70-482:
Advanced Windows Store app development using HTML5 and JavaScript

The following tables itemize changes to exam 70-482. These changes will be made on November 18, 2013, to include updates that relate to Windows 8.1 and Visual Studio 2013 tasks.

1. Develop Windows Store apps (no change: 17%)

Tasks currently measured	Tasks to be added/changed in November 2013
Create background tasks Implement the Windows.applicationmodel.background classes; implement WebUIBackgroundTaskInstance; create a background task to manage and preserve resources; create a background task to get notifications for an app; register the background task by using the BackgroundTaskBuilder class	New subtask: prioritize tasks by using the Scheduler namespace
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
Integrate WinMD components into a solution Consume a WinMD component in JavaScript; handle WinMD reference types; reference a WinMD component	No change

2. Discover and interact with devices (increase: 13% to 16%)

Tasks currently measured	Tasks to be added/changed in November 2013
Capture media with the camera and microphone Use CameraCaptureUI to take pictures or video, and configure camera settings; use MediaCapture to capture pictures, video, and audio; configure camera settings; set media formats; handle media capture events	New subtask: 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	New subtask: enable geofencing
Enumerate and discover device capabilities Discover the capabilities and properties of available devices, for example, GPS, removable storage, accelerometer, and near field communication	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: 18% to 15%)

Tasks currently measured	Tasks to be added/changed in November 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 an app	No change
Implement Play To by using contracts and charms Register an app for Play To; use PlayToManager stream media assets; register an app as a PlayToReceiver	New subtask: programmatically implement PlayTo 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	New subtask: configure and implement push notifications by using Azure Mobile Services

4. Enhance the user interface (no change: 18%)

Tasks currently measured	Tasks to be added/changed in November 2013
Design for and implement UI responsiveness Choose an asynchronous strategy between web workers and promises; implement web workers; nest and chain promises; make custom functions promise-aware	New subtask: improve interface performance by using the Scheduler namespace
Implement animations and transitions Apply animations from the animation library (WinJS.UI.animation); create and customize animations and transitions by using CSS; apply transformations; create animations by using keypoints; apply timing functions; animate with the HTML5 element	Revised subtask: animate with the HTML5 <canvas> element
Create custom controls Create custom controls using WinJS.Namespace, WinJS.Class.define, and HTML; bind to custom controls with data-win-bind; inherit from and extend an existing WinJS control	No change
Design apps for globalization and localization Implement .resjson files to translate text; implement collation and grouping to support different reading directions; implement culture-specific formatting for dates and times; bind JSON properties to resources by using the data-win-res property	No change

5. Manage data and security (no change: 18%)

Tasks currently measured	Tasks to be added/changed in November 2013
Design and implement data caching Choose which types of items (user data, settings, application data) in your app should be persisted to the cache based on requirements; choose when items are cached; choose where items are cached (Windows Azure, remote storage); select a caching mechanism; store data by using indexDB, LocalStorage, and SessionStorage	Revised subtask: choose where items are cached (Windows Azure, Azure Mobile Services, remote storage)
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 file pickers and the folder picker; compress files to save space; access libraries, for example, pictures, documents, and videos	Revised task name: **Save and retrieve files** Additional subtasks: access libraries and KnownFolders, for example, pictures, documents, and videos; manage appearance of the file picker; improve searchability by using Windows Index; integrate SkyDrive with apps; compare files; manage libraries
Secure application data Encrypt data by using the Windows.Security.Cryptography namespace; enroll and request certificates; encrypt data by using certificates	Additional subtask: revoke file permissions

6. Prepare for a solution deployment (no change: 16%)

Tasks currently measured	Tasks to be added/changed in November 2013
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	Revised task name: **Design and implement monetization features in an app** Subtasks: No change
Design for error handling Design the app so that errors and exceptions never reach the user; handle device capability errors; handle promises errors	No change
Design and implement a test strategy Design a functional test plan; implement a coded UI test; design a reliability test plan, including performance testing, stress testing, scalability testing, and duration testing	Additional subtask: simulate in-app purchases
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)	No change
None	New task (section): **Evaluate and configure for Windows Store deployment** Configure app options to submit to the Windows 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)