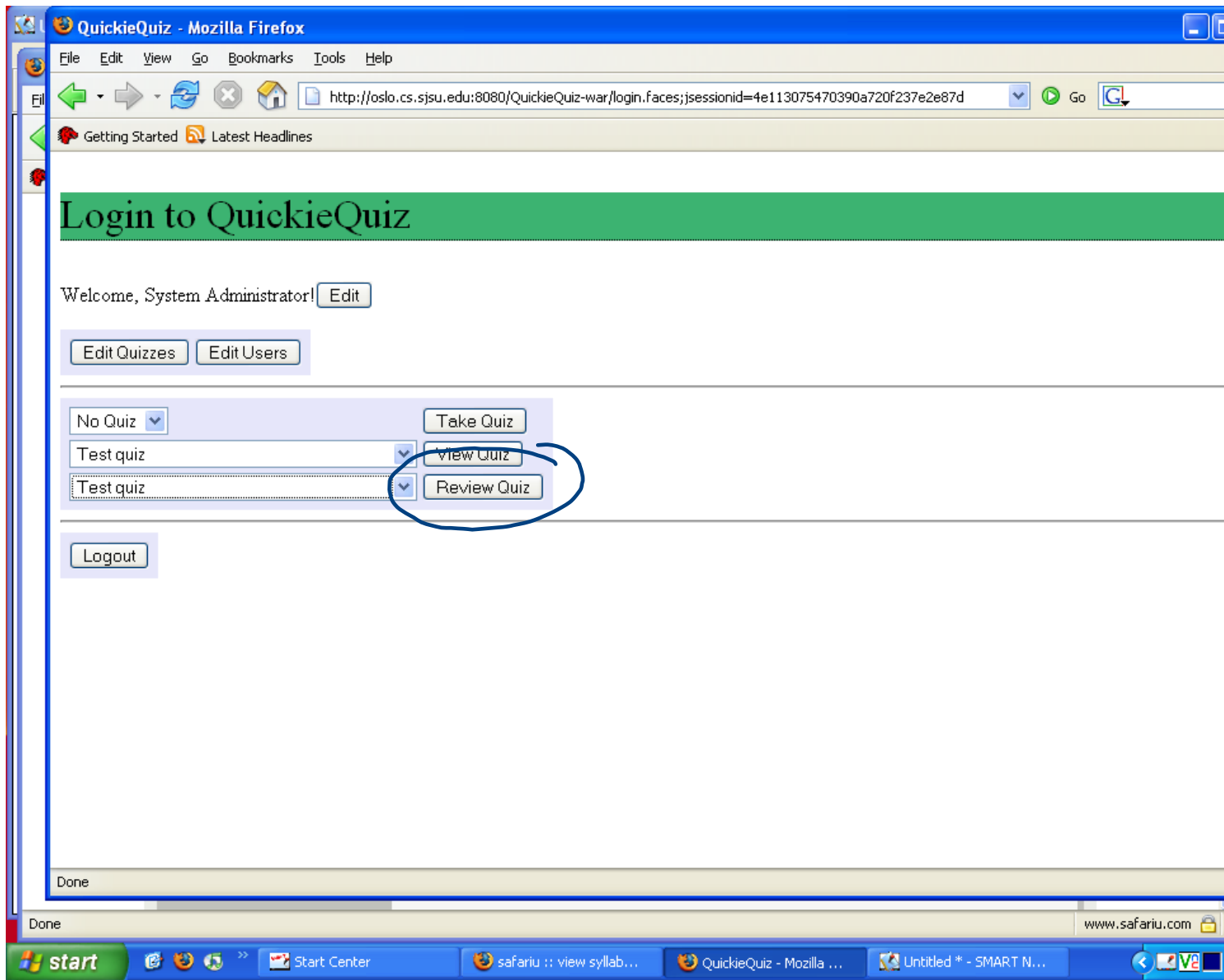


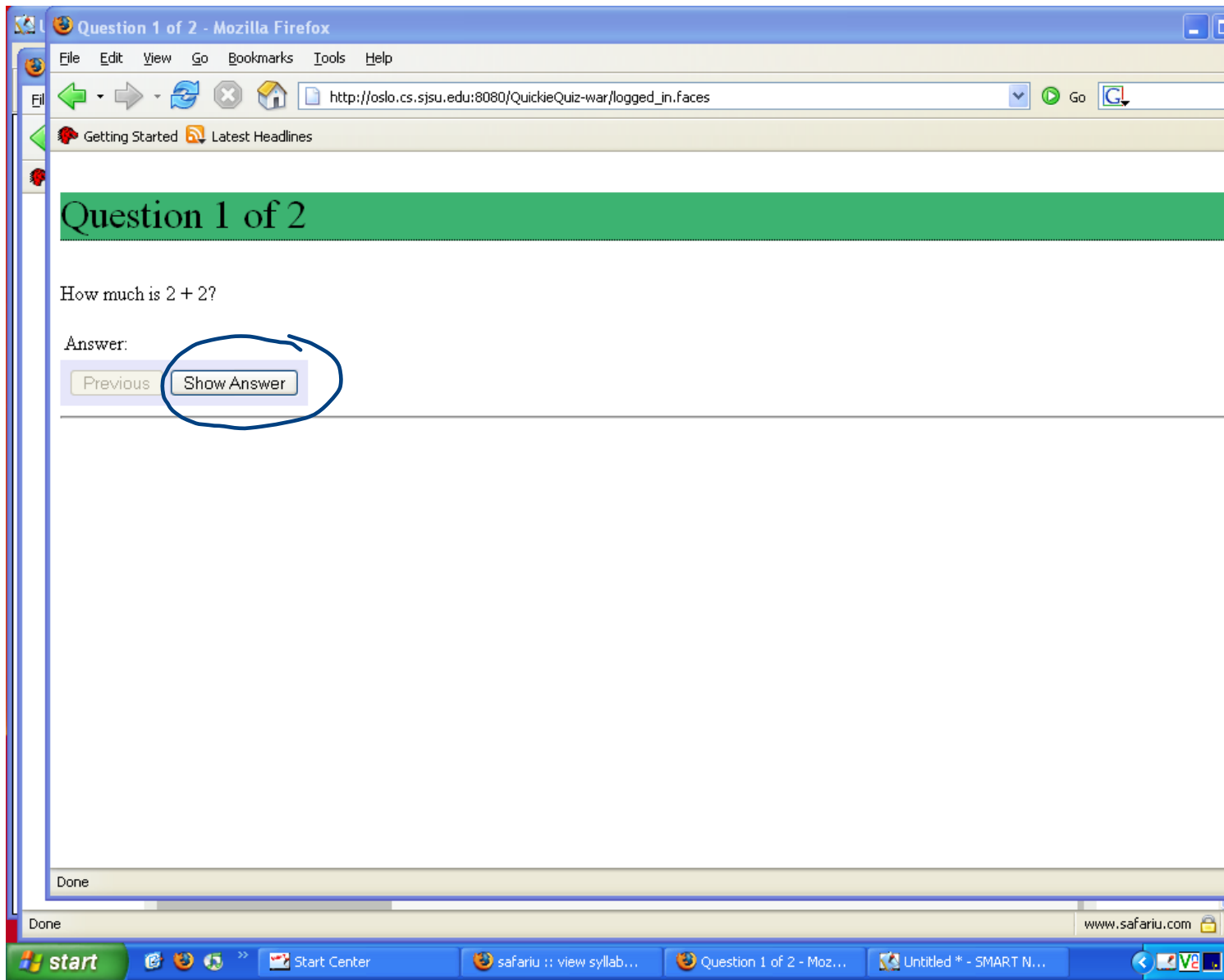
Extreme programming (XP)
= Anti - ~~UML~~ RUP

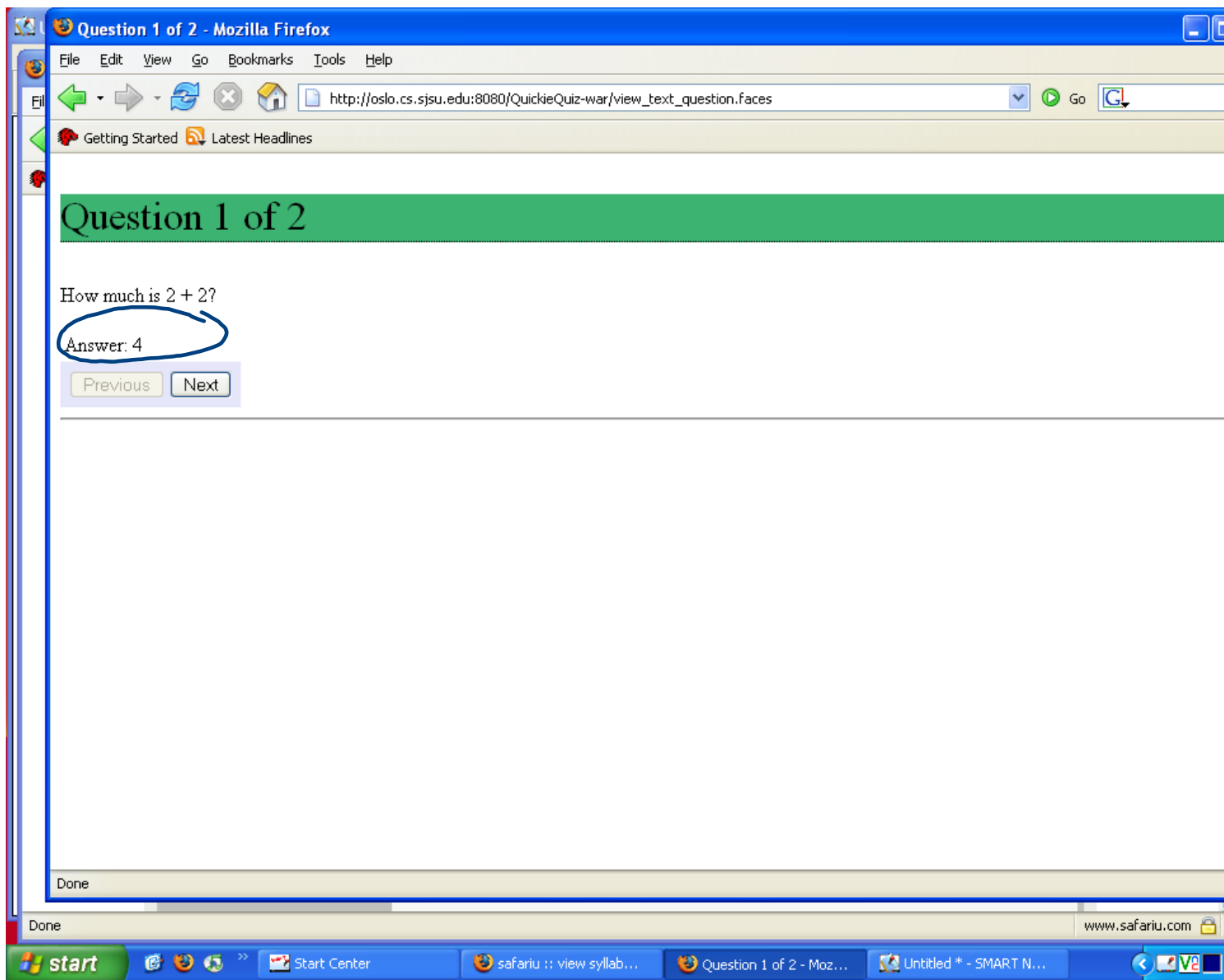
Agile programming

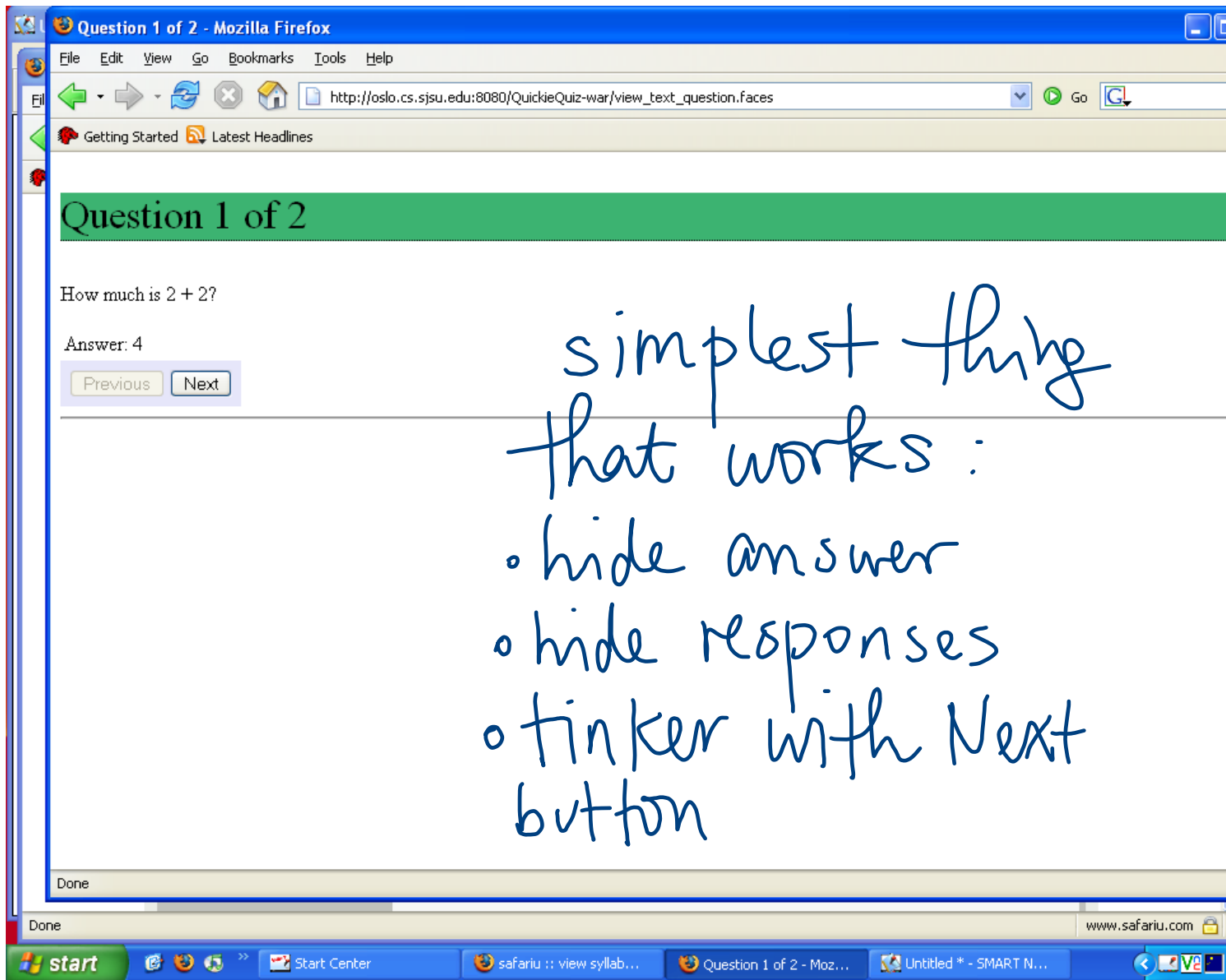
Why is UML rigid?

- All use cases before building system











↳ Comprehensive set of
UML diagrams

Upfront work of questionable
benefit

RUP: ↑ good predictability
if bad: too expensive

The screenshot shows a Mozilla Firefox browser window displaying the Safari U website. The browser's address bar shows the URL: <https://www.safariu.com/view-syllabus.do?method=displayContent&mode=book&syllabusId=767368223543>. The website header features the Safari U logo and the tagline "Your Course – In Print, On the Web – Your Way." Below the header is a navigation menu with links for Home, About, Build, Browse, My Projects, Adopt a Book, and Help. A user is logged in as "Cay Horstmann" with links for "My Account" and "Sign Out".

The main content area displays a breadcrumb trail: [Syllabi](#) > [CS160](#) > [Extreme Programming Explained: Embrace Change, Second Edition](#) > [Exploring XP](#) > [Values](#). Below this is a book entry for "Extreme Programming Explained: Embrace Change, Second Edition" by Kent Beck, published by Addison Wesley Professional. A "Table of Contents" sidebar is visible on the left, listing various sections such as Copyright, Praise for Extreme Programming Explained, Second Edition, The XP Series, Note To Programmers, Foreword to the Second Edition, Foreword to the First Edition, Preface, and 1. What is XP?.

The main content area displays the title "Chapter 4: Values" and the section "Introduction". The introduction text reads: "Paul, the master gardener, has an intuitive sense of what needs to be done next. He knows in his bones what matters and what doesn't. I might think perfectly straight rows are really important. I put a lot of effort into making my rows straight. Along comes Paul and says, 'Why are you working so hard at

Do simplest possible...

↑ good: don't pay for
stuff that's not needed

↓ bad: simple changes
tend to be ugly patches

The screenshot shows a Mozilla Firefox browser window with the title "Question 1 of 2 - Mozilla Firefox". The active tab is "safariu :: view syllabus - Mozilla Firefox". The address bar shows the URL: <https://www.safariu.com/view-syllabus.do?method=displayContent&mode=book&syllabusId=7673682235>. The page content includes a left sidebar with a table of contents and a main content area with the following text:

Section : Courage

Courage is effective action in the face of fear. Some people have objected to using the word “courage”, reserving it for what a patrolling soldier does when going through a darkened doorway. Without intending to diminish the kind of physical courage demonstrated by the soldier, it is certainly true that people involved in software development feel fear. It's how they handle their fear that dictates whether they are working as an effective part of a team.

Sometimes courage manifests as a bias to action. If you know what the problem is, do something about it. Sometimes courage manifests as patience. If you know there is a problem but you don't know what it is, it takes courage to wait for the real problem to emerge distinctly.

Courage as a primary value without counterbalancing values is dangerous. Doing something without regard for the consequences is not effective teamwork. Encourage teamwork by looking to the other values for guidance on what to do when afraid.

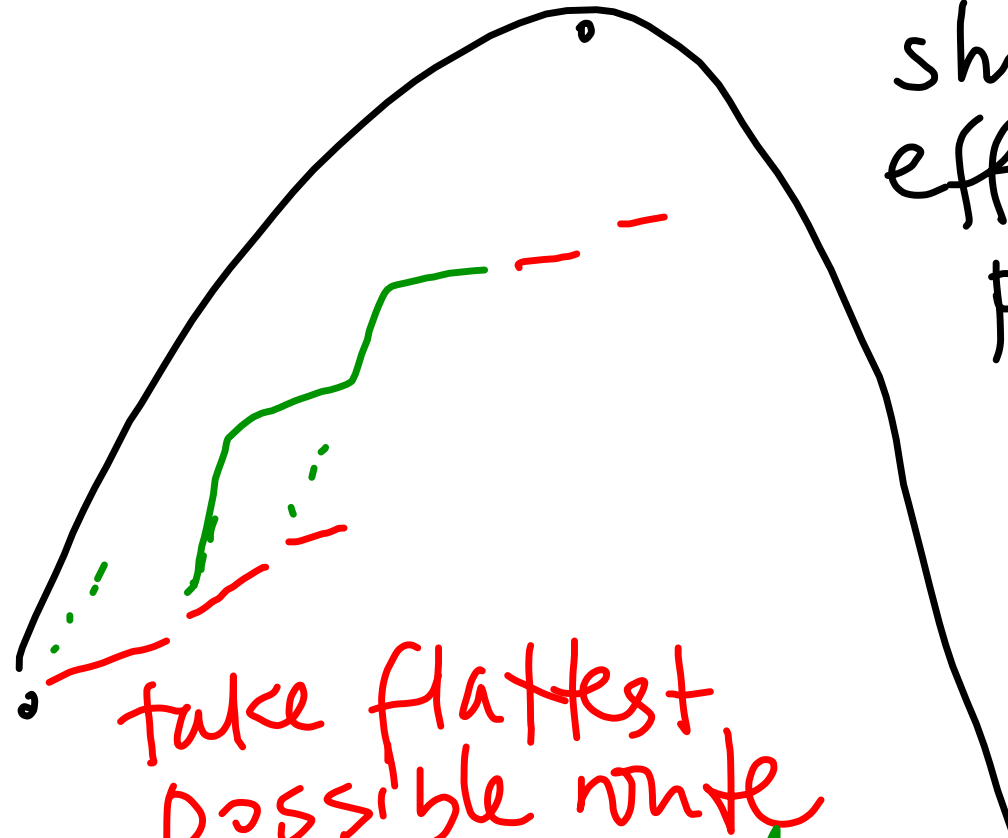
The sidebar contains the following links:

- Getting Started
- Latest Headlines
- Praise for Extreme Programming Explained, Second Edition
- The XP Series
- Note To Programmers
- Foreword to the Second Edition
- Foreword to the First Edition
- Preface
- 1. What is XP?
- Exploring XP
 - 2. Learning to Drive
 - 3. Values, Principles, and Practices
 - 4. Values
 - Introduction
 - Communication
 - Simplicity
 - Feedback
 - Courage
 - Respect
 - Others

Simplicity by itself
is doomed.

Simplicity + Courage
has a chance

"Hill-climbing algorithms"



shortest
effort
path

take flattest
possible route
+ some mechanism
to get out of dead ends

Untitled * - SMART Notebook

safariu :: view syllabus - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

https://www.safariu.com/view-syllabus.do?method=displayContent&mode=book&syllabusId=7673682235

Getting Started Latest Headlines

Programming Explained, Second Edition

- ▶ The XP Series
- ▶ Note To Programmers
- ▶ Foreword to the Second Edition
- ▶ Foreword to the First Edition
- ▶ Preface
- ▶ 1. What is XP?

Exploring XP

- ▶ 2. Learning to Drive
- ▶ 3. Values, Principles, and Practices
- ▶ 4. Values
 - Introduction
 - Communication
 - Simplicity
 - Feedback
 - Courage
 - Respect
 - Others
- ▶ 5. Principles

Section : Feedback

No fixed direction remains valid for long; whether we are talking about the details of software development, the requirements of the system, or the architecture of the system. Directions set in advance of experience have an especially short half-life. Change is inevitable, but change creates the need for feedback.

I remember an all-day presentation I gave in Aarhus, Denmark. One front-row attendee's face got cloudier and cloudier as the day progressed. Finally he couldn't stand it. "Wouldn't it be easier just to do it right in the first place?" Of course it would, except for three things:

- We may not know how to do it "right". If we are solving a novel problem there may be several solutions that might work or there may be no clear solution at all.
- What's right for today may be wrong for tomorrow. Changes outside our control or ability to predict can easily invalidate yesterday's decisions.
- Doing everything "right" today might take so long that

Done www.safariu.com

start Start Center safariu :: view syllab... Question 1 of 2 - Moz... Untitled * - SMART N... 11:06

Feedback from

- users ("on-site customer")
- code reviews
- test suite

Untitled * - SMART Notebook

safariu :: view syllabus - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

https://www.safariu.com/view-syllabus.do?method=displayContent&mode=book&syllabusId=7673682235

Getting Started Latest Headlines

Table of Contents

- ▶ Copyright
- ▶ Praise for Extreme Programming Explained, Second Edition
- ▶ The XP Series
- ▶ Note To Programmers
- ▶ Foreword to the Second Edition
- ▶ Foreword to the First Edition
- ▶ Preface
- ▶ 1. What is XP?
- Exploring XP**
 - ▶ 2. Learning to Drive
 - ▶ 3. Values, Principles, and Practices
 - ▶ 4. Values
 - Introduction
 - Communication
 - Simplicity
 - Feedback
 - Courage
 - Respect

Section : Respect

The previous four values point to one that lies below the surface of the other four: respect. If members of a team don't care about each other and what they are doing, XP won't work. If members of a team don't care about a project, nothing can save it.

Every person whose life is touched by software development has equal value as a human being. No one is intrinsically worth more than anyone else. For software development to simultaneously improve in humanity and productivity, the contributions of each person on the team need to be respected. I am important and so are you.

Done www.safariu.com

start Start Center safariu :: view syllab... Question 1 of 2 - Moz... Untitled * - SMART N... 11:13

Untitled * - SMART Notebook

Raganwald: What I've learned from failure - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://weblog.raganwald.com/2005/01/what-ive-learned-from-failure.html

Getting Started Latest Headlines

Jimmy Buffett

Some of them, in retrospect, would be comical if it wasn't for the human misery, damaged careers, and money wasted on failed projects. Or worse, in my opinion, the opportunity cost of putting good people to work on things that never end up delighting the world. I weep for what might have been.

The four most important causes of failure

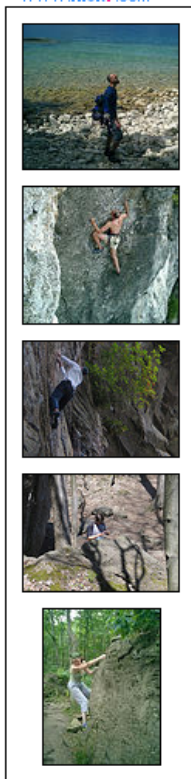
“ Things which matter most must never be at the mercy of things which matter least. ”

Johann Wolfgang Von Goethe (1749-1832)

The first thing I've learned from failure is that the four things which matter most are:

1. The quality of the people doing the development
2. The expected value of the product to its stakeholders
3. The fitness of the proposed solution
4. The quality of project management and expectations communication

In my experience, you need all four working to have a successful project. I've personally failed when even one of those four things was bad and not corrected immediately. If two, three, or all four were wrong, my discovery is that I've been unable to avert disaster. (This list obviously doesn't cover all of the factors needed for business success. I'm just talking about getting the software to ship)



Done

start | Start Center | safariu :: view s... | Question 1 of 2 -... | Raganwald: Wh... | Untitled * - SMA...