https://github.com/saie2009/LFTC_Salagean
Lab6

Finite Automata
S A B -> the set of states
a b -> the alphabet
S -> the initial state
S B -> the final state
The transitions:
S a A
A a A
A b A
A a B
A b B

Sequence:
abbba

EBNF of FA:
S = State{" "State}

S -> aA | e
A -> aA | bA | aB | bB
B -> e

need to be in final state and sequence be empty

Documentation:
-UI
 -> HomeView
  private FAService _faService;

  /// <summary>
  /// Dispays the options to the user
  /// </summary>
  private void ShowMenu()

  /// <summary>
  /// Runs the FA scanner and executse the service methods
  /// </summary>
  public void RunScanner()

-Services
 -> FAService
  private Automata _automata;

  /// <summary>
  /// Returns the set of states
  /// </summary>
  public List<string> GetSetOfStates()

  /// <summary>
  /// Returns the alphabet

```
  /// </summary>
  public List<string> GetAlphabet()

  /// <summary>
  /// Returns the initial statee
  /// </summary>
  public string GetInitialState()

  /// <summary>
  /// Returns the set of final states
  /// </summary>
  public List<string> GetSetOfFinalStates()

  /// <summary>
  /// Returns the list of transitions
  /// </summary>
  public List<Transition> GetTransitions()

  /// <summary>
  /// Verifies wether or not the sequence is accepted by the FA
  /// </summary>
  public bool VerifySequence(string sequence)

  /// <summary>
  /// Reads from a file a given Finite Automata which must follow a predefined structure
  /// </summary>
  /// <param name="fileName"></param>
  private void ReadFA(string fileName)

-Models
 -> Automata
  public List<string> States

  public List<string> Alphabet

  public string InitialState

  public List<string> FinalStates

  public List<Transition> Transitions

 -> Transition
  public string State

  public string AlphabetSequence

  public string Result

  /// <summary>
  /// Prettify the transition display
  /// </summary>
  public override string ToString()
```