

WEINTEK IIOT LTD.

Integrating JS Object and Enhanced Security Mode

Enhancing User Account Control
(UAC) Login and Verification

Demo Project

Contents

- 1. Overview & Operation 1
- 2. Setting up the Screen 7
- 3. Addresses 9

1. Overview & Operation

Overview

Enhancing User Account Control (UAC) login and verification features is crucial in the current network security environment. This demo project aims to achieve this goal through the use of JavaScript objects and enhanced UAC policies.

Password Policies

To increase security, the following password policies will be implemented:

- **Minimum Password Length**
Users must set a password that meets the minimum length requirement to prevent overly simple passwords.
- **Password Complexity Requirements**
Passwords must include at least one number, one letter, and one special character to increase complexity.
- **Failed Login Handling**
The system will limit the number of allowed failed login attempts, locking the account for a specified time after exceeding the limit.

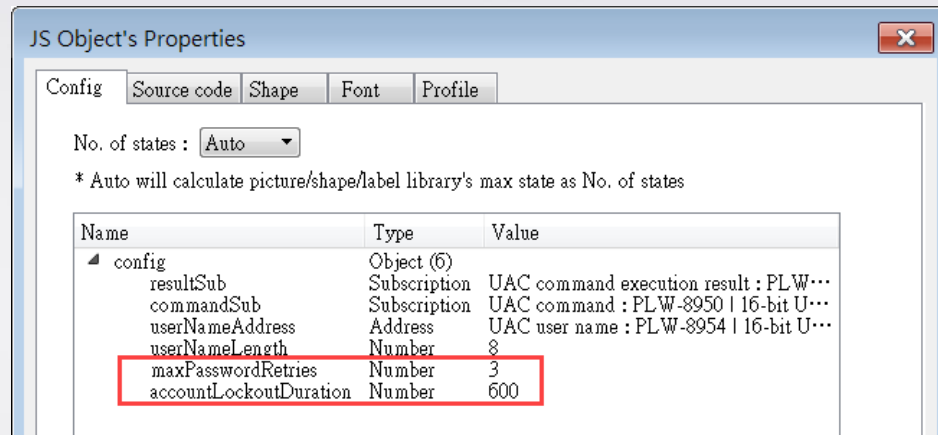
Implementing these policies will significantly improve system security and provide users with a more secure login environment.

Operation

Step 1. Set up the failed login policy.

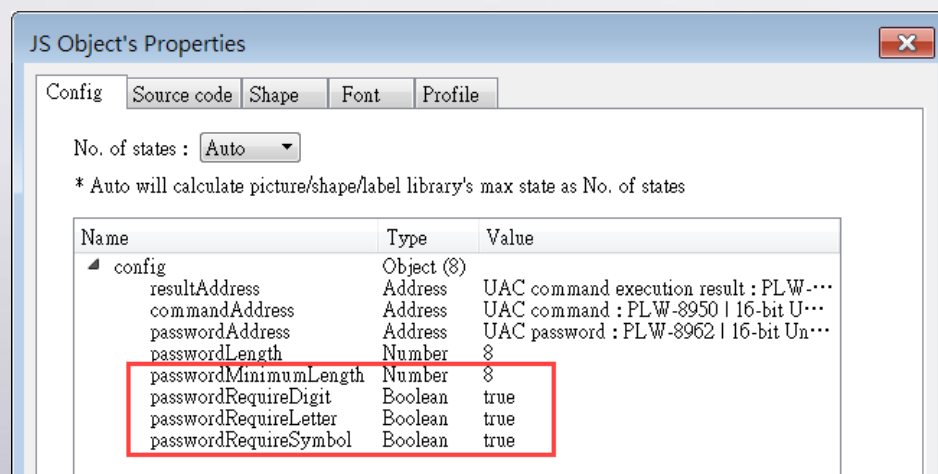
Go to Window 70, in JS object ID CO_0, set the maximum allowed failed login attempts (maxPasswordRetries) and lockout duration

(accountLockoutDuration, in seconds). The example setting allows a maximum of 3 failed attempts, with a 600-second lockout period.



Step 2. Set password requirements when creating an account.

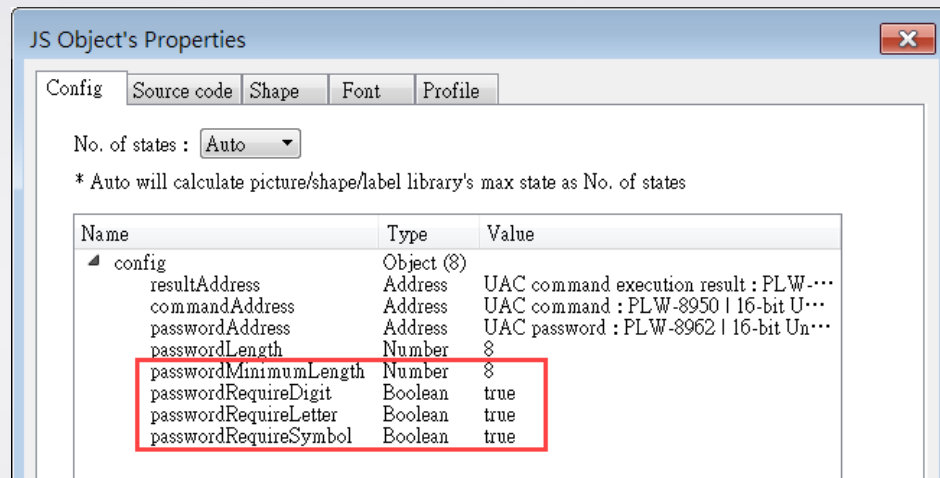
Go to Window 71, in JS object ID CO_0, set the minimum password length (passwordMinimumLength, in characters), and ensure the password includes a number (passwordRequireDigit), a letter (passwordRequireLetter), and a special character (passwordRequireSymbol). This example requires a minimum password length of 8 characters, including the specified elements.



Step 3. Set password rules when changing passwords.

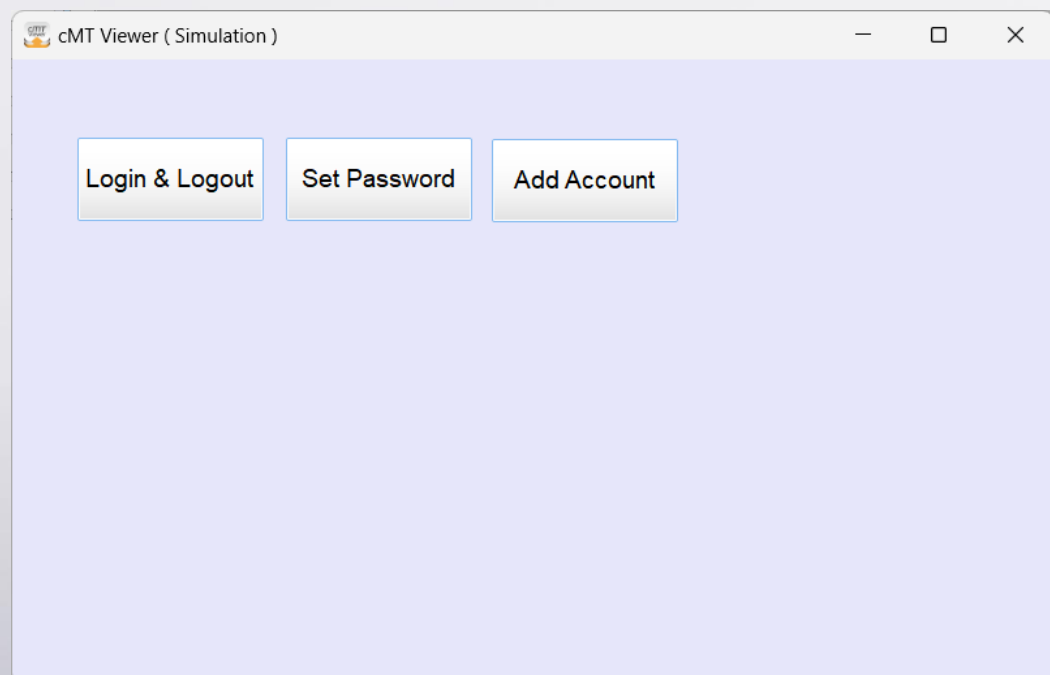
Go to Window 74, in JS object ID CO_0, set the minimum password length (passwordMinimumLength, in characters) and complexity requirements. This example also requires a minimum password

length of 8 characters, including at least one number, one letter, and one special character.

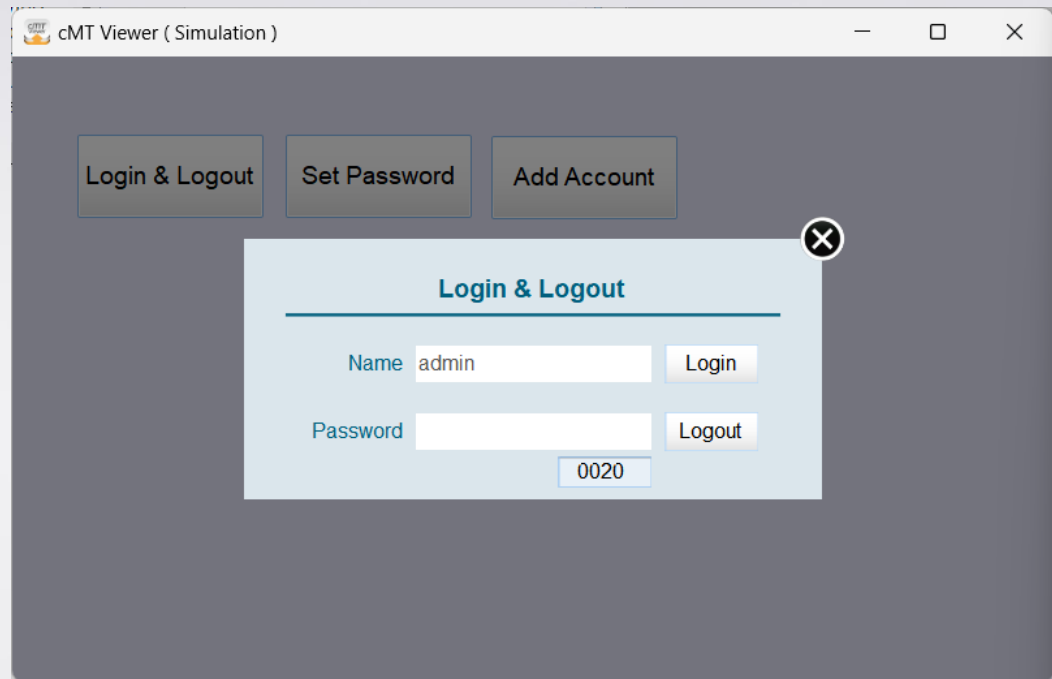


Step 4. Conduct a simulation test.

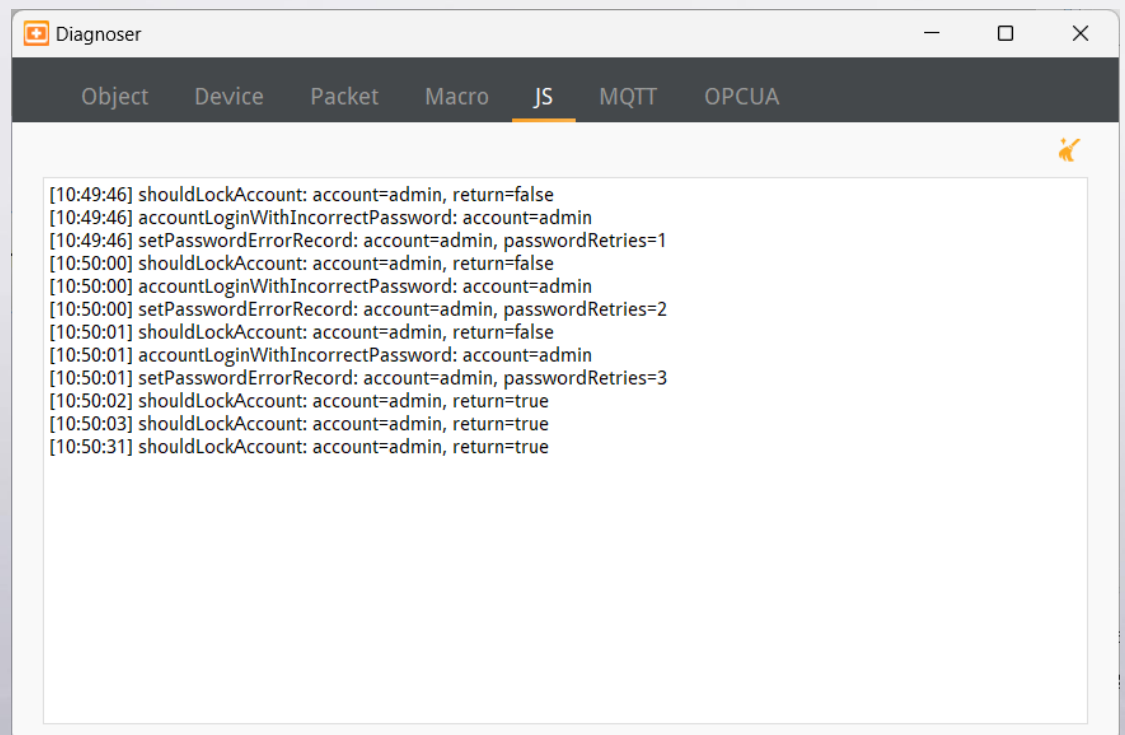
On the main screen, use the buttons to call the Login & Logout window, Set Password window, and Add Account window for testing.



Step 5. Call the Login & Logout window and attempt to log in using the admin account, intentionally enter incorrect password for testing.

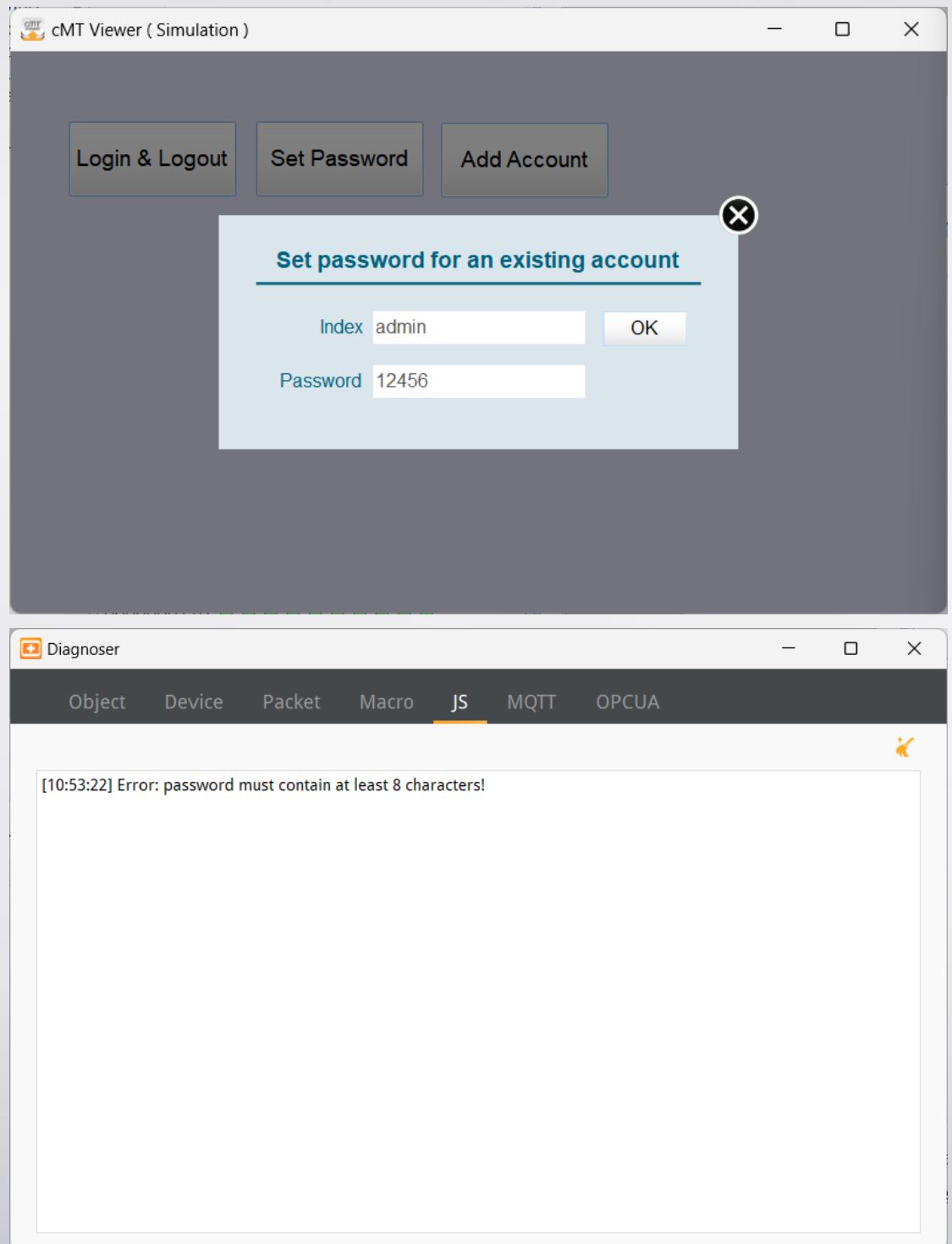


Step 6. Use the cMT Diagnoser's JS console to observe whether the system locks the account after three incorrect password attempts, preventing further login attempts.

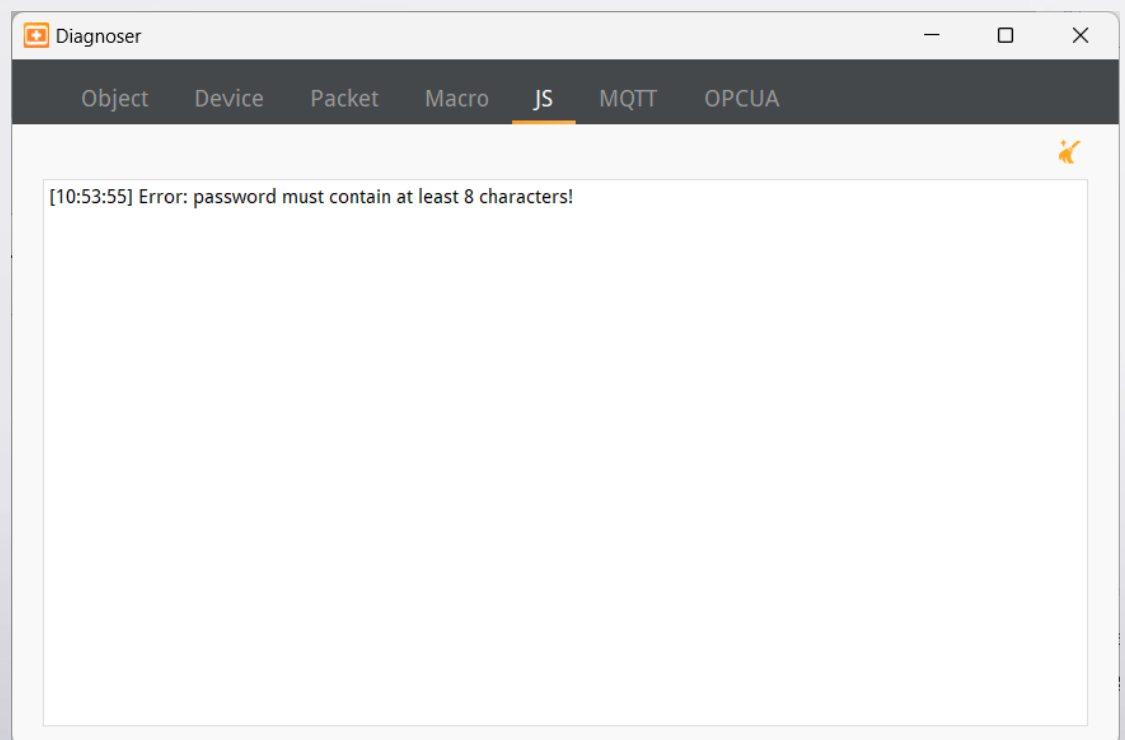
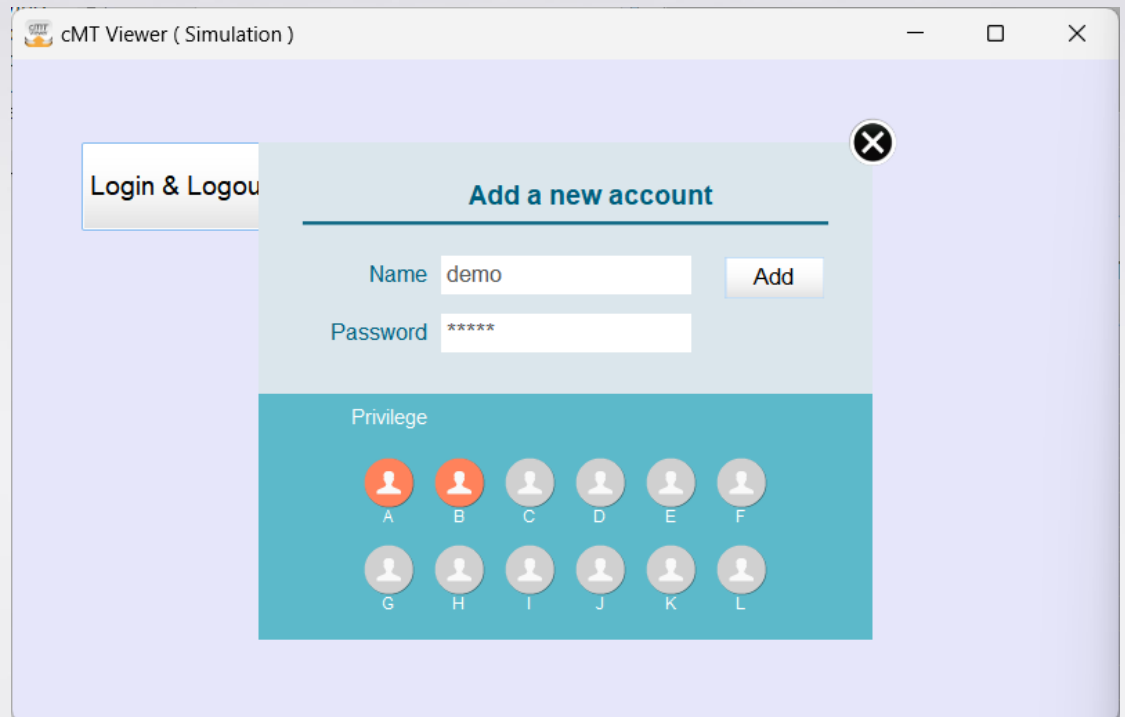


Step 7. Call the Set Password window and try setting a new password for the admin account. Verify that the system blocks setting a password that

does not meet the rules.

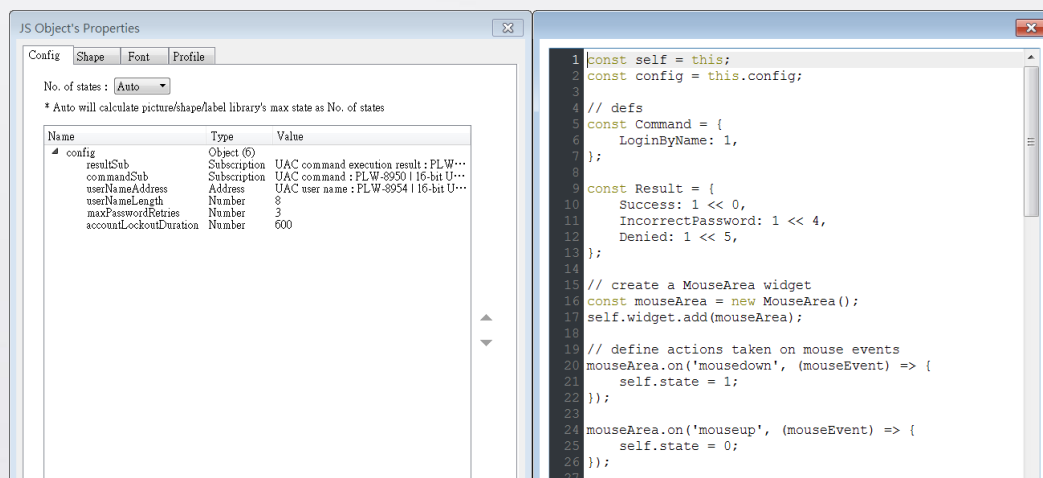


Step 8. Call the Add Account window and try creating a new account. Verify that the system blocks creating a new account if the password does not meet the rules.



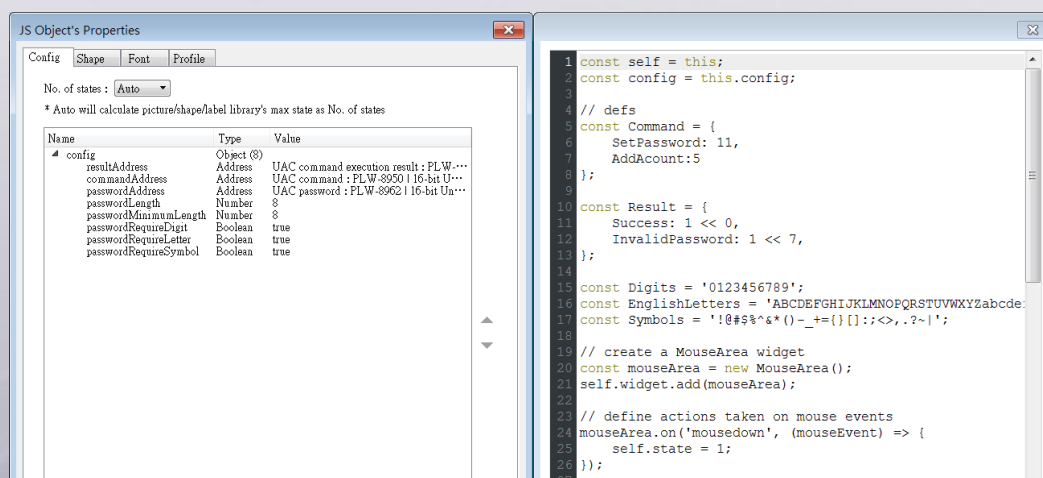
2. Setting up the Screen

Step 1. In Window 70, add and configure a JS object. When the user interacts with this object, it triggers the login process, including reading the username, checking account lockout status, and setting appropriate commands and results.



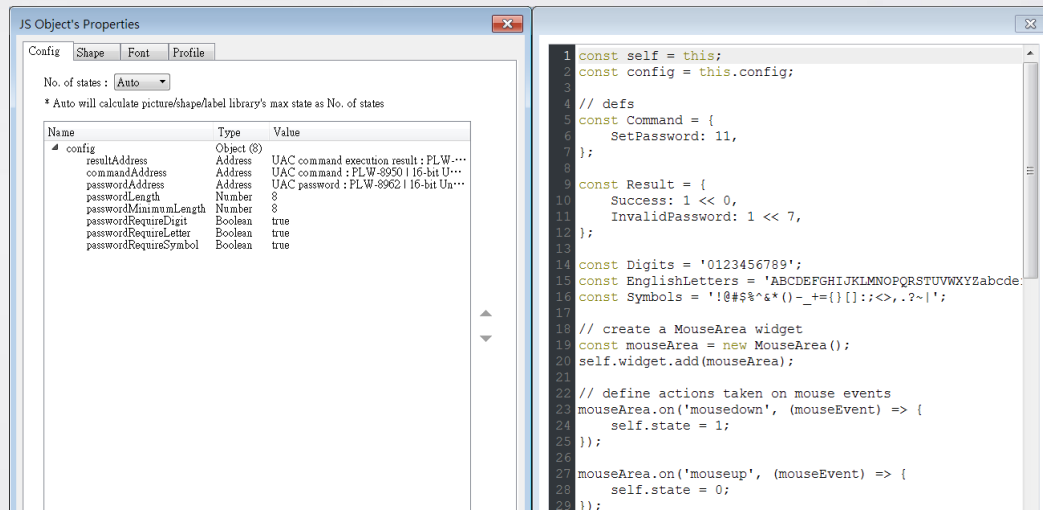
(Only partial commands are shown; see the project file for details.)

Step 2. Go to Window 71, add and configure a JS object. When the user interacts with this object, the system will verify the password's validity and execute the corresponding commands or set the result based on the outcome.



(Only partial commands are shown; see the project file for details.)

Step 3. In Window 74, add and configure a JS object. The click event of this object will trigger the verification process for setting passwords and adding accounts.



(Only partial commands are shown; see the project file for details.)

3. Addresses

The addresses of key objects used in this demonstration are listed below, please adjust as necessary.

Object	Address	Object ID	Description
Window 70			
ASCII	PLW-8954	AE_1	Enters username.
ASCII	PLW-8962	AE_0	Enters password.
JS	PLW-8951 PLW-8950 PLB-8954	CO_0	Checks account lockout status and other processes.
Set Word	PLW-8950	SW_0	Logs out account.
Window 71			
ASCII	PLW-8954	AE_0	Enters username.
ASCII	PLW-8962	AE_1	Enters password.
JS	PLW-8951 PLW-8950 PLB-8962	CO_0	Verifies password validity and adds account.
Set Bit	PLW_Bit-895300~895311	SB_1~SB_12	Sets user permissions.
Window 74			
ASCII	PLW-8954	AE_0	Enters username.
ASCII	PLW-8962	AE_1	Enters password.
JS	PLW-8951 PLW-8950 PLB-8962	CO_0	Verifies password validity and changes password.