Work breakdown

The following is a rough breakdown of work needed to create a functional native Wayland toolkit.

	Graphics		
1		Basic drawing support using shared memory	
2		CPU rendering	
3		Unaccelerated VolatileImage support	
4		Accelerated rendering support (VolatileImage)	
5		GraphicsEnvironment /GraphicsDevice/etc, including multiple screen support	
6		HiDPI	
7		Top-level window transparency	
8		Image formats (native 64bits?), HDR / color profiles?	
9		Color blending with transparency support	alpha, pre-multiplied?
10		AlphaComposite support on client side ?	main Porter-Duff rules like CLR, SRC_OVER or more?
	Input		
11		Mouse	Only 3-button mice with vertical scrolling is supported
12		Keyboard	
13		Touch (hi-resolution scrolling, gestures)	
14		Input methods	
	GUI		
15		Frame decorations	Basic title bar decorations, "native" look-and-feel is a separate task
16		Interactive resize/drag/minimize /maximize	
17		Modal and non-modal dialogs	
18		Correct Z-order for complex window hierarchies with modal dialogs	
19		Fullscreen support	~~But can't specify the device yet~~
20		Tooltips, menus, comboboxes, etc	
21		GTK support (making GTKLookAndFeel work)	
22		Splash screen support	
23		toFront/toBack support	toFront can be implemented through an activation token (as if the window just appeared); toBack can be implemented through a synthesized gesture on the title bar (also needs mouse serial). See GTK implementation.
24		AWT components (java.awt.Button, java.awt.Checkbox, etc)	
25		AWT File dialogs/Print dialogs	
	Robot support		
26		For testing	No input support yet
27		Full support in a production environment	Probably not feasible in full
	Misc		
28		Clipboard support	
29		Drag-n-drop support	

30		Taskbar/tray support	

Legend:

- - feature is functional at the basic level (for example, mouse support means 3-button mice, nothing fancy)
- - task not started yet
- - task is being worked on