

# Lecture 22

## *Prioritizing the requirements*

## *Risk management*

Com S/SE 409/509

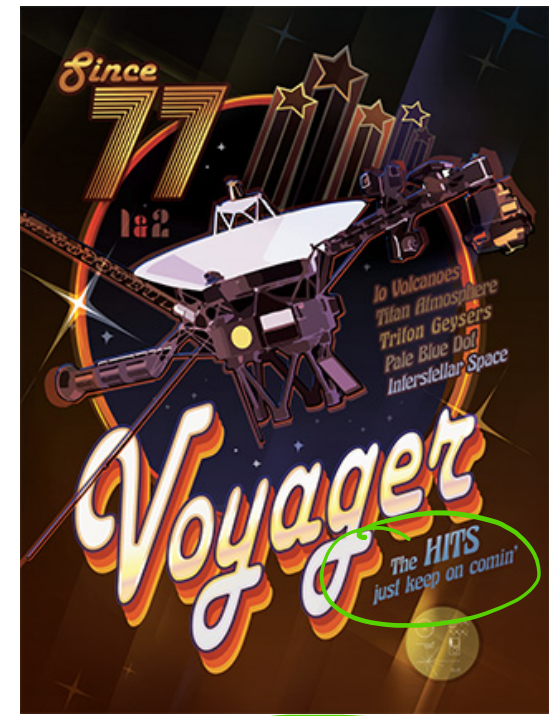
Robyn Lutz

Robyn's Office Hours: **Tues & Thurs, 9:30-10:45**

Wandi's Office Hours: **Mon 10 am. & Wed 7 pm**

Olukorede's Office Hours--**Wed 10**

One more lecture next week: Review & Retrospective  
Exam 2 is scheduled for Sat. 11/21, 9:45-11:45  
More info will be posted in Canvas announcement(s)



<https://voyager.jpl.nasa.gov/downloads/>

## Prioritizing the requirements (Chap. 17)

“One problem with requirements is that there are always too many of them. **Prioritizing** gives you a way to choose **which ones to implement in which versions** of the product.”



### Why prioritize?

- helps manage expectations
- “prepares stakeholders for the fact that you cannot implement all the requirements”
- determines what goes in next version or release

# When to prioritize? What factors to prioritize?

- Early rather than later
  - can assign a priority rating to each use case
  - prioritize as you go along
- Some common prioritization factors
  - ~~Value~~ to customer
  - Value to business organization
  - Ease of implementation
  - Cost to implement
  - Time to implement
  - Legal obligation

# How to prioritize

## 1. Customer satisfaction/dissatisfaction ratings

- How satisfied will you be if we implement this requirement, on a scale of 1-5?
- How dissatisfied will you be if we do not, on a scale of 1-5?

Works better than High/Medium/Low – tendency for all to end up High

## 2. Project triage

(“to sort”)

- 1 • Requirements for next release
- 2 • Requirements not needed/wanted for next release
- 3 • Requirements you’d like if possible.

Agile-backlog

Works better than High/Medium/Low – tendency for all to end up High

## How to prioritize, cont.

### 3. Prioritization Spreadsheet

- calculate a weighted score for each requirement or group of requirements
- column headings: choose 1-4 factors & assign % weight to each (may vote) *or ask clients*
- row headings: requirements or groups of requirements
- score each requirement 1-10 indicating its contribution to this factor
- Ex: Req1 may have score of 2 for “Value to Customer” because customer doesn’t care much; score of 3 for “Minimizing Cost of Implementation” because expensive to develop
- May be useful mechanism for negotiations
- Fig. 17.5 on p. 310 is a larger example

	<u>Factor 1</u>	<u>Factor 2</u> $\leq 4$
<u>Req 1</u>	2	3
<u>Req 2</u>		
<u>...</u>		

## II. Risk assessment

SRS!

“At the moment that you have a complete specification of a product that you intend to build, it’s a good time to pause for a moment & consider the risks involved in proceeding.” – Robertsons

Q: do some requirements contain risk that could affect the success of the project?

# Risks to consider

- Is it a technology that's new to the development team?
- Is client a collaborator or uninterested; are stakeholders hostile or cooperative?
- Are users adequately represented? Are they capable of operating the new product?  
("managers say this is their most frequently encountered risk.")
- Are constraints reasonable? Is product feasible with the schedule & budget?  
("among most common risks cited by projects.")
- Are assumptions reasonable? Should you make contingency plans?  
("Assumptions are really risks.") → *False*
- Is terminology defined so everyone has the same interpretation of requirements?
- Does product's scope include the needed functionality or just the easy stuff?  
➤ Creeping requirements. (see Lecture 21 for solutions!) → *unnecessary?*
- Incomplete (missing) requirements

# Handling risks

- identify & **assess** the risks; keep risks list up-to-date
- **monitor** (“finger on the pulse”) continuously
- raise alert **early** while there’s still time to handle it
- **plan** how to handle risks & then **do it**
  - **mitigate** —
  - **move resources** —
  - **re-prioritize** —
- **learn** from your & others’ mistakes (& successes)

*agile*

*L23: Retrospective*



# THE 10 MOST DEADLY MISTAKES IN SOFTWARE DEVELOPMENT

To err is human, but these mistakes can be fatal. Construx seminars, consulting and resources help you implement the best practices that can prevent your worst nightmares. [www.construx.com](http://www.construx.com)

