MICHIGAN STATE UNIVERSITY

3. Project Schedule and Risk

CSE 498, Collaborative Design

Dr. Wayne Dyksen
Department of Computer Science and Engineering
Michigan State University

Spring 2007

Project Schedule and Risk

☒Risk

☐Project Schedule

☐Teamwork

3-2

Identifying Risks

- What You Don't
  - Know
  - Understand
  - Know How to Do
- Normally
  - Major Project Features
  - "Showstoppers"
- Varies From
  - Not Familiar With But (Probably) Can Learn to
  - Absolutely No Idea How to Implement

What are you worried about?

Or, what should you be worried about?

3-3

Example Risks

Including but not limited to...

- Programming Languages
- Development /Programming Environments
- Software Systems
- Hardware Systems
- Key Application Features
- Etc...

3-4

Prioritizing Risks

- Classify Difficulty
  - High — Showstopper, No Idea How to Do
  - Medium
  - Low — Not Hard, Probably Doable
- Classify Importance
  - High — Showstopper, Must Have
  - Medium
  - Low — Not Vital, Nice to Have
- Prioritize
  - High — Work On *Now*
  - Medium
  - Low — Work on Later

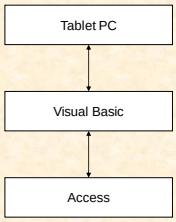
3-5

Case Study: Basketball App

- For Each Player, Track
  - Minutes Played
    - Game Clock Time
    - Consecutive & Total
  - Minutes Rested
    - Wall Clock Time
    - Consecutive
- Must Be Usable
  - On the Bench
  - In Real Time

3-6

Basketball App Architecture



3-7

Basketball App Risks?

- How do I program in VB?
- How do I make a GUI in VB?
- What SDK should I use?
- How do I interface VB with Access?
  - Write Records?
  - Read Records?
  - Traverse Records?
- How do I do clocks?
  - Game Clock?
  - Wall Clock?

How would you classify these difficulties?

3-8

Mitigating Risks

- Use Existing Resources
  - Including But Not Limited To
    - Product Demos
    - Book Sample Code
    - Downloadable Examples
    - Etc...

Nota Bene:
1. Check license if including in project.
2. Document.
3. Inform client.

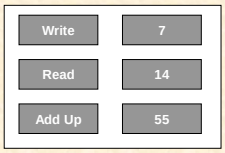
  - Test Drive
    - Install
    - Compile
    - Extend
    - Etc...
- Build New Prototypes
  - Single Purpose
  - Quick-and-Dirty

3-9

Basketball App Prototypes

- Game Clock
  - Start / Stop
  - Counts Down
  - By Minutes/Seconds

- Access Interface
  - Write Number
  - Read Number
  - Add Up Numbers



3-10

Recent Examples

- Women's Fashion Patents Project
  - tiff Images (Bitmaps) of 600+ Patents
  - Wanted
    - Website
    - Search (Find Patent) by Keyword
- Family Medical History Website
  - Enter Family (Tree) Medical History
  - Predict Susceptibility to Diseases
- Hockey Practice Software
  - Web Based
  - Used to Build Hockey Practice Plan

3-11

CSE498 Examples

- Accident Fund
  - Risk: Writing Apps for a SmartPhone
  - Suggestions?
- DaimlerChrysler
  - Risk: Sending Data via Cellular
  - Suggestions?
- GM
  - Risk: Doing Voice Input and Output
  - Suggestions?
- MSU CHM
  - Risk: Reading Barcodes/RFIDs with a Pocket PC
  - Suggestions?

3-12

Project Schedule and Risk

☑Risk

☒Project Schedule

☐Teamwork

3-13

Where do you start?

- Technical Specification
- Prioritized Risks
- Feature Set(s)
- Fixed Milestones
  - Course
  - Client

Tradeoffs... Features vs Time

Are there fixed milestones in the "real" world?

3-14

Course Milestones

Week	Date	Deliverables
4	Jan 29	Technical Specifications
7	Feb 19	Prototype Demo
11-15		Progress Reports & Demos
16	Apr 23	Project Video
16	Apr 25	Everything (Including Documentation)

3-15

Building A Project Schedule

- Start With Fixed Course Milestones
- Identify
  - Tasks
  - Risks
  - Dependencies (Particularly Risk Dependencies)
  - Priorities
- Estimate Times for Tasks
- Assign Task to Team Members
- Use "Short" Deadlines (E.g., 2-3 Days) Why?
- Document and Track
  - E.g., Microsoft Project
  - Project Management = 5% of Team Grade

3-16

Estimating Time for Tasks

- Rough Estimate
  - Intuition
  - Experience
- Refined Estimate
  - Prototype or Partial Build
  - Extrapolation
  - E.g., 2 Days to Build 1 → 6 Days to Build 3
- Keys
  - Be Realistic
  - Include Buffer Time if Unsure
- Adjust Schedule Accordingly

3-17

Typical Build Cycle

Until Project Done Do

1. Divide Next Big Task Into Little Tasks
2. Assign Little Tasks to Team Members
3. Complete Little Tasks
   a. Implement
   b. Test
4. Integrate Little Tasks Into Big Task
5. Test Big Task

Very Important

High Priority Risks Take High Priority

3-18

Revision Control

- Versioning
  - Discrete "Internal" Versions (States)
  - May Correspond to Builds
- Revision Control Systems
  - Check Code In and Out
  - Mark Specific States as Versions
- Motivation
  - Build Breaks System
  - Revert to Earlier Build
  - Avoid Bridge Burning
- Examples
  - Visual SourceSafe
  - GNU RCS (Revision Control System)

Can Be Serious Problem

3-19

Living Schedule

- Schedule Is Dynamic
  - Unforeseen Problems
  - Added Features (Avoid Feature Creep)
  - Etc..
- Track Your Progress
  - E.g., Microsoft Project
  - Project Management = 5% of Team Grade
- Revisit Schedule Often
  - Hold Weekly Triage Meetings
  - Identify Slippage
  - Hold Each Other Accountable (or Contact Matt or me)
  - Set Corrective Action
  - Adjust Schedule

3-20

Project Schedule and Risk

☑Risk

☑Project Schedule

☒Teamwork

3-21

Grading Revisited

- Team (70%)
  - Technical Specification Document 10
  - Prototype Demonstration 10
  - Progress Reports & Demonstrations 15
  - Final Demonstration & Project Video 15
  - Administrator & User Manuals 10
  - Project Management 5
  - Project Web Site 5
  - 70
- Individual (30%)
  - Technical Contribution 10
  - Team Contribution 10
  - Team Evaluation 5
  - Class Meeting Attendance 5
  - 30

3-22

Team Dynamics

- Organize as See Fit
  - Really Hard Stuff
  - Really Important Stuff
- Board of Directors…
  - Hires
    &
  - Fires
- (Be Ready to Discuss During Interviews)

3-23

Team Member Roles

- Client Contact
- Program Manager
- Developer
- Tester
- Etc...

3-24

Team of Peers

Effective Team Members

- Relate as Equals
- Have Specific Roles and Responsibilities
- Empowers Individuals in Their Roles
- Have Specific Skills
- Hold Each Other Accountable
- Drive Consensus-Based Decision-Making
- Give All Members a Stake in the Project

3-25

Potential Problems

Over and/or Under

- Bearing
- Qualified
- Achiever
- Etc…

3-26

Mutual Responsibility

- You are your brother's/sister's keeper.
- Responsible For
  - Your Contribution
    And
  - Your Teammates' Contributions
- What Won't Work
  - "They never asked me to do anything."
  - "They never let me do anything."
  - "He/she never asked to do anything."
  - "He/she never wanted to do anything."
  - Etc...

3-27

Team Evaluation Form

- 5% of Final Grade
- Rate Each Team Member
  - Overall Effort
  - Overall Performance
- Other Questions
  - 8. Describe the contributions of each team member, starting with you. Be specific. Include comments about your/their individual technical contributions as well as your/their contributions to the team as a whole.
  - 9. Whom do you feel did the best (either in effort or overall contribution to the team)? Why? Be specific.
  - 10. Whom do you feel did the worst (either in effort or overall contribution to the team)? Why? Be specific.

3-28

Team Problems

- Can Be
  - Really Hard
  - Awkward
  - Frustrating
  - Etc...
- Addressing Problems
  - ASAP
  - Directly
  - Respectfully
  - Maturely
- Resolving Problems
  - Internally First
  - See Matt and/or Me Now (Don't Wait)
- "Bad" Team Not an Acceptable Excuse

Potential For Bad Effect on 70% of Your Grade

3-29

Project Schedule and Risk

☑Risk

☑Project Schedule

☑Teamwork

3-30