Tools for Iterative Tasks on Mechanical Turk

What is Mechanical Turk?
Amazon's Mechanical Turk (MTurk) is an increasingly popular web service for paying people to do simple human computation tasks. Workers on the system (turkers) are typically paid a few cents for Human Intelligence Tasks (HITs) that can be done in under a minute. MTurk has already been leveraged by industry and academia for image labeling, verifying addresses, and tagging documents. MTurk is quite efficient at many tasks, and can get work done that people won't do for free.

What are Iterative Tasks?
Currently, MTurk is largely used for independent tasks. Task requesters post a group of HITs that can be done in parallel, such as labeling 1000 images. We consider a different model for employing turkers: **Iterative tasks**, in which a succession of turkers do tasks that build each other. For example, turkers can take turns improving a passage of text; verify each other's work by voting on it; and implement the comparison function of an iterative sorting algorithm.

What is TurKit?
TurKit is a toolkit for programming iterative processes on MTurk. The main challenge is making the entire system fault tolerant, so that bugs and system crashes do not lead to wasted money or time. TurKit overcomes this challenge while maintaining a straightforward procedural programming model. In this model, important function calls are performed only once, and their results are stored (memoized) in a database. As a result, a TurKit program can be run repeatedly without redoing previously completed work.

Future Work
For future work, we plan to explore more complicated algorithms using TurKit, such as a parallel sort algorithm that is more robust to human comparison functions that may be noisy or only partially ordered. Also valuable to users of TurKit would be a detailed architectural overview where programmers can understand the program execution log and how TurKit maintains fault tolerance. For example, TurKit should be able to automatically log and maintain a backup of the database, and keep track of which turker's work is currently being compared.