



Stichting NIOC

Stichting NIOC en de NIOC kennisbank

Stichting NIOC (www.nioc.nl) stelt zich conform zijn statuten tot doel: het realiseren van congressen over informatica onderwijs en voorts al hetgeen met een en ander rechtstreeks of zijdelings verband houdt of daartoe bevorderlijk kan zijn, alles in de ruimste zin des woords.

De stichting NIOC neemt de archivering van de resultaten van de congressen voor zijn rekening. De website www.nioc.nl ontsluit onder "Eerdere congressen" de gearchiveerde websites van eerdere congressen. De vele afzonderlijke congresbijdragen zijn opgenomen in een kennisbank die via dezelfde website onder "NIOC kennisbank" ontsloten wordt.

Op dit moment bevat de NIOC kennisbank alle bijdragen, incl. die van het laatste congres (NIOC2025, gehouden op donderdag 27 maart 2025 jl. en georganiseerd door Hogeschool Windesheim). Bij elkaar zo'n 1500 bijdragen!

We roepen je op, na het lezen van het document dat door jou is gedownload, de auteur(s) feedback te geven. Dit kan door je te registreren als gebruiker van de NIOC kennisbank. Na registratie krijg je bericht hoe in te loggen op de NIOC kennisbank.

Het eerstvolgende NIOC vindt plaats op 18 maart 2027 in Arnhem en wordt georganiseerd door HAN University of Applied Sciences.

Reacties over de NIOC kennisbank en de inhoud daarvan kun je richten aan de beheerder:

R. Smedinga kennisbank@nioc.nl.

Vermeld bij reacties jouw naam en telefoonnummer voor nader contact.

Is eXtreme Programming nog extreem?



Copyright © 2003 United Feature Syndicate, Inc.

- Even voorstellen, waarom deze voordracht?
- Wat is XP?
- Onderwijsvormen
- Opzet van het thema XP bij de Hanze
- Ervaringen, voor- en nadelen
- Toekomstige ontwikkelingen
- Conclusies: Is XP nog extreem?

- Jan Baljé (j.w.balje@pl.hanze.nl)
- 6 jaar bij Hanzehogeschool, opleiding Informatica
- “profieleigenaar” Software Engineering
- Waarom deze voordracht?
- Ervaring in de zaal?

- Software ontwikkelmethodiek, ontstaan eind jaren '90 (Beck, Cunningham, Jeffries).
- Probeert (ook) de problemen rond traditionele methodieken te repareren
 - Requirements creep
 - Verschillende visies op eindproduct
 - Overkill van proces en documentatie
 - Pieken in werkuren
- Lichtgewicht ('agile')
- Hype, gospel, controverser

- XP benadrukt
 1. *communicatie*,
 2. *feedback*,
 3. *simplicity* en
 4. *embracing change*
- Zinvolle uitgangspunten extreem doorgevoerd
 - Regelmatig klantcontact → Klant onderdeel ontwikkelteam
 - Vroeg en vaak testen → Testen vooraf schrijven
 - Kennisoverdracht code → Iedereen mag alles wijzigen
 - ...
- 12 *practices* die elkaar aanvullen
- Practices zijn op zich niet nieuw, wel de manier waarop

#1
Whole team

#7
Pair programming

#2
Small releases

Team Practices

- Whole team sits together in one room
- Work at a sustainable pace
- Integrate many times per day
- Small releases
- Regular

Team interaction

XP Room

Dynamic pairs write all production code

overall schedule

RP It. It. It. Release RP It. It. It. Release

RP=Release Planning (1-3 weeks)
It.= Iteration (fixed length, 1-3 weeks)
Release to users every 1-3 months

Integration

customer

Index Card
"Headline"

Conversations

Customer Tests

- Tied to stories
- Automated

#3
Planning game

#8
Collective Ownership

iteration planning

select stories

Yesterday's Weather:
Select as many points as were finished last iteration

#4
Test First

#5
Refactoring

#6
KISS, YAGNI

Design Philosophy

- Design is evolutionary
- Pay as you go: Build today's requirements
- Keep design as simple as possible (but no simpler)
- High quality is both a side effect and an enabling factor
- The code says everything "once and only once"

programming

Incremental Test-First Programming

```
graph TD
    Write[Write a test] --> Implement[Implement just enough to pass]
    Implement --> Refactor[Refactor]
    Refactor --> Write
```

Cycle takes 5-15 minutes

Refactoring

Stepwise design improvement via safe transformations

A	B
f()	

»

A	B
	f()

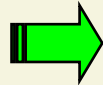
Example: Move Method

- ... gaan je requirements dan niet schuiven?
- ... moet er geen ontwerp gemaakt worden dan?
- ... is de on-site customer wel representatief?
- ... kun je dit wel doen met grote ontwikkelteams?
- ... wat als je ontwikkelaars op verschillende locaties zitten?
- ... hoe zit het met documentatie?

Onderwijsvormen (of: wat is vandaag in de mode?)

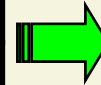
1. "Klassiek"

Vak	t
Vak	t
Vak	t
Vak	t



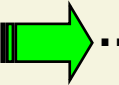
2. Intro projectwerk

Project
Vak t
Vak t
Vak t

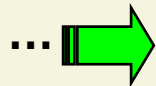


3. Project voorbereiden

Project	
Vak	t
Vak	t
Vak	t



4. 4CID



LT1	LT2	LT3
OT	OT	OT
OT	OT	OT



5. Full-time project

Vak	t	Project/ Leertaak
Vak	t	
Vak	t	

6. GWVG



LT1	LT2	LT3
Vak		t
Vak		t

Week 1-4

XP instructie

XP methodiek, tooling,
refactoring, OO principles,
case (game, pinautomaat)

Design Patterns

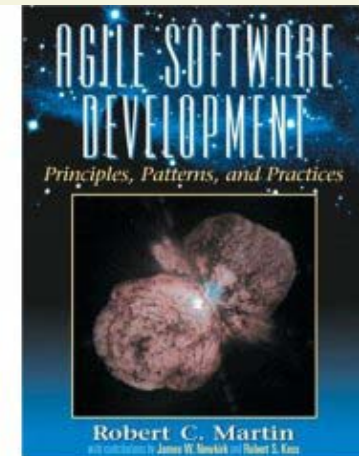
theorie en toepassing van
patterns,
payroll case study

Week 5-8

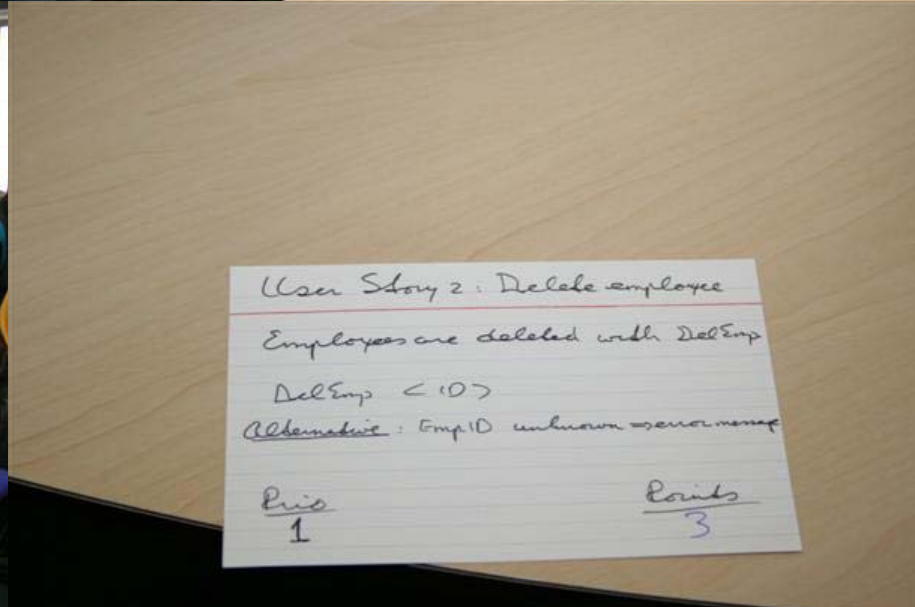
Project Taxi XPress

Projectgroepen van
6 studenten ontwikkelen in
4 weken full-time een
applicatie voor een
taxicentrale.

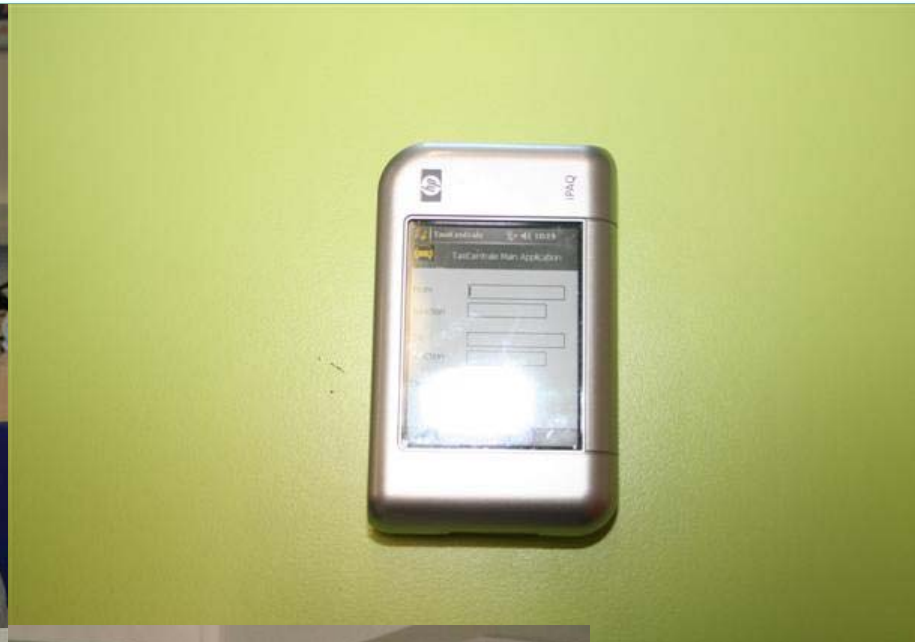
Begeleiding van
opdrachtgever
(a-technisch) en docent
(inhoud, proces)



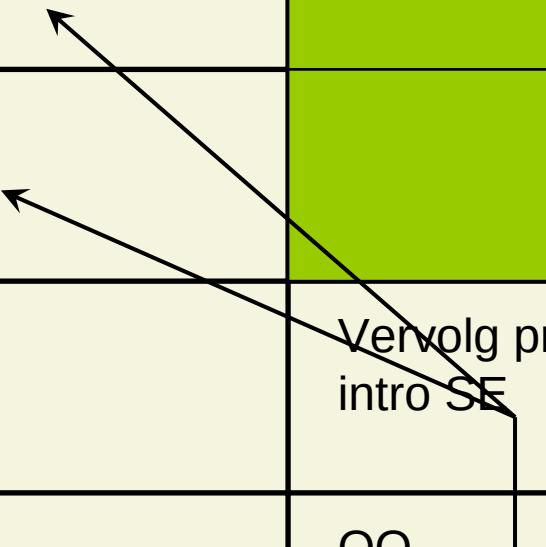
Foto's (1)



Foto's (2)



4	Afstudeerprofiel SE (of Web of ITSM)			
3	Vrije keuzeruimte			
2			Vervolg prog. / intro SE	
1			OO programmeren (BlueJ)	



- Theorie versus praktijk
- License to hack?
- Veel positieve energie
- Hoe ervarener hoe beter
- Werkveld en studenten zijn positief
- Testen wordt weer leuk (eventjes)

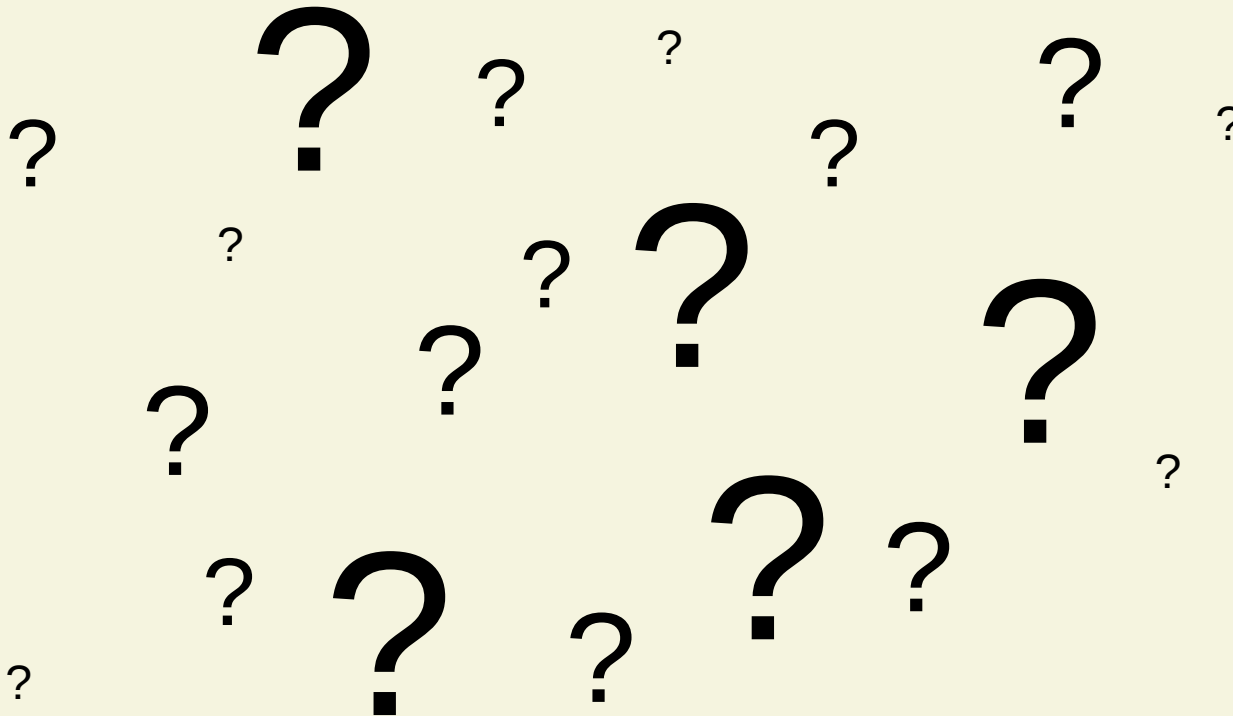
- Design patterns scoren goed
- Studenten ervaren de essentiële SE-ingrediënten op een andere manier
- Veel contact met open source tools
- Vereist flexibele docentinzet
- Feedback op code en XP practices

- Onderdeel van profiel SE
- Tools opfrissen
- GUI Design erbij
- Open Source erbij
- TI-component erbij

XP		Embedded SE
Profiel SE:	XP instructie	o.a. UPEDU
	Design Patt.	
	GUI design	
	Open Source	
Project (incl. PDA)		

Conclusies: Is XP nog extreem?

- XP als methodiek wordt steeds bekender.
- Thema's met full-time projecten worden steeds gangbaarder.
- XP is een leuk alternatief om studenten te laten werken aan Software Engineering.
- Een methodiek is geen vervanging van individuele kwaliteit of ervaring.



Sites

- http://en.wikipedia.org/wiki/Extreme_programming
- <http://www.xprogramming.com/>
- <http://www.xp123.com>
- en vele, vele andere...

Boek

- *Agile Software Development, Principles, Patterns, and Practices* door Robert Martin, Prentice Hall 2002, ISBN 0135974445
- en vele, vele andere...

