

CMPT270 Intersession 2017

Assignment #4 (of 4_5)

Simulation of a DVD Player with Internet Access

This assignment is in two parts (Assignment #4 and assignment #5). You are to design and then build a software simulation of a DVD player that also has WIFI capability and can connect to Netflix and play streaming video (we will keep it to Netflix at the moment). You are given the complete specification for the remote control until. It is modeled in the Analysis Classification model provided to you. Your task will be:

- In assignment #4 to complete the analysis and design of the DVDPlayer class which is the model (in the MV two-tier architecture) so that we can test out its functioning against the remote control before actually building the hardware DVD Player.
- Implement the classes *RemoteControl* and *DVDPlayer* to demonstrate that the remote control can correctly and completely control the DVD player. For output of the DVD player functioning you can either direct status messages to the console, or build a GUI to demonstrate the DVD status. In either case, this is a view and must be totally separate from the DVDPlayer class.

Part 1: Assignment #4.

System Requirements:

The remote control has been chosen. The following descriptions provide what is expected behaviour from the DVD player when each button is pressed:

- HomeBtn: places the DVD player into home state and displays the home screen. If connected to the internet, the connection is terminated. If a DVD is playing, it is stopped.
- InternetBtn: If at home screen and internet service is available, connect to the last internet site visited (ex. Netflix). If there is no last site, go to "list of sites" screen. If no internet service is available, do nothing.
- NavRightBtn: Move the cursor display to the right
- NavLeftBtn: Move the cursor display to the left
- NavUpBtn: Move the cursor display up one line
- NavDownBtn: Move the cursor display down one line
- PlayBtn: if a DVD is inserted and we are not in internet mode, begin playing the DVD. If connected to an internet site, play (or select) the highlighted item (ex. Movie)
- StopBtn: if playing a DVD, stop playing. If streaming an Internet movie stop and return to the movie select screen.
- PauseBtn: Pause playing the DVD or Internet movie
- FFwdBtn: Fast-forward the DVD or Internet movie
- FRevBtn: Fast-reverse the DVD or Internet movie
- SkipSceneFwdBtn: skip one scene forward on the DVD; if an internet movie, play in fast-forward
- SkipSceneRevBtn: skip one scene backward on the DVD; if an internet movie, play in fast-reverse
- EjectBtn: If a DVD is in the drawer, open the drawer.
- CloseDrawerBtn: Close the drawer if open.
- OpenDrawerBtn: Open the drawer if closed.

- AcceptBtn: (Internet only) Accept the highlighted item on the Internet site
- PowerBtn: If DVD is currently ON, turn it OFF. If it is OFF, turn it ON.

Assignment Requirements:

- Create a UML State-Change-Model for the DVDPlayer. You will be implementing from this. The model requires the following:
 - The diagram model (can be hand-drawn and scanned into your doc or you can use a computer-based tool to produce your model diagram). You can start from what I have provided. This model needs to be fully labeled (internal events for each state and labeled transitions between states)
 - A description of each state, event, guard condition and action (behavior method)
 - Identify the valid start and stop states.
- Create a Classification implementation model (class model with all supporting attributes and behavior added) for the DVDPlayer.
 - Include all attributes and their data types.
 - Include a description of the parameters and behaviour of each method.

Remember this model represents exactly what you will be implementing. Since we are focusing on the DVDPlayer class, I really don't care about the design and implementation of the DVDSimulationDisplay class or the RemoteControl class, other than I can choose which button I press; the display correctly shows what the player has (or is) doing, and that there is only one public function in the player called by the remote control. Also there should be only one public function in the display class called by the player.

Marking scheme (80 marks):

State-change-model diagram: 20 marks (correctness and accurate representation using UML modeling symbols)

State-change-model documentation: 20 marks (concise and accurate description of each state, event, guard condition and behaviour requirement)

Classification Implementation model: 20 marks (correct use of the UML symbols and their meanings; accuracy when matched to the specification of the narrative and the state-change-model)

Classification Implementation model documentation: 20 marks (descriptions and details for each class, listing the attributes and behaviour descriptions including their signatures).

Part 2: Assignment #5.

In this part, you implement your model in Java. This part will not be marked according to how well the implementation behaves according to the description of the DVD. It will be marked according to how well you implemented your model from assignment #4. You must:

- Accurately implement your model (behaviors, states, events and relationships)
- Keep the model separate from the view – no view code in the model; no model code in the view
- You may use either a GUI or console for the view (or a combination). For example, you can have a window GUI representing the buttons of the remote, while displaying text on the console advising what the DVD is doing. Do not incorporate input/output into the console if you use a GUI for the remote.
- You may

- Make minor adjustments to your model while implementing: you may discover that you identified attributes that are, in fact, redundant or unnecessary. You may also discover the need for attributes you did not identify in the model. Also methods may be identified in your model that you now find unnecessary, or introduce methods that are not in the model because you discovered they were necessary.
- Any deviations from the model must be well documented (i.e. why you removed the characteristic, or why you introduced a characteristic)

Marking scheme:

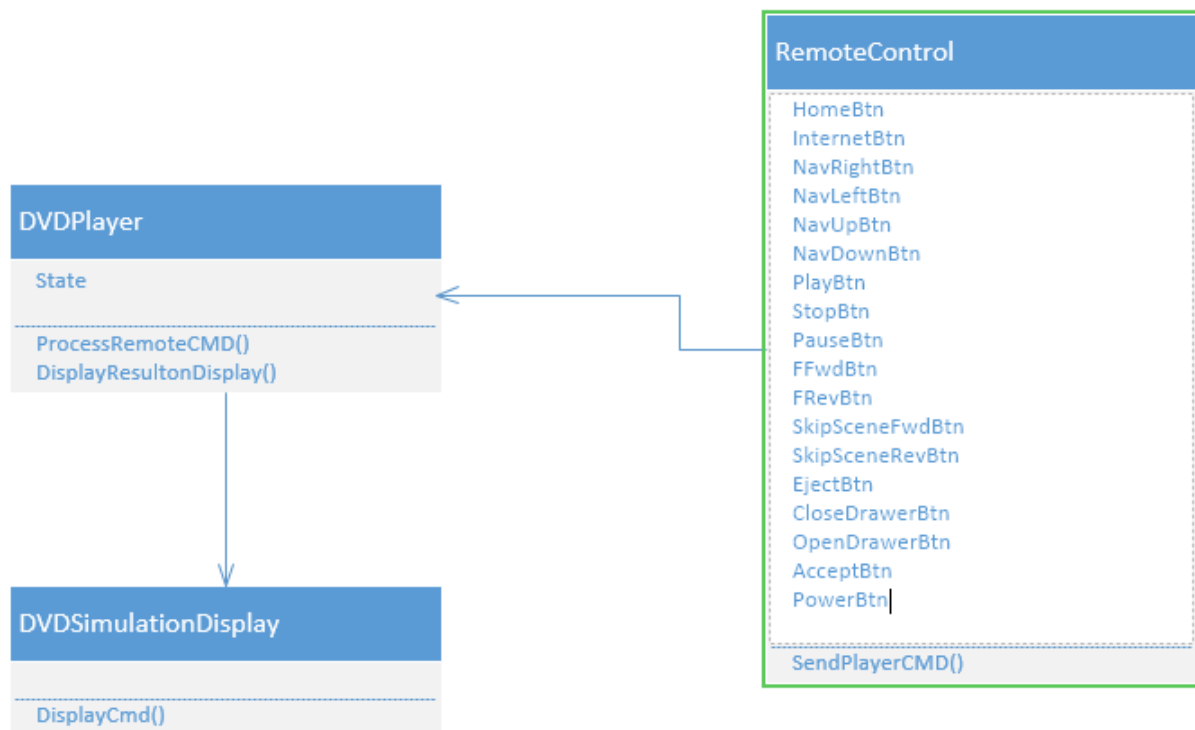
This will be pretty straightforward:

80 – How well you matched your model and correctly implemented the model requirements

20 – Your documentation explains any special implementation decisions, and reasons for adding/removing characteristics.

Enjoy.

Classification Model – Preliminary



This model is your starting point. You can continue from here to complete the model.