



Processes, best practices and real dev

Admin

- <snipped>

-

Turning in your project

- First, how's the project going?
- Second project submission
 - Create a GitHub repository
 - One per group.
 - Each group member check in his/her code. (include commit comments)
 - Each of you email me with a small document which includes
 - A guide to how you implemented your part, which files I can find your implementation in, a user manual explaining what I need to do
 - If you had to update the tests, send me the new test file and explain what/why things changed

Pair programming note

- I was asked about pair programming earlier in the semester
 - I gave my experiences
 - Here are some observations from pycon:
 - when mixture of skill sets – pair programming works great
 - slows things down when there are all senior members

Development process

- From pycon, attendee thoughts on process
 - *don't trust evolution to produce good process*
 - *large company process **hard** to change*
 - *sometimes too many senior people isn't ideal*
 - *Fresh look/perspective helps*
 - *the process on paper vs the process as it exists*
 - *Not always the same.*

Centralized Repositories

- Having everything in a central location means that if that location isn't there, it doesn't exist.
 - Advantage: git
 - Network outages? Don't happen often but when they do
 - Murphey says it will happen the day before 'gold master' upload packaging.
 - And Murphey gets what he wants.

Security

- You should be thinking about security in every application you write
 - Unless it is a complete toy.
 - The sound player for your tabletop games
 - Security thoughts
 - A game like League of Legends?
 - A banking app?
 - Voting machine?
 - Car
 - Airplane?
 - Refer to news articles as needed.

Security

- Think security from the beginning
 - Add it to your unit tests
 - Security should be in your specifications – and your tests
- The trouble with security
 - Sometimes good security practices and bad practices look awfully similar
 - Sometimes good can become bad
 - Encryption in 1999 vs encryption today

Liskov Substitution and Security

- Security is strengthened if you don't violate the Liskov Substitution principle
- Liskov Substitution as she presented it:
 - Let $q(x)$ be a property provable about objects x of type T . Then $q(y)$ should be provable for objects y of type S where S is a subtype of T .
- As Bob Martin paraphrased it:
 - Subtypes must be substitutable for their base types.
- As Security researchers paraphrase it:
 - Subtypes should never give more permissions than the parent/base type.

Principle of Least Authority

- What is the Principle of Least Authority?

Principle of Least Authority

- What is the Principle of Least Authority?
 - Never give an object/module/program more authority or permissions than it needs to do its job.
 - Otherwise someone will be tempted to use those extra permissions
 - 'it would be a shame if they went to waste.'

Designing for Security

- Suppose you have a high security program
 - What is a better way to secure it
 - Passphrase?
 - Or
 - Biometrics?
 - And why?

Designing for Security

- Suppose you have a high security program
 - Considerations when choosing security/locking mechanism
 - Passphrase?
 - Users can sometimes be idiots
 - See top leaked passwords
 - But protected by 5th amendment in US
 - Biometrics?
 - Hard for users to mess up
 - But finger prints can be lifted and with effort fingerprint molds can be used
 - High res images for retina scans
 - Or worse

Ethics dicussions

- Software developers and engineers
- Doctors, Lawyers and more
 - All governed by code of ethics -why?

Ethics discussions

- Software developers and engineers
- Doctors, Lawyers and more
 - All governed by code of ethics -why?
 - They hold our privacy/health in their hands
 - Have power over us

transparency

secrecy

service

So computer science and ethics

- If those people hold power over others how much more us
 - Our code runs the world today
 - From airplanes to automobiles
 - From coffee makers to communications
 - From elevators to entertainment
 - Banking to buildings
 - Everything is software
 - If our software goes bad
 - Lives can be ruined
 - Lives can be lost
 - Its all fun and games till someone get killed (Baltimore?)

Ethics: why we need it

- When a big failure happens – we want to come back
 - Democrats after the US Civil war
 - Got exactly one man elected president from the civil war to Woodrow Wilson in 1912
 - Bankers after the crash of 1929
 - Never operated as freely again
 - Do we really Software quality commission as part of the government?

Ethics and software engineering

- So in Texas and Ontario
 - Software engineering is a licensed profession
 - Pass a test
 - Abide by a code of ethics
 - Maintain status through regular training
 - Movement has fizzled out in recent years.

Ethics

- One possibility as an alternative to ethics:
 - Teach everyone
 - Scribes – once a privileged field, now everyone has the skill.
 - S Its not rocket science, its just computer science
 - if an 10 year old can do it in an hour seminar an undergraduate can
 - (if they work as hard as that 10 year old)
 -

Ethics

- One possibility as an alternative to ethics:
 - Teach everyone
 - Scribes – once a privileged field, now everyone has the skill.
 - S Its not rocket science, its just computer science
 - if an 10 year old can do it in an hour seminar an undergraduate can
 - (if they work as hard as that 10 year old)
 - **So... ethical code? Or each everyone?**

Usability

- Usability is an important topic in real life software development
 - We have no usability specialist here in our small department
 - But no one needs to use software with poor usability any more
 - Unless monopoly
 - See Banner – BSU
 - Or Government
 - Though those are disappearing

Usability Testing

- What is usability testing?

Usability Testing

- What is usability testing?
 - Evaluate a product (particularly the user interface) by testing it on users
 - Part of user centered-design

Usability Testing

- When do we do usability testing?

Usability Testing

- When do we do usability testing?
 - Just like voting in Chicago (used to be) early and often
 - Test
 - when you get the idea
 - and
 - Whenever you are testing code – test usability

Usability testing

- So you have an idea for software and you want to know if you can sell it
 - First talk to your (potential) customers – focus group
 - Don't ask:
 - **Would you use this?**
 - Often yes
 - Unsaid: I already do
 - Ask
 - Are you using something like this?
 - What does this question do?

Usability testing

- So you have an idea for software and you want to know if you can sell it
 - First talk to your (potential) customers – focus group
 - Don't ask:
 - **Would you use this?**
 - Often yes but Unsaid: I already do
 - Ask
 - Do you using something like this?
 - What does this question do?
 - **Finds competitors!**
 - Now you can find what they don't do
 - Do this face to face if possible

Focus group

- What do you get from these questions:
 - You find out:
 - is your idea unique
 - is it desirable
 - competition?
 - What is still needed?

Next in Usability

- So now you have an idea that people want
 - Need to design the layout
 - We're nerds – we have our own biases.
 - But put real users in front of it
 - They may not see it the same way
 - Search is not the same as a good 'information architecture'
 - cf. BSU extranet/intranet
 -

Usability Design

- Possible approach:
 - Put all of the information on note cards - have target audience arrange note cards to do design
 - Can find what might be missing
 - 'The right place' Don Norman: psychology of everyday things.
- Then once you have things in 'the right place'
 - Make it pretty

Usability testing the prototype

- Now you have the real thing on the screen
 - Its time to do more usability testing
 - Who should be the one interacting with the users here?

Usability testing the prototype

- Now you have the real thing on the screen
 - Its time to do more usability testing
 - Who should be the one interacting with the users here?
 - Not you!
 - Preferably a neutral 3rd party like a consultant
 - Why?

Usability testing the prototype

- Now you have the real thing on the screen
 - Its time to do more usability testing
 - Who should be the one interacting with the users here?
 - Not you!
 - Preferably a neutral 3rd party like a consultant
 - Why?
 - So people can give brutal feedback and not feel bad
 - You want it – because they sure will be brutal in their posts after using the software

During the usability testing

- Pay attention to what they are saying – and their faces
 - And who they are
 - What does it mean when your user says “looks like Microsoft”

During the usability testing

- Pay attention to what they are saying – and their faces
 - And who they are
 - What does it mean when your user says “looks like Microsoft”
 - If linux person – polite way of saying “looks like crap”
 - Need to redesign
 - If BSU IT people might be high praise.