Planning and the Software Lifecycle

CSCE 740 - Lecture 2 - 08/23/2016

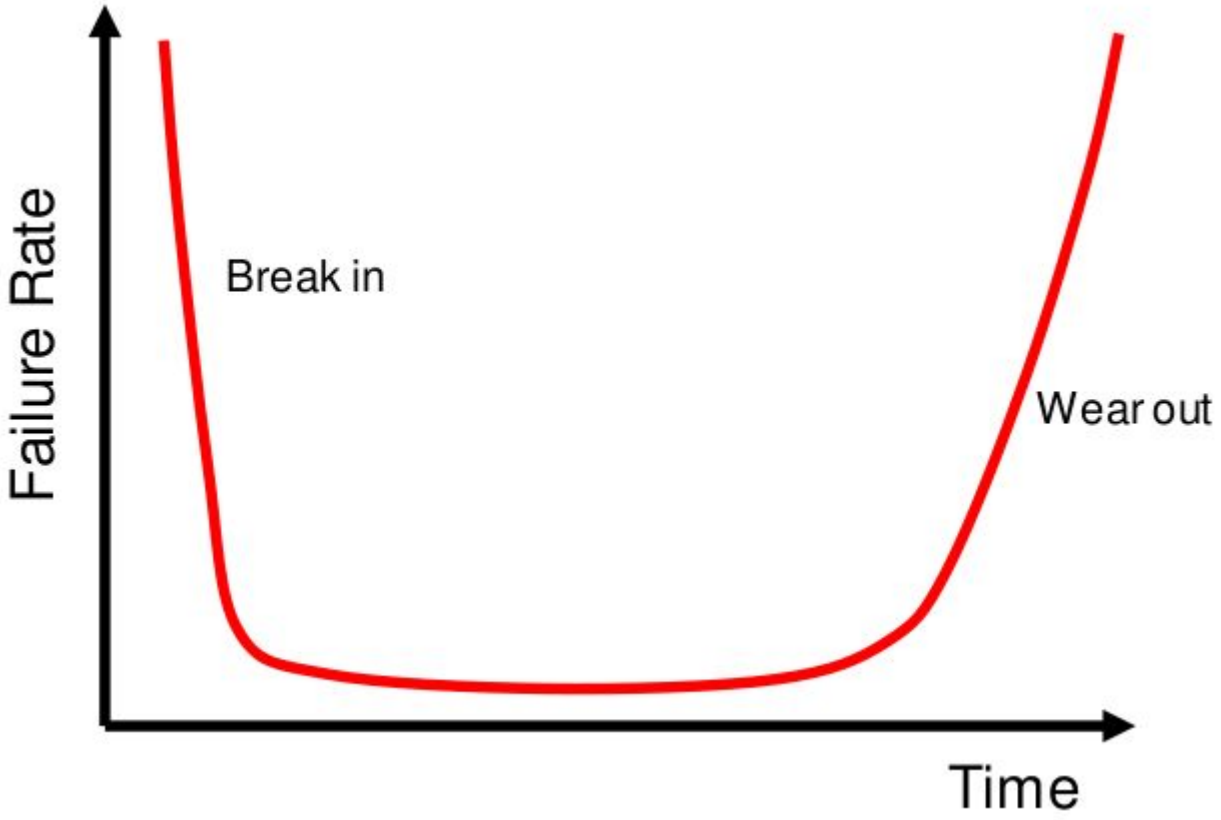
Today's Goals

- Understand risk factors in software development.
- Introduce software development processes
  - Definitions - processes and process models
- Choosing a process
  - AKA: planning and risk management
- Discuss the Waterfall Model

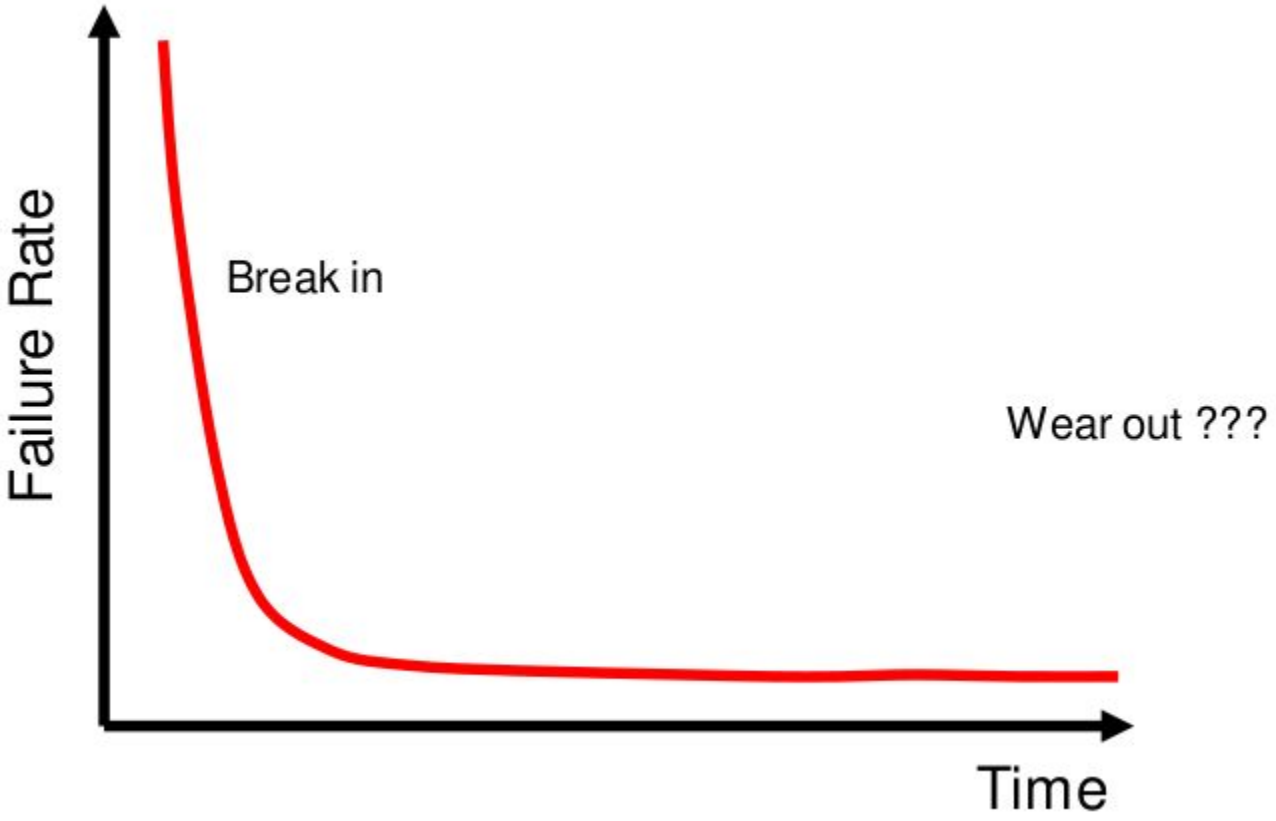
Development is %S##% Hard

- Complexity
  - More complex than a skyscraper.
- Changeability
  - Software is “easy” to change.
- Invisibility
  - We cannot see the progress of development.
- Conformity
  - Software must be molded to fit external constraints.

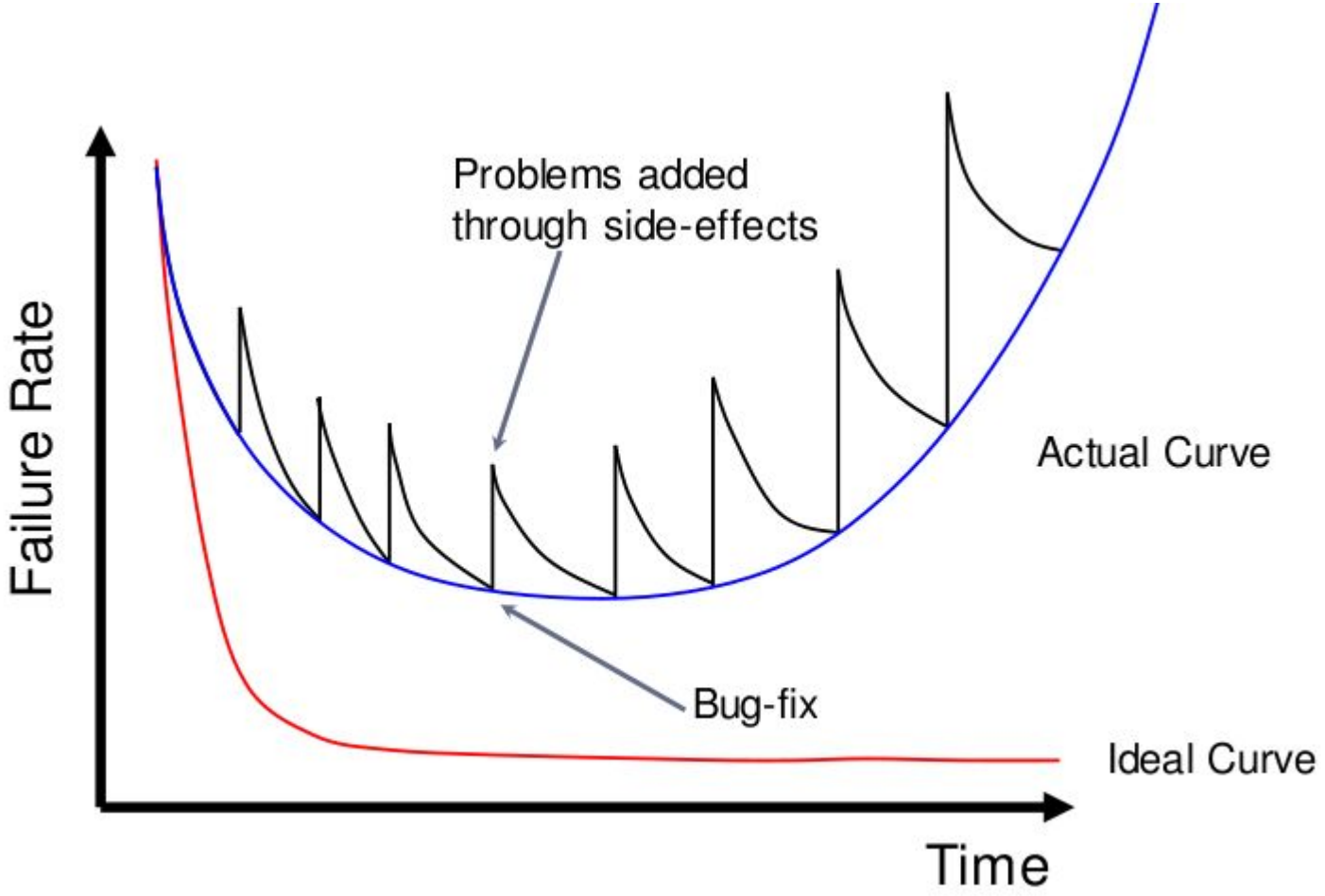
Hardware Failure



Software Failure



Software Failure



Prevailing Paradigm:
Code and Fix

Life Cycle of Software

Any product - software included - has a life cycle: a timeline that can be split into the required phases of existence.

What are the phases of the software lifecycle?

Project Planning

Requirements Definition

Software Design

Implementation

Testing

Release

Operation and Maintenance

The Need for Planning

Why do we get stuck in the code & fix loop?

We know the phases of the lifecycle. We know there are activities that must be performed:

- Specification, Design, Coding, Testing, Evolution

Lack structure and guidance:

- When are we done? When do we move on?
- Activities must be planned and modeled if they are to be managed.

Asking Questions

Development begins by asking questions.

1. Why is the system being developed?
2. What will be done?
3. When will it be accomplished?
4. Who is responsible?
5. Where are they located within the organization?
6. How will the job be completed (technically and managerially)?
7. How much of each resource is needed?

Risk Management

The principle task of a manager is to minimize (avoid or mitigate) risk.

- The “risk” in an activity is a measure of the uncertainty of the outcome of that activity.
  - Risk is related to the amount and quality of available information.
  - What are the risk factors? What will be their impact? How likely are they to arise?

Defining a Process

Process: a flow of events that describes how something works.

- In our case - defines a timeline of activities required to build software.
- Structures who is doing what, when, and how.

Typical Engineered Systems

What do these have in common?

Risk Management

High-risk activities cause schedule and cost overruns.

A visible process provides the means to track, assess, and mitigate risk.

Processes provide quality and predictability by removing risk.

Engineering Process Model

Set of sequential, discrete phases:

- Specification
  - Set out the requirements and constraints.
- Design
  - Produce a paper model of the system.
- Manufacture
  - Build the system.
- Test
  - Check the system meets the specifications.
- Install
  - Deliver the system to the customer.
- Maintain
  - Repair faults over time.

Software Process Models

Why is software different from other engineered products?

- Specifications are often incomplete and vague (complex functionality, intangibility)
- Blurred distinction between specification, design, and manufacture phases.
- No physical realization of the system for testing.
- Software does not wear out.

Choosing a Process - Characteristics

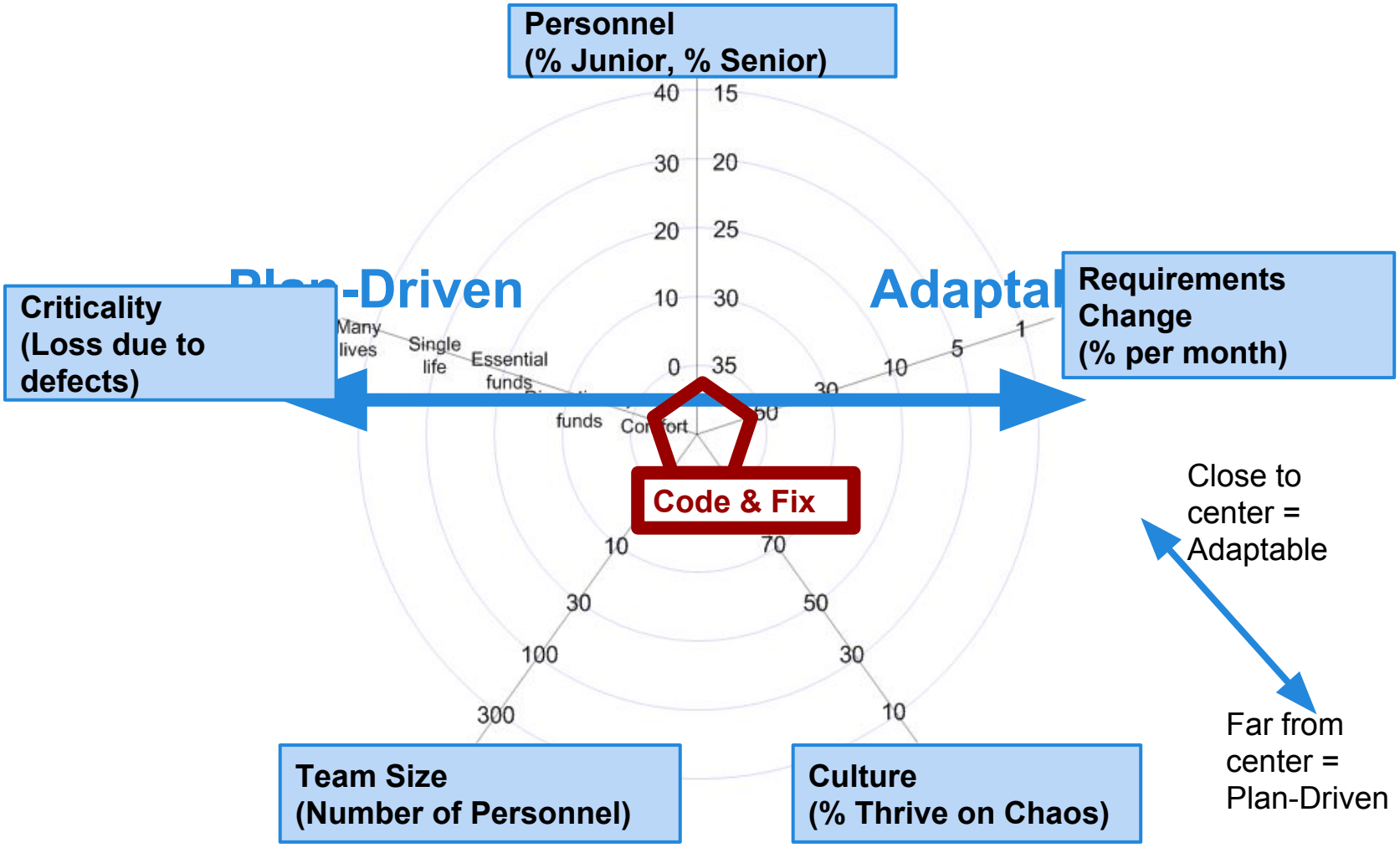
What are characteristics of a good process?

- Understandability
  - Is the process defined and understandable?
- Visibility
  - Is the progress of the process externally visible?
- Supportability
  - Can the process be supported by tools?
- Acceptability
  - Is the process acceptable to those involved in it?

Process Characteristics

- Reliability
  - Are process errors discovered before resulting in product errors?
- Robustness
  - Can the process continue in spite of unexpected problems?
- Maintainability
  - Can the process evolve to meet organizational needs?
- Rapidity
  - How fast can the system be produced?

Considering Developmental Factors



Risk Identification

What are some risk factors to consider?

- Technical Requirements
  - Are the requirements stable?
- Design
  - Does the design depend on unrealistic assumptions?
- Schedule
  - Do we depend on the completion of other projects?
- Budget
  - How reliable are the cost estimates?

Risk Identification

What are some risk factors to consider?

- Quality
  - Are quality considerations built into the design?
- Work Environment
  - Do people cooperate and communicate?
- Staff
  - Are we understaffed? Experience? Contractors?
- Customer
  - Does the customer have realistic expectations?

Understanding Risk

Quantify the impact of risk factors:

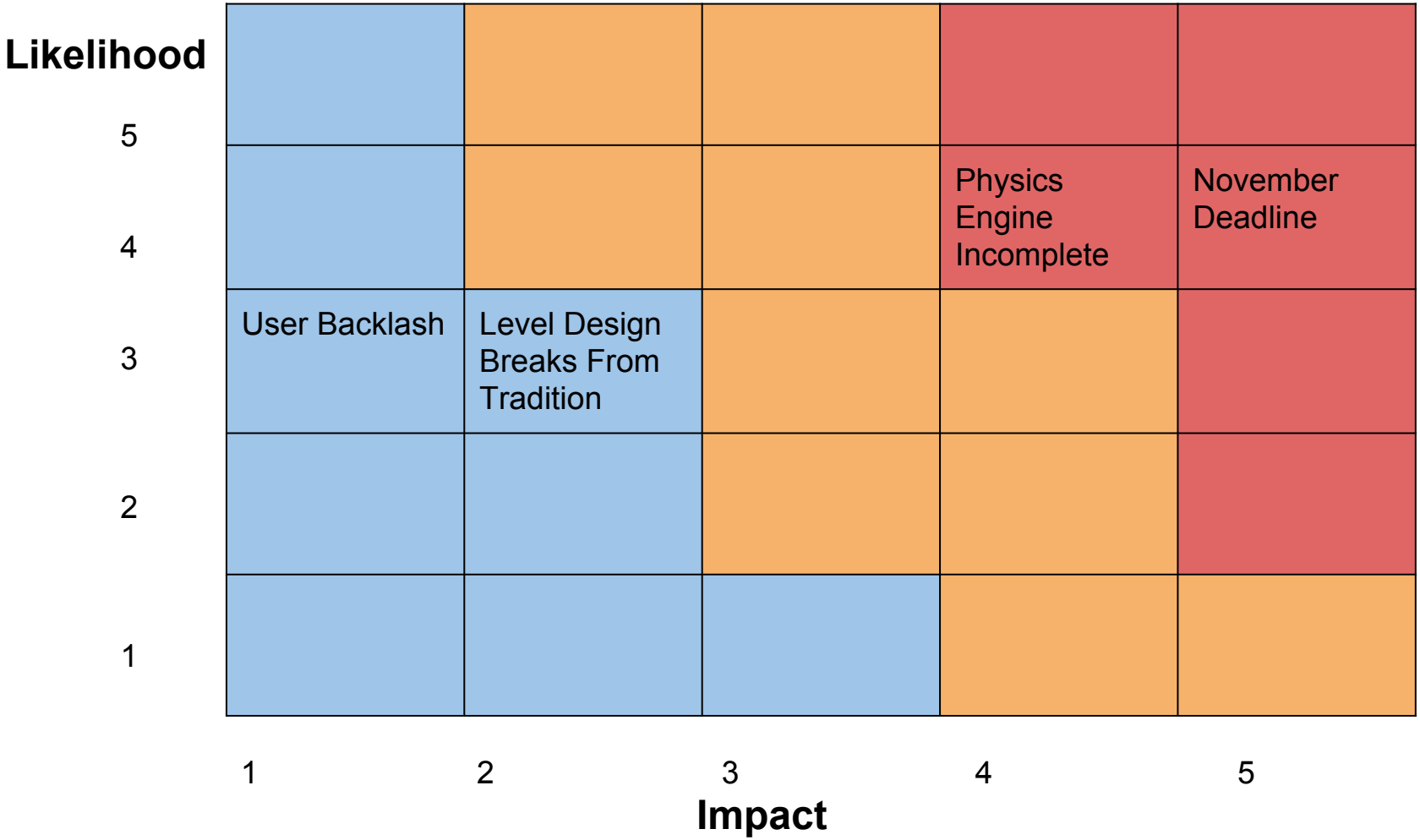
- What are the risk factors?
- When will they occur?
- (Ranked from 1 -Very Low - to 5 - Very High)
  - How likely are they to occur?
  - What is their impact?
  - How difficult are they to detect?

Risk Factor Example

Nintendo has hired us to build Super Mario Bros 26. What are some of the risk factors?

Risk Factor	Likelihood	Impact	Phase
Level Design Breaks From Tradition	3	2	Design
Licensed Physics Engine Must Be Completed	4	4	Implementation
Client Requires November Release	4	5	All pre-release phases
User Backlash	3	1	Post-Release

Risk Severity Matrix



Dealing With Risk

How can you deal with risk?

- Mitigate risk
  - Reduce the likelihood or impact of an event.
- Avoid risk
  - Change the plan to eliminate the risk entirely.
- Transfer risk
  - Employ an external firm on a fixed-price contract.
- Accept risk
  - Take a chance that a risk will not occur.

Choosing a Process

How do we choose a process?

- Consider project characteristics.
- Consider project risks and how they can be mitigated.
- Determine the degree of rigor required.
- Define a task set for each development phase.
  - Task set = {engineering tasks, work products that will be produced, quality assurance, schedule of project milestones}

There is no one-size-fits-all software process.

Code and Fix Model

- Short, interleaved specification and design phases.
- Interleaved implementation, testing, and maintenance phases.

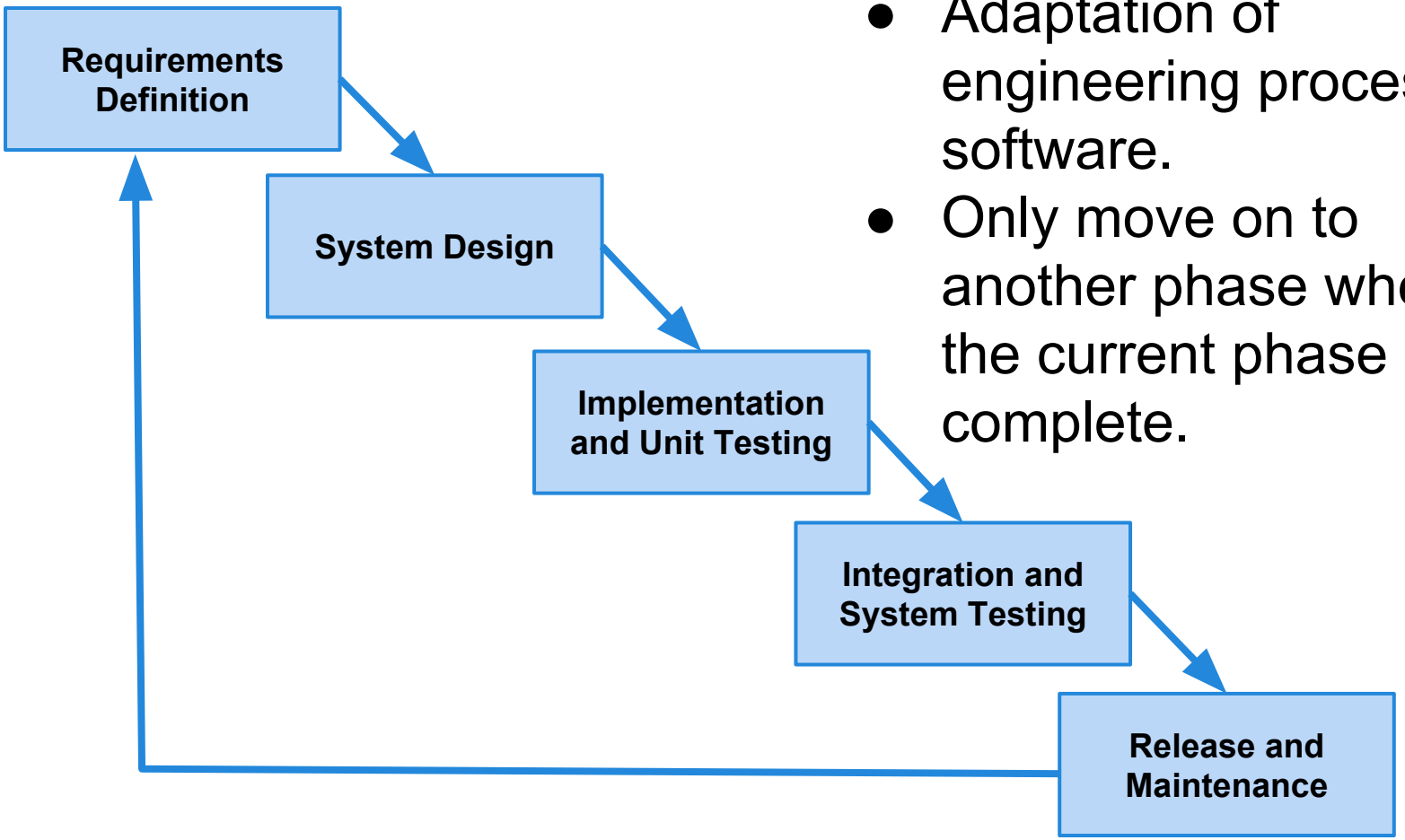
Applicability

- Useful for small, simple projects.
- Allows faster time-to-market.

Potential Problems

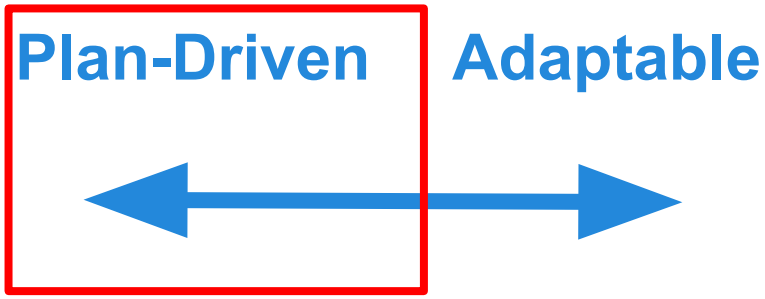
- Quality
- Maintainability

The Waterfall Model



- Adaptation of engineering process to software.
- Only move on to another phase when the current phase is complete.

The Waterfall Model



Benefits of Waterfall?

- Spending more time on earlier phases prevents problems from being discovered later.
- Brings discipline and structure.
- Clear understanding of project progress.
- Places emphasis on documentation.

Waterfall Model Task Set

Phase	Output Document
Product Conception	Feasibility Study
Requirements Definition	Requirements Document
System Specification	Functionality specification, acceptance test plan, draft of user manual
Architectural, Interface, and Detailed Design	Architectural, Interface, and Detailed Design Specifications. System, Integration, and Unit test plans
Coding	Source Code
Unit, Module, Integration, and System Testing	Testing Reports
Acceptance Testing	Final system and documentation

Applicability

What type of projects should use Waterfall?

- Projects where the requirements are stable.
- Projects where impact of failure is severe.
- Projects with high turnover or inexperienced developers.

Problems

What are some problems with Waterfall?

- It is hard to know when you are done with a phase.
- Inflexible model that does not accommodate change.
- Hard to respond to unexpected risk.
- You rarely know all requirements that early.
- Implementation details often emerge only during implementation.

Some Other Process Models

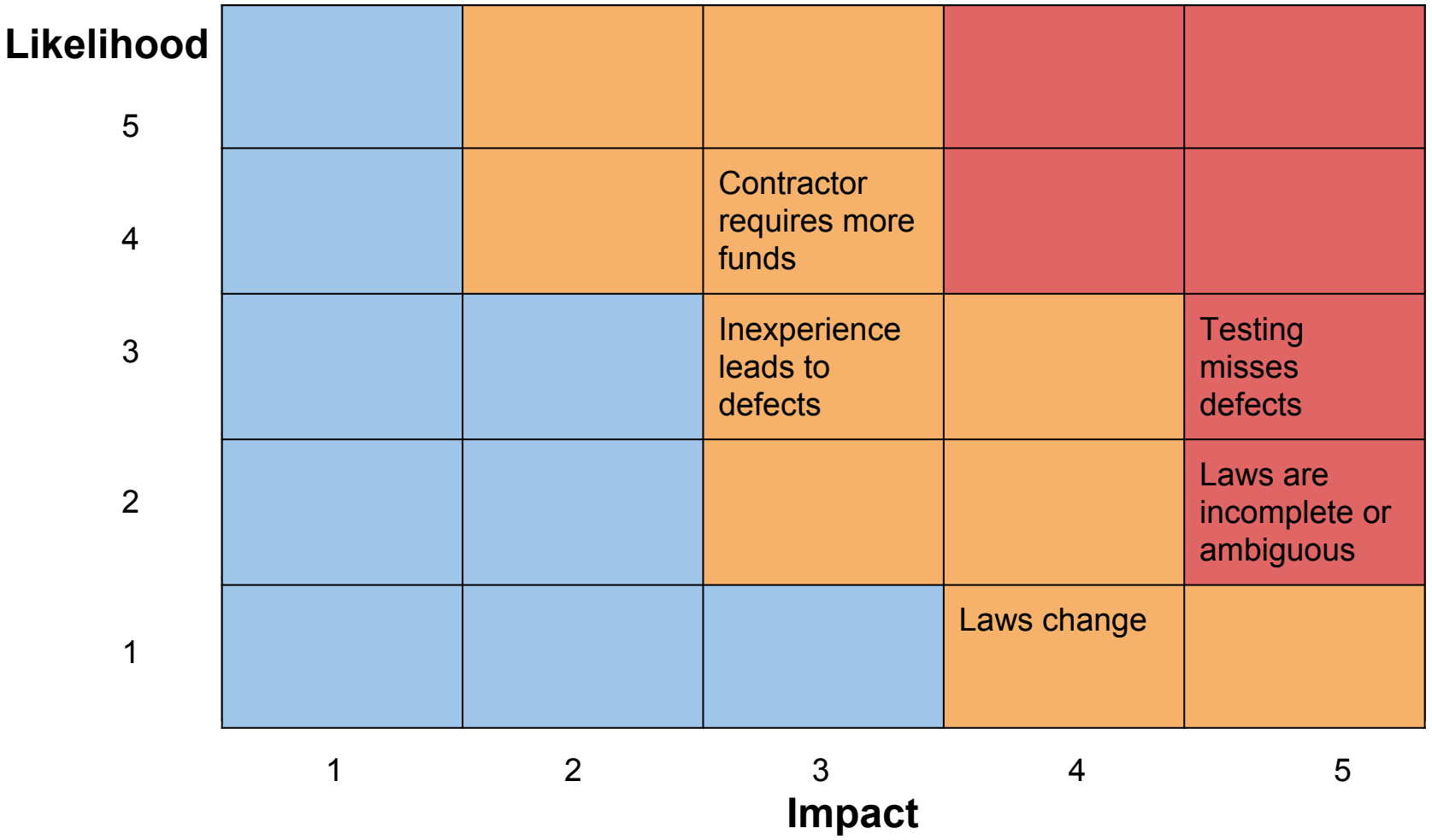
- Evolutionary Development
  - Specification and development are interleaved in evolving series of prototype builds.
- Spiral Model
  - Based on cycles where you identify objectives, alternatives, and constraints.
- Incremental Development
  - Deliver your system in small, planned increments (one working feature per release).

Activity

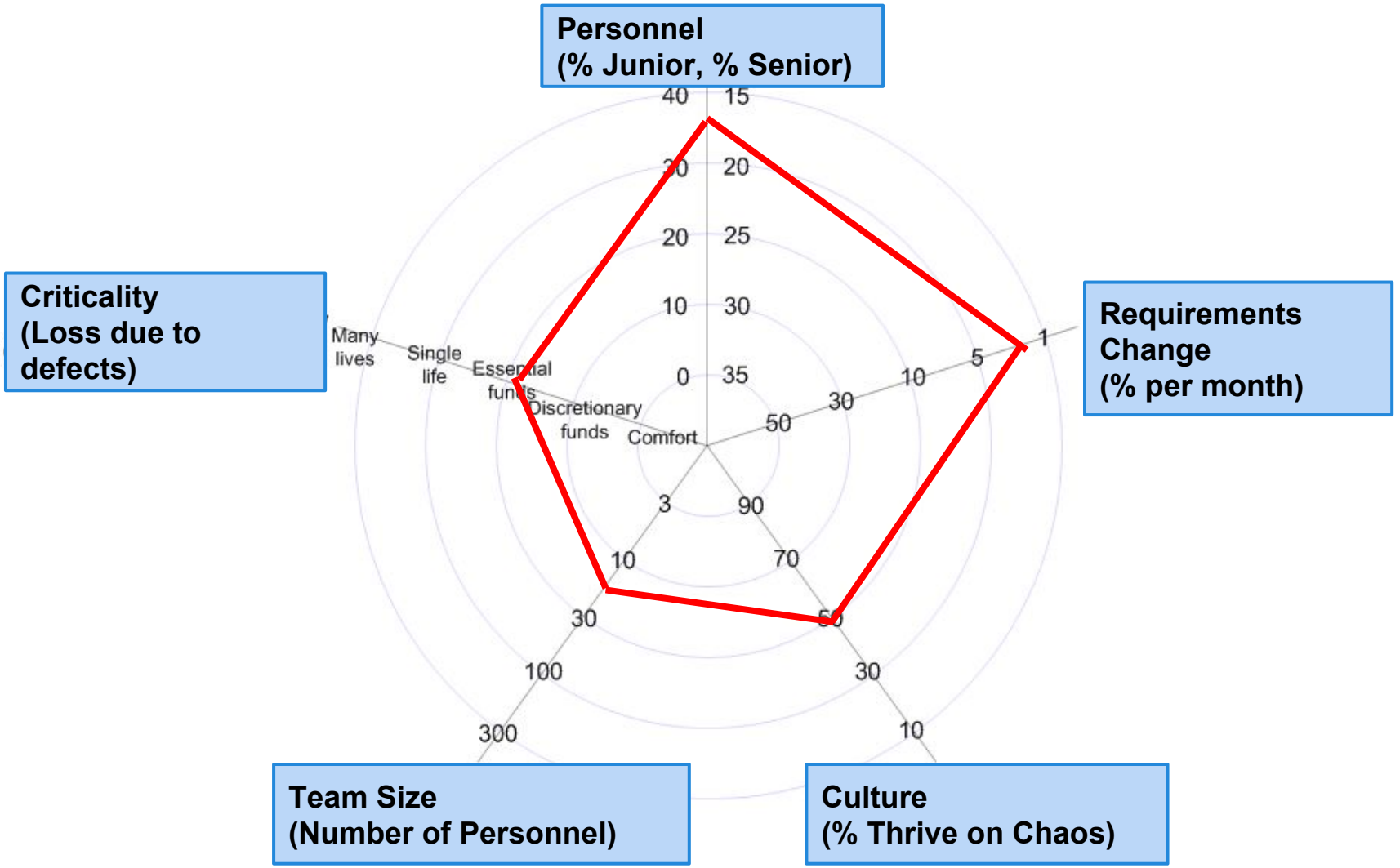
You are involved in the development of a new software product. The product is an insurance application intended to determine what insurance products a potential customer is eligible for. The eligibility requirements are captured in various laws and regulations. You have a team of 15 developers, about 33% of which were newly hired for this project. Half of the team members have experience with projects that required adaptability. Your organization has hired a contractor to perform all levels of testing - the contractor has retained the ability to request additional funds to license additional testing tools.

Your organization has chosen the waterfall model as their development process. The product will be long-lived and good documentation is a must. In addition, the existing laws and regulations were thought to constitute a good start for the requirements of the project.

Activity - Risk Severity Matrix



Activity - Developmental Factors



We Have Learned

- Understanding and mitigating risk factors.
- What is a process?
- How we choose a process:
  - Desired Qualities
  - Organizational Factors
  - Risk Mitigation Potential
- The Waterfall Model
  - Separate and distinct phases of specification and development.

Next Time

- Agile processes
- Plan your team selection!
- Reading:
  - **Paper: Embracing Change with Extreme Programming** (on Moodle)