Generating a Proposal Idea

A few initial places to look for an idea:

(1) Think of things related to your current activities—(Super)UROP/UAP, extension of previous project, technical project related to MIT campus or to a student group.

(2) Ask any one of the recitation instructors, especially if they have research interests in your field of expertise.

(3) Consult the 6.UAT Hall of Fame on the website for more inspiration.

A few ideas from Tony to get you started (he’d be happy to supervise a summer/fall UROP if you propose a good solution):

1) Suppose an American is attempting to Google for a keyword in some other language (e.g. a European city or food item). Offer possible corrections for misspellings due to pronunciation of foreign words by considering the International Phonetic Alphabet and what combinatorial options result. (This was mentioned in L2.)

2) A person (who is not a trained actor) who is reciting something from memory often speaks differently from a person who is speaking off the cuff, as in a conversation. Come up with a way to tell when someone is in “recitation”/autopilot mode as opposed to someone who is talking naturally.

3) There are 2 elevators in the Student Center, but you have to call each of them separately. Come up with a mechanism that will call both with the push of one button. (Due to Ishaan Gulrajani, student in Fall 2015.)

4) Lack of volume is a big issue for many 6.UAT students. Come up with a solution to give a student real-time visual feedback on how loud they are at any moment, so that they can immediately adjust their volume level.
A few example ideas from past proposals:

1) Java tutor (by Casey O'Brien—this talk is in the Hall of Fame on the website). The context is that MIT students typically know the Python programming language well, but don't have an easy way of picking up the Java programming language, which is required for one of our CS foundational courses. Her proposal was to build a website that would teach Java to students who know Python, by giving them exercises that involve "translating" Python code to Java code. The website would have a server that could automatically check the students' solutions and provide feedback.

2) Microfluidics (Francis Chen’s proposal talk in Spring 2014). Microfluidic devices are machines that can manipulate tiny volumes of liquid. They have the potential to automate manual lab work by mixing and incubating reagents mechanically. My proposal was to build a prototype microfluidic system, involving both hardware and software, which would be able to execute biological protocols. (Note that this relied heavily on previous work, which made the scope reasonable.)

3) Solar panels (from a student’s talk in Fall 2015). Solar panel arrays can be inefficient—one faulty cell can easily lower the output of an entire array. The student proposed building a solar panel array that could automatically "cut out" a faulty cell, using sensors and switches, to increase output.