

## Information Risk Management – Seminar 3: The Effect of Risk on the SDLC

### Outline:

- Announcements
- Feedback
- Reading & Questions
- Waterfall, Agile and DevOps
- Questions & Next Week

### Announcements:

- Peer Review Forms
- Open Office Session – Of Interest?
- Proposed 07/04/22 19:00 – 20:00

### Feedback:

- Make sure you answer All sections
- Think about word count & most efficient presentation format
- Word Count!
- Referencing
- Spell Checking

### Reading & Questions:

#### Richards list:

- Insufficient resources
- Miscommunication
- Not incorporating risk design or planning
- Inability to change
- Poor management of change

#### My list (group list):

- Inadequate estimation of project time, cost, scope and other resources
- Developing the wrong user functions and properties
- Modules are developed by different programmers
- Too much complex system
- Continually changing requirements

### Waterfall, Agile and DevOps:

- Waterfall: One of the earliest software development lifecycle (SDLC) approaches, it is a practice of developing in stages: Gather and analyse software requirements, design, develop, test and deploy into operations. The output of one stage is required to imitate the next stage.
- Agile: Agile methodology allows for exploration of new ideas and quicker determinations about which of those ideas are viable. Additionally, Agile methods are designed to adapt to changing business needs during development. There are two primary frameworks used in Agile: Scrum and Kanban.
- DevOps: DevOps is an engineering culture that aims to unify development and operations in ways that lead to more efficient development. When done effectively, DevOps practices result in faster, more dependable software releases that are aligned to business operations. It is not a standard or framework, but instead an organisational collaboration that has given rise to set of best practices through continuous integration, continuous delivery and continuous testing.

#### Key differences between Waterfall, Agile and DevOps

##### Waterfall:

- Waterfall is best used on software development projects that are well defined, predictable and unlikely to significantly change.
- This usually applies to simpler, small-scale projects. Its sequential nature makes it largely unresponsive to adjustments, so budgets and delivery timelines will be affected when business requirements change during the development cycle.
- Waterfall projects have a high degree of process definition, little or no variability in output and they do not accommodate feedback during the development cycle.

##### The three main components:

1. Well defined
2. Predictable
3. Unlikely to have significant change.

##### Agile:

- Agile methods are based on iterative, incremental development that rapidly delivers a viable business product. Incremental development breaks the product into smaller pieces, building some of it, assessing and adapting.

- Agile projects do not start with complete upfront definitions; variability through development is expected. And, importantly, continuous feedback is built into the development process.
- Their inherent adaptability allows them to flexibly realign product development, and it's often cheaper to adapt to user feedback based on building something fast than it is to invest in trying to get everything right up front.
- While Agile brings flexibility and speed to development, the practices that really accelerate delivery of scalable, reliable code are in the DevOps realm.

Can Waterfall, Agile and DevOps coexist?

- DevOps practices can be joined in part with Waterfall development. For example, the development team can use tools to automate the build. The siloed, staged nature of Waterfall however, means most DevOps practices are not applicable.
- DevOps culture grew out of Agile and helps to speed time to market. If your team and project use Agile development, you should consider DevOps practices based on:
  - → The size of your project
  - → The skill set of your team members.
  - → Your organisational structure and ability to work to modify that structure and culture as necessary
  - → Whether you operate in the cloud or host locally
  - → IT budget

SDLC

Discussion:

- Waterfall for the COTS solution (off the shelf)
- Agile for the Open-Source solution (could be implemented as well as in cooperation with DevOps).
- In house-solution could benefit from the DevOps approach
- Many companies using the Agile and DevOps approach nowadays, but especially for small companies and shorter project and waterfall approach could be a good choice because of the easiness as well as financial reasons.

Next Week:

- No Seminar but Office Hours available and Open Office?
- Lecture Cast on DR, BC & future trends
- For next Seminar task: Research for disaster recovery and business continuity, how the learned relates to real life situations and thoughts about future of risk.

Assignment questions:

- Assessment of the risk in the status report:
- → Difficulty to quantify risks
- → Creativity is asked
- → Assumptions could be used
- → Case studies can be use
- → All risks should be discussed