HW1， Project Proposal

1. **Description**

Give a couple of sentences – **4 sentences maximum** – describing this project and what it does. If your project solves a specific problem, describe the problem here. This is the first thing that will be read, so make sure that it is clear, concise, and gives the gist of the project.

Reducing a project to four sentences can be hard. If you are having problems, you can write a four-sentence description as follows. The first sentence states the problem. The second states why the problem is a problem. The third is a startling sentence. The fourth states the implication of the startling sentence. As an example, a description for the Piazza website would be:  
*Students need to communicate effectively with each other and with the instructor for active learning. They currently do so through emails which is cumbersome. Piazza is a website that offers an effective means for such communication. Piazza could greatly improve the learning environment.*

**2. Motivation**

**3. Comparison with similar software**

**4. Programming languages, libraries, frameworks, platforms**

List the languages that you think you will be using and justify your choice. You can make this project multi-lingual. In addition, if there is a specific version of the language that you want to use, please state it: e.g. C# 4.0, Java 1.7, etc.

Also list the libraries, frameworks, etc. that you will be using. If you will be building upon some open-source project, list that here too. Put down the platform that your project will run on: Windows, Linux, OS X, web, Android, etc. or all/some of them.

**5. Risks/Challenges**

Enumerate the challenges inherent in your project.

General examples:

learning a new language (e.g., Erlang is not for the faint of heart?)

using a new framework, getting the framework to install, etc. (e.g., are you familiar with Ruby on Rails?)

requiring specialized hardware (e.g., access to big parallel computer)

cross-platform compatibility (e.g., Firefox AND IE?)

new or untested ideas (e.g., research is fun, but risky)

hard to meet requirements (e.g., is it possible to raytrace at 60 frames per second?)

Your proposal should be specific. Also list ways to mitigate the risks, if possible.

**6. User stories and iterations**

Describe *at least* 4 iterations of the project. Each iteration should be 2 weeks. Make sure that each user story describes something that is *well-defined* and *accomplishable*.

*If you are planning on learning a new language or framework, do* ***not*** *just put that down; that is not a sufficient description. Instead, put down goals such as learn Ruby by doing the Sudoku Solver puzzle on* [*http://www.rubyquiz.com/quiz43.html*](http://www.rubyquiz.com/quiz43.html)*. That way you actually have something to deliver. And we can actually tell that you have accomplished something.*

At this point, planning will mostly be guesswork, but being specific (despite the uncertainty) will help you better understand the project.

Iteration 1

| **actual** | **estimated** | **story description** |
| --- | --- | --- |
| **actual** | **estimated** | **story description** |
| 4 units | 2 units | write a proposal |
|  | 4 units | form the team |
|  | 2 units | prepare for [Iteration 1](https://wiki.cites.illinois.edu/wiki/display/cs428sp15/Iteration+1) meeting |

Iteration 2

| **actual** | **estimated** | **story description** |
| --- | --- | --- |
| **actual** | **estimated** | **story description** |
| 8 units | 5 units | user can create account |
|  | 10 units | user can export document to a file |
|  | 5 units | user can set their preferences |

Iteration 3 ...

**7. Meeting schedule**

Describe how many hours you think this project requires for it to be completed by the end of the semester for a team of **about 6-8 students**. Make a schedule of when you plan to meet to fulfill those hours each week. A good rule of thumb is to spend at least 6 hours each week. If you list more availability, you have a higher chance to attract other students to your project.

**8. Skill sets**

Describe the skills that you and your partners will be contributing to the project.

**9. Process**

XP as in class. (This is the default but you can propose changes, e.g., if you don't want pair programming, describe how you will review your code.)

**10. Tools**

* Version Control
* Project Management
* IDE
* …