

Usability and software developers

*A tale of **Usability Professionals**,
Software Developers and
Managers*

Presented by:

Rob Knight

Lead Technical Architect, PRWD

Northern User Experience

20th January 2009

<http://northern-user-experience.org/>

Introduction

What are the problems faced by **usability professionals** and **software developers** working together as a team?

Clash of Cultures

Software Developers

Think in terms of tools

Value reusability

Want to create elegant systems

Usability Professionals

Think in terms of users

Value speciality

Want to create elegant interfaces

How software developers think

Reusability is a core **discipline**

It means:

Lower **cost**

Fewer **bugs**

Easier to **maintain**

Lines of code

“Measuring programming progress by lines of code is like measuring aircraft building progress by weight”

- Bill Gates

The Pareto Principle

80% of the work comes from
implementing **20%** of the features

The Pareto Principle

But the most efficient amount of effort is

zero

Examples

A **website** reuses features of the **web browser**

A **web browser** reuses features of the **GUI**

A **GUI** reuses features of the **Operating System**

This is why we're stuck with some outdated concepts – replacing them is too difficult

Enter UCD

User-centered design leads to specialisation and the creation of **new things**

The results are better for **users**

But not so good for **developers?**

Chinese Whispers

Do the developers really understand **why usability matters?**

Without a direct line of communication, the input of the usability professional can be **distorted**

Developers may feel that user-centered designs are **unreasonable** if they cannot see the benefit

La Resistance

UCD might force developers out of their **comfort zone**

“This isn't a design, it's just a set of pictures!”

The developers aren't seeing the **bigger picture**

Management

Fortunately for the project, one man can link the whole thing together...



DILBERT

Or maybe not

Often, managers don't understand **usability** or **software development**

As a result, managers cannot impose usability on a software development team

They just want to get something **delivered**

Who suffers most?

When a project goes wrong, the **blame game** can begin

External consultants are first in the firing line

Developers may be able to blame '**unworkable**' designs

Who suffers most?

Developers can **hold up** a project if they're not happy

By the time the project is apparent, it may be **too late** for you to do anything about it

The Solution

Communication

Collaboration

Consideration

Know your team

Work **with** the developers and not against them

Learn their **capabilities** and their **limitations**

Understand that usable solutions have **costs for the developers**

Not manufacturing

Software development is **not manufacturing**

Software design is **not product design**

It is a two-way process

Not manufacturing

Software development is **not manufacturing**

Software design is **not product design**

It is a **two-way process**

Agile

Software developers have been embracing **new methods** of working

Agile development focuses on **achievement** of concrete goals

Emphasis on prototyping reduces the cost of trying **something different**

The Perfect Fit?

Agile

Prototype, revise,
rebuild

Focus on high-value
components

Do what works

Collaborate with
stakeholders

UCD

Prototype, revise,
redesign

Focus on key user
goals

Do what users want

Bring the users into
the process

The Perfect Fit?

Agile

Prototype, revise,
rebuild

Focus on high-value
components

Do what works

Collaborate with
stakeholders

UCD

Prototype, revise,
redesign

Focus on key user
goals

Do what users want

Bring the users into
the process

Prioritise

Work out what's **most important**

Collaborate with developers in estimating the cost

Work on completing the **top priorities** before the small details trip you up

Summary

Insist on **direct lines of communication**

Explain the **user's point of view**

Make it **fun!**