Closing the application via Task list

User should be able to close all applications via device's task list. Task list is opened by pressing the Application key for a while. Then, use can select the application to be closed and press C key. This is important from the usability and user experience point of view as applications running in the background may consume a lot of memory and battery.

Being able to close the application via Task list is a requirement in Symbian Signed, check for test case number "Task List (UNI-05)" . Application that cannot be closed via Task list, won't pass the tests and therefore cannot be Symbian Signed.

- Application must not be closed via task list if its showing a dialog. So make sure that no dialog is showing during closing application via task list.

Solution

In order for an application to be closed through the Task list, it should handle the *EEikCmdExit* command either in the application UI or views. Independently of the current state, one of the application's *HandleCommandL()* methods should react to *EEikCmdExit* and call *Exit()* on the application UI.

Note that an application cannot be closed through the Task list if it runs a confirmation dialog on exit, for example:

```
…
if(ConfirmExitL())
 {
 SaveL();
 Exit();
 }
…
```

where *ConfirmExitL()* displays a confirmation dialog. Consider removing the confirmation dialog, or running it only if the application is in the foreground. The above example could be rearranged as follows:

```
…
TBool reallyExit = IsForeground()?ConfirmExitL():ETrue;
if(reallyExit)
 {
 SaveL();
 Exit();
 }
…
```

where *IsForeground()* is provided by the *CAknAppUiBase* class.

It is also important to allow the framework to be able to close the application without user intervention, in cases of low memory for instance.

Enabling app-closed notifications using Symbian C++ details the notification mechanism for the same. The event can also be captured in the HandleWsEventL function of the appui class if required.

```
void CYourAppUi::HandleWsEventL (const TWsEvent &aEvent, CCoeControl *aDestination)
 {
 switch (aEvent.Type())
  {
  case EEventUser:
   {
   //Check if this is a System's call to the app for shut down
   if (( *reinterpret_cast<TApaSystemEvent*>(aEvent.EventData())) ==
```


(C) Copyright Nokia 2014. All rights reserved.

```
EApaSystemEventShutdown)
    {
                // Do something if required
                }
   }
            break;
  default:
   break;
       CAknAppUi::HandleWsEventL(aEvent, aDestination);
       }
 }
```

See also

- How to conform with Symbian Signed criteria
- How to retrieve a list of currently running tasks

(C) Copyright Nokia 2014. All rights reserved.