

Kravhantering sammanfattning

- BDD/ATDD/Specification By Example
 - User stories
 - Acceptanctestfall/Testscenarion
- Icke-funktionella krav
- Technical Stories
- Dokumentation

Drive: The surprising truth about what motivates us

Viktigaste slutsatserna ur filmen?

Varför tror ni Spotify fokuserade så mycket på autonomy?

Dagens uppgift:

Muddiest point?

Design - översikt

- Vad är design?
- Domain Driven Design
- Designprinciper
- Design Patterns
- Vad är arkitektur?
- Architectural Patterns
- UML

Vad är design?

google define: design

google define: design

“the act of working out the form of something”

Vad är design?

Design

Varför?
 När?
 För vem? (Mottagare?)
 Resultat?
 Artefakter?

När – Up-Front Design

Analys

- Vart tog Analysen vägen?

Alternativ?

Incremental Design
Evolutionary Design
Emergent Design

Incremental Design

1. Välj ett problem
2. Lös det på **enklast möjliga sätt (KISS!)**
3. Välj ett nytt problem
4. Lös det på **enklast möjliga sätt**
5. Förenkla och tydliggör (Abstrahera, Generalisera)
6. Gå till 3

Vad betyder TDD?

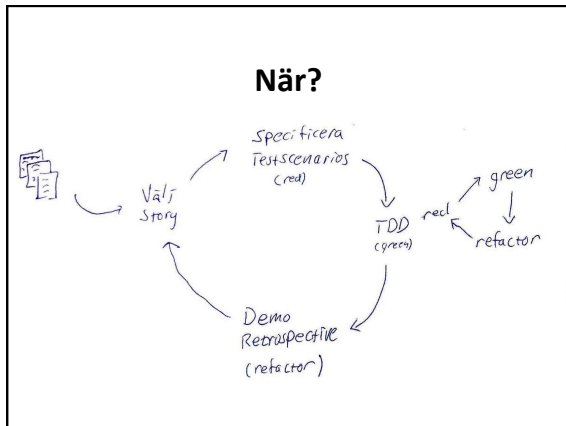
TDD

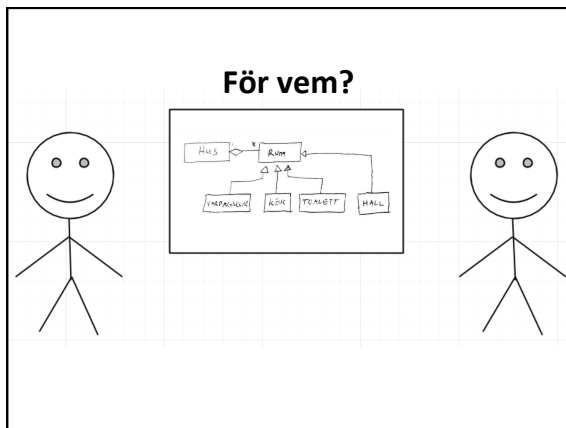
Test Driven **Design**

Coding by intention
Programming by intention

Designelement

Metod
Klass
Komponent
Tjänst
Arkitektur
Etc...





Vad är en bra design?

Designprinciper

- Abstraction
- Decomposition and modularization
- Coupling and Cohesion
- Encapsulation and information hiding
- Consistency (Ubiquitous language)

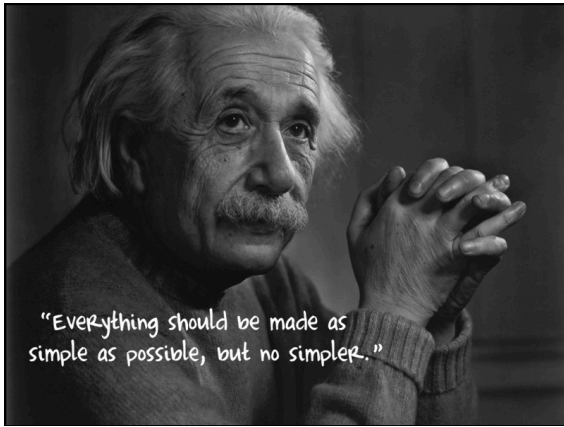
Designprinciper

- Reveal Intent - *Naming*
- SRP: Single Responsibility Principle
 - One and one reason only to change
- DRY: Don't Repeat Yourself
 - Duplication: Missed opportunity for abstraction!
- KISS: Keep it simple, stupid!
- YAGNI: You ain't gonna need it!

Designprinciper – att tänka på

No one gets it right the first time!

Analysis Paralysis



Design pattern

- "Each pattern describes a problem that occurs over and over again in our environment, and then describes the core of the solution to that problem..."
- Christopher Alexander

Software Design Pattern

- "a general reusable solution to a commonly occurring problem within a given context"

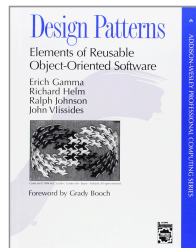
http://en.wikipedia.org/wiki/Software_design_pattern

Design patterns

- Factory
- Builder
- DI
- ServiceLocator

GoF

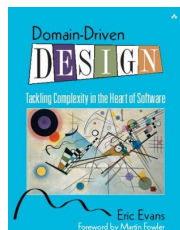
- Creational
 - Builder,
 - Factory method, ...
- Structural
 - Adapter, Decorator, ...
- Behavioral
 - Iterator, Observer, ...



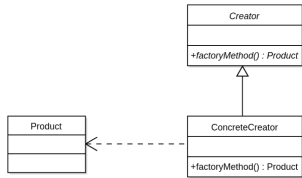
“workarounds for missing features in C++”?

Some DDD Design patterns

- Entity
- Value-Object
- Aggregate
- Repository
- Service
- ...

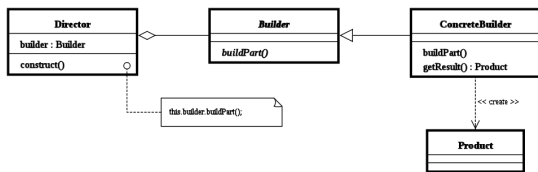


Factory method



- http://en.wikipedia.org/wiki/Factory_method_pattern

Builder



- http://en.wikipedia.org/wiki/Builder_pattern

Pattern Language

“A pattern language is a method of describing good design practices within a field of expertise”

http://en.wikipedia.org/wiki/Pattern_language

