MICHIGAN STATE UNIVERSITY

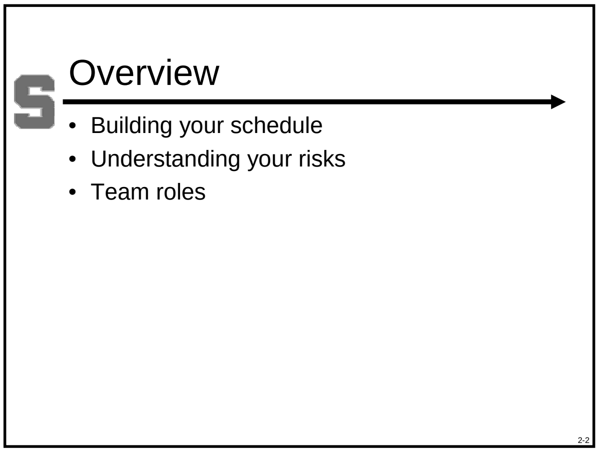
3. Project Schedule and Risk

CSE 498, Collaborative Design

Wayne Dyksen
Brian Loomis

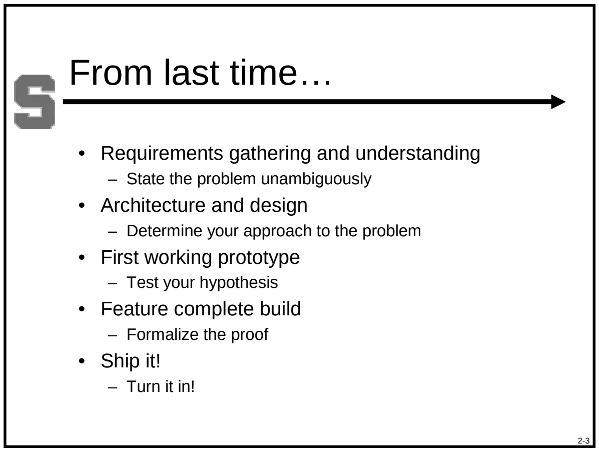
Department of Computer Science and Engineering
Michigan State University

Spring 2005



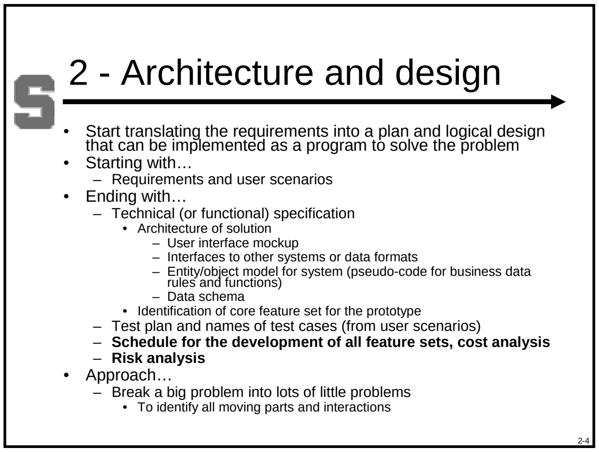
Overview

- Building your schedule
- Understanding your risks
- Team roles



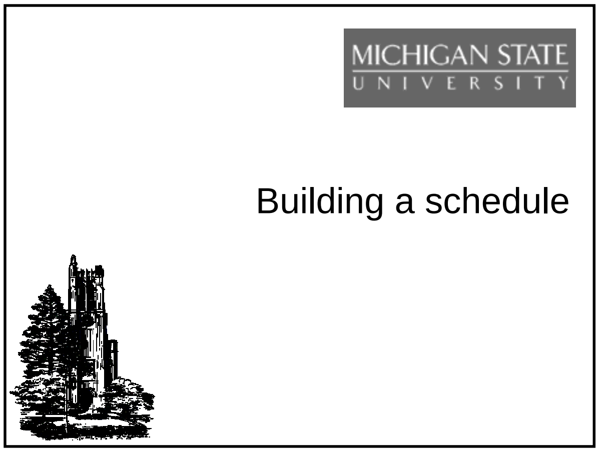
From last time…

- Requirements gathering and understanding
  - State the problem unambiguously
- Architecture and design
  - Determine your approach to the problem
- First working prototype
  - Test your hypothesis
- Feature complete build
  - Formalize the proof
- Ship it!
  - Turn it in!



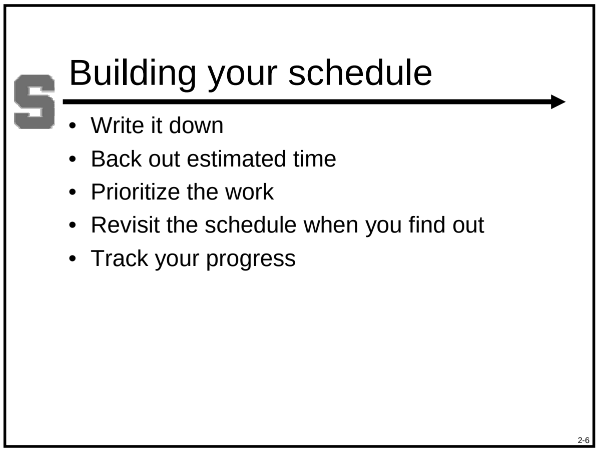
2 - Architecture and design

- Start translating the requirements into a plan and logical design that can be implemented as a program to solve the problem
- Starting with…
  - Requirements and user scenarios
- Ending with…
  - Technical (or functional) specification
    - Architecture of solution
      - User interface mockup
      - Interfaces to other systems or data formats
      - Entity/object model for system (pseudo-code for business data rules and functions)
      - Data schema
    - Identification of core feature set for the prototype
  - Test plan and names of test cases (from user scenarios)
  - **Schedule for the development of all feature sets, cost analysis**
  - **Risk analysis**
- Approach…
  - Break a big problem into lots of little problems
    - To identify all moving parts and interactions



MICHIGAN STATE UNIVERSITY

Building a schedule



Building your schedule

- Write it down
- Back out estimated time
- Prioritize the work
- Revisit the schedule when you find out
- Track your progress

Write it down

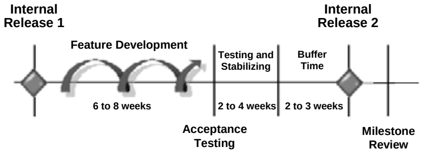
- Customer interaction points
  - What is their schedule?
  - When would be useful to show a prototype, or deliver items?
- Know all the outputs
  - Class due dates (what is due on each?)
  - What goes into each?
- Estimate time
  - Start with a template of what you want to achieve and how you think it will go (be realistic)
  - How long will it take to do X?
  - Tasks only of 2-3 days in duration
  - What has to happen first (dependencies)?
  - Who will do it? (level for skills, duration and role coverage)
  - Put in some buffer time

A living schedule

- Prioritize your builds (plan for multiple major sync points)
  - What is most important to the customer?
  - What has to happen in the first release?
  - What is the highest risk? (Address that first)
- Revisit the schedule regularly (triage meetings)
  - If the customer needs to move a date
  - If a feature set is taking much longer than anticipated
  - If a team member gets sick
- Track your progress
  - Know when you slip, identify what you need to move forward

Internal Cycles

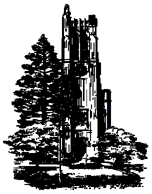
Getting the product to a known state and incrementally building upon it



Exercise

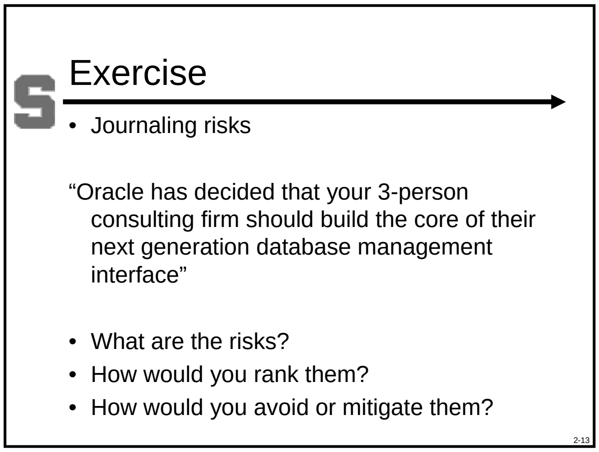
MICHIGAN STATE UNIVERSITY

Risk Management



Risk Management

- What don't you know about the problem?
  - When competitors release their product…
  - When your star developer will get sick…
  - When funding might get cut…
  - When a tool might not work as expected…
- What can you do before it happens, how can you mitigate it after it happens?
  - Risk exposure = probability times impact
- How do you track it?
  - Keep a top-10 list



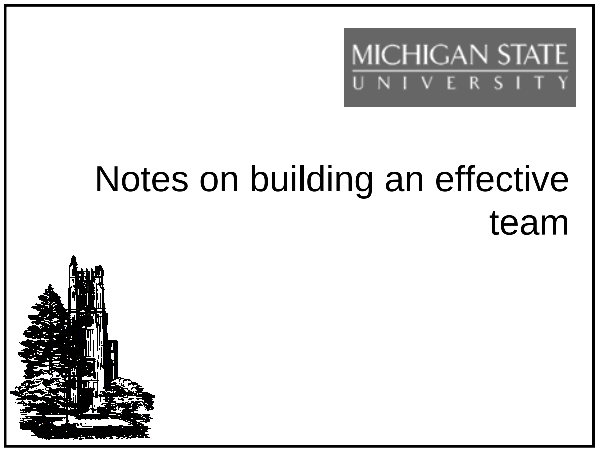
Exercise

- Journaling risks

"Oracle has decided that your 3-person consulting firm should build the core of their next generation database management interface"

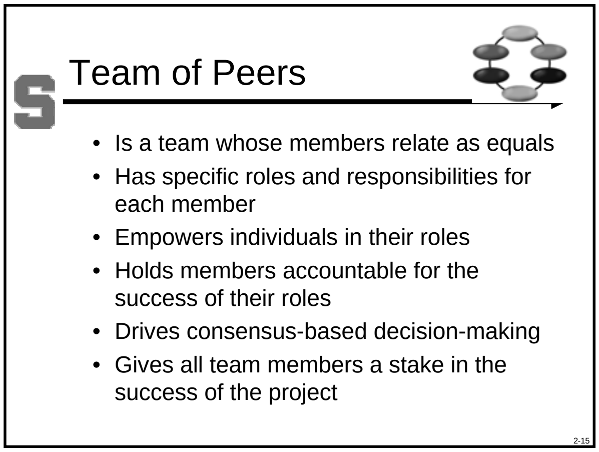
- What are the risks?
- How would you rank them?
- How would you avoid or mitigate them?

2-13



MICHIGAN STATE UNIVERSITY

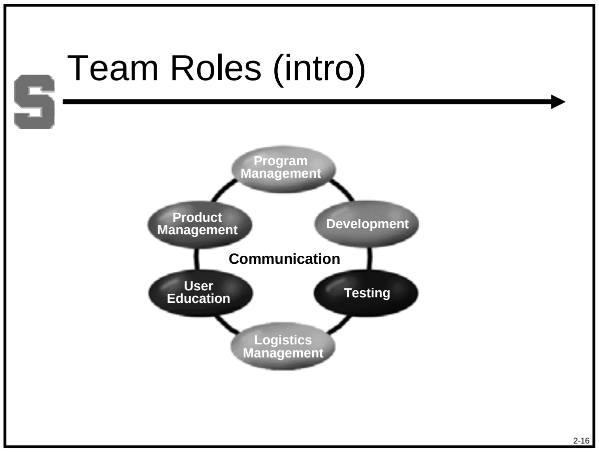
Notes on building an effective team



Team of Peers

- Is a team whose members relate as equals
- Has specific roles and responsibilities for each member
- Empowers individuals in their roles
- Holds members accountable for the success of their roles
- Drives consensus-based decision-making
- Gives all team members a stake in the success of the project

2-15



Team Roles (intro)

Program Management

Development

Testing

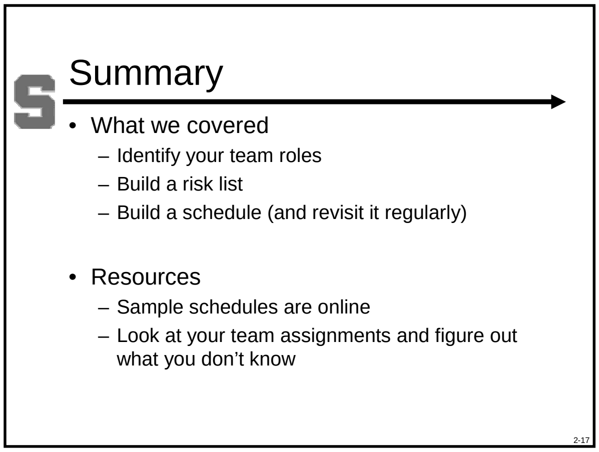
Logistics Management

User Education

Product Management

Communication

2-16



Summary

- What we covered
  - Identify your team roles
  - Build a risk list
  - Build a schedule (and revisit it regularly)
- Resources
  - Sample schedules are online
  - Look at your team assignments and figure out what you don't know

2-17