Project Proposal Template

Student Name(s): Your name(s)
Proposed System Under Test (SUT): Name of the system under test
Link to SUT Source Code: Link to the system under test
SUT Size: Number of lines of code in the system under test

SUT Description
What is the SUT, and what does it do? Did you make the SUT, or is this someone else's code? Are you familiar with the SUT? If the SUT is sufficiently large (at least 10,000 lines per person) and you're unfamiliar with the source code, up to 25% bonus is available - say if you're expecting bonus.

State of SUT
How old is the SUT? Is it being actively developed? Is there documentation available? Is the source code messy?

Attributes
From an ACC perspective, what attributes does the SUT have? Name all relevant attributes, even if your testing doesn't intersect with them. Generally you'll have 3-7, never more than 12.

Components
From an ACC perspective, what components does the SUT have? Name all relevant components, even if your testing doesn't intersect with them. Generally you'll have several; rarely more than 12, never more than 20.

Capabilities:
From an ACC perspective, what capabilities does the SUT have? Each capability should clearly list which attribute and component the capability intersects with. In general, there may be hundreds of capabilities. In this case, list **only** those capabilities you're planning to test with either your unit tests or automated testing techniques. You must list **at least one** capability.

　　　Capabilities which intersect more than one attribute/component should be split into multiple capabilities which each intersect with one attribute and component. Capabilities which intersect with neither indicate one of the following problems:
• An attribute is missing (simply add it)
• A component is missing (simply add it)
• The SUT has a capability which is irrelevant to its attributes or components (it happens in real software!) If this is the case, additionally include why it doesn't relate to your listed attributes or components, why this seemingly-irrelevant feature is in the SUT, and why it's still worth testing.

Unit Tested Capabilities(s):

Which capabilities above will be tested with unit tests? These capabilities must collectively involve at least 1,000 lines of the codebase per person. For example, if you have 2 people on the project, these capabilities need to collectively touch 2,000 lines. It's ok if you're not sure if you hit enough code, or if you hit too much; we can revise this later and add/remove capabilities.

Automatically Tested Capabilities(s):

Which capabilities above will be tested with automated approaches? What kinds of inputs will be needed? How will you know if a bug is found? Try to answer these questions to the best of your ability; it's ok if you're not sure. The only restriction is that your inputs need to be more complex than basic random numbers or strings.