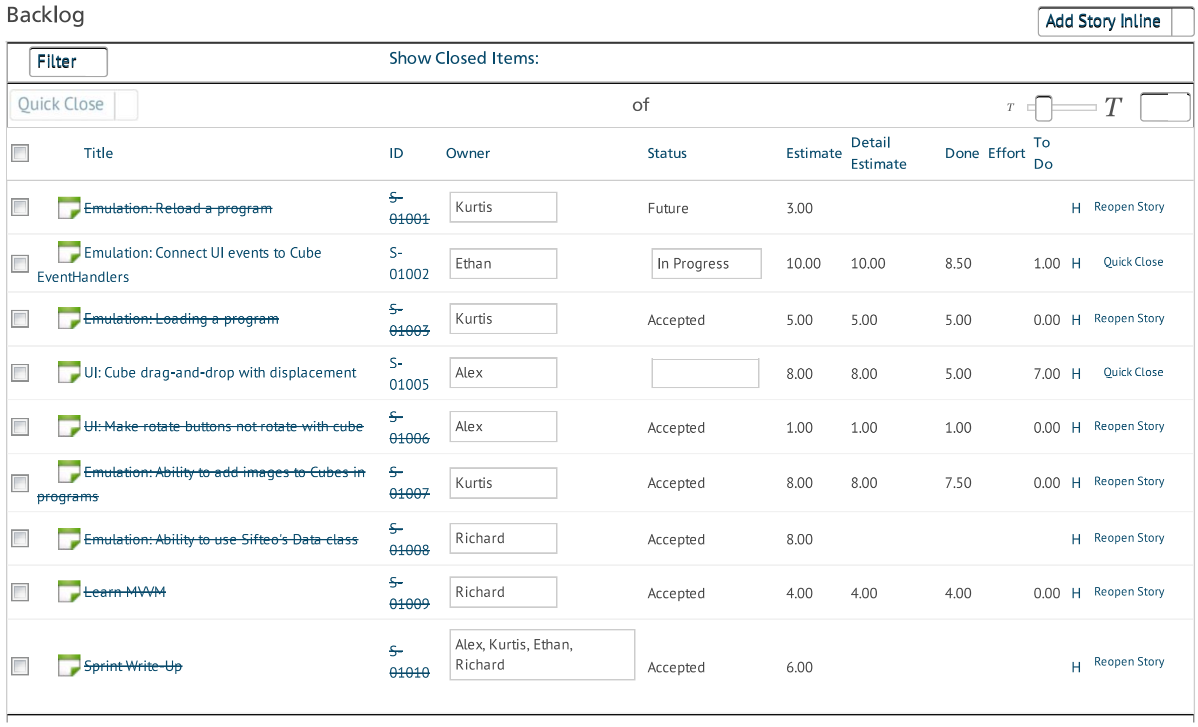
Milestone 3

Singularity Software

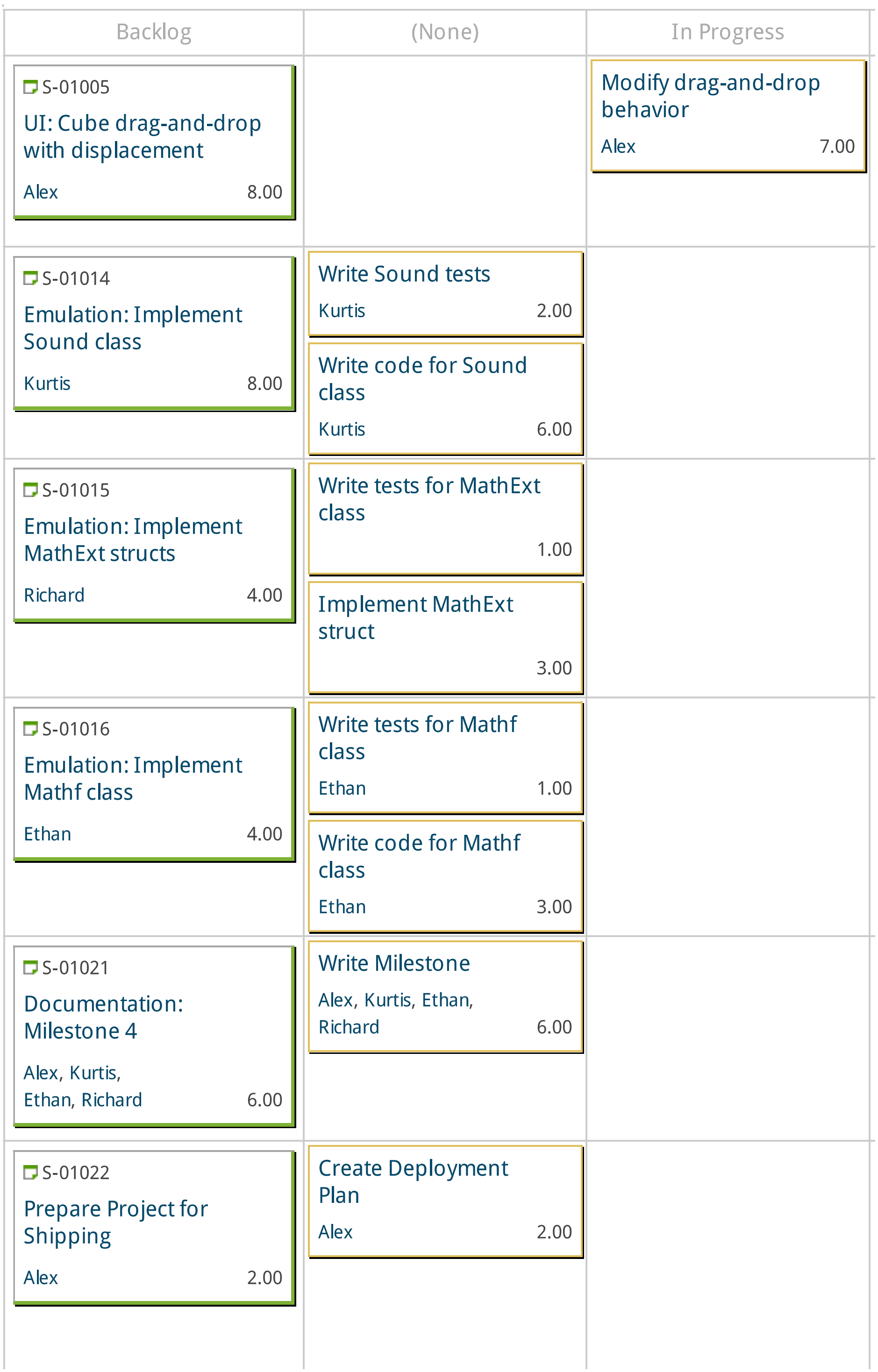
April 11, 2012

Sprint 2 Backlog



☐	Title	ID	Owner	Status	Estimate	Detail Estimate	Done	Effort	To Do		
☐	~~Emulation: Reload a program~~	~~S-01001~~	Kurtis	Future	3.00					H	Reopen Story
☐	Emulation: Connect UI events to Cube EventHandlers	S-01002	Ethan	In Progress	10.00	10.00	8.50		1.00	H	Quick Close
☐	~~Emulation: Loading a program~~	~~S-01003~~	Kurtis	Accepted	5.00	5.00	5.00		0.00	H	Reopen Story
☐	UI: Cube drag-and-drop with displacement	S-01005	Alex		8.00	8.00	5.00		7.00	H	Quick Close
☐	~~UI: Make rotate buttons not rotate with cube~~	~~S-01006~~	Alex	Accepted	1.00	1.00	1.00		0.00	H	Reopen Story
☐	~~Emulation: Ability to add images to Cubes in programs~~	~~S-01007~~	Kurtis	Accepted	8.00	8.00	7.50		0.00	H	Reopen Story
☐	~~Emulation: Ability to use Sifteo's Data class~~	~~S-01008~~	Richard	Accepted	8.00					H	Reopen Story
☐	~~Learn MVVM~~	~~S-01009~~	Richard	Accepted	4.00	4.00	4.00		0.00	H	Reopen Story
☐	~~Sprint Write-Up~~	~~S-01010~~	Alex, Kurtis, Ethan, Richard	Accepted	6.00					H	Reopen Story

Sprint 3 Backlog



Backlog	(None)	In Progress
S-01005 UI: Cube drag-and-drop with displacement Alex 8.00		Modify drag-and-drop behavior Alex 7.00
S-01014 Emulation: Implement Sound class Kurtis 8.00	Write Sound tests Kurtis 2.00 Write code for Sound class Kurtis 6.00	
S-01015 Emulation: Implement MathExt structs Richard 4.00	Write tests for MathExt class 1.00 Implement MathExt struct 3.00	
S-01016 Emulation: Implement Mathf class Ethan 4.00	Write tests for Mathf class Ethan 1.00 Write code for Mathf class Ethan 3.00	
S-01021 Documentation: Milestone 4 Alex, Kurtis, Ethan, Richard 6.00	Write Milestone Alex, Kurtis, Ethan, Richard 6.00	
S-01022 Prepare Project for Shipping Alex 2.00	Create Deployment Plan Alex 2.00	

Test-Driven Development

Framework

We used the Silverlight Unit Test Framework made available by Microsoft at http://silverlight.codeplex.com/releases/view/78435. We chose it because it ios designed by the same people who work on the Silverlight runtime and was therefore easy to integrate into our solution. This easy integration kept the amount of time required for TDD setup low.

Effects on Development

We found that TDD didn't really slow down our development process significantly. Because we're still unfamiliar with many of the intricacies of the Sifteo API, there was and continues to be a lot of time spent simply understanding what each API class does before we start to implement it. In this regard, TDD was very helpful because it forced us to understand each class as we implemented the tests for it. This in turn tended to ensure that we understood each class in small increments instead of struggling to comprehend the entire class all at once.

We found that TDD didn't really slow down our development process significantly. Because we're still unfamiliar with many of the intricacies of the Sifteo API, there was and continues to be a lot of time spent simply understanding what each API class does before we start to implement it. In this regard, TDD was very helpful because it forced us to understand each class as we implemented the tests for it. This in turn tended to ensure that we understood each class in small increments instead of struggling to comprehend the entire class all at once.

TDD didn't really have an opportunity to improve our design decisions because most of the development we're doing at this point directly mirrors the Sifteo API structure. Mainly, the process helped us ensure more complete coverage of the API.