

## Front-end Classes/ Patterns

```
1. // ###
2. // Front-end classes
3. // ###
4.
5. class Login {
6.     constructor( /*...*/ ) {}
7.     sendLogin() {}
8. }
9.
10. class Register {
11.     constructor( /*...*/ ) {}
12.     sendRegister() {}
13. }
14.
15. class TwitterLogin {
16.     constructor( /*...*/ ) {}
17.     sendKey() {}
18.     sendSecret() {}
19. }
20.
21. class credentialsVerifier {
22.     constructor( /*...*/ ) {}
23.     checkEmail() {}
24.     checkPasswordMatch() {}
25.     checkUserName() {}
26.     checkCapcha() {}
27.     checkStrings() {}
28.     sendCredentials() {}
29.     resolveError(){}
30. }
31.
32. class TwitterRezolver {
33.     constructor( /*...*/ ) {}
34.     sendKeyData() {}
35.     sendSecretData() {}
36.     resolveError(){}
37. }
38.
39. class CredentialsCommunication {
40.     constructor( /*...*/ ) {}
41.     sendLogin() {}
42.     sendRegister() {}
43.     sendTwitter() {}
44.     updateName() {}
45.     updateUserName() {}
46.     updatePassword() {}
47.     updateEmail() {}
48.     updateMarking() {}
49.     throwError(){}
50. }
51.
52. class User {
53.     constructor( /*...*/ ) {}
54.     sendToCollector(){}
55. }
56.
57. class Settings {
58.     constructor( /*...*/ ) {}
```

```

59.  checkName() {}
60.  checkUserName() {}
61.  checkPassword() {}
62.  checkEmail() {}
63.  sendName() {}
64.  sendUserName() {}
65.  sendPassword() {}
66.  sendEmail() {}
67.  sendMarking() {}
68. }
69.
70. class preferencesCollector {
71.   constructor( /*...*/ ) {}
72.   sendCollection(){}
73. }
74.
75. class preferencesGathering {
76.   constructor( /*...*/ ) {}
77.   validateCollection(){}
78.   sendCollection(){}
79. }
80.
81. class fakeNewsRezolver {
82.   constructor( /*...*/ ) {}
83.   updateInfo() {}
84. }
85.
86. class metricsBuilder {
87.   constructor( /*...*/ ) {}
88.   showMetrics() {}
89. }
90.
91. // ###
92. // Front-end observer pattern used
93. // when you edit the level of the marking
94. // ###
95.
96. var markingObserver = function() {
97.
98.   return {
99.     subscribeMarkingObserver: function(observer) {
100.
101.       },
102.     unsubscribeMarkingObserver: function(observer) {
103.
104.       },
105.     notifyMarkingObserver: function(observer) {
106.
107.       },
108.     notifyAllMarkingObservers: function() {
109.
110.       }
111.     };
112.   };
113.
114.   var mkObserver = function() {
115.     return {
116.       notify: function(index) {
117.         console.log("Observer " + index + " is notified!");
118.       }
119.     }

```

```
120.     }
121.
122.     // ###
123.     // Front-end builder pattern used
124.     // when you make the metrics
125.     // ###
126.
127.     class Metrics {
128.         constructor(/*...*/) {
129.
130.         }
131.         static get Builder() {
132.             class Builder {
133.                 constructor(/*...*/) {
134.
135.                 }
136.                 withNumbers() {
137.
138.                 }
139.                 withColors() {
140.
141.                 }
142.                 build() {
143.                     return new Metrics(this);
144.                 }
145.             }
146.             return Metrics;
147.         }
148.     }
```