

70-485: Advanced Windows Store app development using C#

The following tables itemize changes to exam 70-485. These changes will be made on December 2, 2013, to include updates that relate to Windows 8.1 and Visual Studio 2013 tasks.

1. Develop Windows Store apps (increase: 17% to 18%)

Tasks currently measured	Tasks to be added/changed in December 2013
Create background tasks Implement the Windows.applicationmodel.background classes; implement IbackgroundTask interface	No change
Consume background tasks Use timing triggers and system triggers; keep communication channels open; request lock screen access; use the BackgroundTransfer class to finish downloads	No change
Create and consume WinMD components Create a WinMD component in C#; consume a WinMD component; handle WinMD reference types; reference a WinMD component	No change

2. Discover and interact with devices (no change: 16%)

Tasks currently measured	Tasks to be added/changed in December 2013
Capture media with the camera and microphone Use CameraCaptureUI to capture pictures or video; use MediaCapture to capture pictures, video, or audio; configure camera settings; set media formats; handle media capture events	Added subtasks: implement advanced photo capabilities, such as sequence mode, thumbnails, and focus mode
Get data from sensors Determine the availability of a sensor (Windows.Devices.Sensors); add sensor requests to the app manifest; handle sensor events; get sensor properties; determine location via GPS	Added subtask: enable geofencing
Enumerate and discover device capabilities Discover the capabilities of available devices (for example, GPS, accelerometer, near field communication, and camera)	No change
None	New task (section): Implement device access USB; Bluetooth; Human Interface Device (HID); 3D printer support; Point of Service (PoS) devices

3. Program user interaction (decrease: 17% to 16%)

Tasks currently measured	Tasks to be added/changed in December 2013
Implement printing by using contracts and charms Implement the Print contract; create a custom print template; construct a print preview; handle print pagination; implement in-app printing; expose printer settings within your app	No change
Implement Play To by using contracts and charms Register your app for Play To; use PlayToManager to stream media assets; register your app as a PlayToReceiver	Added subtask: programmatically implement Play To functionality
Notify users by using Windows Push Notification Service (WNS) Authenticate with WNS; request, create, and save a notification channel; call and poll the WNS	Added subtask: configure and implement push notifications by using Azure Mobile Services

4. Enhance the user interface (increase: 16% to 17%)

Tasks currently measured	Tasks to be added/changed in December 2013
Design for and implement UI responsiveness Choose an asynchronous strategy for your app; implement the Task Parallel library for multi-processor utilization; convert asynchronous operations to tasks	No change
Create animations and transitions Apply animations from the animation library; create and customize animations and transitions, including XAML transitions; implement storyboards and transformations; utilize built-in animations for controls	No change
Create custom controls Choose the appropriate base control to create a custom control template; style a control through control templates; design the control template to respond to changes in viewstate	Modified subtask: design the control template to respond to changes in windowing modes
Design Windows Store apps for globalization and localization Implement .resw files to translate text; implement collation and grouping to support different reading directions; implement culture-specific formatting for dates and times	No change

5. Manage data and security (increase: 17% to 18%)

Tasks currently measured	Tasks to be added/changed in December 2013
Design and implement data caching Choose which types of items (user data, settings, app data) in an app should be persisted to the cache according to requirements; choose when items are cached; choose where items are cached (Windows Azure, remote storage); select the caching mechanism	Modified subtask: choose where items are cached (Windows Azure, Azure Mobile Services, remote storage) New subtask: store data by using LocalStorage and SessionStorage
Save and retrieve files from the file system Handle file streams; save and retrieve files by using the StorageFile and StorageFolder classes; set file extensions and associations; save and retrieve files by using the file picker classes; compress files to save space; access libraries, including pictures, documents, and videos	Modified subtask: access libraries and KnownFolders, for example, pictures, documents, and videos New subtasks: manage appearance of the file picker; improve searchability by using Windows Index; integrate SkyDrive with apps; compare files; manage libraries; secure application data
Secure app data Encrypt data by using the Windows.Security.Cryptography namespace; enroll and request certificates; encrypt data by using certificates	New subtask: revoke file permissions

6. Prepare for a solution deployment (no change: 17%)

Tasks currently measured	Tasks to be added/changed in December 2013
<p>Design and implement trial functionality in an app Set up a timed trial; set up a feature-based trial; set up in-app purchases; transition an app from trial to full</p>	<p>Revised task name: Design and implement monetization features in an app</p> <p>New subtask: implement in-app purchases</p>
<p>Design for error handling Design the app so that errors and exceptions never reach the user; application class for global collection; handle device capability errors; handle asynchronous errors</p>	<p>No change</p>
<p>Design and implement a test strategy Recommend a functional test plan; implement a coded UI test; recommend a reliability test plan (performance testing, stress testing, scalability testing, duration testing); implement unit testing in an app</p>	<p>New subtask: simulate in-app purchases</p>
<p>Design a diagnostics and monitoring strategy Design profiling, tracing, performance counters, audit trails (events and information), and usage reporting; decide where to log events (local vs. centralized reporting)</p>	<p>No change</p>
<p>None</p>	<p>New task (section): Evaluate and configure for Windows Store deployment Configure app options to submit to store, such as age restrictions, privacy statement, permissions, images, and contact information; create application files, resource files, and application bundles; verify application readiness by using the Windows Application Certification Kit (WACK)</p>