ILLINOIS INSTITUTE OF TECHNOLOGY

Transforming Lives. Inventing the Future.

www.iit.edu

SOFTWARE QUALITY MANAGEMENT

CSP587

Prof. Dennis Hood

Computer Science

Reading

- Risk Management
- Reading
  - Ch. 11 – Risk Management
- Objectives
  - Gain an appreciation for the role of risk in decision making
  - Study risk management as an element of both project and software quality management
  - Understand how to anticipate risk and to mitigate exposure
  - Develop an approach for assessing risk exposure and using it to build and execute effective plans

Topics for Discussion

- Explain the relationship between opportunity and risk, and how our perception of them impacts decision making
- Assess the risk of an AI decision making system being trained with a poor set of training data
- Explain how risk assessment is used to justify greater investments in SQM
- Describe risk exposure and risk prevention (mitigation) as costs – use an example to illustrate

Week 12
Risk Management

Anticipating the Future

- Plans are built around expectations of future events – with a good deal of confidence that our actions and efforts can deliver those positive outcomes (opportunities)
- However, past experience tells us that something may disrupt that plan (risks) - and we should take proactive steps to mitigate our exposure
- Some of us are risk-takers, while some of us are more risk-averse … SQMs (and PMs) need to be effective risk managers

Risk Management

- Assessment
  - Identify assets requiring protection and value
  - Identify threats and likelihood of occurence
  - Assess exposure (likelihood X impact)
  - Consider mitigation possibilities and costs
  - Mitigate where feasible
- Life-cycle risk assessment
  - Secondary assessment following system and data architecture decisions

Cost vs. Benefit

- Costs
  - Risk Exposure is an expected cost, the financial impact if the bad thing happens
  - Risk Mitigation has a price tag, the financial cost of the mitigating action (e.g., training, engineering, etc.) and/or things (e.g., tools, time, etc.)
- Benefits
  - Just thinking about risks reduces associated exposure
  - Mitigation reduces exposure, and the magnitude of the reduction is a quantification of the benefit

The Risk Management Process

- Identify risk
- Assess Exposure = Like.(%) x Impact($)
- Identify possible mitigations
- Estimate the reduction in exposure, benefit, and compare it to the mitigation costs
- Implement mitigation if the math supports it
- Study the outcomes for process improvement

Strive for Efficiency

- Risks are predictable and tend to re-occur
  - Makes identification much easier
- So are mitigations
  - Estimating costs, likelihoods, and impacts is also simplified
- Capturing objective data improves future decision making
  - Study to understand the threats, their origins, and the best mitigations for them

Examples of Common Risks

- Under-estimating the challenge … and then over-estimating the team's ability to deliver
  - Good solution: capture history then rely on it
  - Bad solution: double all time and $ estimates
- Working under the pressure of tight deadlines
  - Deliver in iterations with narrow scopes based on feasibility and customer priority
- Change (e.g., scope creep)
  - Incorporate change management
- Inadequate analysis and incomplete testing
  - Rapid prototyping and object orientation
- Risky programming
  - Training, testing, and reuse

Technology Readiness Assessment

- Conducted by an independent review team of SMEs to assess the level of maturity of a technology
- Technology Readiness Levels (NASA)
  1. Basic principles observed and reported
  2. Technology concept and/or application formulated
  3. Analytical and experimental proof of concept
  4. Component validation in a test environment
  5. Component validation in a relevant environment
  6. Model / prototype demonstration in a relevant env.
  7. Demonstration in an operational environment
  8. System completed and qualified via testing and demo
  9. Proof of success through mission operations

AI Risk Management

- An intelligent system is more difficult to predict (impossible?)
- Its environment will change over time … can the designers guarantee that the system will evolve appropriately?
- Recognizing that the system is making poor decisions may be difficult and delayed … diagnosing the root cause and correcting it may be harder and more time consuming