

# Varför testar vi?

Att skaka fram förankrade testuppdrag

SAST, Stockholm, 11-sep-2014

Rikard Edgren

LearningWell Karlstad

[rikard.edgren@learningwell.se](mailto:rikard.edgren@learningwell.se)

# Agenda

1. Mitt största misstag
2. Testuppdrag
3. Exempel
4. Att få fram information
5. Att använda testuppdragen
6. Sammanfattning

# Testuppdrag

- ▶ Syfte: Om du inte vet vad testningen ska bidra med, så är det väldigt svårt att göra bra testning.
- ▶ Definition: Testuppdrag är svaren på frågan **Varför testar vi?**
- ▶ Det är **personer** som ger uppdraget, vet du vilka de är?
- ▶ Riktigt dåligt exempel:
  - Testavdelningen ansvarar för testningen av produkten
- ▶ Vaga exempel:
  - Hitta viktiga problem
  - Ge kvalitetsrelaterad information
  - För att vi måste

# Testuppdrag - exempel

- ▶ Dokumenterade uppdrag:
  - hitta problem som vi vill fixa innan release
  - ta fram beslutsunderlag för ledningen
- ▶ Men där fanns också outtalade uppdrag:
  - ge feedback snabbt till utvecklare och kravställare, så att de kan komma fortare fram
  - uppmärksamma billiga förbättringsförslag
  - verifiera att produkten följer utlovad tillgänglighets-standard
  - informera projektledaren om riskers uppgång och fall
- ▶ Dessa uppdrag var olika viktiga vid olika tillfällen. Det första uppdraget: “hitta viktiga fel” var dock det viktigaste, och det brukar det vara i de flesta sammanhang.

# “så att”-knepet

- ▶ När du har vagt testuppdrag, till exempel:  
*testa produkten*
- ▶ lägg till "så att" och fyll på:  
*så att vi hittar viktiga fel*
- ▶ kanske ytterligare en gång:  
*så att de kan åtgärdas och ge nöjdare kunder och färre supportärenden*
- ▶ Då börjar du få ett uppdrag som verkligen är meningsfullt, och där personer kan fylla i med mer saker:  
*så att vi kan ta välinformerade beslut,  
så att produktrisker har utforskats, så att vi inte får obehagliga  
övertaskningar*
- ▶ Prova småord som "även om", "såvida inte", "till exempel" m.m.

# Ta reda på det som är viktigt

- ▶ Prata med intressenterna!
- ▶ Fråga "Vad vill ni veta?", flera gånger om det krävs.
- ▶ Träna på att ställa följdfrågor.
  
- ▶ Ta reda på vad "viktiga problem", "kvalitet", "risker" betyder.
  
- ▶ Undersök de informationskällor som är relevanta.
  - Specifikationer
  - Kvalitetsmål
  - Teknologier
  - Branschkunskap
  - Riktiga kunder
  - ..., se [37 källor för testidéer](#)

# Software Quality Characteristics

Go through the list and think about your product/features. Add specifics for your context, and transform the list to your own.

# IT-bility. Is the product easy to install and support?

- System requirements: ability to run on supported configurations, and handle different environments or missing components.

## Capability. Can the product perform valuable functions?

- **Completeness:** all important functions wanted by end users are available.
- **Accuracy:** any output or calculation in the product is correct and presented with significant digits.
- **Efficiency:** the product uses resources efficiently.
- **Interoperability:** the product can be rolled-out by IT department to different types of (restricted) users and environments.
- **Conformance:** the product and its artifacts are easy to maintain and support for customers?
- **Data:** how does the product interact with software and environments?
- **Extensibility:** ability for customers or 3rd parties to add features or change behavior.

**IT-vänlig. Är produkten lätt att installera och underhålla?**

## Reliability. Can you trust the product in many and different situations?

- **Stability:** the product shouldn't cause crashes, unhandled exceptions or hangs.
- **Robustness:** the product handles foreseen and unforeseen errors gracefully.
- **Stress handling:** how does the system cope when exceeding various limits?
- **Recoverability:** it is possible to recover and continue using the product after a fatal error.
- **Data:** how does the product interact with software and environments?
- **Safety:** efficiency, switch-offs, power-saving modes, telecommuting.
- **Disaster recovery:** compliance standards, regulations, laws or ethics.
- **Trust:** how does the product interact with software and environments?

**Kompatibilitet. Hur väl interagerar produkten med sin omgivning?**

## Usability. Is the product easy to use?

- **Affordance:** product invites to discover possibilities of the product.
- **Intuitiveness:** it is easy to understand and explain what the product can do.
- **Minimalism:** there is nothing redundant about the product's content or appearance.
- **Learnability:** it is fast and easy to learn how to use the product.
- **Memorability:** once you have learnt how to do something you don't forget it.
- **Discoverability:** the product's information and capabilities can be discovered by exploration of the user interface.

**Support. Kan kundernas användning och problem understöddas?**

**Användbarhet. Är produkten lätt att använda?**

- **Openness:** the product logs actions at appropriate levels and in usable format.
- **Interoperability:** the product logs actions at appropriate levels and in usable format.
- **Control:** ability to independently set states, objects or variables.
- **Clarity:** ability to observe things that should be tested.
- **Errors:** there are informative error messages, difficult to make mistakes and easy to repair after making them.
- **Consistency:** behavior is the same throughout the product, and there is one look & feel.
- **Tailorability:** default settings and behavior can be specified for flexibility.
- **Accessibility:** the product is possible to use for as many people as possible, and meets applicable accessibility standards.
- **Documentation:** there is a Help that helps, and matches the functionality.

**Testbarhet. Är det lätt att verifiera och testa produkten?**

## Charisma. Does the product have "it"?

- **Uniqueness:** will users get interested and try out what they can do with the product?
- **Satisfaction:** do users get hooked, have fun, in a flow, and fully engaged when using the product?
- **Professionality:** should the product use the latest and greatest technologies/ideas?
- **Attraction:** the product exceeds expectations and meets the needs you desire.
- **Curiosity:** do the product and its information have the right attitude and style?
- **Entrancement:** are (first) impressions impressive?
- **Hype:** are there compelling stories about the product's inception, construction or usage?
- **Expectancy:** are there compelling stories about the product's inception, construction or usage?
- **Attitude:** are there compelling stories about the product's inception, construction or usage?
- **Directness:** are there compelling stories about the product's inception, construction or usage?
- **Story:** are there compelling stories about the product's inception, construction or usage?

**Karisma. Har produkten "det"?**

**Underhåll. Kan produkten underhållas och utökas till låg kostnad?**

## Security. Does the product protect against unwanted usage?

- **Authenticity:** the product should under no circumstances disclose information about the underlying systems.
- **Authenticity:** the product should under no circumstances disclose information about the underlying systems.
- **Privacy:** ability to withstand penetration attempts.
- **Security:** product will not transport virus, or appear as one.
- **Secrecy:** no possibility to illegally copy and distribute the software or code.
- **Invulnerability:** security standards the product adheres to.
- **Compliance:** security standards the product adheres to.

**Säkerhet. Skyddar produkten mot oönskat användande?**

## Portability. Is transferring of the product to different environments enabled?

- **Reusability:** can parts of the product be re-used elsewhere?
- **Portability:** can the product be used in a different environment?
- **Interoperability:** support a different environment?
- **Compatibility:** common interfaces or official standards?
- **Localization:** product.
- **Localization:** product.
- **Localization:** product.
- **User Interface-robustness:** will the product look equally good when translated?

**Flyttbarhet. Är det möjligt att flytta produkten till andra miljöer och språk?**

## Performance. Is the product fast enough?

- **Capacity:** the many limits of the product, for different circumstances (e.g. slow network).
- **Resource Utilization:** appropriate usage of memory, storage and other resources.
- **Responsiveness:** how does the product interact with software and environments?
- **Availability:** how does the product interact with software and environments?
- **Throughput:** how does the product interact with software and environments?
- **Endurance:** how does the product interact with software and environments?
- **Feedback:** is the feedback from the system on user actions appropriate?
- **Scalability:** how well does the product scale up, out or down?

**Prestanda. Är produkten tillräckligt snabb?**

"Kvalitetsegenskaper för programvara" (thetesteye.com)



# Detaljerade testuppdrag

- ▶ Det behöver inte vara fel med detaljerade testuppdrag:
  - kolla om webbplatsen kommer klara förväntad belastning till julhelgen
  - undersök säkerhetsrisker för inloggning och användarkonton
  - vi har inte råd med fler supportärenden kring felifyllda formulär, testa all felhantering och begriplighet för mormor
  
- ▶ Brasklapp:
  - Detaljerna kan dölja helheter som är viktigare
  - Det man säger sig vilja ha, är kanske inte det man behöver



# Informera om vad du tror

- ▶ Det är lätt hänt att man tror man har förstått det hela, men även testare kan ha fel...
  - ▶ Berätta därför om vad du tror att testuppdraget är.
  - ▶ Det är då testuppdragen förbättras och förankras.
- 
- ▶ I min erfarenhet är ett samtal vid kaffebryggaren mer effektivt än en godkänd test policy.

# Använda testuppdragen

- ▶ Om du fått fram en del detaljer vid undersökningarna, så har du:
  - så gott som halva teststrategin klar
  - ett giltigt mind set för alla testare
  - ett embryo till rapportering
- ▶ Du har intressenter som är med på noterna, och rapporteringen kommer vara lättare än någonsin.
- ▶ Du har bäddat för värdefull testning.

# Sammanfattning

- ▶ Försök ta reda på vad som är viktigt.
- ▶ Berätta vad du tror är viktigt.
- ▶ Ändra dig när du vet mer.
  
- ▶ Frågor???
  
- ▶ Litteraturtips:
  - [Den lilla svarta om teststrategi](#) (Edgren, opublicerat)
  - [Kvalitetssegenskaper för programvara](#) (Edgren, Emilsson, Jansson)
  - [Heuristic Test Strategy Model](#) (James Bach)

