Kodu – Windows 8 Project - Estimates

**Priority 1 Features**

# NEW: Water – Simple

**Priority: 1**

Total Man Days: 23.5d-26.5d

* **Estimates**
  + **Code:** 16.5d-19.5d
    - Separate water height map with editing controls similar to Terrain Editor (raise, lower, smooth): 4.5d-5.5d
    - Vector field added/applied to water height map, implementation allowing for separate resolution of vector field: 2.5d
    - Visualization of vector field during edit mode (thousands of tiny arrows need to render): 2.5d
    - Water current paint system (similar to terrain painting, allow current direction and intensity to be painted on vector field): 4.5d-5.5d
    - Incorporate water vector field into movement of waterborne objects (somewhat dependent on how many objects need to interact with the water): 2.5d-3.5d
  + **Art:** 2d
    - UI icons for Water paintbrushes: 1.5d
    - Vector field visualization (Current arrows): 0.5d
  + **Design:** 5d
    - Level Support
* **Estimate Confidence:** Medium
* **Notes**
  + Assumes no new Water sound effects. Waves will need to be disabled for levels that want to use this feature.

# NEW: Current Generator

**Priority: 1**

Total Man Days: 11d (without Air Current Generator and with Visible Fan)

* **Estimates**
  + **Code:** 2d (6.5d-7.5d with Air Current Generator)
    - Water Current Generator Objects (Augment water vector field with a given direction/intensity/radius): 2d
    - Air Current Generator Objects (Completely separate from water system, as the air will not have a vector field applied. Placed objects affect any flying game object within a given radius and applies a wind force in the direction specified): 4.5d-5.5d (CUT)
  + **Art:** 6d (2d with Invisible Fan)
    - Spinning Fan-type Prop (Visible in Gameplay): 6d
      * Concept Art: 1.5d
      * Model: 1d
      * Texture/Shader Setup: 2d
      * Rigging: 0.5d
      * Animations: 0.5d
      * Tile Icon: 0.5d
    - Spinning Fan-type Prop (Invisible – Edit Mode only): 2d
      * Concept Art: 0.25d
      * Model: 0.25d
      * Texture/Shader Setup: 0.5d
      * Rigging: 0.25d
      * Animations: 0.25d
      * Tile Icon: 0.5d
  + **Design:** 3d
    - Design support
  + **Estimate Confidence:** Medium
* **Notes**

# NEW: Water – Gameplay (Tower Defense)

**Priority: 1**

Total Man Days: 77.5d (71.5d if one of the Characters is replaced with simpler Object)

* **Estimates**
  + **Code:** 10d (9d if one of the Characters is replaced with simpler Object)
    - Locked camera angle/single screen: 1d
    - 1 New Character: 2.5d
    - 1 New Character: 2.5d
      * Of if replaced by Object (Example: Barge): 1.5d
    - 1 Barricade Object: 1d
    - 1 Object: 1d
    - Design Support: 2d
  + **Art:** 39.5d (33.5d if one of the Characters is replaced with simpler Object)
    - All Objects include the following:
      * Concept Art
      * Model
      * Texture/Shader Setup
      * Rigging
      * Animations
      * Tile Icon
    - TBD Character (Example: Octopus): 13d
    - TBD Character (Example: Orca): 13d
      * Or if replaced by Object (Example: Barge): 7d
    - TBD Barricade Object (Example: Coral): 4d
    - TBD Object (Example: Floatsam (Floating Trash)): 6d
    - FX: 1d
    - Unique Action Tile support: 1.5d
    - Level Art support: 1d
  + **Design:** 23d
    - Level 1 (incl. Design Spec, Level Building, Kode and Polish): 8d
    - Level 2 (incl. Design Spec, Level Building, Kode and Polish): 7d
    - Level 3 (incl. Design Spec, Level Building, Kode and Polish): 6d
    - Tile Design Spec: 1d
    - New Content Design: 1d
  + **Audio:** 5d
  + **Estimate Confidence:** High
* **Notes**
  + See separate Design Spec (SGI – Kodu – Water Tower Defense.docx) for more details on the Levels.

# NEW: Telemetry (New Features only)

**Priority: 1**

Total Man Days: 2.5d-3.5d

* **Estimates**
  + **Code:** 2.5d-3.5d
  + **Estimate Confidence:** High
* **Notes**
  + The estimates assume that there is an existing backend and we are only integrating the client.

# DLC Support

**Priority: 1**

Total Man Days: 16.5d-19.5d

* **Estimates**
  + **Code:** 15.5d-18.5d
    - Steam integration: 3.5d-4.5d
    - Windows LIVE Marketplace integration: 3.5d-4.5d
    - License check system (works with Steam and Windows LIVE): 3.5d-4.5d
    - License check system UI: 2.5d
    - Upsell: 2.5d
    - Steam distribution package: 2.5d (Assuming that Microsoft will do this)
    - Windows LIVE distribution package: 2.5d (Assuming that Microsoft will do this)
  + **Art:** 1d
  + **Estimate Confidence:** Low
* **Notes**
  + In order to provide estimates with higher confidence, we would have to acquire Steam and Windows LIVE licenses and start looking into specifics. The cost can be lowered by 4d-5d if Microsoft takes on the Distribution packages tasks.

# Windows 8 Integration (incl. Marketplace)

**Priority: 1**

Total Man Days: 4.5d

* **Estimates**
  + **Code:** 3.5d
  + **Art:** 1d
  + **Estimate Confidence:** Medium – Need to learn more about the App Store approval process and learn from other projects going through approval.
* **Notes**
  + No specific code work is necessary to get Kodu running on Windows 8. However, the App Store has a list of certification requirements (http://msdn.microsoft.com/en-us/library/windows/apps/hh694083.aspx) While they aren’t as strict as Xbox 360 TCRs, there will likely be things we need to fix. We may also need guidance from Microsoft as it looks like it is non-standard to get a Windows 7 app listed in the App Store. We also need Art for App Store icons, screens, etc.

# Windows 8 Touch Support - Basic

**Priority: 1**

Total Man Days: 19.5d-20.5d

* **Estimates**
  + **Code:** 13.5d-14.5d
    - Finish 'Pinch & Zoom' (Basic implementation complete): 0.5d
    - Finish 'Pinch & Rotate' (Basic implementation complete): 0.5d
    - Finish 'Swipe <Direction>' (Basic implementation complete): 0.5d
    - Finish 'Tap and Hold' (Basic implementation complete): 0.5d
    - Finish 'Single Tap' (Basic implementation complete): 0.5d
    - Finish Designer support for which gestures gets exposed as tiles and what the gestures functionality is during edit mode: 1d
    - Tuning/Usability tweaks (Left hand vs. Right hand, etc.): 3.5d
    - Implement Double tab: 1.5d
    - Hook up Virtual Keyboard: 1d
    - Implement Virtual Controller: 2.5d-3.5d
    - Move in direction of tap/mouse click (Also needed to support New Levels - Tower Defense): 1.5d
  + **Art:** 4d
  + **Design:** 2d
  + **Estimate Confidence:** Medium
* **Notes**
  + The Virtual Keyboard assumes that we’d hooking into an OS-level virtual keyboard providing that one exists. If this requires Windows 8 API calls, it might cause issues with backwards compatibility with Windows 7. In case we have to build our own, it will require more time for Code and Art.

# Windows 8 Touch Support – New GUI

**Priority: 1**

Total Man Days: 11d

* **Estimates**
  + **Code:** 5d
    - Buttons that can be clicked or modified: 1.5d
    - 1 Configuration (Multi-button On/Off): 1.5d
    - Adding GUI Buttons to the Level property editor: 1d
    - Adding ability to label the buttons and score: 1d
  + **Art:** 4d
  + **Design:** 2d
  + **Estimate Confidence:** High to Medium
* **Notes**
  + We are assuming that we can expand on the fact that the touch version of Kodu already has tiles for touching buttons for (a/b/x/y).  
    Please see the New GUI Design spec (SGI – Kodu – New GUI.docx) for more details.

# NEW: Windows 8 Touch Support – Terrain Editing

**Priority: 1**

Total Man Days: 24d

* **Estimates**
  + **Code:** 11d
    - Heavily dependent on Design and how different the UI needs to appear
    - Includes Iteration time
  + **Art:** 5d
  + **Design:** 8d
    - Paper Design: 3d
    - Iteration & Prototyping: 5d
  + **Estimate Confidence:** Medium
* **Notes**

# Linked Levels

**Priority: 1**

Total Man Days: 8.5d for Simple Implementation, 11d for Medium Implementation, 17d for Full Implementation

* **Estimates**
  + **Code:** 7d for Simple Implementation, 9.5d for Medium Implementation, 14.5d for Full Implementation
    - Simple Implementation: 7d
      * Fix user changes not being saved when loading a new level: 3d
      * Give player option of loading levels that were linked when entering edit mode: 2d
      * Add ‘package’ uploading and downloading: 2d
    - Medium Implementation: +2.5d
      * Add “Score Persistent” toggle to level settings: 1d
      * Preserving score between level loads: 1.5d
    - Full Implementation: +7.5d
      * Persistent XML that gets written to for any persistent objects that linked levels will read from: 3.5d
      * Ability to link levels that will load off of the same persistent XML: 2.5d
      * Adding Persistent flag to actor properties to mark this object as persistent: 1.5d
  + **Art:** 0.5d
  + **Design:** 1d for Simple Implementation, 1d for Medium Implementation, 2d for Full Implementation
  + **Estimate Confidence:** High to Medium – More investigation would be needed for the Full Implementation.
* **Notes**
  + There are a few unanswered questions that might not be able to get answered until we start implementing the feature. Eg., if the user downloads a 3 level package and they make modifications to a single level, a new save is currently created. The goto levels in this newly saved level may not properly reference the other levels in the package unless local copies of them are also made. The user may need to make local copies of the package before they can upload it to share with their modifications. For the Full Implementation, more investigation is needed as it raises a few extra questions, eg. What happens when going to edit mode? All the persistent settings revert back to level settings? If you are playing level 1 and get to level 2 with half health. Entering edit mode and playing from level 2 again, does that revert your health to full? As well, we don’t think that it’s feasible to create tiles for the persistent objects in the current frame work as all the tiles are specified in an XML file to start. If users place a Kodu in level 1 with certain settings, placing it in level 2 should inherit that setting? Another question is if they edit an object in level 2, will that change get reflected in level 1 as well? Or is there going to be a main level where all the persistent objects gets edited and the other levels just inherit from this?  
    Please see the Design Spec (SGI – Kodu – Linked Levels.docx) for more Details.

# NEW: Expose Settings Variable

**Priority: 1**

Total Man Days: 6d-7d

* **Estimates**
  + **Code:** 2.5d-3.5d
  + **Art:** 1.5d
  + **Design:** 2d
    - Design system and Tiles
  + **Estimate Confidence:** Medium
* **Notes**
  + Estimates assume 5 pre-selected settings. If the desire is to have everything dynamically updated based on the Character they are being applied to, the Code estimates will be higher by at least several days.