SSW-555: Agile Methods for Software Development

Software Project Challenges
Week 1

To Start with A Survey…

A. Software Development > Coding

B. Software Development = Coding

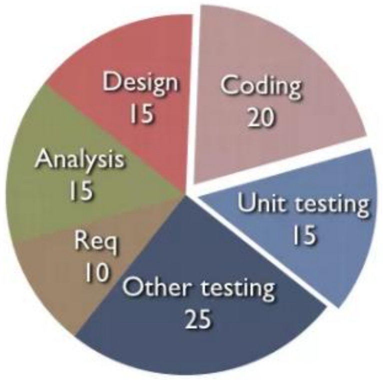
Software Development > Coding

- Programming (coding) of software is an important part of any software effort.

- 100% effort
- 80% effort
- 50% effort
- 25% effort
- 15% effort

Software Development > Coding

- Programming (coding) of software is an important part of any software effort.



- 100% effort
- 80% effort
- 50% effort
- **25% effort**
- **15% effort**

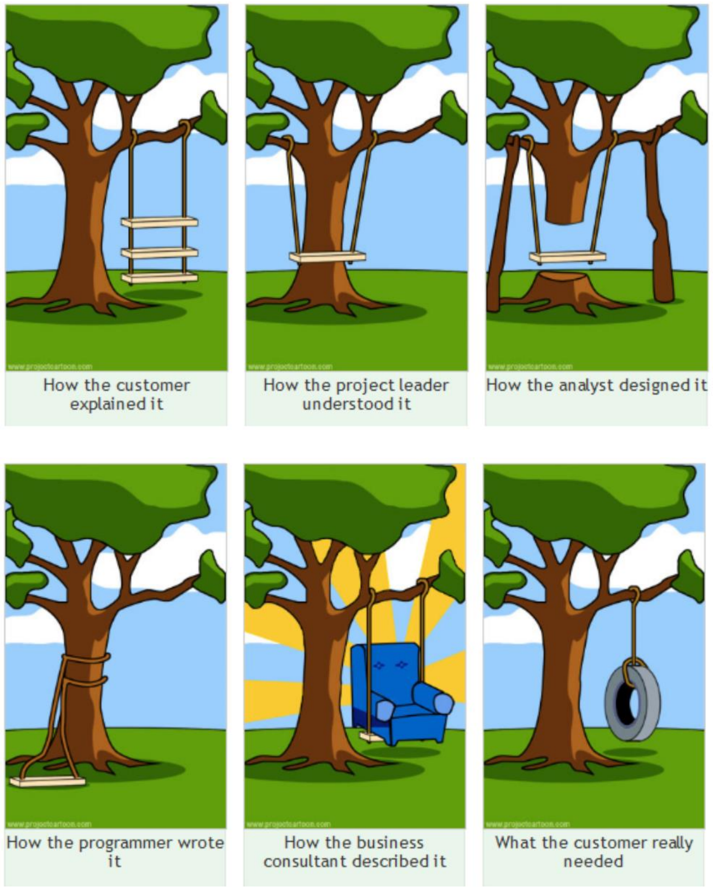
- It is usually less than **1/4** of the total effort.
- On large projects, it is less than **1/6** of the total effort.

If your project fails…

- Poor planning

If your project fails…

- Inadequate understanding of the requirements

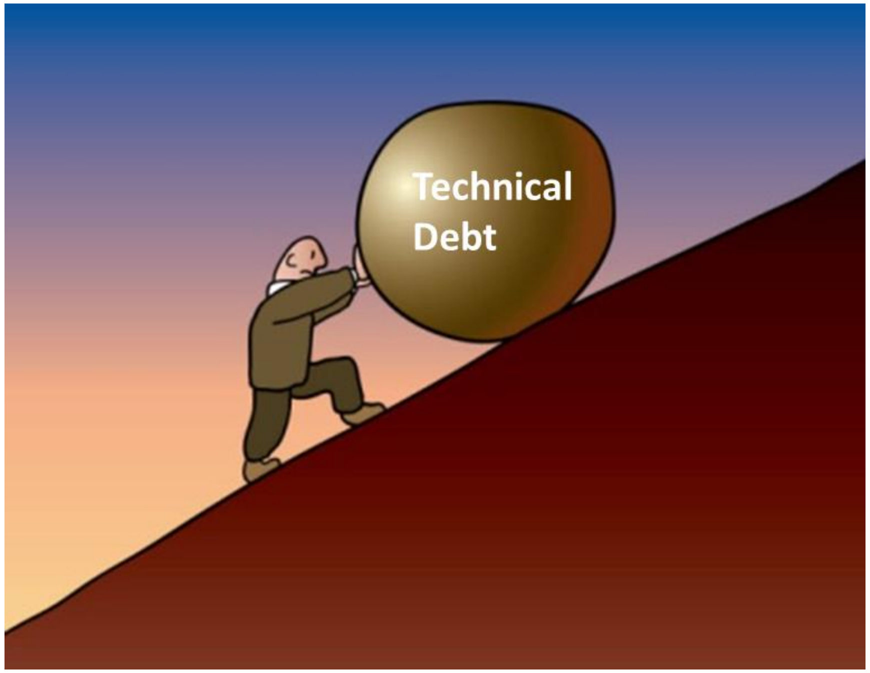

How the customer explained it
How the project leader understood it
How the analyst designed it
How the programmer wrote it
How the business consultant described it
What the customer really needed

If your project fails…

- Inadequate attention to quality

If your project fails…

- Failure to respond to problems until too late



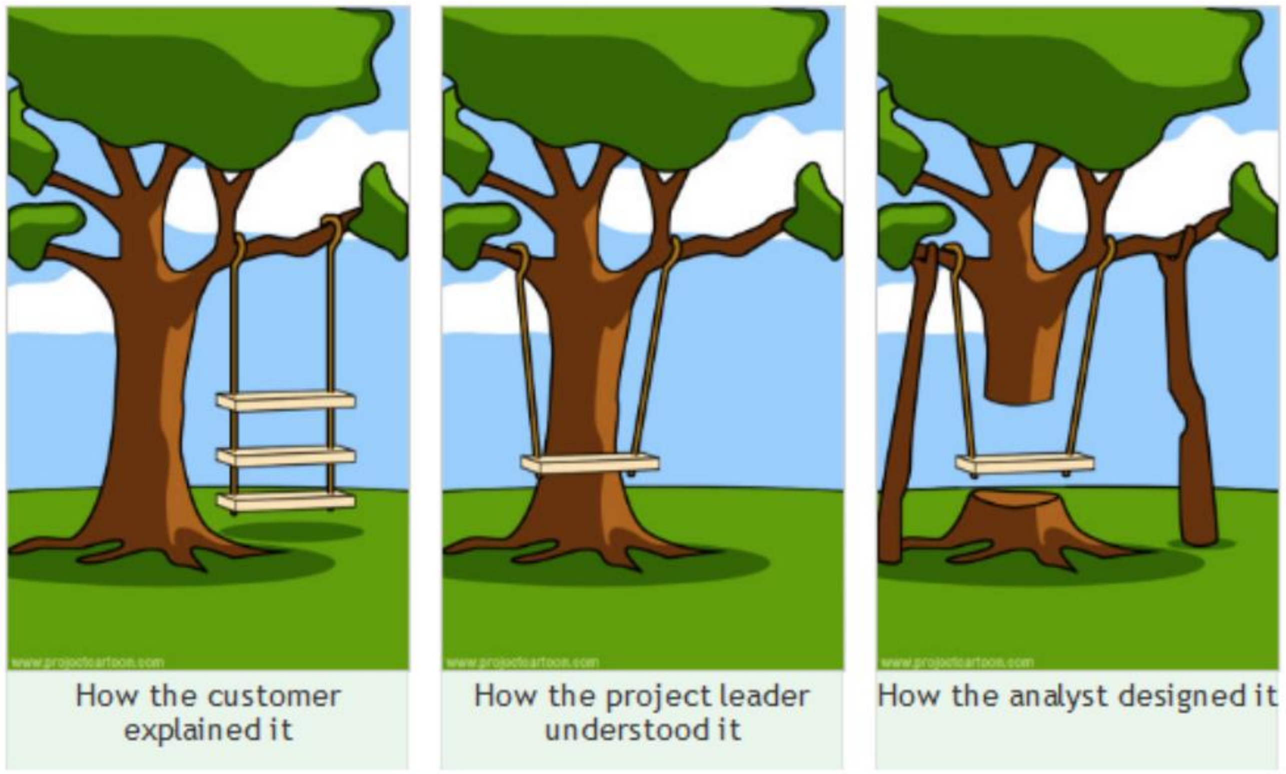
Software project challenges

1. Feasibility and profitability
    - Should the project be done?

Software project challenges

2. Requirements

- **What needs to be done?**


www.projectcartoon.com
How the customer explained it
www.projectcartoon.com
How the project leader understood it
www.projectcartoon.com
How the analyst designed it

Software project challenges

3. Planning and controlling

- **Who does what and when?**

Software project challenges

4. Implementation

- Creating the software

```
<li><a href="index.html">Home</a></li>
<li><a href="home-events.html">Home Events</a></li>
<li><a href="multi-col-menu.html">Multiple Column Menu on Larger Viewports
<li class="has-children"> <a href="#" class="current">Header Options</a>
  <ul>
    <li><a href="tall-button-header.html">Tall Button Header</a></li>
    <li><a href="image-logo.html">Image Logo</a></li>
    <li class="active"><a href="tall-logo.html">Tall Logo Image</a>
  </ul>
</li>
<li class="has-children"> <a href="#">Carousels</a>
  <ul>
    <li><a href="variable-width-slider.html">Variable Image Wi
    <li><a href="testimonial-slider.html">Testimonial Slider<
    <li><a href="featured-work-slider.html">Featured Work Sli
    <li><a href="equal-column-slider.html">Equal Column Slid
    <li><a href="video-slider.html">Video Slider</a></li>
    <li><a href="mini-bootstrap-carousel.html">Mini Slider
  </ul>
```

Software project challenges

5. Delivery and maintenance

- Delivery to users, responding to needed changes

Software project challenges

6. Support

- **Environment and tools needed for project**

WINNER
Fit Out Project of the Year - Office (Large)
Fit Out Awards 2013
2013 GOLD WINNER IDA INTERNATIONAL DESIGN AWARDS
A'DESIGN AWARD WINNER 2014 PLATINUM

Software project challenges

7. Teamwork

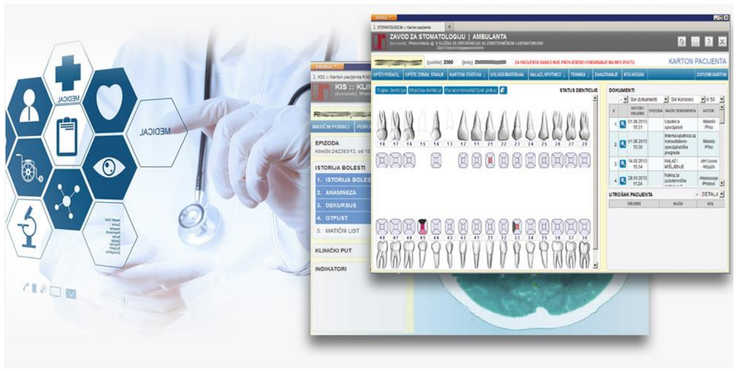
- **Making sure everyone is on the same page**

Software project challenges

1. Feasibility and profitability
   Should the project be done?
2. Requirements
   What needs to be done?
3. Planning and controlling
   Who does what and when?
4. Implementation
   Creating the software
5. Delivery and maintenance
   Delivery to users, responding to needed changes
6. Support
   Environment and tools needed for project
7. Teamwork
   Making sure everyone is on the same page

Two examples to consider

Medical practice information System



- Cloud-based
- Highly reliable and secure
- Additional products/services
- Size: 1 MLOC
- 700 staff-months by 40 staff over 21 months

Running App for Apple Watch

- Speed to market is imperative
- Reliability needs are low
- Cost must be low
- Size: 10 KLOC
- 8 staff-months by 2 staff over 4 months

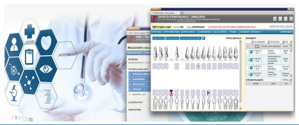
Feasibility and profitability

- What is the market?
  - Who will pay for and use the system?
  - How much will they pay?
  - How will the market change?
- How expensive will the project be?
  - Effort
  - Calendar time
  - Available resources
  - Purchased or leased resources
- What are the risks?

Requirements

- Which features should be produced?
- What are the non-functional requirements?
  - Reliability
  - Security
  - Maintainability
  - Efficiency: run-time /space
  - Usability
  - Price
  - Time to market
- Who knows what is needed?
- Who decides what to implement?

Medical system vs. Running App Non-functional Requirements



	Medical System	Running App
Reliability	On 24/7	Not an issue
Security	Patient-sensitive info..	Not an issue
Maintainability	Several updates over few years	Not an issue
Efficiency	Some run-time requirements	Space/power limitations
Usability	Work with clients and users	Expect to change it later
Price	Fairly expensive	Very low
Time to market	Not too much urgency	critical

Planning and controlling

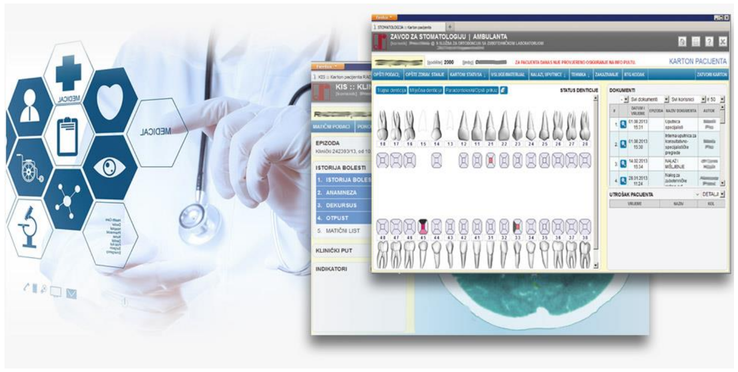
- Long-term planning
  - Release schedule
  - Lifetime of product or service
- Project planning
  - Who does what
  - Relationship and communication with stakeholders
  - Scheduling of tasks
- Project management
  - Reviewing and tracking progress
  - Deciding when something is “done”
  - Taking corrective action
- Project communication
  - What information is shared across project
  - How information is shared across project
  - How decisions are made and communicated

Implementation

- High-level design and architecture
  - Creation
  - Communication to project staff
  - Maintenance
- Low-level design
  - Similar to high-level design
- Programming
  - Creation methods (e.g. pairs, coding standards)
  - Review
  - Maintenance
- Verification and Validation
  - Types of reviews and tests
  - Who does what

Medical System Implementation

- High-level design and architecture
  - Carefully developed and maintained
- Low-level design
  - Use best practices (e.g. design patterns)
- Programming
  - Need good standards
- Verification and Validation
  - Extensive review and testing
  - Need separate team for testing



Running app implementation

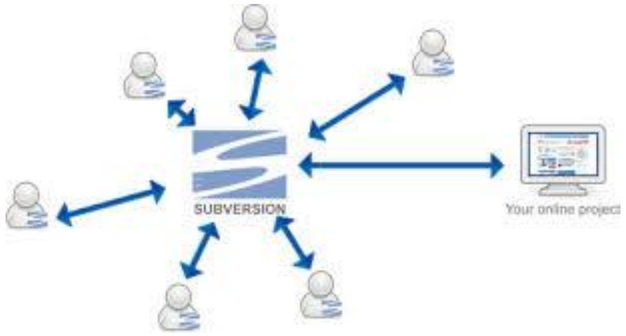
- High-level design and architecture
  - Throw-away sketch
- Low-level design
  - Some comments in the code
- Programming
  - As fast as possible
- Verification and Validation
  - Let the users find the bugs

Delivery and Maintenance

- How often releases are delivered and installed?
- Who does what?
- How issues are detected, recorded, reported and tracked?
- How change requests are managed?

Support

- Software development methods
  - Training, mentors and guides
- Tools
  - Version control and configuration management
  - Editors and programming environments
  - Compilers and code generators
  - Static analysis tools
  - Testing tools
  - Bug tracking tools
- Physical care
  - Project workspace
  - Individual workspace



Bugzilla@Mozilla – Main Page
version 4.0.6+
Home | New | Browse | Search | Search | [?] | Reports | Requests | New Account | Log In | Forgot Password
Welcome to Bugzilla
Get Help
File a Bug
Search
Open a New Account
Enter a bug # or some search terms
Quick Search
Quick Search help | Install the Quick Search plugin
Bugzilla User's Guide | Release Notes | Bugzilla Etiquette | Bug Writing Guidelines
Home | New | Browse | Search | Search | [?] | Reports | Requests | New Account | Log In | Forgot Password
Privacy Policy

Teamwork

- Software development is not a solitary activity
- Communication between team members is essential
  - Some knowledge needs to be shared immediately
  - No one knows everything
- Collaboration amongst team members is essential
  - Group activities
  - Dependencies between activities and components (STC)
- Project success depends on the success of the entire team

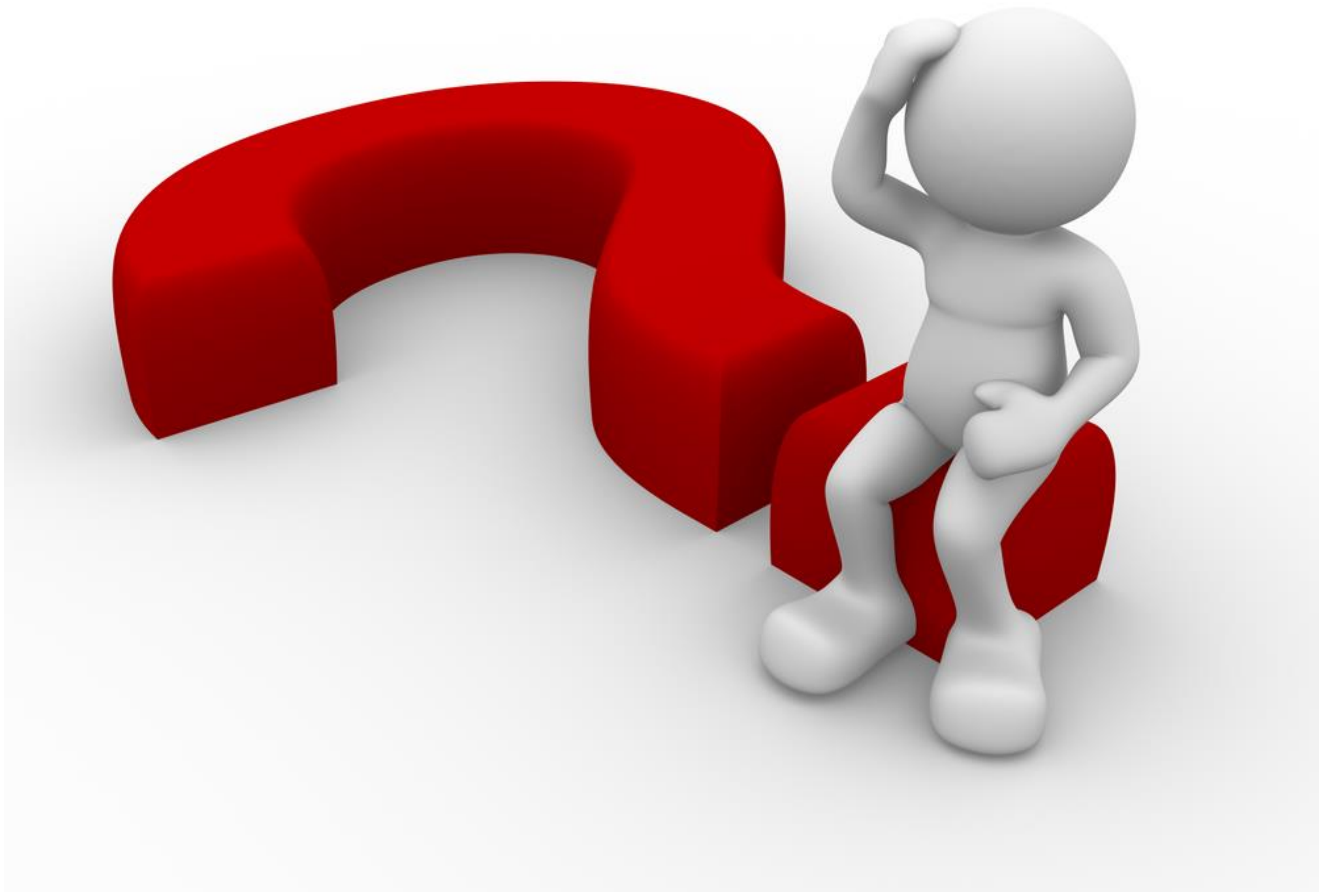
Medical System Teamwork

- Software development
  - Teams of developers working on different sub-systems
- Communication
  - Regular meetings
  - Published documents for design, plans, user support
- Collaboration activities
  - Code reviews
  - Protocols for shared sub-system modifications
- Entire team
  - Operations and support teams are critical

Running App Teamwork

- Software development
  - Two developers talk to one another
- Communication
  - Two developers talk to one another
- Collaboration activities
  - Two developers talk to one another
- Entire team
  - Two developers talk to one another

Questions?



STEVENS
INSTITUTE of TECHNOLOGY
THE INNOVATION UNIVERSITY®

stevens.edu

Thank You