State Machine Security Door Example



In this lab your task is to implement a workflow which will provide the back-end implementation for a secured door system. The goal for the state machine is to track the opening and closing of the door and to deal with alerting the staff if the door does not close due to an obstruction.

Note: Actual door security systems probably don't work this way but it makes a good illustration

The door system consists of the following actors

* + The door hardware with a card key reader
  + The card key
  + The card key encoder machine
  + The human who wants to enter the door

Next we want to consider the states of the door

* + Closed and Locked
  + Closed and Unlocked
  + Open
  + Unknown

Then we want to consider the events that cause the door to move between these states

|  |  |  |
| --- | --- | --- |
| Given | When | Then |
| The door is closed and locked | An unauthorized card is read | The door remains locked, the LED flashes unauthorized |
| The door is closed and locked | An authorized card is read | The door is opened |
| A closed unlocked door | A human opens the door | The door hardware notifies the server that the door is open. |
| An open door | The door is closed | The door hardware sends a closed message to the server |
| An open door | The door open timeout is exceeded | An alert is sent to the security staff to check the door. Maintenance may have to repair the door or replace the door hardware. |

From what we know so far we can draw a State Transition diagram using traditional notation.



Another way to describe the states and transitions would be to fill in the following table which also allows you to specify entry/exit/transition actions and guard conditions.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| State | Enter Actions | Exit Actions | Transition Event | Transition Enabled When | Transition Action | Destination State |
| Initial |  |  | Start Tracking message is received to create workflow |  |  | Closed Locked |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

## Requirements

* + Our workflow must be able to receive messages from door hardware installed at various locations around the building.
  + The workflow must track the status of the door at all times and warn staff if a door is open longer than the timeout.
  + Because it controls door access, the workflow must be highly available
  1. Given these two requirements the Door Control State Machine should be implemented as a Workflow Service hosted with IIS and Windows Server AppFabric.
  + Workflow Services can receive messages from the door hardware using any WCF transport.
  + Workflow Services can scale out to provide high availability
  + Windows Server AppFabric supports durable timers which will allow the workflow to notify staff when door timeouts are exceeded

## Service Operations and Messages

* 1. When planning a Workflow Service you must also consider the shape of the messages. Important concerns to keep in mind are
  + Correlation keys – which identifier you use to uniquely identify the instance of the workflow you want to send a message to.
  + Creating forward compatible versions following the [Data Contract Versioning Best Practices](http://msdn.microsoft.com/en-us/library/ms733832.aspx)
  + Every transition will occur because of a message or a timeout
  1. **Note:** The recommended approach for messages is to create a DataContract type which contains the contents of your message.

|  |  |  |  |
| --- | --- | --- | --- |
| Transition | Operation | Request Message | Response Message |
| Start Tracking | StartTracking | StartTrackingRequest | StartTrackingResponse |
| Card Authorized | AuthorizeCard | AuthorizeCardRequest | AuthorizeCardResponse |
| Door Opened | DoorOpened | DoorOpenedNotification | (none – OneWay) |
| Door Closed | DoorClosed | DoorClosedNotification | (none – OneWay) |
| Manual | SetDoorState | SetDoorStateRequest | SetDoorStateResponse |
| Stop Tracking | StopTracking | StopTrackingRequest | StopTrackingResponse |

### Correlation Key

* 1. Each door to a room is assigned a number. If room 100 has two doors, the key for the first door is 100-1 and the second door is 100-2.
  2. Every message will include the (room)-(door number) value as the correlation key.

# Implementation Notes

* 1. There is no template for StateMachine Workflow Service so I have to start with a Sequential and convert it
  2. Need to enable service metatdata to add a reference
  3. State Machine Workflow Service
  4. Where to drop the first Receive – Entry Action of initial state or transition?
  5. C:\Users\rojacobs\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\ESM6VO6X\MC900389390[1].wmfC:\Users\rojacobs\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\DF10ZJWG\MM900283848[1].gif