

There are 3 interconnected systems in our Model:

- Library1
- Bank1
- User1

The interactions between these components are modeled using channels and broadcasts in the UPPAAL tool. The purpose of this model is to simulate the behavior of the system and verify specific properties related to its functionality. We have created the model keeping in mind the real-life website models in which the user system is analogous to the User Interface, the Library System is analogous to the Server and the bank system takes care of the payments behind the scenes.

## Library System:

The Library system is the central hub for user interactions within the membership issuance process. Upon receiving a "login" channel, the system transitions to the "UserLoggingIn" state, where user credentials are verified. If authenticated, the system enters the "UserAuthenticated" state and activates the "authenticated" channel; otherwise, it signals the "logout" channel. From the "UserAuthenticated" state, users can navigate to "BuySelectedMembership" or "ShowUserMembership" states based on selection from the "view" or "buy" channels. The "back" channel allows users to return to the "UserAuthenticated" state. In the "BuySelectedMembership" state, users may proceed to the "ProcessPayment" state using "next" channel.

In the "ProcessPayment" state, activation of the "banklogin" channel leads to the "PaymentProcessing" state. The variable "amount" is set according to the cost of Gold or Platinum Membership, accessed through the cost array with "Gold\_or\_Platinum\_Membership" index - for which 0th index represents cost of Gold membership, and 1st index represents cost of Platinum Membership. When "paymentprocessed" channel is signaled by the bank system, if successful (boolean "paid" set to True), the system transitions to the "PaymentSuccessful" state, leading to user logout and returning to the Library "Home"; else if boolean paid is set False, then the user goes back to "ProcessPayment" state.

## User System:

The User system initiates in the "MainHome" state, offering the option to log in through the "login" channel. Upon entering credentials in the "SignIn" state, the system authenticates with the Library. If successful ("authenticated" channel activated), the system enters the "SignedIn" state; otherwise, it returns to "MainHome" via the "logout" channel. In the "SignedIn" state, users

can choose to return to the main home, buy a membership ("buy" channel), or view their membership ("view" channel). The "back" channel allows users to revert to the "SignedIn" state from "BuyMembership" or "ViewMembership."

In the "BuyMembership" state, users choose between Gold and Platinum memberships (controlled by the "Gold\_or\_Platinum\_Membership" variable) and move to the "GoToPayment" state. From there, the "next" channel leads to the "BankLogIn" state. After bank log in is completed, "banklogin" channel is activated and user goes to "Payment" state. In the "Payment" state, successful processing and a "paid" boolean set to true lead to user logout and return to the main home. If the payment is unsuccessful, the system returns to the "BankLogIn" state.

Channels such as "login," "logout," "authenticated," "buy," and "view" are connected to the Library system, while "banklogin" and "paymentprocessed" interact with the Bank system.

## Bank System:

The Bank system begins in the "BankHome" state. Activation of the "banklogin" channel by the User transitions to the "BankLoggedIn" state. If the amount is less than or equal to the balance, the system updates the balance, sets the "paid" boolean to 1, and proceeds to the "PaymentSuccess" state. If the amount exceeds the balance, the system enters the "PaymentFail" state and sets the "paid" boolean to 0. From both states, the "paymentprocessed" channel is activated, leading back to the "BankHome" state.

## Assumptions

1. Library authentication is not explicitly shown, as it is not relevant to the required models.
2. Bank Login is assumed to be successful on every try, as otherwise, it would add unnecessary complexity to the model.
3. Bank Login is not explicitly shown, as it is not relevant to the required model.
4. It is assumed that the user can only go back or log out from certain positions, as it is not explicitly mentioned in the instructions.
5. It is assumed that the user is always either logged in or trying to log in, as it is not explicitly mentioned in the instructions.

6. We have not included book issuing in the model, as it was not explicitly mentioned in the instructions.
7. We have assumed the costs of membership and the user's bank balance, as they were not explicitly mentioned in the instructions.
8. We have assumed that the user always attempts to pay using the only allowed method.
9. We have assumed there is only 1 user as it was not explicitly mentioned in the instructions.

The template is completely deadlock-free.

## Safety Properties:

1. Users can only choose between viewing membership and buying membership, as mentioned in the instructions. The same applies to the library.
2. A payment request can either be successful or fail. It is not possible for it to fail and succeed simultaneously.

## Liveness Properties:

1. If the payment is done successfully, eventually, the library and user will be informed about it.
2. If the user is logged into the bank, eventually, the transaction will be processed.