Name: Gaurav Patil

Reducing Screen complexity of current GUI

❖ Title:

GUI for screen complexity

❖ Problem Statement:

To Redesign Existing Graphical User Interface with screen complexity

❖ Learning Objective:

1) To study principles of good screen design
2) To apply the screen complexity rules to a GUI to improvise it.
3) To analyse the human considerations in Interface and screen design.

❖ Learning Outcomes:

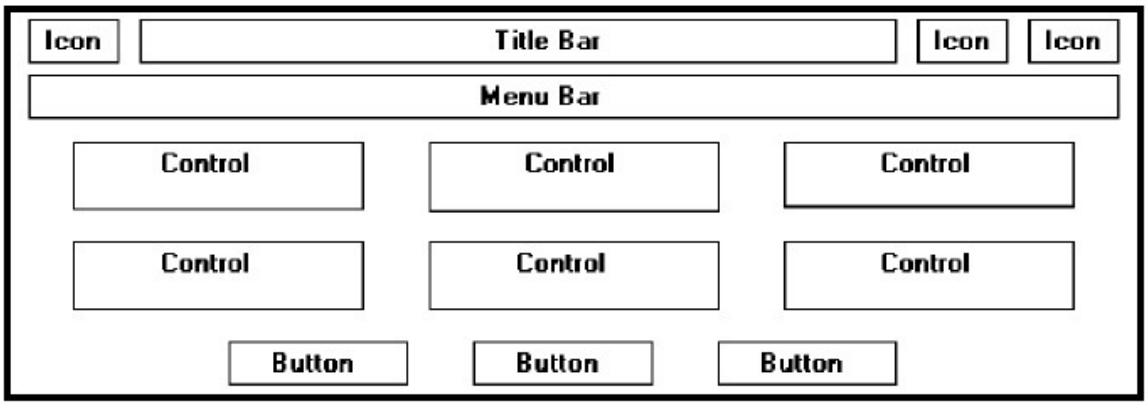
1) Design better screens in interfaces based on visually pleasing structure
2) Learn to organize the elements on an interface screen by properly calculating the screen complexity.
3) Learning the factors that affect the screen design quality with respect to user expectations

❖ Software and Hardware Requirement:

Any GUI screen from a selected application

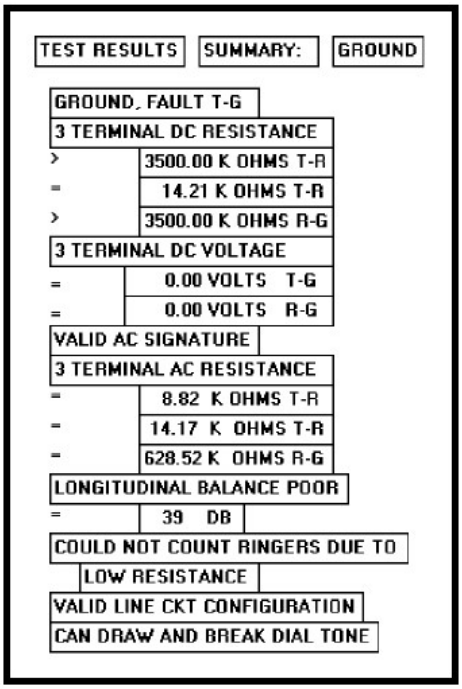
❖ Theory:

Any GUI screen from a selected application.
General structure of the elements on the screen to measure complexity factors.



To calculate the complexity first determine the following:
(1) the number of elements on the screen
(2) the number of horizontal (column) alignment points

(3) the number of vertical (row) alignment points
(4) Total:
(5) Complexity:
An example is given below:



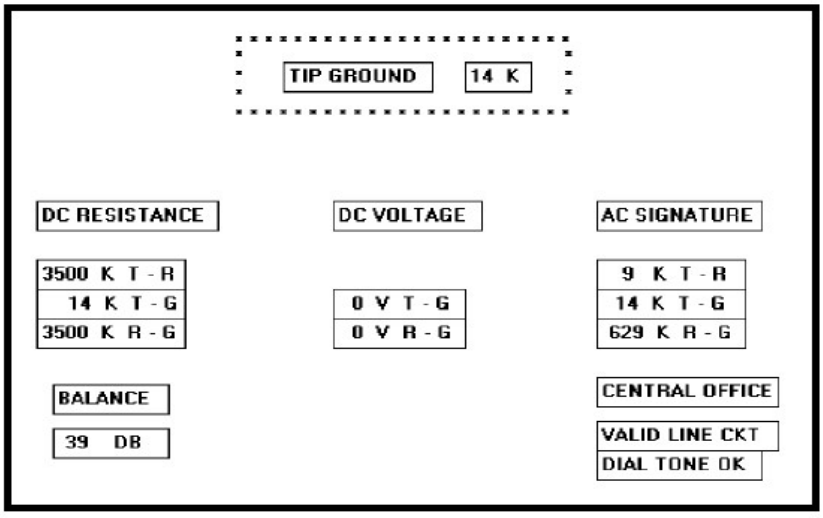
Original Design of the GUI

In the above screen the elements are not placed in a proper symmetry, which creates user confusion and loss of interest in the interface.

✓ The first requirement is to identify the text boxes and their places on the screen and then place them in a proper order , also group them as per requirement.

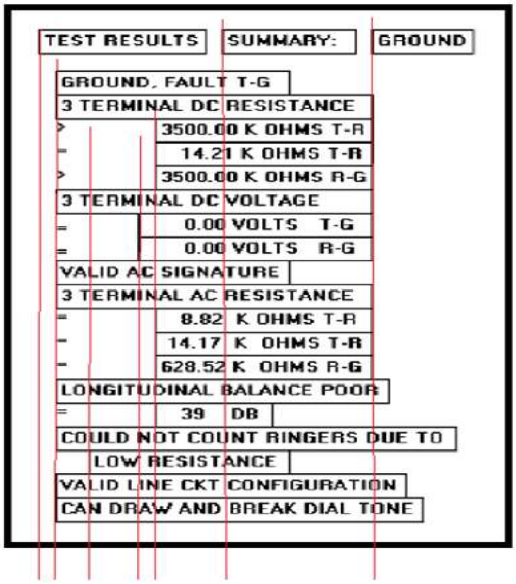
✓ The re-designed screen for the above example is shown in the figure below.

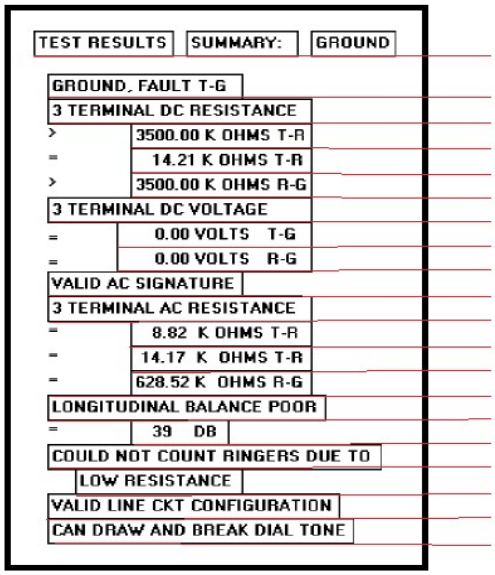
✓ To validate the improved screen, complexity of the screen is calculated which shows the optimization of screen space as well as the user friendly interface.



Re-designed Screen design

Calculation of complexity:





Calculate Screen Complexity:

Original Design

1. 22 elements
2. 7 horizontal (column) alignment points
3. 20 vertical (row) alignment points
4. 49 = Total

V_Complexity = $-N \sum_{n=1}^{m} p_n \log_2 p_n$ **;**
where p_n**= (Vertical Element Cut / Total)**

H_Complexity = $-N \sum_{n=1}^{m} p_n \log_2 p_n$ **;**
where p_n**= (Horizontal point Element Cut / Total)**

Total Complexity = V_Complexity + H_Complexity

Redesigned Screen

1. 18 elements
2. 6 horizontal (column) alignment points
3. 8 vertical (row) alignment points
4. 32 = Total

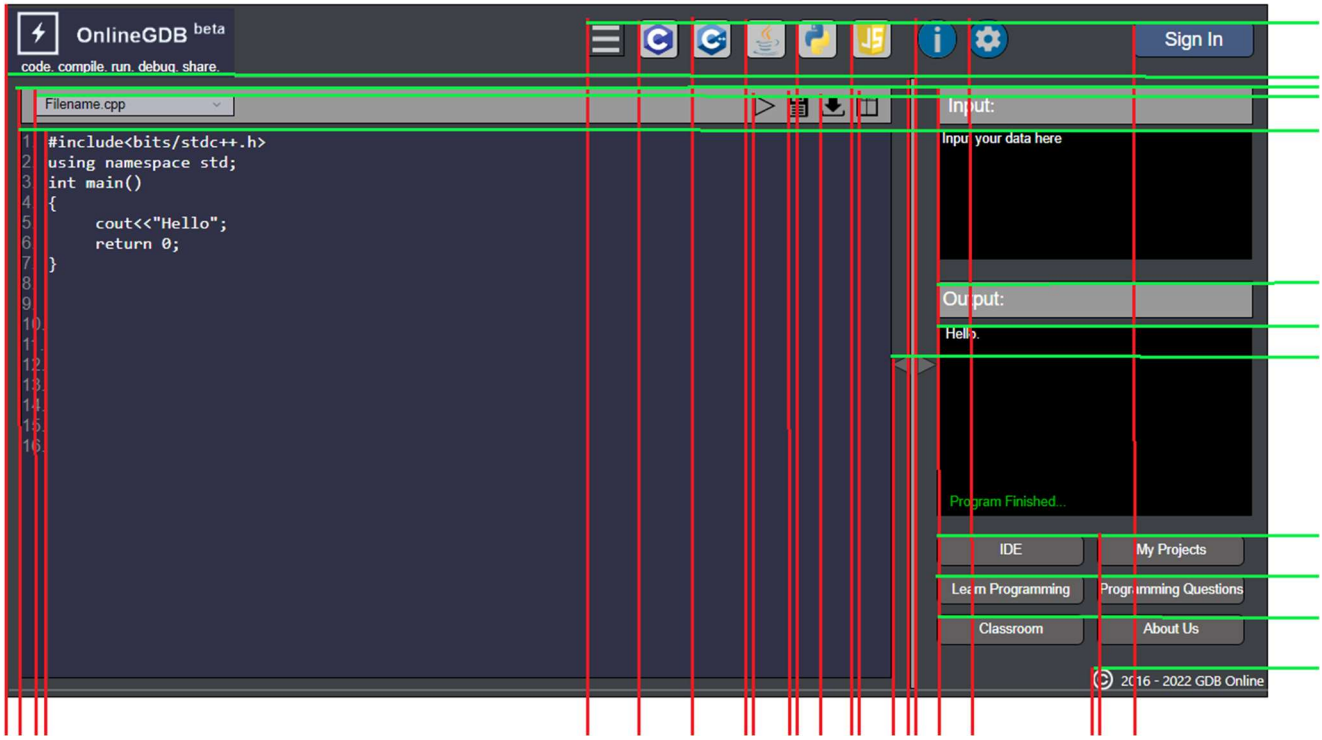
V_Complexity = $-N \sum_{n=1}^{m} p_n \log_2 p_n$ **;**
where p_n**= (Vertical Element Cut / Total)**

H_Complexity = $-N \sum_{n=1}^{m} p_n \log_2 p_n$ **;**
where p_n**= (Horizontal point Element Cut / Total)**

Total Complexity = V_Complexity + H_Complexity

❖ Output

Original Design



1. 32 elements
2. 12 horizontal (column) alignment points
3. 22 vertical (row) alignment points
4. 66 = Total

V_Complexity = $-N \sum_{n=1}^{m} p_n \log_2 p_n$ **;**
where p_n**= (Vertical Element Cut / Total)**

H_Complexity = $-N \sum_{n=1}^{m} p_n \log_2 p_n$ **;**
where p_n**= (Horizontal point Element Cut / Total)**

Total Complexity = V_Complexity + H_Complexity

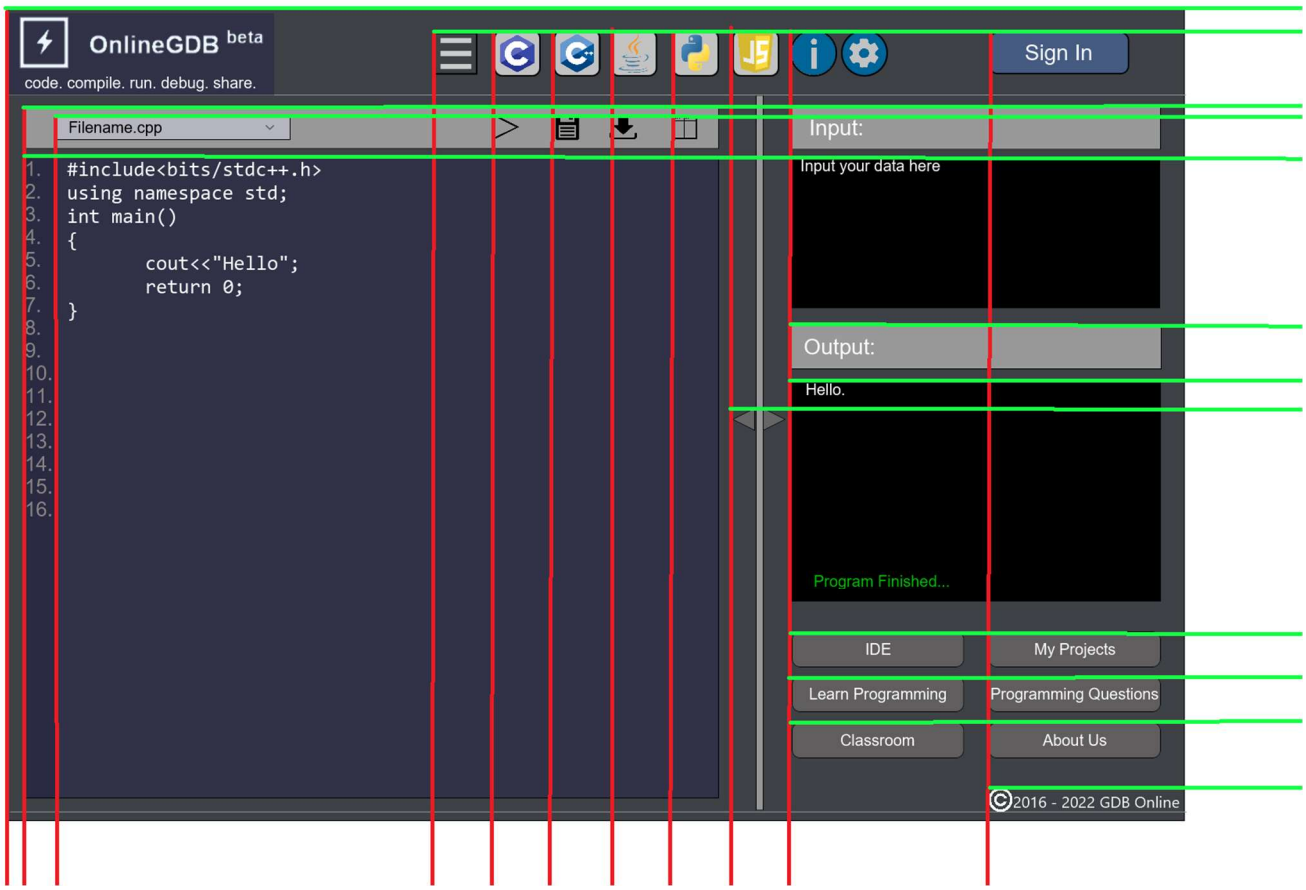
After calculating,

V_Complexity = 16.90626667
H_Complexity = 14.30942033

So the total complexity is

Total Complexity = 31.215687

Redesigned Screen



1. 32 elements
2. 11 horizontal (column) alignment points
3. 12 vertical (row) alignment points
4. 55 = Total

V_Complexity = $-N \sum_{n=1}^{m} p_n \log_2 p_n$ **;**
where p_n**= (Vertical Element Cut / Total)**

H_Complexity = $-N \sum_{n=1}^{m} p_n \log_2 p_n$ **;**
where p_n**= (Horizontal point Element Cut / Total)**

Total Complexity = V_Complexity + H_Complexity

After calculating,

V_Complexity = 15.33484599
H_Complexity = 14.86033981

So the total complexity is

Total Complexity = 30.1951858

Difference between original and redesigned UI,

Total Difference = 31.215687 – 30.1951858 = 1.0205012

❖ Conclusion:

Hence, we are able to improve design by using the complexity analysis of a GUI we created in assignment 2 by 1.0205012 bit using the principal of perception.