Prototyping

01/27: Prototyping

CSE 498, Collaborative Design
Wayne Dyksen
Department of Computer Science and Engineering
Michigan State University
Spring 2010

Prototypes

- Developed
  - Early
  - Rapidly
- Implement Subset of the Requirements
- Done for Variety of Reasons
- Are Not Finished Goods
- “Hacking” (Good Sense)

Why? Answer Questions

Help Determine
- Specifications
  - Functional
  - Design
  - Technical
- Usability
- How Existing Code Works
- Programming Language(s)
- Development Environment(s)
- Operating Environment(s)
- Etc…

Why? Determine Schedule

Determine how long will it take to…
- Learn the new programming language.
- Learn the development environment.
- Learn the existing code.
- Convert the existing code.
- Convert the existing database.
- Get libraries working.
- Deploy the application.
- Etc…

Why? Reduce Risk

- Operability
  - How do we make a game clock?
  - Where do we store the data?
- Interoperability
  - How does the game clock work with other tablets?
  - How do the tablets all write to the same database?
- Scalability
  - Will the game clock propagate in real time?
  - Will the database engine keep up?
- Reliability
  - What happens if the clock tablet dies?
  - What happens if the database tablet dies?
- Etc…

Speed (to Write)

- Critical
- 2-3 Day Tasks
- Use Whatever Works
  - RAD Languages
  - SDK’s
  - IDE’s
  - Design Tools
  - Wizards
  - Sample Code
  - Etc…
- Stop When Question(s) Answered
Tradeoffs: Speed \(\text{(to Write)}\) vs...
- Testing
- Documentation
- Security
- Software Engineering Best Practices
- Usability
- Performance
- Coding Standards
- User Interface Standards
- Using Real Data
- Etc...

Hence Normally Not Appropriate in Final Deliverable

Challenge/Danger
- “Hack” Solution
  - It works.
  - It’s “a” way to do something.

\[vs\]
- “Correct” Solution
  - It works.
  - It’s the “right” way to do something.

(There may be more than one “right” way to do something.)

Hence Normally Not Appropriate in Final Deliverable

Prototypes: Case Studies
- Basketball
  - Play Effectiveness
  - Player Timer
  - Radio Stats
  - Real Time Play Stats
  - Plus/Minus

Basketball Play Effectiveness
- Coaches Desired
  - Determine Effectiveness of Plays
  - Record All Plays with Result
  - Produce Report of Effectiveness
    - Each Play
      - # of Success / # of Attempts
- I Learned (During First Meeting)
  - Done After Game from DVR
  - Lots of Plays (~200) in Play Book
  - ~60-80 Plays Run Per Game
  - Plays Categorized
    - Early Offense 1,2 (E.g., Fast Breaks)
    - Offense 1,2 (E.g., Half Court Plays)
    - Special Situations 1,2 (E.g., Out of Bounds)
  - Overwhelming

Basketball App Architecture

Basketball Play Effectiveness

- BPE Application
- Visual Basic
- Access
- Windows XP Desktop

Basketball already had all three of these components.

Risks
- Learning Basketball Processes?
- Programming in Visual Basic?
- Access?
- Building a GUI with Access/VB?
- Interfacing VB with Access?
- Generating Reports in Access?
- Etc…
What I Learned From AV1 (1 of 2)…

- Wanted to Identify Plays Within a Possession
- Plays Categorized Series / Set
  - Set is Variation on Series ("Parameterized")
  - E.g.
    - Series: Thumbs
    - Sets: Up, Down, Circle
    - Plays: Thumbs Up, Thumbs Down, Thumbs Circle
  - 1, 2 Notation
    - EO1 = Early Offense Series
    - EO2 = Early Offense Set
- ST (Special Teams) Missing

Nota Bene
- Just Screen Layout
- No Code (Underneath)
- Would NOT Have Entries in All Fields

Huge Impact On Design

What I Learned From AV1 (2 of 2)…

- Results Coded
  - XV Missed Pointer (X1, X2, X3)
  - ON Made 1 Pointer (O1, O2, O3)
  - FF Foul on the Floor
  - TO Time Out
  - Etc…
- Wanted to Record Notes on Defense
- Didn’t Care About Player Times

What I Learned From AV2…

- Wanted to Grade Effectiveness of Plays
- Wanted to Record Player Steals and Assists (Remember this…)
- Needed to Navigate Plays and Possessions

What I Learned From AV3…

BB Stats AV1
Fields
- P# Play Number
- T Time
- C# Clip Number
- EO Early Offense
- O Offense
- SS Special Situations
- R Result

Nota Bene
- Just Screen Layout
- No Code (Underneath)
- Never Have All Entries Filled at Once

BB Stats AV2
Fields
- PO# Possession Number
- PL# Play Number
- SS Special Situations
- DF Defense

Nota Bene
- Just Screen Layout
- No Code (Underneath)
- Would NOT Have Entries in All Fields

BB Stats AV3

What I Learned From AV3…

- Wanted Grades to Be A, B, C, D, F
- Wanted Results to Be X1, O1, X2, O2,…
- Wanted Results Associated With Players
- Wanted Series/Set Combined
- Wanted to Record Player Rebound
- Did NOT Want to Record Player Steals and Assists

What I Learned From Beta 1…

- Entering a Play
  - Some Things Calculated Automatically
    - Play/Possession Number
    - Score
  - Most Things Entered Via Pull-Down Menus
    - Series / Set
    - Result
  - But time Entered Manually (On Keyboard)
- Need Mouse-Only Input
- Need Easy Way to Adjust Clock

Player Timer

- 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
  - Portable and Not Require Electrical Outlet
  - Feel Like a Pen and a Clipboard
Basketball App Architecture

Player Time

- Player Timer Application
- Visual Basic
- Access
- Windows XP Tablet PC

Player Timer Prototypes

- Game Clock
  - Start / Stop
  - Counts Down
  - By Minutes/Seconds
- Access Interface
  - Write Number
  - Read Number
  - Add Up Numbers

Player Timer Huge Mistake

Knew Exactly What They Wanted, So…

- Designed “Final” Version
  - User Interface
  - Data Base Schema
  - Etc…
- Coded “Final” Version
- Lab Tested “Final” Version
- Field Tested “Final” Version
  - At a Scrimmage
  - Totally Unusable
- Scrapped “Final” Version

Software Updates

- Enable Clock Adjustments (While Clock Stopped)
- Enable Check In/Out By Touching
  - Check In/Out Button
  - Player Name
  - Player Slot
- Allow > 5 Players Checked In (While Clock Stopped)
- Enable Pending Check In (While Clock Running)
- Eliminate Almost All Modal Dialog Boxes
Prototyping

Your Prototypes

• What?
• Why?
• How?
• When?
• Where?

What’s next?

Project Plan Presentations
• 02/01, 02/03, 02/08, 02/10
• Template On Web
• Schedule Announced on Sunday, 01/31 (If Conflicts, Say So Now)
• Everybody Turns in PowerPoint and Document By 3:00pm on Monday, 02/01
• Dress is business casual.
• “Formal” Team Pictures Right After Meeting

Karin Petty