Pre-mortems
Keeping your project off the autopsy slab

Christopher Cowell
SDET, HealthSparq
1. Pretend that your project has failed.
2. Ask why it failed.
3. Do something now to prevent that failure.
1. Why am I here?
2. Define terms
3. 11 steps
4. Extra stuff
1. Why am I here?
2. Define terms
3. 11 steps
4. Extra stuff
The Best Instant Coffee

Why you should trust us

I'm a former holder of the Appalachian Trail speed record who relies on caffeine as much for making miles on the trail as for meeting deadlines in my role as a Wirecutter staff writer. After a bad experience with caffeine pills years ago, I turned to instant coffee as my go-to for liquid courage in the backcountry. I’ve tried dozens of instant brands, suffering through the watery, the overly sweet, or the not fully dissolvable.

Wanting the help of professional palates, we recruited Arno Holzschuh, a 14-year...
SDET at HealthSparq, Puppet, Jive, Oracle
Lots of failed projects
Didn’t invent pre-mortems
Pre-mortems worked great for me
Maybe useful for you too?
Formal process >> casual banter
christopher.w.cowell@gmail.com
Projects fail due to foreseeable problems.

This happens *all the time*.
User Score

2.6

Generally unfavorable reviews
based on 1970 Ratings
Vaccinate your project against failure.
1. Why am I here?
2. Define terms
3. 11 steps
4. Extra stuff
“It depends upon what the meaning of the word ‘is’ is.”
Pre-mortem?

Formal risk identification and mitigation process
Pre-mortem?

Formal risk identification and mitigation process
Risk?

Anything that might reduce a project’s success
Risk?

Anything that might reduce a project’s success
Anything?

EVENT

- Server catches on fire
- Someone quits
- Requirements change
Anything?

PERSON

● Incompetent
● Stubborn
● Negative
Anything?

CULTURE

- Competitive, not collaborative
- Skeptical
- Fast and sloppy
Anything?

CONSTRAINT

- Not enough time
- Not enough people
- Technical decisions imposed from outside
Anything?

TECH

- Buggy libraries
- Friction between components
- Services don’t do the needful
SParse1

# handle numeric address spec
R$* < @ [ $+ ] > $* $: >98 $1 < @ [ $2 ] > $3 numeric internet spec
R$* < @ [ $+ ] > $* $#esmtp @ [$2] $: $1 < @ [$2] > $3 still numeric: send

# handle virtual users
R$+ < @ 0 = w . > $: < $(virtuser $1 @ $2 $@ $1 $: $@) > $1 < @ $2 . >
R<@> $+ + $* < @ $* . >
   $: < $(virtuser $1 + * @ $3 $@ $1 $: $@) > $1 + $2 < @ $3 . >
R<@> $+ + $* < @ $* . >
   $: < $(virtuser $1 @ $3 $@ $1 $: $@) > $1 + $2 < @ $3 . >
R<@> $+ < @ $+ . > $: < $(virtuser $2 @ $3 $@ $1 $: $@) > $1 < @ $2 . >
R<@> $+ $: $1
R< error : $- $+ > $* $#error $@ $(dequote $1 $) $: $2
R< $+ > $+ < @ $+ > $: $>97 $1

R$=L < @ $= w . > $#local $: @ $1 special local names
R$+ < @ $= w . > $#local $: $1 regular local name
LACK OF KNOWLEDGE

- Known unknowns
- Unknown unknowns
Risk?

Anything that might reduce a project’s success
Reduce?

Catastrophic failure is not the only kind of failure
Pre-mortem?

Formal risk identification and mitigation process
Mitigation?

Small expense now prevents a big expense later

- insurance
- umbrella

Not free, but worth it
“Mitigation” is a weird word.

- reduction?
- minimization?
- lessening?
- shrinking?
- alleviation?

In conclusion: English is dumb.
No really, English is dumb.

- Tomb = \textit{\textbackslash toom}\textbackslash 
- Womb = \textit{\textbackslash woom}\textbackslash 
- Bomb = \textit{\textbackslash bawm}\textbackslash 
- Comb = \textit{\textbackslash k\textasciitilde o\textbackslash m\textbackslash 

Esperanto FTW.
1. Why am I here?
2. Define terms
3. 11 steps
4. Extra stuff
1. Decide scope

- Whole project?
- One feature?
- QA only?
2. Decide logistics

- Who?
- (A)synchronous?
- Local vs. remote?
3. Send instructions

- Explain process
- Speeds things up
- Increases buy-in
4. Imagine failure

- What is failure?
- Partial vs. complete
- Minor vs. disaster
5. Brainstorm reasons for failure

- One list per person
- No peeking!
- No filters
- *Meteor!*
6. Make a master list

- Consolidate into one list
- De-duplicate
- Clarify
- Spur new risk ideas
- Timebox
7. Filter

- Low impact?
- Unlikely?
- Out of your control?
- Too vague?
- Prefer consensus
“Reasons” → “Risks”

pretend backward-looking → real forward-looking
8. Vote

- What do you think are the top risks?
- Define “top” however you want
9. Reorder

- Top vote-getters move to the top
10. Action items to mitigate top risks

- How many?
- Assign owners
- Set due dates
Action items are the whole point
QA-2934 Build meteor-proof shield

Code Coverage

Ready for Engineering
Build meteor-proof shield
To Do

[ ] Build meteor-proof shield
def build_meteor_proof_shield
    # TODO implement this
end
To Do

build meteor-proof shield
1 Pinned Item

chris Today at 5:10 PM
[ ] build meteor-proof shield
Action items are the whole point
● Code arrived late so QA ran out of testing time
  ○ **Erik** assigned to HJ-464: create HipChat plugin to sends alerts for unreviewed PRs
  ○ **Chris** to add “QA tasks” agenda item to weekly grooming meeting, so the whole team understands how much time QA needs for testing
● We hired lots of Devs but not enough QA, so QA became overloaded
  ○ **Phong** to add discussion topic for 1:1 with Ryan
● QA didn't apply 80/20 rule when deciding what to test, so tested the wrong things
  ○ **Chris** assigned to HJ-451: propose process for assessing risk of new stories
11. Track action items

• Use any technique you want
1. Decide scope
2. Decide logistics
3. Send instructions
4. Imagine failure
5. Brainstorm reasons for failure
6. Make a master list
7. Filter
8. Vote
9. Reorder
10. Make action items
11. Track action items
1. Why am I here?
2. Define terms
3. 11 detailed steps
4. Extra stuff
Contraindications

- Failure is an option
- Success is hard to define
- Familiar processes, technology, and/or people
- Huge team
Pre-mortems just shift the odds.
Pre-mortems get cheaper and better with practice.
Pareto principle *totally* applies.
Fringe benefits

- Team bonding
- Team learning
- Schadenfreude
- QA proves its worth
Q&A and Discussion

christopher.w.cowell@gmail.com