Нумерация:

Номер требования пишется в квадратных скобках и состоит из следующих частей:

1. сокращение REQ;
2. название подсистемы. Возможные значения: FRONT – требования, относящиеся к интерфейсу программы, IO – требования к подсистеме ввода и вывода, TRANSLATE – требования к подсистеме трансляции.
3. порядковый номер. Для каждой подсистемы используется отдельный порядковый числовой номер для требования.

Между частями номера требования пишется символ «\_».

Требования к интерфейсу пользователя:

1. интерфейс должен быть графическим и должен быть выполнен в зелёно-белых цветах;
2. элементы управления должны быть выполнены в «плоском» стиле, то есть у них не должно быть теней и иллюзии объёма;
3. вертикальное разделение основной рабочей области на два текстовых поля. В поле слева вводится текст программы на языке C#, в поле справа выводится эквивалентная программа на языке Java. Текст в правом поле должен быть доступен только для чтения;
4. каждое поле для ввода кода программ на языках C# и Java занимают половину ширины рабочей области и изменяют ширину автоматически вместе с изменением ширины окна;
5. поля для ввода текста должны показывать соответствующие плейсхолдеры;
6. под левым текстовым полем должна располагаться кнопка для вызова диалога открытия файла с исходным кодом с расширением CS или TXT;
7. под правым текстовым полем должна располагаться кнопка для вызова диалога сохранения переведённого кода в файл с расширением JAVA или TXT;
8. трансляция кода из C# в Java происходит при нажатии кнопки «Транслировать» в меню программы или под полем ввода кода на C#;
9. если поле для ввода кода на языке C# пустое, кнопка «Транслировать» должна становиться неактивной;
10. если поле для вывода кода на языке Java пустое, кнопка для вызова диалога сохранения кода в файл должна становиться неактивной;
11. ошибки в коде на языке C# должны выделяться красным цветом (например, красной чертой под текстом, или красным фоном);
12. при изменении текста в поле ввода кода на языке C# подсветка ошибок должна сбрасываться;
13. горизонтальное разделение рабочей области. В верхней части находятся поля для ввода и вывода, в нижней части находится консоль с логами работы транслятора. Текст в консоли доступен только для чтения;
14. на форме должна быть кнопка для очистки текста в консоли;
15. если консоль пустая, то кнопка для очистки текста в ней должна становиться неактивной;
16. при выходе из программы при наличии переведённого кода у пользователя должно запрашиваться подтверждение действия.

Требования к подсистеме ввода/вывода:

1. программа должна позволять читать код на языке C# из файлов формата TXT и CS;
2. программа должна сообщать, что выбранный входной файл является пустым;
3. программа должна позволять сохранять код на языке Java в файлы формата TXT и JAVA.

Требования к подсистеме трансляции:

1. транслятор должен состоять из четырёх компонентов: лексического анализатора, синтаксического анализатора, семантического анализатора и генератора кода;
2. транслятор должен уметь находить синтаксические и семантические ошибки в коде на языке C# и сообщать о них;
3. транслятор должен заменять несуществующие в Java типы на эквивалентные (например, «nuint» на «int»);
4. транслятор должен вести логи своей работы.