

```

1 @startuml
2 'https://plantuml.com/class-diagram
3 interface Observer {
4     +update(String message)
5 }
6
7 interface Observable {
8     +subscriber(User observer)
9     +unsubscribe(User observer)
10    +notifyObservers(String string)
11 }
12
13 class User {
14     -int id
15     -String name
16     +update(String message)
17 }
18
19 class Creator {
20
21 }
22
23 Observer <|-- User
24 User <|-- Creator
25
26 abstract Video {
27     -int id
28     -String title
29     -String description
30     -int duration
31     -User createdBy
32     -Comment[] comments
33     -Reaction[] reactions
34
35     +play()
36     +pause()
37     +share(User user)
38     +comment(User user, String text)
39     +saveToWatchLater(User user)
40     +next()
41 }
42
43 class LongVideo {
44     +toggleCaption()
45     +next()
46 }
47
48 Video <|-- LongVideo
49
50 class Reel {
51     +next()
52 }
53

```

```

54 Video <|-- Reel
55
56 interface Iterator<Video> {
57     +boolean hasNext()
58     +Video next()
59 }
60
61 class VideoIterator {
62     +boolean hasNext()
63     +Video next()
64 }
65
66 Iterator <|-- VideoIterator
67
68 interface IterableByUser {
69     +Iterator<Video> search(String keywords)
70     +Iterator<Video> getRecommendations(User user, String keywords)
71 }
72
73 class Channel {
74     -Creator owner
75     -Video[] videos
76     -User[] subscribers
77
78     +subscriber(User user)
79     +unsubscribe(User user)
80     +notifyObservers(String string)
81 }
82
83 Observable <|-- Channel
84
85 Channel "1" o-- "0..*" Video
86 Channel "1" *-- "0..*" User
87
88 Creator "1" *-- "0..*" Channel
89
90 class Reaction {
91     +click()
92 }
93
94 class Like {
95     -User likedBy
96 }
97
98 class DisLike {
99     -User dislikedBy
100 }
101
102 Reaction <|-- Like
103 Reaction <|-- DisLike
104 Video "1" o--"0..*" Reaction
105
106 class Comment {

```

```

107     -String text
108     -User commentedBy
109 }
110 Video "1" o-- "0..*" Comment
111 Comment o-- User
112
113
114 interface RequestVideo {
115     +Video getVideo(int id)
116     ' +Iterator<Video> search(String keywords)
117     ' +Iterator<Video> getRecommendations(User user, String keywords)
118 }
119
120 IterableByUser <|-- RequestVideo
121
122 class VideoServer {
123     -Channel<String,Video[]> channels
124     -RecommendationEngine recommendationEngine
125
126     +Video getVideo(int id)
127     +bool createChannel(String name, Creator user)
128     +Video createVideo(Channel channel, Video video, Creator user)
129     +bool deleteVideo(Video video, Creator user)
130     +Iterator<Video> search(String keywords)
131     +Iterator<Video> getRecommendations(User user, String keywords)
132 }
133
134 RequestVideo <|-- VideoServer
135 'IterableByUser <|-- VideoServer
136
137 class RecommendationEngine {
138     +Iterator<Video> recommendVideos(User user)
139 }
140
141 VideoServer o-- RecommendationEngine
142
143 VideoServer *-- Channel
144 VideoServer o-- User
145
146 class ProxyEdgeVideoServer {
147     -VideoServer videoServer
148     -Video<key,Video> videos
149
150     +Video getVideo(int id)
151     +Iterator<Video> search(String keywords)
152     +Iterator<Video> getRecommendations(User user, String keywords)
153 }
154
155 RequestVideo <|-- ProxyEdgeVideoServer
156 ProxyEdgeVideoServer o-- VideoServer
157 ProxyEdgeVideoServer o-- Video
158
159

```

160

161 @enduml

```

1 @startuml
2 start
3 if ( Open \nYouTube\n App) then
4 :Search;
5 :Input Keyword;
6 :Select Videos;
7 if (Starts Playing) then
8     :Let it Play;
9     else
10    :Pause;
11    endif
12
13    if (Toggle Captions) then
14        if (Caption on) then
15            : Choose \nLanguage;
16        else
17            :Default;
18        endif
19    else
20        : Default \n(Caption off);
21    endif
22
23
24 else
25     if ( Browse \nRecommendations) then
26         :Watch Videos?;
27         :Select Video ;
28         ->Yes;
29
30         if (Starts Playing) then
31             :Let it Play;
32         else
33             :Pause;
34         endif
35
36         if (Toggle Captions) then
37             if (Caption on) then
38                 : Choose \nLanguage;
39             else
40                 :Default;
41             endif
42         else
43             :Default \n(Caption off);
44         endif
45     else
46         :Watch Reels?;
47         :Click on Reels;
48         if (Starts Playing) then
49             :Let it Play;
50         else
51             :Pause;
52         endif
53

```

```

54         : Scroll\n(Up/Down);
55     endif
56 endif
57
58 split
59
60 if (Likings) then
61 :Like;
62 :Saved to Liked\n Videos Library;
63 :Like Count\n Increases;
64
65 else
66 :Dislike;
67 :Dislike Count\n Increases;
68 endif
69
70 split again
71
72 if (Comment on\n Video?) then
73 :Yes;
74 : Comment\n Published;
75 :Comment Count\n      Increases;
76 else
77 :No;
78 endif
79
80 split again
81
82 if (Share Video?) then
83 :Yes;
84
85 if (Copy Link) then
86 :Yes;
87 :Send to \nFriends;
88 else
89 :No;
90 : Share on \nSocial Media;
91 endif
92
93 else
94 :No;
95 endif
96
97 split again
98
99 if (Subscribe?) then
100 :Yes;
101
102 if (Notification\n settings\n enabled) then
103         :Keep it on;
104         else
105         :Turn it off;
106         endif

```

Youtube (Quiz3)

```
107 else
108 :No;
109 endif
110
111 split again
112
113 if (Save for\n Later?) then
114 :Yes;
115 :Saved;
116 else
117 :No;
118 endif
119 end split
120 :Close the Video;
121 : Return To \n YouTube\n home screen ;
122 :Close App;
123
124 stop
125 @enduml
126
```

Youtube (Quiz3)

```
1 @startuml
2
3 start
4 :Open YouTube App;
5 :Log In to your Account;
6   if (Switch Channel)
7     :Yes;
8     :Select Channel;
9   else
10    :No;
11  endif
12
13 :Click Upload Button (+);
14 Repeat
15 if (Select Video File)
16   :File is Valid;
17 else
18   :Invalid File;
19   :Select a Valid Video;
20 endif
21 Repeat while
22 :Provide Video Details (Title, Description, etc.);
23
24 Repeat
25 :Validation;
26 if (File Valid?)
27
28   :Yes;
29   if (Edit Video?)
30     :Yes;
31     :Edited;
32   else
33     :No;
34
35   endif
36 :Publish Video;
37
38 stop
39 else
40   :Error: Try Again;
41 endif
42 backward:Check Video Details \n(Title, Description, etc.);
43 Repeat while
44
45 @enduml
46
```