

# WHY ME?

---

- ☐ Self taught software tinkerer
- ☐ Love the craft & growing techies (to replace me)
- ☐ ~2 decades of failed, successful, mind blowing and soul haunting projects across finance, education, agriculture, health, genealogy, HR, ERP (time management)
- ☐ Served clients in UG, UK, US, Australia, Europe, SA
- ☐ 12 years experience starting, setting up, growing & running a Ugandan custom software development shop
- ☐ Experience both sides – funder, product owner, consultant, client consultant, customer, sales, technology service provider, architect, tech lead, developer, tester

# CONTEXT SETTING

---

- What is software delivery

  - The process of getting a software product to market

- Ideation → Requirements → Design → Implementation → Deployment → Support & Maintenance → Evolution

- Including end user training, production monitoring, Security Management, end-user Security Management, end-user devices, performance management, change management, change management

- Don't forget the important stuff – contracts, pricing/funding, sales/marketing, pricing/funding, sales/marketing, stakeholder needs, competition, regulatory competition, regulatory compliance

# SAMPLE PROJECTS

*If your dreams  
do not scare you,  
they are too  
small*

*~Richard  
Branson*

*Aim for the stars,  
and you will  
probably reach  
the sky*

## □ Employee Management solution

- Started 2004 as timesheet
- Added recruitment & employee onboarding, invoicing & billing, data exports to external systems, calculators
- 2011 – no more changes could be made
- PHP 3 → 4 → 5, MySQL 3 → 4 → 5

## □ eHealth Solution – UgandaEMR

- 2015 – inherited 350 sites at 1.6.x, 2 forms
- 2016 – first upgrades to 1.9.x with 4 forms
- 2017/8 – Migrations to 2.0, 8 forms, 20 reports ~850 sites
- 2019/20 – 3.0, 10 forms, 30 reports, ~1000 sites

# WHY DO WE HAVE TO BE AGILE?

*ag·ile*

*“able to move quickly*

*and easily”..*

*“able to think and*

*understand quickly”*

- ☐ Increased executive expectations for impact of technology projects
- ☐ Shorter delivery timelines
- ☐ Project scope/features are getting more complex due to integration & need to respond to more knowledgeable customers
- ☐ Cross cutting stakeholders – no longer confined to a single department/sector
- ☐ Relevant technology skillsets are in short supply
- ☐ Customer behavior is changing at a very fast rate

# UGANDA SPECIFIC CHALLENGES?

---

- ❑ Not special but some areas are compounded
  - ❑ Convincing stakeholders of your project relevance
  - ❑ Power & Internet
  - ❑ Low appreciation of impact of technology
  - ❑ Internal staff challenges
  - ❑ Culture – we always have to be right
  - ❑ Difficult to make pragmatic technology choices that meet client project needs



# PROJECT WON! CONTRACT SIGNED

---



# YOUR TYPICAL PROJECT

---

- ❑ After a long protracted negotiation
  - ❑ Fixed cost
  - ❑ Fixed timeline
  - ❑ Defined scope
- ❑ What happens along the way
  - ❑ Requirements keep changing along the way
  - ❑ Stakeholder influencing on scope – we did not think about that
  - ❑ Timelines are too short
  - ❑ No single product owner
  - ❑ Changing Technology platforms & choices
  - ❑ Changing business environment and problem



# HOW DID WE GET HERE

---

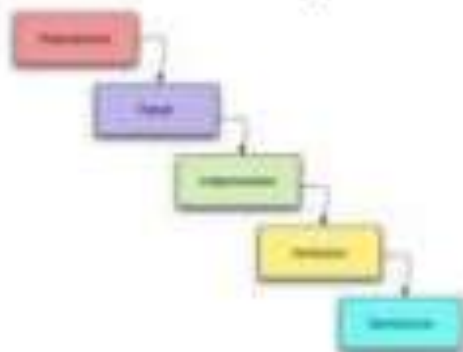
- ❑ Waterfall – rigid process Requirements → Design → Development → QA → Production
- ❑ Rational Unified Process – Waterfall in the large & small

## Software Development

### Waterfall

70s, 80s

Sequential Process  
All design front-up  
Process heavy



### RAD

Rapid Application Development  
80s, 90s

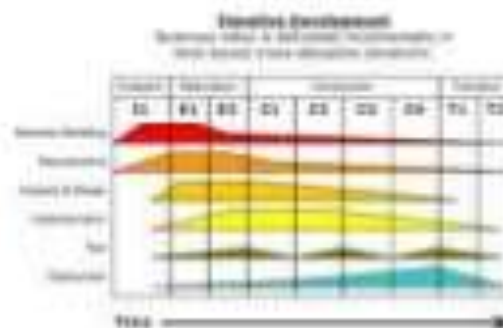
Rapid Prototyping  
Prototype not plan  
Process Light



### RUP

Rational Unified Process  
90s, 00s

Framework for iterative development  
Can be process heavy



### Agile

00s, 10s

Iterative and incremental  
Can be process light



# ***AGILE MANIFESTO***

*Uncovering better ways of developing  
software by doing it and helping others do it*

*2001*

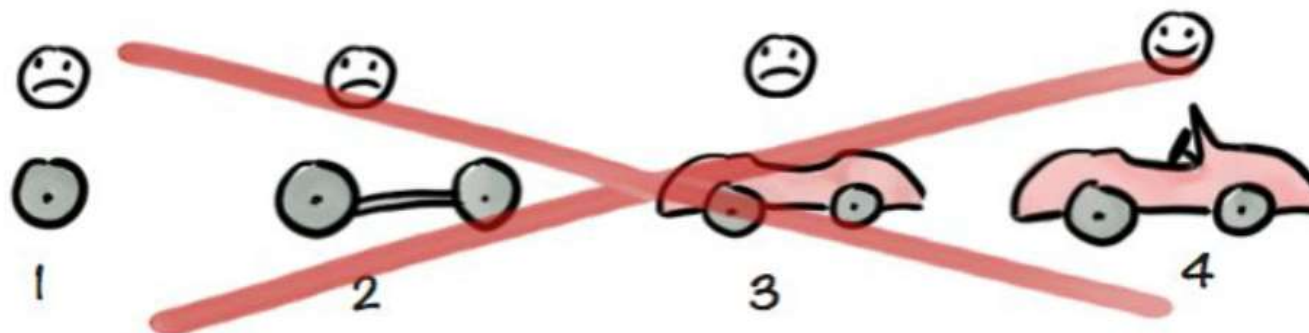


**Kevin Whinnery**  
@kevinwhinnery

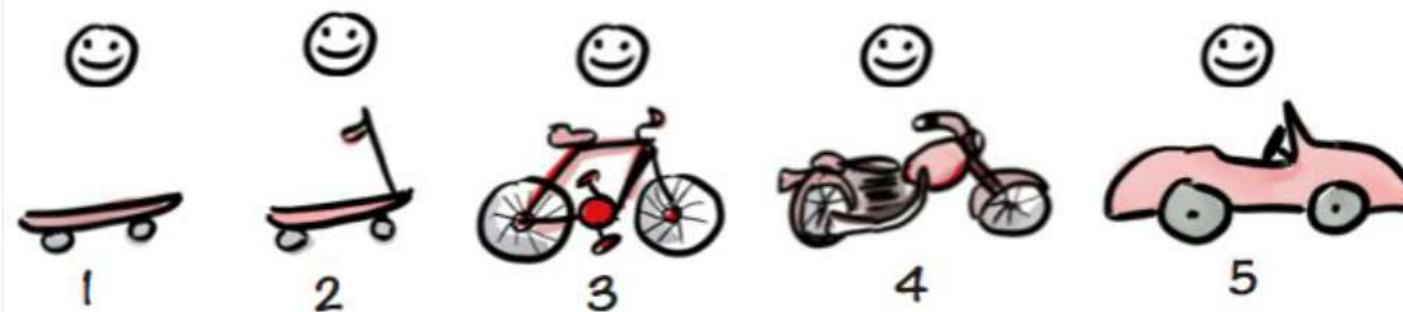


It's coming up in my feed again, so I'll remind you that "ship something even if it sucks!" is not "agile" or "lean". Do this instead:

Not like this....



Like this!



3:12 AM · 04 Sep 17

- # AGILE VALUES
- ❑ Individuals and interactions *over processes and tools*
  - ❑ Working software *over comprehensive documentation*
  - ❑ Customer collaboration *over contract negotiation*
  - ❑ Responding to change *over following a plan*

*“While there is value in  
the items on  
the right, the items on  
the left are valued  
more...”*

So how do I apply these  
values?



# LET US APPLY SOME AGILE SAUCE

---



# INDIVIDUALS AND INTERACTIONS OVER PROCESS AND TOOLS

---

- Define what success looks like
  - Find your end user & determine how your solution helps them
  - Find out why management needs this solution
- Use process and tools for recording your results
  - Engage the stakeholders while keeping out of their way
  - How does your team work together internally?



# WORKING SOFTWARE OVER COMPREHENSIVE DOCUMENTATION

---

- Give me something to play with NOW!
  - Show stakeholders how you are tackling their challenges
  - Demonstrate future state
  - SHOW PROGRESS!!!
- Document requirements
  - Document feedback
  - Document your progress, status update and future plans

*Documentation is a love letter that you write to your future self*

*- Damian Conway*

# CUSTOMER COLLABORATION OVER CONTRACT NEGOTIATION

---

- Identify what value is to the customer
  - Be creative within your constraints (time/scope/budget)
  - Define & work with SMART deliverables
  - Be Realistic
- Make sure your contract is airtight
  - Keep your contract deliverables in sight – document any changes
  - Pareto 80/20 rule
  - Search for win-win-win outcome

# RESPONDING TO CHANGE OVER FOLLOWING A PLAN

---

- Adapt and validate your plan as you learn more and move forward
  - The more things change the more they remain the same
- Have a plan
  - A plan is a guide, framework
  - Keep the lighthouse (goal) in sight as you change
  - If you fail to plan, plan to fail

*The only constant thing is change*



# IN SEARCH OF THE PERFECT MOUSE

---



# CEREMONIES AND TOOLS

---

- ☐ Standup – daily checkin keep collaborating even after standup
- ☐ Retrospective – if you do not know where you have come from, you will not understand where you are and where you have to go
- ☐ Show case – release early, release often, status check
- ☐ Sprint planning
- ☐ Green build – code builds and runs
- ☐ QA – does it do what the customer expects?

# AIDS TO SUCCESSFUL AGILE – NON TECH

---

- ☐ Use Prototyping tools to give your users a sense of what the final solution will feel like and work
- ☐ Spend time collaborating with your clients in a manner and language that they understand – remember they have other full-time jobs not just your project
- ☐ Discipline – keep time, promises, manage expectations, pay attention
- ☐ Document, document, document – in a way that makes meaning to your end-users

*On time is late!!!*

*Good developers write excellent code*

*Great developers write no code*

*Zen developers delete code*

# AIDS TO SUCCESSFUL AGILE – TECH

---

- ❑ Automate & simplify deployment – to dev, staging and demo sites
- ❑ Fanatical, Meticulous version control – use GitFlow/Trunk based development
- ❑ Testing – unit, integration, stress, load, Travis CI
- ❑ Security – the price is too high at the end
- ❑ Document, document, document for your development team?
- ❑ Use/Leverage prebuilt frameworks, and libraries
- ❑ Use an IDE – JetBrains, VS Code, Sublime

*Every problem is a special case of a more general problem and you are not the first one to solve it*



# AIDS TO SUCCESSFUL AGILE - TECH

---

- ☐ Grit, persistence, perseverance, focus
- ☐ The best tool for the job is the one you know how to use
- ☐ Stick to the fundamental principles, experiment with implementation – where necessary & possible
- ☐ Frugal innovation – stay hungry, keep lean
- ☐ Kaizen ~ Solomon King – search for improvements, never stay still with the status quo
- ☐ Contribute back to #OpenSource – give back to the community you work in
- ☐ Look after your health – what you eat, exercise
- ☐ Connect with others outside tech – perspective

# AIDS TO SUCCESSFUL AGILE – TECH - PRINCIPLES

---

- ☐ Architecture:
  - ☐ Use the simplest architecture possible
  - ☐ 12 Factor App
  - ☐ OWASP Security Considerations
  - ☐ API First Development – everything is a client
- ☐ Phoenix Servers
- ☐ Schema.org for data designs
- ☐ Open Source - give back
- ☐ Don't hesitate to spend money on a tool that will save you time

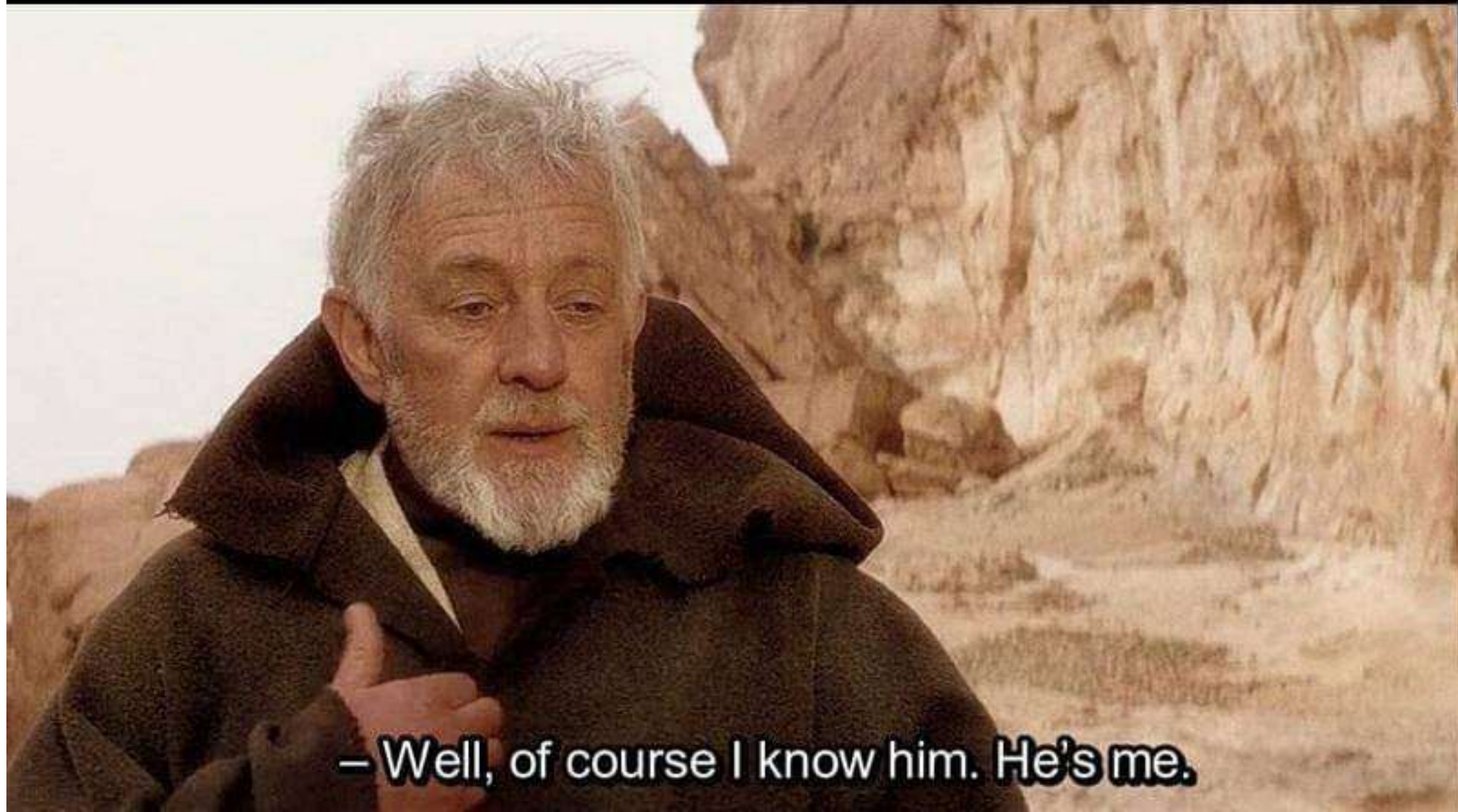
# CHALLENGES

---

- ☐ Draft fatigue – too many back and forth cycles
- ☐ As requirements change, then goals and deliverables are forgotten
- ☐ Team dynamics – changing requirements are stressful
- ☐ Technology tools & choices – do not try to re-invent the wheel as much as you can, solve the problem at hand + know how to use your tools

*If I am given 4 hours to cut down a tree, I will spend 3 hours sharpening  
my axe  
~ Abraham Lincoln*

**When you read some incredibly bad code, thinking "What moron wrote this...", but halfway through it starts to become familiar.**



# READING LIST – BUSINESS & TECH

---

- ☐ Martin Fowler – Refactoring, Patterns of Enterprise Architecture & <http://martinfowler.com>
- ☐ Andy Hunt/Dave Thomas – The Pragmatic Programmer
- ☐ Basecamp ShapeUp - <https://basecamp.com/shapeup>
- ☐ Blue Ocean Strategy
- ☐ Clean Code
- ☐ Imposters Handbook by Rob Conery (I am somewhere)
- ☐ It does not have to be crazy at work
- ☐ Blogs/Articles in whatever language or problem domain you are working in



# PARTING QUESTIONS

---

☐ Is High Quality Software Worth the Cost by Martin Fowler?

<https://martinfowler.com/articles/is-quality-worth-cost.html>

☐ Yes it is, but ensure that this cost can be linked to business goals & value such higher revenue, market share growth/maintenance, new market entry etc

☐ What about Scrum, Kanban, SAFe, etc etc

☐ The best tool for the job is the one you know how to use and which has quick support

☐ Remember the principles



# IN CLOSING

---

*DO NOT BE TOO BUSY*

*Enjoy your project*

*Keep learning*

*Read, Read, Read*

*Stand on the shoulders of Giants*

*Share your stories/experiences with others*

*Discipline, Discipline, Discipline*

*AND MOST OF ALL*

*Deliver and satisfy your clients*

---



# THANK YOU FOR LISTENING

For questions or suggestions

@ssmusoke

<http://ssmusoke.com>