Prototyping

@gavinballard

About Me

Software Eng
Lawyer
Freelance
Senior Dev
(Nearly) Co-Founder

What's a Prototype?

A disposable artifact that can be created quickly and easily in order to answer a specific question about your project.

What's a Prototype?

A disposable artifact that can be created quickly and easily in order to answer a specific question about your project.

Disposable

“If you don’t have the discipline to throw away the prototype, you’re defeating the very purpose of prototyping.”

– Jeff Atwood

Artifact

Paper
Whiteboard
Wireframes
Interactive
Proof of Concept

Quickly and Easily

Familiar Tools
Small Team
Short Timeframe

Ask Specific Questions

Not a hacky version of final product.
Identify a specific question.

Bad Question

“What language should we use?”

Better Question

“Does language X have the performance, ease of use and interoperability for this project?”

Answering the Question

Narrow to Python, Ruby, C++
Build quick (2-day) performance test
Report back

Bad Question

“How is the user interface going to work?”

Better Question

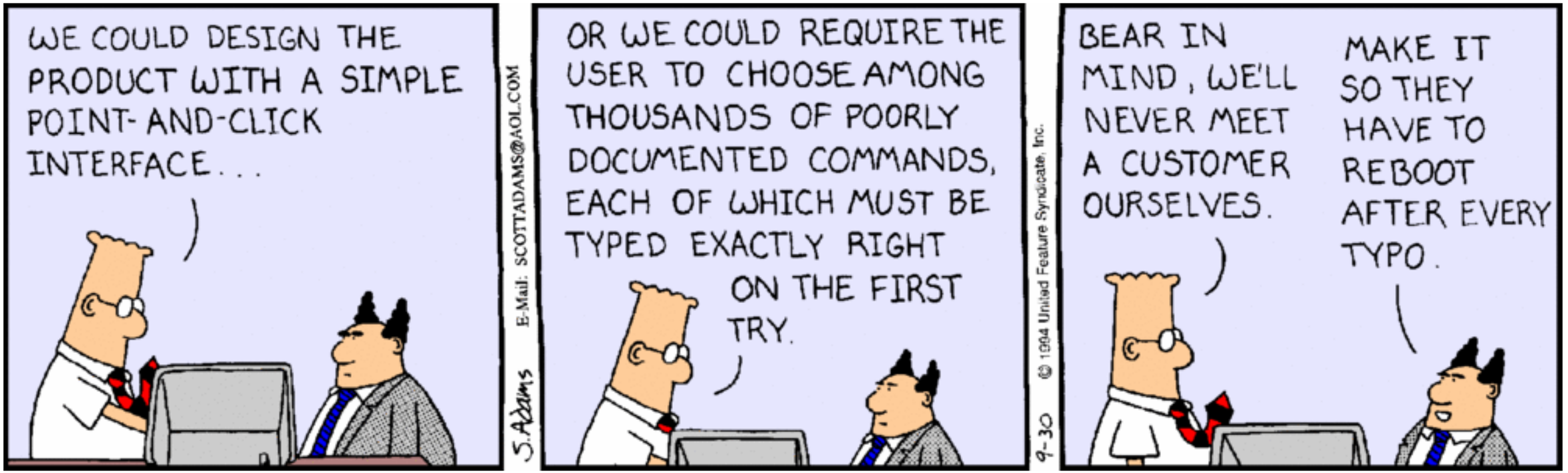
“Would the client be happy to use a command line interface?”

Answering the Question

Write a shell script that gives fake return values

Sit down with the client and/or users and see if it meets needs

Client Understanding



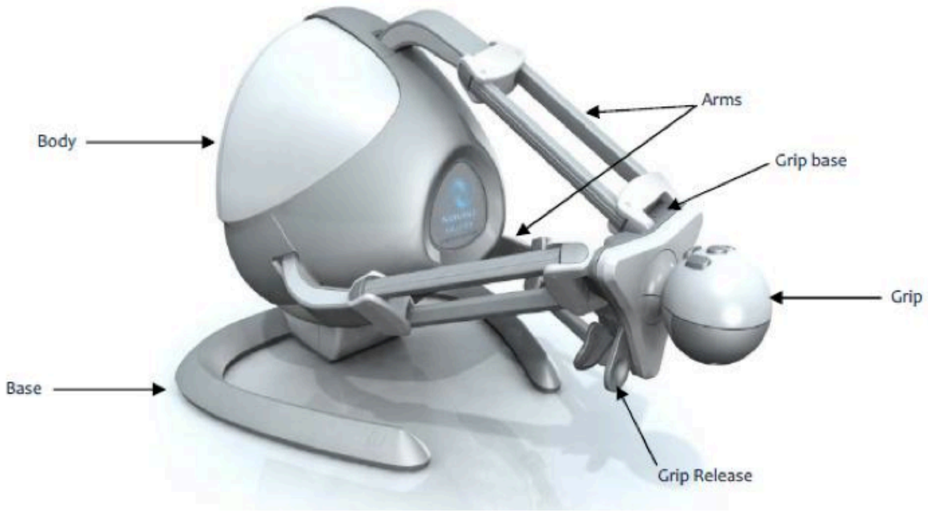
Your Understanding

“Canyons”

Client: MUVES

Objective: Develop a game to help automate stroke rehabilitation

“Canyons”



“Canyons”

“Canyons”

C#
Ogre Game Engine
Mogre C# Bindings
3D Studio Max

Prototyping

Good: Built simple version of game to see if C# and Ogre game engine suitable.

Prototyping

Bad: Didn’t throw away prototype.

Prototyping

Bad: Didn’t pay enough attention to what the prototype told us.

Prototyping

Bad: Didn’t stick to specific prototype goal.

Prototyping

Could have: Prototyped at least one other game engine.

Prototyping

Could have: Prototyped completely different style of game.

Prototyping

Should have: Used the prototypes to increase communication with client.

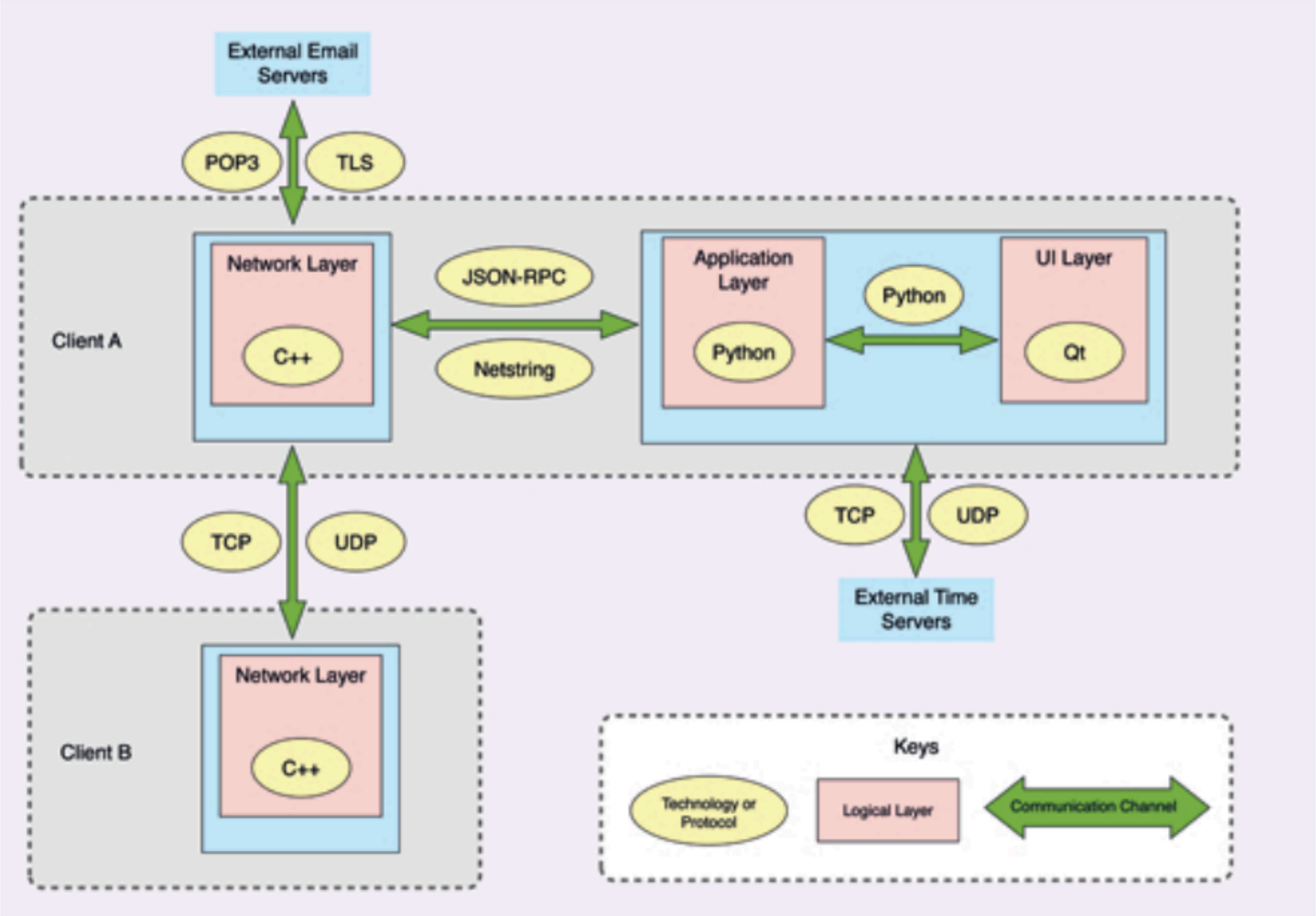
“Family Fun”

Client: UniMelb Researchers

Objective: “Interactive, synchronised, decentralised, distributed communication system.”

“Family Fun”

“Family Fun”


External Email Servers
POP3
TLS
Client A
Network Layer
C++
JSON-RPC
Netstring
Application Layer
Python
Python
UI Layer
Qt
TCP
UDP
TCP
UDP
External Time Servers
Client B
Network Layer
C++
Keys
Technology or Protocol
Logical Layer
Communication Channel

“Family Fun”

Large number of moving parts:
- Interactive graphical frontend
- Email Parsing
- Synchronisation Code
- VoIP
- P2P Network Layer

“Family Fun”

Python/Qt (front-end)
C++ (network layer)
VoIP library

Prototyping

Good: Created two prototypes to determine which graphics framework to use.

Prototyping

Good: Threw these prototypes away!

Prototyping

Good: Didn’t spend lots of time or development effort.

Prototyping

Good: Prototyped the UI with pen and paper for the client.

Prototyping

(Really) Bad: Could have cut project size in half with a prototype.

Prototyping

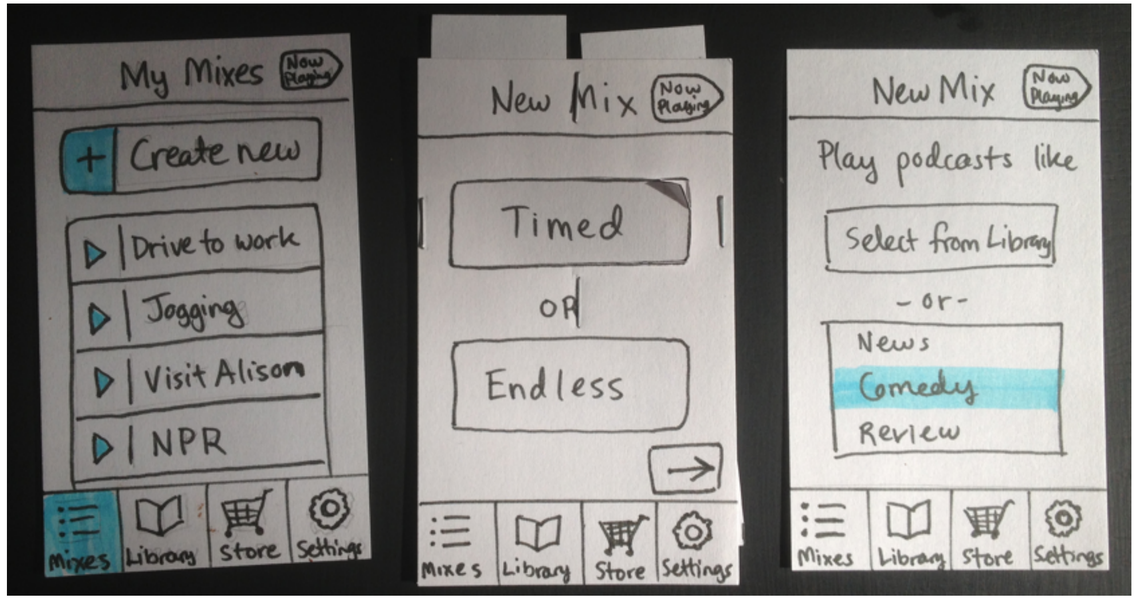
(Really) Bad: Could have cut project size in half with a prototype.

Prototyping

Bad: UI Prototype wasn’t interactive and didn’t explore other options.

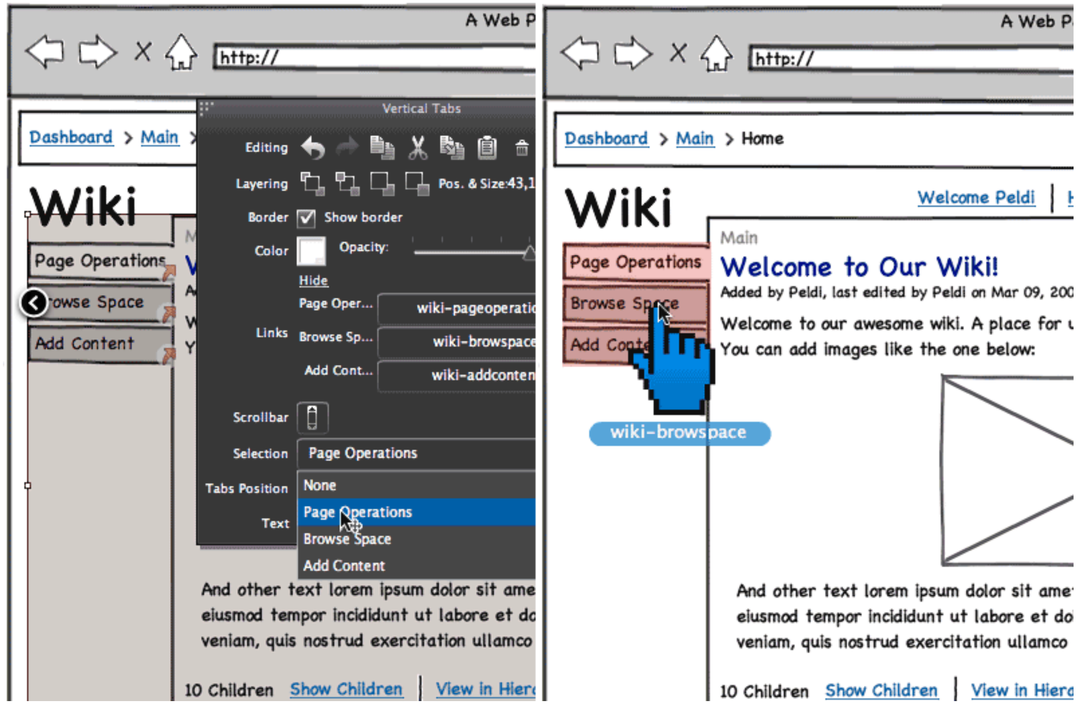
Prototyping Tools

Paper, Whiteboard



Prototyping Tools

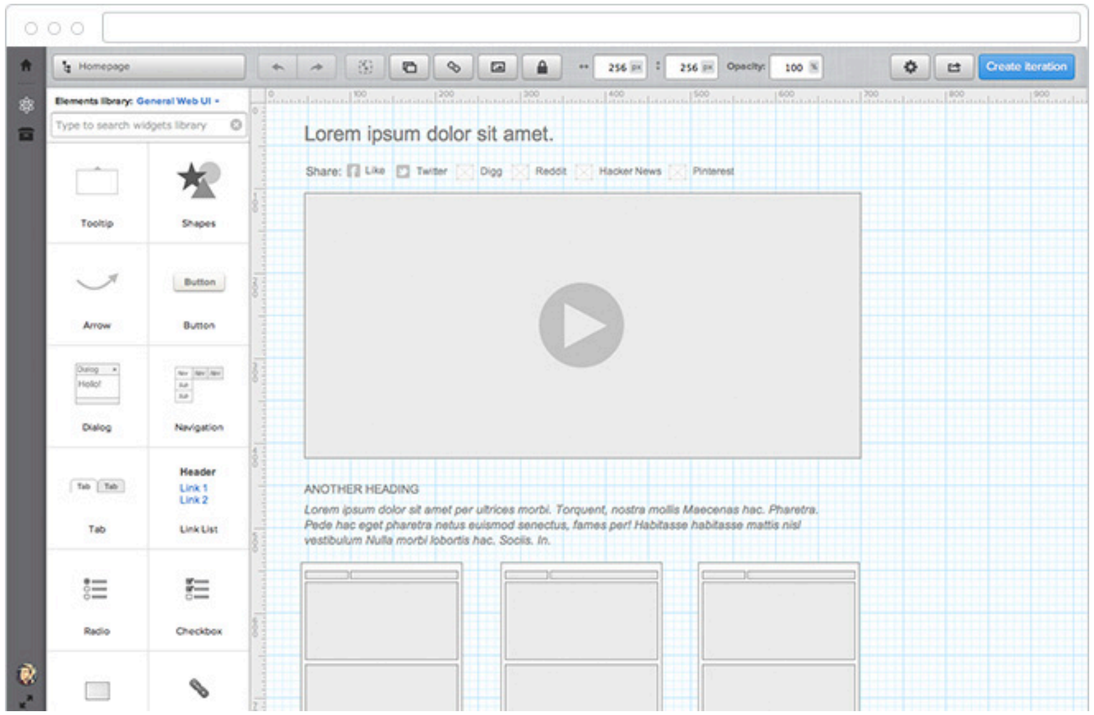
Balsamiq



http://www.balsamiq.com

Prototyping Tools

UXPin



http://www.uxpin.com

Questions?

gavin@gavinballard.com