Software Development Life Cycles

CSCI 3300 / CSCI 5300

What SDLC models did we explore on Friday?

Waterfall

- Follow through stages in order: analyze, design, code, test
- Requirements are clear and will not change
- Technology is understood and is not dynamic.
- Later stages often take longer than planned (errors from earlier stages are fixed in later stages)
- When would you use this model?

Big Design Up-Front

- Similar to Waterfall
  - Analysis -> Design
- Different from Waterfall
  - Several iterations of code and Test

Iterative

- Multiple iterations of: analyze, design, code, test
- When would you use this mode:
  - Requirements are almost complete and understood
  - Time to market constraint
  - Resources with needed skills are not available right away (contractors)

Agile

- Adaptive approach:
  - No detailed planning
  - Future tasks are defined in terms of features that will be supported
- Frequent customer interaction:
  - Communication
  - Minimal documentation
  - Close collaboration in agile teams

Spiral

- Budget constraints
- Long term project commitment may change over time
- Customer is not sure of the requirements
- Complex requirements – need evaluation to clarify
- New product line that can be released in phases to get feedback
- Significant changes in the product are expected during development

Which model should we use?

We are developing an "physics calculator" application. This calculator will support a fixed set of built-in equations and will allow for user-defined equations. It will be used by future physics students to check their work. The physics professor who needs this done is available now to work with the team on requirements and is going on sabbatical when the development will take place.

A. Waterfall
B. Big-Design up front
C. Iterative
D. Agile
E. Spiral

We are developing a medical records software, working with a local hospital. We locked in the funding for this project for 6 months, and are uncertain if the funding will continue after 6 months. There are so many details in this project, that we know we don't have all the requirements. The doctors seem to have an idea of what they want, but those ideas are not clearly communicated. It is possible that we may have to hand off the project to another development team at a in 6 months.

A. Waterfall
B. Big-Design up front
C. Iterative
D. Agile
E. Spiral

We work for a mobile application development company. Our clients are usually very involved in the process. We have regular meetings with the clients to review our progress and to demonstrate what we have accomplished so far. It is not unusual for clients to think of new features they want and to change their mind about the requirements. Our customers give us high reviews because they are impressed at how quickly they get to see the first versions of the apps.

A. Waterfall
B. Big-Design up front
C. Iterative
D. Agile
E. Spiral

We are working on an extensions to existing structural design modeling software. The original design is available and there are new features that we need to implement. The features are fairly well understood. Some of the features requires an expert in structural design, so we'll reach out to a local engineering company for an expert. Of course, engineering time is very expensive, so we'll need to be efficient in our use of this resource.

A. Waterfall
B. Big-Design up front
C. Iterative
D. Agile
E. Spiral