

# Software Craftersmanship

Crafting the Software

---

by Evren Tan

# Who am I?

---



# What is Software Craftsmanship?

## What Wikipedia says;

Software craftsmanship is an approach to software development that emphasizes the coding skills of the software developers. It is a response by software developers to the perceived ills of the mainstream software industry, including the prioritization of financial concerns over developer accountability.

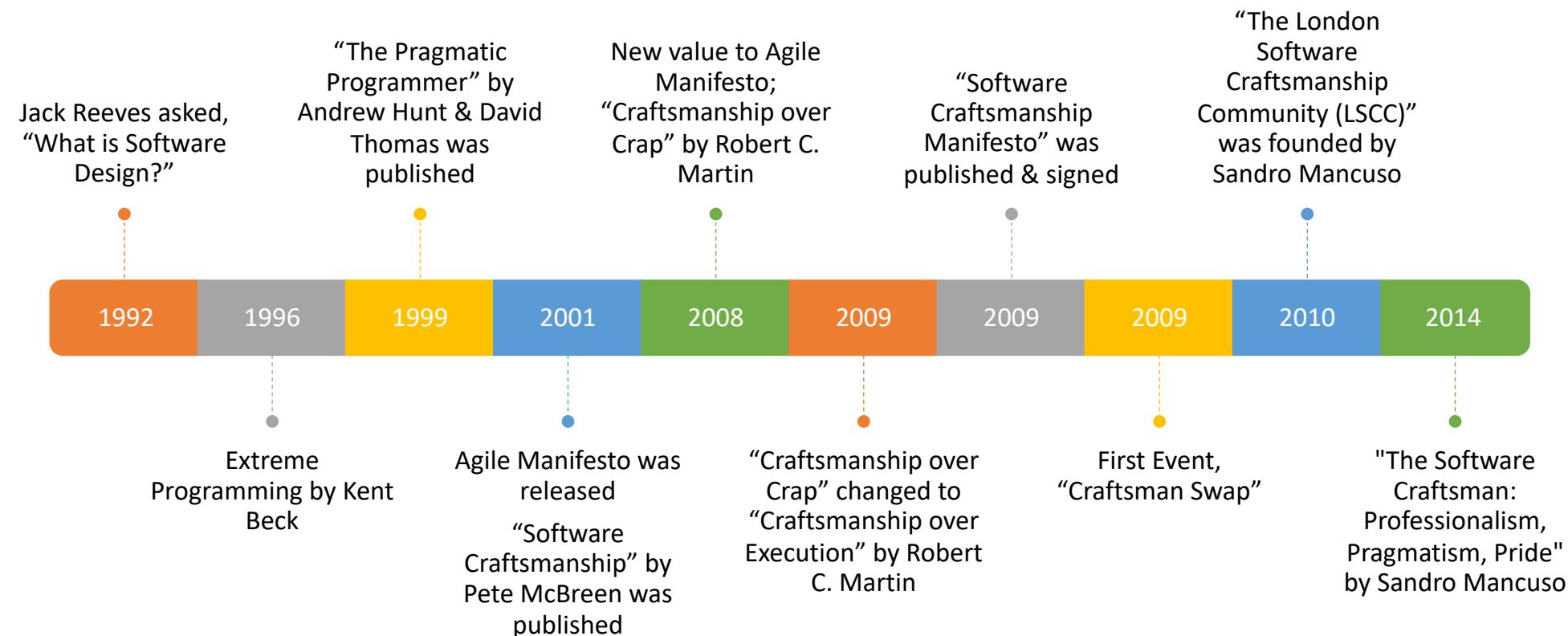
## What Sandro Mancuso says;

Software craftsmanship is about professionalism in software development.

## What Evren Tan says;

Software crafting is all about adding continuous value to oneself and to the community with continuous improvement.

# History





Software Craftsmanship  
Manifesto

# Software Craftsmanship Manifesto

---

Not only working software, but also **well-crafted software**

---

Not only responding to change, but also **steadily adding value**

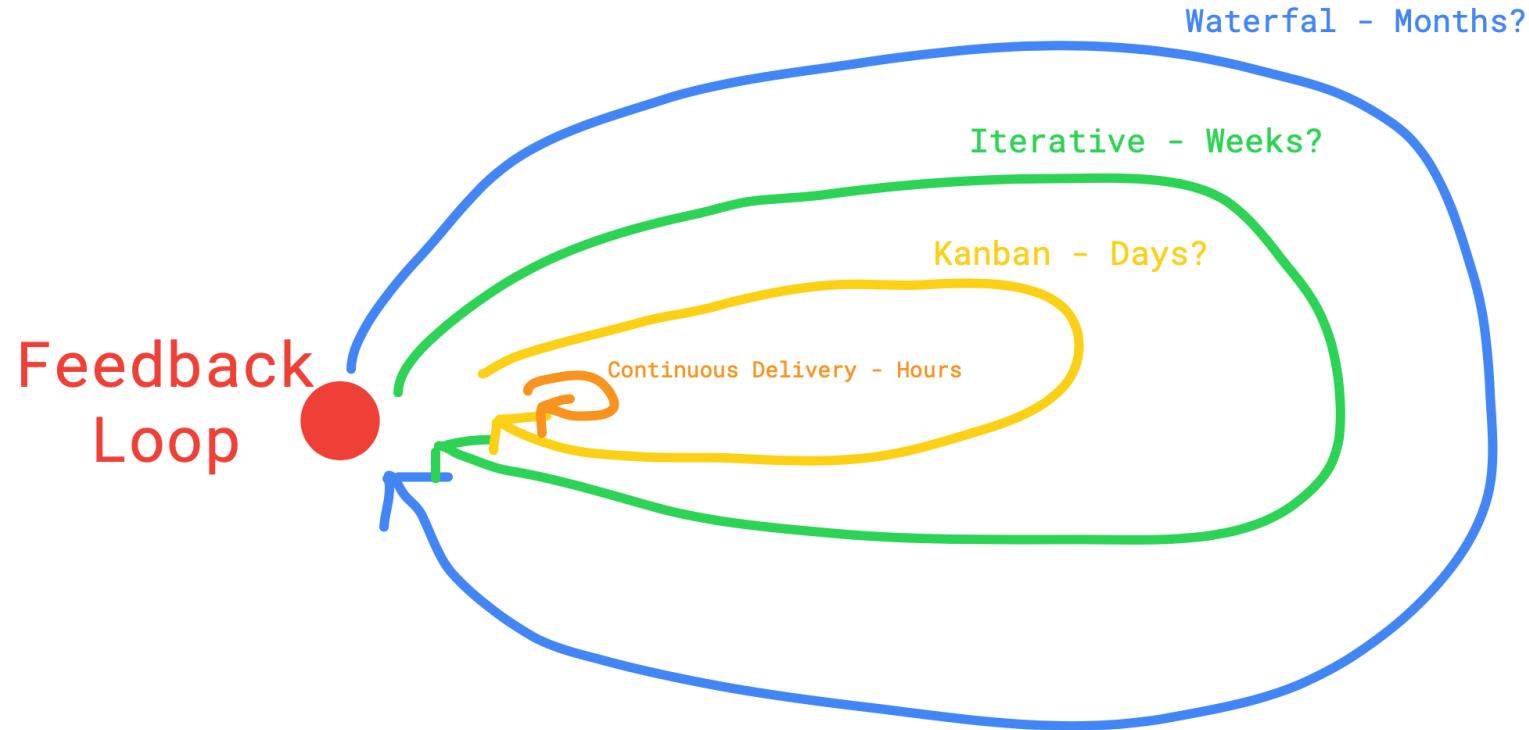
---

Not only individuals and interactions, but also **a community of professionals**

---

Not only customer collaboration, but also **productive partnerships**

# The Keyword – Feedback Loop



# Techniques for Crafting the Software

Test Automation

Pair Programming

Code Reviews

Refactoring

Continuous Integration

Pragmatic Programming



# What is the Goal?



Improve Know-How



Maintainable & Readable Code



Scalable Architecture



Reliable Application



Improve Quality / Reduced # of Bugs



Minimize OPEX



# What Attributes should a Software Crafter Have?



Professionalism

Owning Your Career  
Express Yourself  
Be Pragmatic  
Leadership



Continuous Improvement

Read  
Practice  
Follow



Be a Social Crafter

Join Communities  
Contribute



Balance Your Life

Pomodoro Technique

# Working Software? Well-Crafted Software



Doing the Right Thing vs Doing the Thing Right? 🤔



Technical Debts? ↗



Accountability – “Programming is more like gardening” from “The Pragmatic Programmer” 🌱



Legacy Code? 🥴



Time Management? ⏳

# What is NOT Software Crafting?

Software Engineering

Anti-Agile Pattern

Forcing Every Developer for Crafting

Not Writing Crap Code



PYPL –  
Popularity of  
Programming  
Language



# Reading Suggestions

Software Craftsman, The: Professionalism, Pragmatism, Pride by Sandro Mancuso

Clean Architecture: A Craftsman's Guide to Software Structure and Design by Robert C. Martin

Clean Code: A Handbook of Agile Software Craftsmanship by Robert C. Martin

Refactoring: Improving the Design of Existing Code by Martin Fowler

The Pragmatic Programmer: From Journeyman to Master by Andy Hunt & Dave Thomas

