

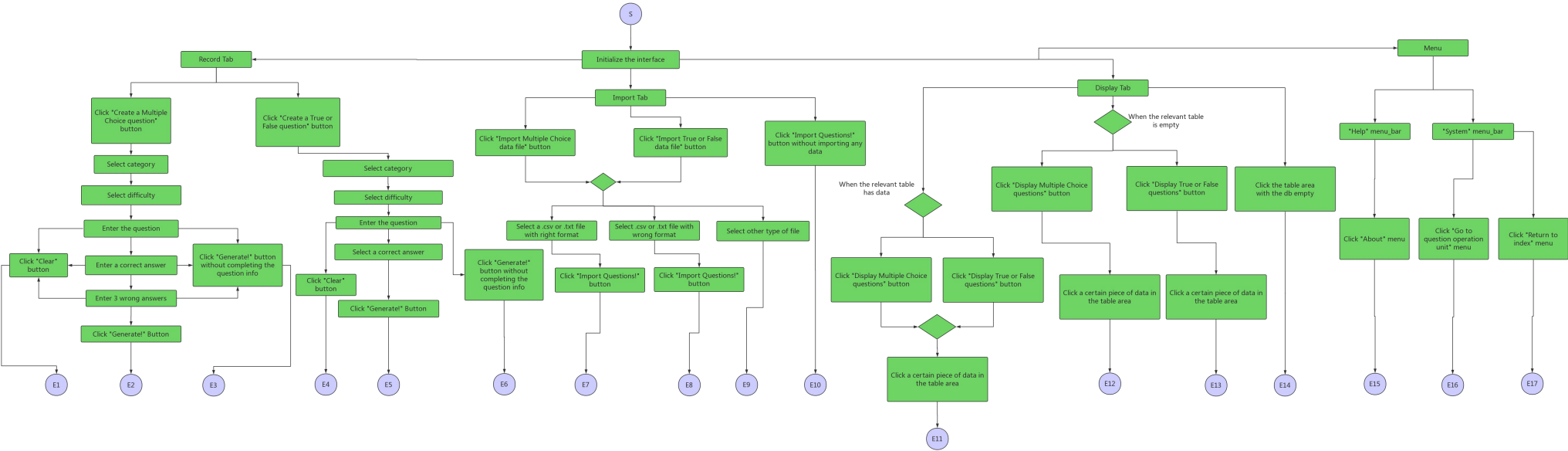
Team 8 - Portfolio B

Unit 1 - Add Question

Flow Control and Test Case

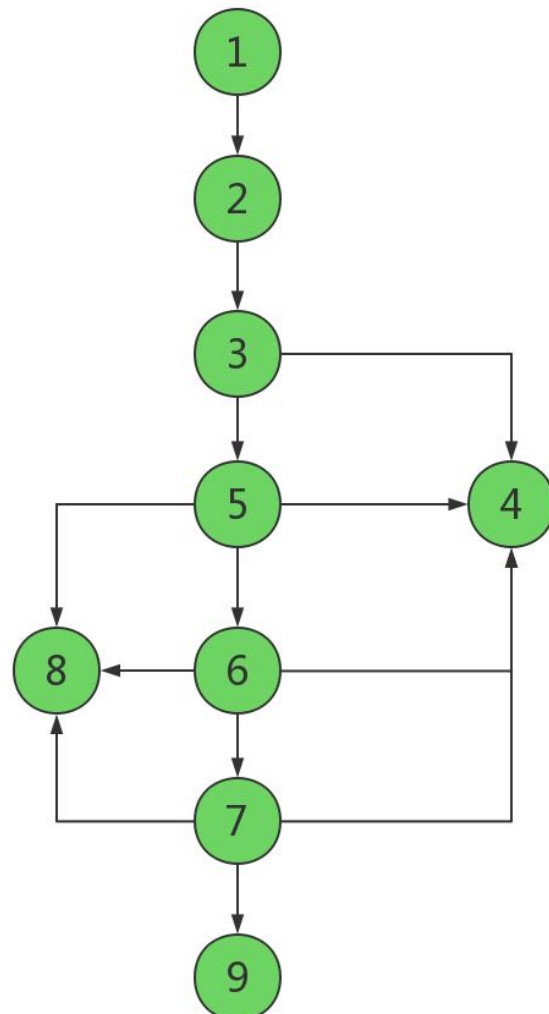
Author: Tian ZHANG

Overview Of All The Flows For Unit 1



Flow 1: Record a Multiple Choice question manually

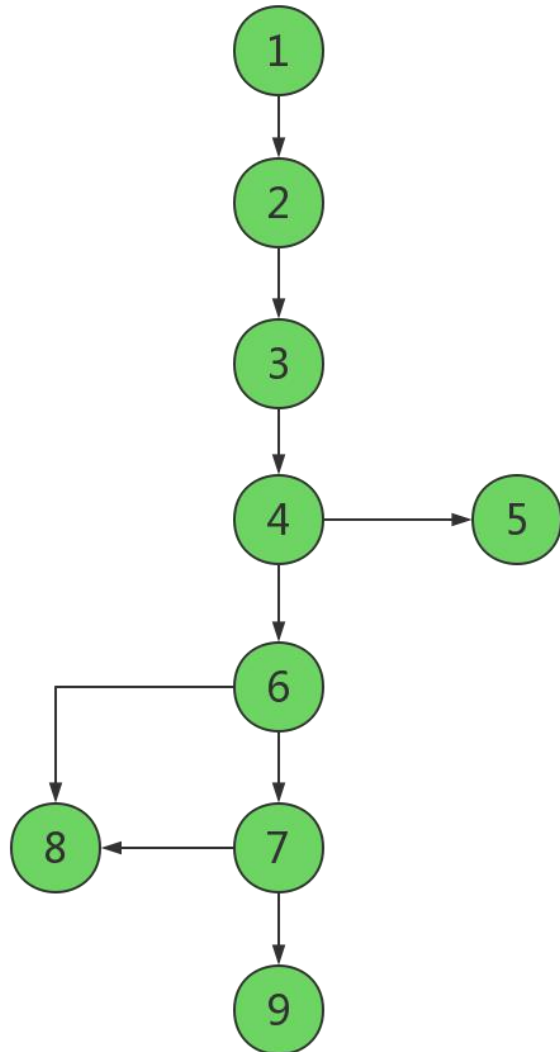
1. Initialize the interface
2. Select the category of the question
3. Select the difficulty of the question
4. Press "Generate!" button without completing the question info
5. Enter the question
6. Enter the correct answer
7. Enter 3 wrong answers
8. Press "Clear" button
9. Press "Generate!" button



Test Case Id: Add Question 01			
Test Purpose: COMSC Engagement team can record a multiple choice question manually			
Environment: Written in python 3.7.4 under PyCharm 2019.3.3			
Preconditions: COMSC Engagement team is logged into the system			
Test Case Steps:			
Step No	Procedure	Response	Pass/Fail
1	Initialize the interface	Interface runs successfully	P
2	Select the category of the question	Can choose different category	P
3	Select the difficulty level of the question	Can choose different difficulty	P
4	Press "Generate!" button without completing the question info	Error message box popping up (Question/ Correct Answer/ Wrong answers cannot be empty, please try again.)	P
5	Enter the question	Question input shows correctly	P
6	Enter the correct answer	Correct answer input shows correctly	P
7	Enter 3 wrong answers	Wrong answers input shows correctly	P
8	Press "Clear" button	All input deleted	P
9	Press "Generate!" button	Message box popping up showing the success hint; All input clear; Generates "system.db" file with the question recorded correctly	P
Comments: Initial function of this unit: to record a Multiple Choice question manually, GUI finished at v1.1, function finished at v1.2, exceptions added at v1.4.			
Related Tests: Add Question 02			
Author: Tian ZHANG		Checker:	

Flow 2: Record a True or False question manually

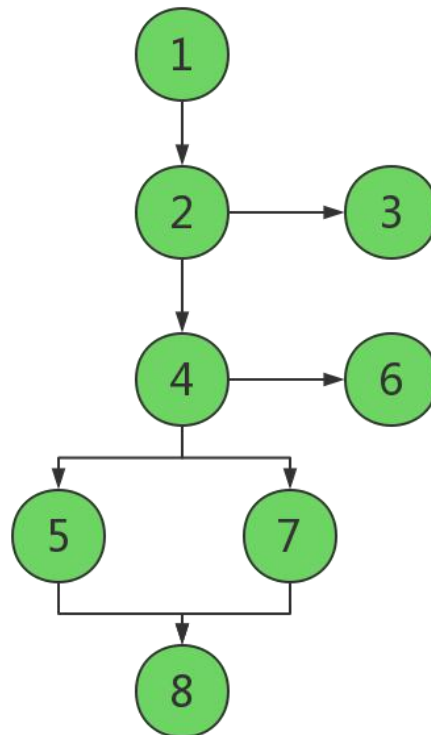
1. Initialize the interface
2. Click "Create a True or False question" button
3. Select the category of the question
4. Select the difficulty of the question
5. Press "Generate!" button without completing the question info
6. Enter the question
7. Select the correct answer
8. Press "Clear" button
9. Press "Generate!" button



Test Case Id: Add Question 02			
Test Purpose: COMSC Engagement team can record a true or false question manually			
Environment: Written in python 3.7.4 under PyCharm 2019.3.3			
Preconditions: COMSC Engagement team is logged into the system			
Test Case Steps:			
Step No	Procedure	Response	Pass/Fail
1	Initialize the interface	Interface runs successfully	P
2	Click "Create a True or False question" button	The interface changes to TF question recording interface	P
3	Select the category of the question	Can choose different category	P
4	Select the difficulty level of the question	Can choose different difficulty	P
5	Press "Generate!" button without completing the question info	Error message box popping up (Question cannot be empty, please try again.)	P
6	Enter the question	Question input shows correctly	P
7	Select the answer with the radio checkbox	Can select either True or False correctly, one at a time	P
8	Press "Clear" button	All input deleted	P
9	Press "Generate!" button	Message box popping up showing the success hint; All input clear; Generates "system.db" file with the question recorded correctly	P
Comments: Initial function of this unit: to record a True or False question manually, GUI finished at v1.1, function finished at v1.2, exceptions added at v1.4.			
Related Tests: Add Question 01			
Author: Tian ZHANG		Checker:	

Flow 3: Import Multiple Choice questions from a .csv file

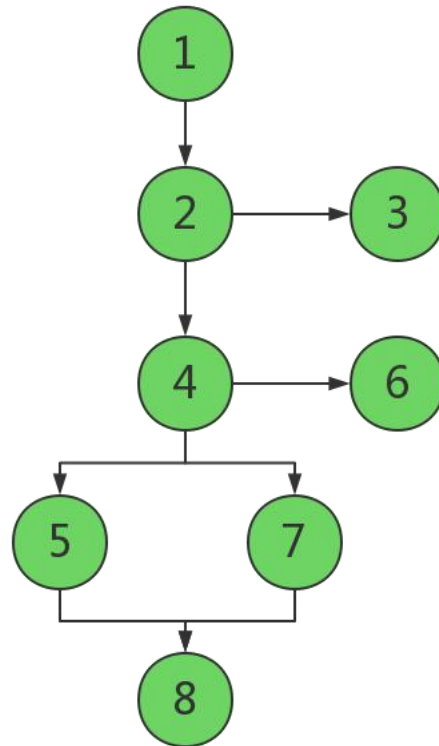
1. Initialize the interface
2. Switch to "Import Tab"
3. Click "Import Questions" button without importing anything
4. Click "Import Multiple Choice data file" button
5. Select file "template_mc_csv.csv"
6. Select other type of file
7. Select file "template_tf_csv.csv"
8. Click "Import Questions" button



Test Case Id: Add Question 03			
Test Purpose: COMSC Engagement team can import multiple choice questions from a .csv file			
Environment: Written in python 3.7.4 under PyCharm 2019.3.3			
Preconditions: COMSC Engagement team is logged into the system			
Test Case Steps:			
Step No	Procedure	Response	Pass/Fail
1	Initialize the interface	Interface runs successfully	P
2	Switch to "Import Tab"	The interface changes to import question tab's interface	P
3	Click "Import Questions" button without importing anything	Error message box popping up (No data imported, please try again)	P
4	Click "Import Multiple Choice data file" button	A file browser popping up	P
5	Select file "template_mc_csv.csv"	File selected successfully, the full path of the file shows in the label below	P
6	Select other type of file	Error message box popping up (File type not supported, please try again); No path shows below; File not selected	P
7	Select file "template_tf_csv.csv"	File selected successfully, the full path of the file shows in the label below	P
8	Click "Import Questions" button (with "template_mc_csv.csv" imported)	Progress bar raises from 0 to top; Questions imported into "system.db" correctly; Message box popping up giving a success hint; File path clear	P
8-2	Click "Import Questions" button (with "template_tf_csv.csv" imported)	Error message box popping up (Wrong data format, choose the right file and try again); Import terminated; File path clear	P
Comments: Considering the requirement to add multiple questions at a time, added the function of importing questions at v1.3, exceptions added at v1.3 and v2.3 (prevent the system from importing wrong format of data), progress bar added at v2.2.			
Related Tests: Add Question 04, 05, 06			
Author: Tian ZHANG		Checker:	

Flow 4: Import Multiple Choice questions from a .txt file

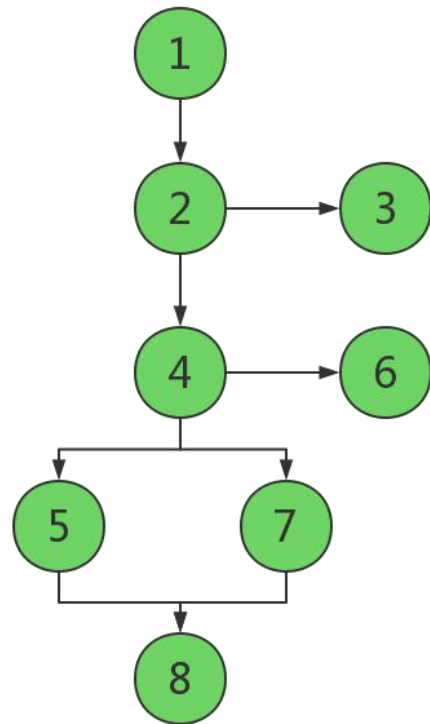
1. Initialize the interface
2. Switch to "Import Tab"
3. Click "Import Questions" button without importing anything
4. Click "Import Multiple Choice data file" button
5. Select file "template_mc_txt.txt"
6. Select other type of file
7. Select file "template_tf_txt.txt"
8. Click "Import Questions" button



Test Case Id: Add Question 04			
Test Purpose: COMSC Engagement team can import multiple choice questions from a .txt file			
Environment: Written in python 3.7.4 under PyCharm 2019.3.3			
Preconditions: COMSC Engagement team is logged into the system			
Test Case Steps:			
Step No	Procedure	Response	Pass/Fail
1	Initialize the interface	Interface runs successfully	P
2	Switch to "Import Tab"	The interface changes to import question tab's interface	P
3	Click "Import Questions" button without importing anything	Error message box popping up (No data imported, please try again)	P
4	Click "Import Multiple Choice data file" button	A file browser popping up	P
5	Select file "template_mc_txt.txt"	File selected successfully, the full path of the file shows in the label below	P
6	Select other type of file	Error message box popping up (File type not supported, please try again); No path shows below; File not selected	P
7	Select file "template_tf_txt.txt"	File selected successfully, the full path of the file shows in the label below	P
8	Click "Import Questions" button (with "template_mc_txt.txt" imported)	Progress bar raises from 0 to top; Questions imported into "system.db" correctly; Message box popping up giving a success hint; File path clear	P
8-2	Click "Import Questions" button (with "template_tf_txt.txt" imported)	Error message box popping up (Wrong data format, choose the right file and try again); Import terminated; File path clear	P
Comments: Considering the requirement to add multiple questions at a time, added the function of importing questions at v1.3, exceptions added at v1.3 and v2.3 (prevent the system from importing wrong format of data), progress bar added at v2.2.			
Related Tests: Add Question 03, 05, 06			
Author: Tian ZHANG		Checker:	

Flow 5: Import True or False questions from a .csv file

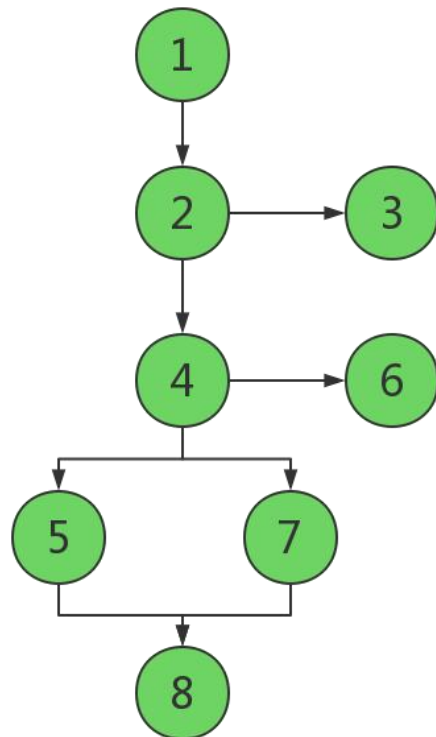
1. Initialize the interface
2. Switch to "Import Tab"
3. Click "Import Questions" button without importing anything
4. Click "Import True or False data file" button
5. Select file "template_tf_csv.csv"
6. Select other type of file
7. Select file "template_mc_csv.csv"
8. Click "Import Questions" button



Test Case Id: Add Question 05			
Test Purpose: COMSC Engagement team can import true or false questions from a .csv file			
Environment: Written in python 3.7.4 under PyCharm 2019.3.3			
Preconditions: COMSC Engagement team is logged into the system			
Test Case Steps:			
Step No	Procedure	Response	Pass/Fail
1	Initialize the interface	Interface runs successfully	P
2	Switch to "Import Tab"	The interface changes to import question tab's interface	P
3	Click "Import Questions" button without importing anything	Error message box popping up (No data imported, please try again)	P
4	Click "Import Multiple Choice data file" button	A file browser popping up	P
5	Select file "template_tf_csv.csv"	File selected successfully, the full path of the file shows in the label below	P
6	Select other type of file	Error message box popping up (File type not supported, please try again); No path shows below; File not selected	P
7	Select file "template_mc_csv.csv"	File selected successfully, the full path of the file shows in the label below	P
8	Click "Import Questions" button (with "template_tf_csv.csv" imported)	Progress bar raises from 0 to top; Questions imported into "system.db" correctly; Message box popping up giving a success hint; File path clear	P
8-2	Click "Import Questions" button (with "template_mc_csv.csv" imported)	Error message box popping up (Wrong data format, choose the right file and try again); Import terminated; File path clear	P
Comments: Considering the requirement to add multiple questions at a time, added the function of importing questions at v1.3, exceptions added at v1.3 and v2.3 (prevent the system from importing wrong format of data), progress bar added at v2.2.			
Related Tests: Add Question 03, 04, 06			
Author: Tian ZHANG		Checker:	

Flow 6: Import True or False questions from a .txt file

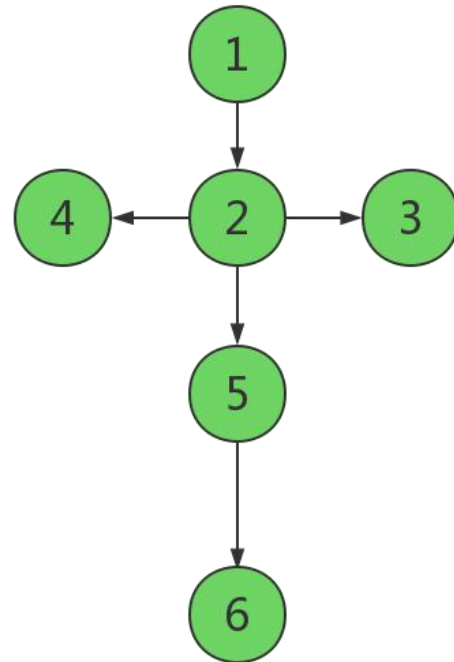
1. Initialize the interface
2. Switch to "Import Tab"
3. Click "Import Questions" button without importing anything
4. Click "Import True or False data file" button
5. Select file "template_tf_txt.txt"
6. Select other type of file
7. Select file "template_mc_txt.txt"
8. Click "Import Questions" button



Test Case Id: Add Question 06			
Test Purpose: COMSC Engagement team can import true or false questions from a .txt file			
Environment: Written in python 3.7.4 under PyCharm 2019.3.3			
Preconditions: COMSC Engagement team is logged into the system			
Test Case Steps:			
Step No	Procedure	Response	Pass/Fail
1	Initialize the interface	Interface runs successfully	P
2	Switch to "Import Tab"	The interface changes to import question tab's interface	P
3	Click "Import Questions" button without importing anything	Error message box popping up (No data imported, please try again)	P
4	Click "Import Multiple Choice data file" button	A file browser popping up	P
5	Select file "template_tf_txt.txt"	File selected successfully, the full path of the file shows in the label below	P
6	Select other type of file	Error message box popping up (File type not supported, please try again); No path shows below; File not selected	P
7	Select file "template_mc_txt.txt"	File selected successfully, the full path of the file shows in the label below	P
8	Click "Import Questions" button (with "template_tf_txt.txt" imported)	Progress bar raises from 0 to top; Questions imported into "system.db" correctly; Message box popping up giving a success hint; File path clear	P
8-2	Click "Import Questions" button (with "template_mc_txt.txt" imported)	Error message box popping up (Wrong data format, choose the right file and try again); Import terminated; File path clear	P
Comments: Considering the requirement to add multiple questions at a time, added the function of importing questions at v1.3, exceptions added at v1.3 and v2.3 (prevent the system from importing wrong format of data), progress bar added at v2.2.			
Related Tests: Add Question 03, 04, 05			
Author: Tian ZHANG		Checker:	

Flow 7: Display all the Multiple Choice questions recorded

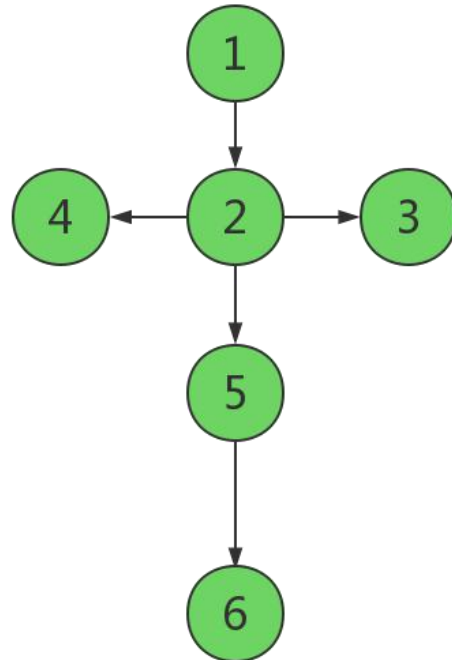
1. Initialize the interface
2. Switch to "Display tab"
3. Click the table area with the relevant table empty
4. Click "Display Multiple Choice questions" button with no db file or empty db file
5. Click "Display Multiple Choice questions" button with data in relevant table of the db file
6. Click one certain piece of data



Test Case Id: Add Question 07			
Test Purpose: COMSC Engagement team can see all the Multiple Choice questions displayed			
Environment: Written in python 3.7.4 under PyCharm 2019.3.3			
Preconditions: COMSC Engagement team is logged into the system			
Test Case Steps:			
Step No	Procedure	Response	Pass/Fail
1	Initialize the interface	Interface runs successfully	P
2	Switch to "Display Tab"	The interface changes to display question tab's interface	P
3	Click the table area with the relevant table of the db file empty	Nothing happens, no error message shows in python console	P
4	Click "Display Multiple Choice questions" button without any data in the db file	Error message box popping up (No question found, please record the question first)	P
5	Click "Display Multiple Choice questions" button with data in the db file	All the questions recorded displayed in the table below the button	P
6	Click a certain piece of data in the table area	Question selected; Details of the question showed in the label below the table area	P
Comments: This tab has been developed for the purpose that the user could check all the questions recorded without using extra db browser. Main function finished at v2.0, auto-word-wrap added at v2.1 (to prevent the interface from cutting off the output), exceptions for clicking the empty table added at v2.3.			
Related Tests: Add Question 08			
Author: Tian ZHANG		Checker:	

Flow 8: Display all the True or False questions recorded

1. Initialize the interface
2. Switch to "Display tab"
3. Click the table area with the relevant table empty
4. Click "Display True or False questions" button with no db file or empty db file
5. Click "Display True or False questions" button with data in relevant table of the db file
6. Click one certain piece of data



Test Case Id: Add Question 08			
Test Purpose: COMSC Engagement team can see all the True or False questions displayed			
Environment: Written in python 3.7.4 under PyCharm 2019.3.3			
Preconditions: COMSC Engagement team is logged into the system			
Test Case Steps:			
Step No	Procedure	Response	Pass/Fail
1	Initialize the interface	Interface runs successfully	P
2	Switch to "Display Tab"	The interface changes to display question tab's interface	P
3	Click the table area with the relevant table of the db file empty	Nothing happens, no error message shows in python console	P
4	Click "Display True or False questions" button without any data in the db file	Error message box popping up (No question found, please record the question first)	P
5	Click "Display True or False questions" button with data in the db file	All the questions recorded displayed in the table below the button	P
6	Click a certain piece of data in the table area	Question selected; Details of the question showed in the label below the table area	P
Comments: This tab has been developed for the purpose that the user could check all the questions recorded without using extra db browser. Main function finished at v2.0, auto-word-wrap added at v2.1 (to prevent the interface from cutting off the output), exceptions for clicking the empty table added at v2.3.			
Related Tests: Add Question 07			
Author: Tian ZHANG		Checker:	

Flow 9: Check author information

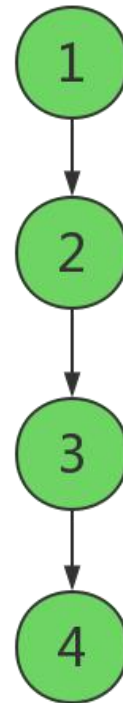
1. Initialize the interface
2. Click "Help" menu bar
3. Click "About" menu



Test Case Id: Add Question 09			
Test Purpose: COMSC Engagement team can check the author's information			
Environment: Written in python 3.7.4 under PyCharm 2019.3.3			
Preconditions: COMSC Engagement team is logged into the system			
Test Case Steps:			
Step No	Procedure	Response	Pass/Fail
1	Initialize the interface	Interface runs successfully	P
2	Click "Help" menu bar	A menu shows up	P
3	Click "About" menu	A message box with author info popping up	P
Comments: Simple menu to show author information. Finished at v1.1.			
Related Tests: Add Question 10			
Author: Tian ZHANG		Checker:	

Flow 10: Move to/ from other units

1. Initialize the interface of "backstage_index.py"
2. Click "Create Question" button
3. Click "System" menu
4. Click "Return to index" menu



Test Case Id: Add Question 10			
Test Purpose: COMSC Engagement team can move to/ from other units			
Environment: Written in python 3.7.4 under PyCharm 2019.3.3			
Preconditions: COMSC Engagement team is logged into the system			
Test Case Steps:			
Step No	Procedure	Response	Pass/Fail
1	Initialize the interface of "backstage_index.py"	Interface runs successfully	P
2	Click "Create Question" button	Current GUI exit, GUI of Unit1 shows up	P
3	Click "System" menu	List of System menu displayed	P
4	Click "Return to index" menu	Current GUI exit, GUI of "backstage_index.py" shows up	P
Comments: A necessary function for the unit integration, created at v1.5.			
Related Tests: Add Question 09			
Author: Tian ZHANG		Checker:	