FINAL PROJECT—HANG MAN GAME

**-Console**

INPUT

Open file to select the word

User input a character. (click push button)

Validate a character A-Z(needed??-> using pushbutton)

PROCESS

Match the word whether the character is in the word or not

If the character is in the word, replace the blank into the character

If there is no character, leave the position as blank.

1. Main\_menu
2. Fruits\*
3. Animals\*\*
4. Movie Characters\*\*\*
5. Movie Names\*\*\*\*
6. Randomly select a word
7. findPosition
8. input one character (->validation needed??)
9. replace
10. hint
11. newline function
12. check finish
13. sortNcount

void main\_menu( int& option)

{

Display : 1. Fruits\*

2. Animals\*\*

3. Movie\*\*\*\*

4. Movie Characters\*\*\*

Ask user to input the option

Cin>>option

(validate option is 1-4)

}

String RandomWord(int option, ifstream& instream)

{

// num = (rand() % 50-70 +1);

Switch(option)

{

case 1: open instream file “Fruits”

num = (rand() % 70 +1);

for( int I = 0 ; i<num ; I ++)

newline();

cin>>word;

break;

case2 : open Vegetables file

do same

case 3,4 same as case 1 (open character, and movie name file instead of fruits file.

}

Return words.

}

FindPosition ( int start, int index, string word, char letter)

{

For(int I = start ; i< word.length() ; i++)

{

Index = Word.find(letter);

}

If( index>=0 )

Return index;

Else

Return -1;

}

void input\_letter(char& letter, char blankArr[], char wordarray[], int wordcount[],int used, int size);

{

Cin>>letter;

Find the letter in the word ( use function find position)

If there are letters in the word Wordcount-- until wordcount=0;

If letter is ‘?’

* hint function

if the letter already used

display : the word is already used.

}

void replace (int index, int start, char blankArr[], string word, int& count)

{

if( index != -1)

blankArr[index] = word[index];

else if ( index == -1 && start == 0) //if start != 0, this means hint use letter

{

Display: no matching letter exist in the word

count ++;

}

}

void sortNcount (string word, int wordcount[], char wordArray[], int& used)

{

Make the array which alphabets are in the word with the amount of letter.

}

Void Newline (istream&)

{

New line

}

OUTPUT

Output the letter in the position whenever the letter is in the word.

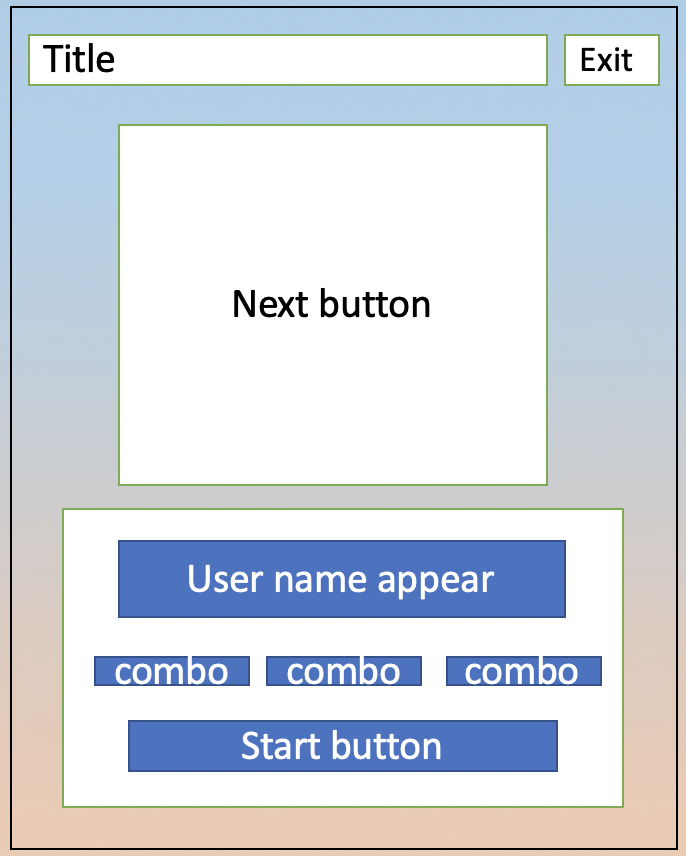
\_\_ \_\_ \_\_ A \_\_ \_\_ A

Until the blank is full or if the turn is greater than 7 <Username> die ;(

**-Qt Gui**

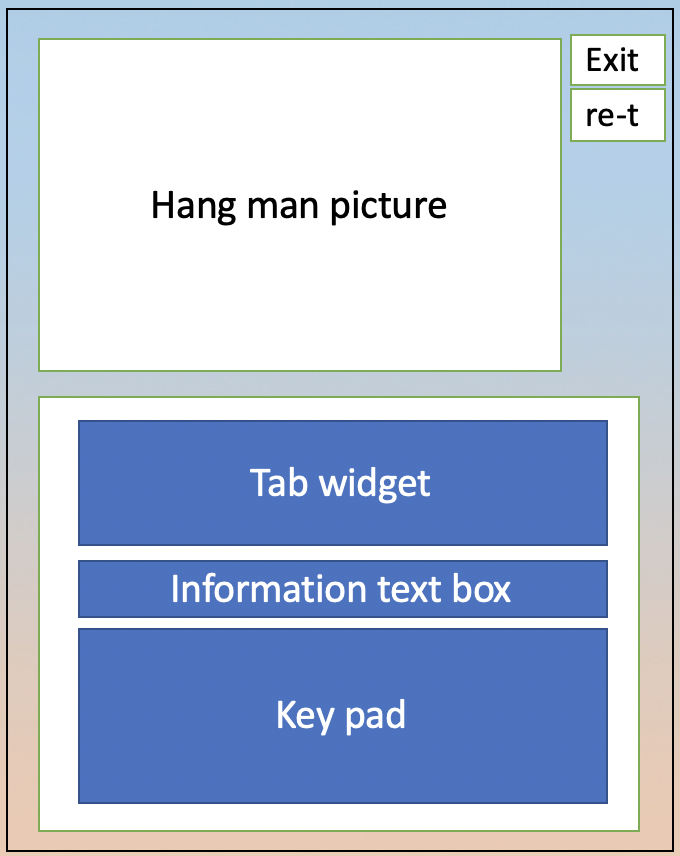
Page Design

Page 1 –



First page generate username with their birthday. ->go to next

Page 2-



Second page is actual hangman page.

Input

-User push the button – categories, key pad, hint, exit or restart

Process

-validate the user input and replace the letter and # in the tab widget.

Output

-Output the letter if the letter is in the word & “Correct” in the information text box

-It there is no letter in the word, output “No matching word” in the information text box.