
   CALL captureImage(source, isFrontImage, setState)

   // Step 2: AI Food Recognition
   detectedFood ← classifyFood(_frontImage)
   PRINT "Detected Food: " + detectedFood

   // Step 3: Save to Database
   CALL updateTotalNutrients()

   // Step 4: Receive Nutrition Information
   calories ← getCalories()
   PRINT "Calories: " + calories

   // Step 5: Provide Insights
   insights ← getInsights(dailyIntake)
   IF insights IS NOT NULL THEN
      PRINT insights
   END IF
END FUNCTION
```

Algorithm 1(Take Picture):

```
FUNCTION captureImage(source, isFrontImage, setState)
   CREATE imagePicker INSTANCE OF ImagePicker
   ASSIGN image TO imagePicker.pickImage(source)
   IF image IS NOT NULL THEN
      IF isFrontImage IS TRUE THEN
         SET _frontImage TO new File(image.path)
      ELSE
         SET _nutritionLabelImage TO new File(image.path)
      ENDIF
      CALL setState()
   ENDIF
END FUNCTION
```

Algorithm 2 (AI Food Recognition):

```
FUNCTION classifyFood(imageFile)
```

```
    IF model NOT loaded THEN RETURN "Error: Model not loaded"

    image ← Decode and resize image to 224x224
    input ← Normalize pixel values (0-1) into tensor format

    output ← Run model on input
    maxIndex ← Index of highest confidence score in output

    RETURN _labels[maxIndex]  // Return detected food name
END FUNCTION
```

Algorithm 3 (Save to Databases):

```
FUNCTION updateTotalNutrients()
    SET totalPlateNutrients TO {
        'calories' ← 0.0,
        'protein' ← 0.0,
        'carbohydrates' ← 0.0,
        'fat' ← 0.0,
        'fiber' ← 0.0
    }
END FUNCTION
```

Algorithm 4(Receive Nutrition Information) :

```
FUNCTION getCalories()
    energyNutrient ← FIND first item in parsedNutrients WHERE name = "Energy"
    IF energyNutrient NOT FOUND THEN SET energyNutrient.quantity TO "0.0"

    RETURN Convert energyNutrient.quantity TO number
END FUNCTION
```

Algorithm 5 (Provide Insights):

```
FUNCTION getInsights(dailyIntake)
    FOR each nutrient IN nutrientData DO
        IF nutrient["Nutrient"] IN dailyIntake THEN
            dvValue ← Convert nutrient["Current Daily Value"] TO number
            IF dailyIntake[nutrient["Nutrient"]] / dvValue > 1 THEN
                RETURN "Exceeded daily intake of " + nutrient["Nutrient"
    END FOR
END FUNCTION
```