**Program Exercise 2 Technical Design Document**

**Name:** Rosalynn Alejandro

**Date Created:** June 8, 2025

**Program Description:**

This program will scan a users email for common words/phrases found in spam email, and score how many times a word or phrase was found, display the likelihood that it is spam, and also display the words/phrases found in the email.

**Functions used in the Program (list in order as they are called):**

1. **Function Name:** main()

**Description:** This function will ask the user for input and display the spam\_score, spam likelihood, and the matched\_phrases

**Parameters:** none

**Variables:**

1. message, this is the users email input
2. matched\_phrases, this is the list of phrases found in the users email
3. spam\_score, this variable is used to tally up the spam words and phrases found in the users email

**Logical Steps:**

1. Initialize the variables
2. While true statement so program will rerun
   1. Ask user for input
   2. Call the scan() function
   3. Display spam\_score
   4. Display spam likelihood
   5. Display the matched phrases
   6. Ask if the user wants to scan another email

**Returns:** none

2. **Function Name:** scan()

**Description:** This function scans the users email for common words and phrases, tallies up the spam\_score, and creates a list of matched\_phrases

**Parameters:** message, spam\_score, matched\_phrases

**Variables:**

1. spam\_score, used to tally up the spam words and phrases found in the users email
2. matched\_phrases, used to create a new list of matched words and phrases from the phrases list

**Logical Steps:**

1. This function contains a list of 30 words and phrases commonly used in spam emails.
2. Uses a for loop for number if items in phrases (which is 30) to scan the users email for each item in the phrases list.
3. For each word/phrases found, 1 is added to the spam\_score
4. Appends a new list of matched\_phrases that the main function can display to the user

**Returns:** spam\_score, matched\_phrases

**Logical Steps:**

1. Calls a main() function
   1. Get user email as input to scan for common words and phrases
   2. Call a scan() function to calculate the spam\_score and create a list of found common words and phrases in the email
   3. Display the spam\_score, spam likelihood, and the list of found phrases in the email
   4. Ask user if they would like to scan another email
2. Use a scan() function to calculate
   1. Contains a list of 30 common words and phrases
   2. Add 1 to spam\_score for each word and phrase found in the users email
   3. Add each word or phrase found in the users email from the phrases list in a new list, matched\_phrases

**Link to your repository:** <https://github.com/ficklative/COP2373>

**Output Screenshot: (make sure big enough so I can see)**

A screenshot of a computer program

AI-generated content may be incorrect.