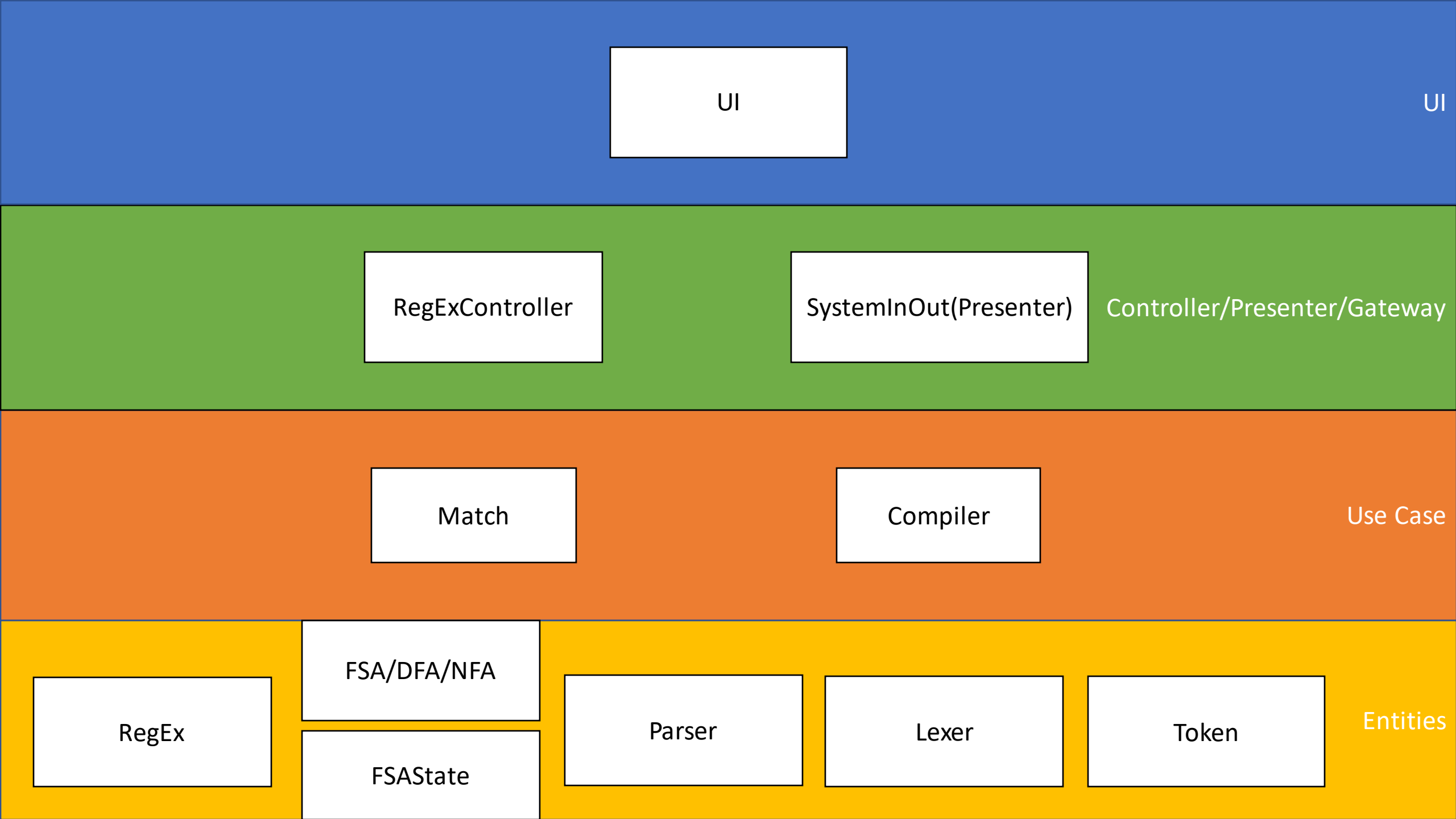


CRC model



Name: RegEx # input a string as a regex expression, output the match

Entities

- Responsibilities:

- Save input string to a RegEx instance and output the match result by working with Parser, Compiler and Match
- Convert a RegEx statement to NFA with Parser(Nondeterministic Finite Automaton)

- Collaborators:

- Parser
- Compiler
- Match
- NFA

Name: Token #abstract class

Entities

- Responsibilities:

- Save the TokenType to an instance attribute
- Have a string to save and be evaluated in value

- Collaborators:

- TokenType

Name: TokenType

Entities

- Responsibilities:

- Several Supported TokenType stored(*Char*, KleeneStar, KleenePlus, Operator, Number, Delimiter)

- Collaborators:

- TokenType

Name: FSA #Interface

Entities

Subclasses: DFA, NFA

- **Responsibilities:**

- Store a list of FSASState(nodes in FSA)

- **Collaborators:**

- FSASState

Name: FSAState

Entities

- **Responsibilities:**

- Store and Show the state of FSA.

- **Collaborators:**

- FSA

Name: Lexer # Convert String to a list of tokens

Entities

- Responsibilities:

- Analyze the input and stored string and convert it to a list of Tokens

- Collaborators:

- Parser
- Token



Name: Parser # Convert a list of tokens to NFA

Entities

- Responsibilities:

- Analyze and convert a list of tokens to NFA

- Collaborators:

- NFA
- Token
- Lexer

Name: Compiler

Use Case

- Responsibilities:

- **compile** (Compile a Token into FSA object with Lexer and Parser)

- Collaborators:

- Parser
- Lexer
- FSA

Name: Match # match the string

Use Case

- Responsibilities:

- **hasMatch** (check if there's any match return a boolean)
- **matchGroups** (return a list of groups in the test that matches the RegEx)
- Match string using FSA
- Getters of Match attribute

- Collaborators:

- Compiler
- RegEx

Name: RegExController

Controller

- Responsibilities:

- **getMatches**(This method will mainly get a regular string input and create RegEx with Match and Compiler and give a list of Match result for compiled RegEx)
- **simplifyRegex**(This method will mainly get a RegEx input and use Match and Compiler to find the simplified RegEx result)
- Get the result to the SystemInOut for output at basic command line interface

- Collaborators:

- Compiler
- Match
- SystemInOut
- FSA

Name: SystemInOut(Main)

Presenter

- **Responsibilities:**

- (Now)A Basic Command Line interface that interacts with user and input the value to create RegEx instance by RegExController and output the required content by user

- **Collaborators:**

- RegExController

Name: UI

UI (Frameworks & Drivers)

- **Responsibilities:**

- The interface that interacts with users

- **Collaborators:**

- SystemInOut