

# A torn testers tales from the trenches

Good vs evil

**Who's working with Ux/Cx?**

**Who consider themselves one of the good guys?**

**Jörgen – QA nerd**

## People tend to go to conferences to:

- Get away from office for a while
- Meet old colleagues
- Feel they invest in themselves
- Learn how others are doing their work
- Get entertained
- Get a confirmation they don't have to change
- Find some little gem of an idea

**“Ok, that sounds good, but it would never work for us.”**

**“We are to far from that. I wont take responsibility for that.”**



**”A bug is something that bugs  
someone that matters”**



Prioritized stakeholders



Prioritized stakeholders





**Subjective**  
**Context dependent**

# Software quality characteristics

## Förmågor. Erbjuder produkten värdefulla funktioner?

- Komplet: alla viktiga funktioner som önskas av användare är tillgängliga.
- Korrekt: utdata eller beräkningar i produkten är korrekta och presenterade med signifikanta siffror.
- Effektiv: utför sina funktioner på ett effektivt sätt (utan att göra något den inte ska).
- Interoperabilitet: olika funktioner interagerar med varandra på bästa sätt.
- Samtidighet: möjlighet att utföra multipla parallella uppgifter, och att kunna köra samtidigt som andra processer.
- Dataagnostisk: stödjer allehanda dataformat och hanterar databrus.
- Utökningsbar: möjlighet för kunder eller tredjepartsleverantörer att lägga till egna funktioner eller förändra beteendet.

## Pålitlig. Kan du lita på produkten i många och svåra situationer?

- Stabilitet: produkten ska inte krascha, orsaka undantag eller skriptfel.
- Robust: produkten hanterar (o)förutsedda fel på ett behagligt sätt.
- Stresstålighet: hur beter sig systemet när olika gränser överskrids?
- Återhämtning: det är möjligt att starta om och fortsätta efter ett allvarligt fel.
- Dataintegritet: all sorts data behålls intakt genom hela produkten.
- Säkerhet: produkten medverkar inte till att skada personer eller egendom.
- Katastrofhantering: vad händer om någonting riktigt, riktigt allvarligt inträffar?
- Trovärdighet: är produktens beteende konsekvent förutsägbart och trovärdigt?

## Användbarhet. Är produkten lätt att använda?

- Inbjudan: produkten inbjuder till att upptäcka möjligheter för produkten.
- Intuitiv: det är lätt att förstå och förklara vad produkten kan göra.
- Minimalistisk: produktens innehåll eller utseende innehåller inget onödigt.
- Lättlärd: det går snabbt och enkelt att bli van vid produkten så användas.
- Ihågkombarhet: har du gjort något som du ska utföra något, glömmet du inte bort det.
- Utforskningsbar: information om möjligheter och produkter kan upptäckas genom att använda gränssnittet.
- Snabbt: en erfaren användare kan göra enkla uppgifter väldigt snabbt.
- Interaktivitet: produkten har lättinställbara tillstånd och utmärkta möjligheter till interaktion via användargränssnittet.
- Kontroll: användarens känsla av kontroll över programvaran genom hela användarscenariot.
- Klarhet: är allting explicit och detaljerat uttryckt, med ett språk som kan bli förstått och inte är tvivelaktigt?
- Felhantering: felmeddelanden är informativa, det är svårt att göra misstag och lätt att reparera.
- Konsekvent: beteenden är samma genom hela produkten, och det är igenkännligt utseende och känsla.
- Anpassningsbart: förvalda inställningar och beteenden kan specificeras för att uppnå flexibilitet.
- Tillgänglighet: produkten kan användas av så många människor som möjligt och möter aktuella tillgänglighetsriktlinjer.
- Dokumentation: det finns en Hjälpa som hjälper och som beskriver den aktuella funktionaliteten.

## Karisma. Har produkten "det"?

- Unikhet: produkten är säregen och har någonting som andra inte har.
- Tillfredsställelse: hur känner du dig efter att ha använt produkten?
- Professionalism: andas produkten rätt typ av professionalism och känns den anpassad för syftet?
- Attraktivitet: är alla aspekter av produkten estetiskt tilltalande för ögon och andra sinnen?
- Väcker nyfikenhet: kommer användarna bli intresserade att prova sig fram i produkten?
- Uppslukande: blir användarna fast, satta i trans, och helt uppslukade när de använder produkten?

- Hajp: använder produkten för mycket eller för lite av den bästa och senaste tekniken och idéerna?
- Över förväntan: produkten överträffar dina förväntningar och möter de behov du inte visste du hade.
- Attityd: har produkten och dess information den rätta attityden och pratar till dig med rätt språk och stil?
  - Förstaintryck: är (första) intrycken av produkten imponerande?
  - Historier: finns det intressanta historier om produktens uppkomst, konstruktion eller användande?

## Säkerhet. Stoppar produkten oönskat användande?

- Autentisering: produktens igenkännande av användare.
- Auktorisering: produktens hantering av vad autentiserade användare kan se och göra.
- Privatheit: förmågan att inte avslöja data som är skyddad mot icke-auktoriserade användare.
- Säkerhetsluckor: produkten ska varken erbjuda eller inbjuda till sårbarhetsintrång.
- Hemlighetsfull: produkten ska under inga omständigheter avslöja information om underliggande system.
  - Osårbarhet: förmågan att motstå försök att penetrera säkerheten.
  - Virusfri: produkten ska inte vidarebefordra virus eller intrycket av att vara ett.
  - Piratkopieringsskydd: ingen möjlighet att illegalt kopiera och distribuera programvaran eller koden.
  - Reglementsnylig: produkten uppfyller tillämpliga säkerhetsstandarder.

## Prestanda. Är produkten tillräckligt snabb?

- Kapacitet: produktens möjlighet (begränsningar) att hantera olika situationer (t.ex. slöta nätverk)?
- Resursnyttjande: välanvändande av processor, minne, lagring och andra resurser.
- Respons: (den upplevda) snabbheten när funktioner utförs.
- Upptid: systemet är tillgängligt för användande vid alla tidpunkter det ska vara.
- Genomströmning: produktens förmåga att processera många saker samtidigt.
  - Uthållighet: kan produkten hantera last över lång tid?
- Återkoppling: svarar systemet tillräckligt snabbt när användare utför funktioner?

- Skalbarhet: hur bra skalar systemet upp, ner, eller ut? IT-vänlig. Är produkten lätt att installera, underhålla och supporta?
- Systemkrav: förmågan att köra på stödda konfigurationer och att hantera olika miljöer eller saknade komponenter.

- Installerbarhet: produkten kan installeras på avsedda plattformar och använder en rimlig storlek av hårddiskutrymme.

- Uppgradering: det är enkelt att uppdatera till en ny version utan att tappa konfigureringsdata och inställningar.

- Avinstallation: alla filer (förutom användar- eller systemfiler) och andra resurser ska tas bort under en avinstallation.

- Konfiguration: kan installationen konfigureras på olika sätt eller ställen för att stödja kundens användande?

- Utrullningsbarhet: produkten kan rullas ut av IT-avdelningen till olika typer av (begränsade) användare och miljöer.

- Underhåll: är produkten och dess artefakter lätta att underhålla och supporta för kunderna?
  - Testbarhet: hur effektivt kan den utrullade produkten bli testad av användarna?

## Kompatibilitet. Hur väl interagerar produkten med andra programvaror och miljöer?

- Hårdvarukompatibilitet: produkten kan användas tillsammans med applicerbara konfigurationer av hårdvarukomponenter.

- Operativsystemkompatibilitet: produkten kan köra på avsedda versioner av operativsystemen och följer deras riktlinjer.

- Programkompatibilitet: produkten, och dess data, fungerar ihop med andra program som kunderna använder.

- Konfigurationskompatibilitet: produktens förmåga att smälta in med omgivningens konfigurationer.
  - Bakåtkompatibilitet: kan produkten göra allt den förra versionen kunde?

- Framåtkompatibilitet: kommer produkten att kunna använda artefakter eller gränssnitt i framtida versioner?

- Hållbarhet: påverkan på miljön, t.ex. energieffektivitet, elbesparingslägen, möjlighet till distansarbete.

- Reglementsnylig: produkten uppfyller applicerbara standarder, regelverk, lagar, riktlinjer eller andra hållningssätt.

## Support. Kan kundernas användning och problem understöddas?

- Identifierbarhet: det är lätt att identifiera delar av produkten och deras version, eller specifika fel.
- Diagnostisering: är det möjligt att få detaljerade kringkundsituationer?
- Undersökning: är det lätt att sätta fingret på fel (t.ex. loggfil) och hjälpa?
- Avlusning: kan du observera programmets interna tillstånd när det behövs?
- Möjliggör: möjligt att använda produkten på fler sätt än vad det designades för.

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

- Spårbar: produkten loggar händelser på lämplig nivå, och i användbart format.
  - Styrbar: möjligheten att sätta tillstånd, objekt och variabler.
  - Observerbar: möjligheten att observera det som testas.
- Övervakningsbar: kan produkten indikera hur och vad den gör?
  - Isolerbar: möjligheten att testa en del för sig själv.
- Oföränderlig: förändringar till programvaran görs kontrollerat, och inte för ofta.

- Automation: finns det publika eller interna programmatiska gränssnitt som kan användas?
  - Information: möjligheten för testare att lära sig det man behöver veta...
  - Granskningsbar: kan produkten, och sättet den skapats på, valideras?

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

- Flexibel: är det lätt att ändra produkten för att möta kunders behov?
  - Utökningsbar: blir det lätt att lägga till funktionalitet i framtiden?

- Enkel: koden är inte mer komplex än vad som krävs, och försvårar inte testdesign, testekekvering och resultatolkning.
  - Läsbar: koden är dokumenterad och lätt att läsa och förstå.
  - Genomskinlig: Är det lätt att förstå de underliggande strukturerna?
    - Modulär: koden är uppdelad i lätthanterliga bitar.
    - Omskrivningsbar: är du nöjd med enhetstesterna?

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

- Återanvändning: kan delar av produkten användas på andra ställen?

- Anpassningsbar: är det lätt att förändra produkten så den fungerar i en annan miljö?
  - Kompatibel: följer produkten vanliga beteenden, eller officiella standards?
    - Internationalisering: det är lätt att översätta produkten.

- Lokalisering: är alla delar av produkten anpassade för att möta behoven av det nya landet/kulturen?
  - Robusta gränssnitt: kommer produkten se lika bra ut när den översatts

# FURPS+

**Functionality**

**Usability**

**Reliability**

**Performance**

**Security**

FUDD

+

Affects  
experience

**Speed of  
change**





Public Domain



Good  
guy









**Business**



**IT**



**Operations**





**Business**

**Agile**



**IT**



**Operations**



**Business**



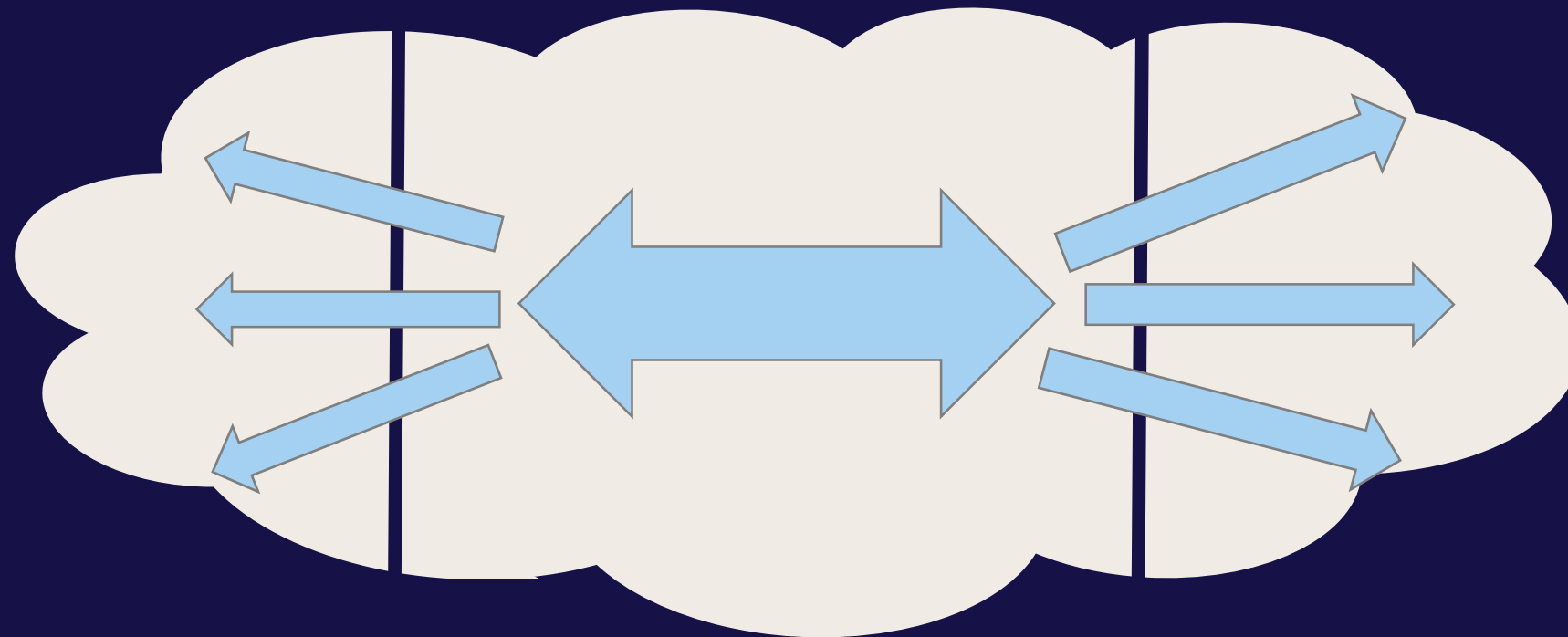
**IT**



**Operations**

**DevOps**

**Agile**



**SEP**

# The evilness of Ux/Cx

**Any success can turn to  
nightmare at any time**

**It's not about designing user  
interaction**

**Experience is not constructed.  
It's maintained.**

# Continuous appreciation - robustness

Incident analysis reveals:

Performance problems

Initial

Creeping

Reliability problems

Slow adaption speed

Security breaches

Not my  
problem?

*But it is*



# Organization success – Speed & versatility

Prepare architecture and infrastructure for testing

Use automated CHECKING

Make sure application development never is  
from business

Not my  
problem?

*But it is!*

Are your customers only  
your users?

# Metrics for success

# Performance and reliability

Make sure to do performance tests  
Not once, but again and again

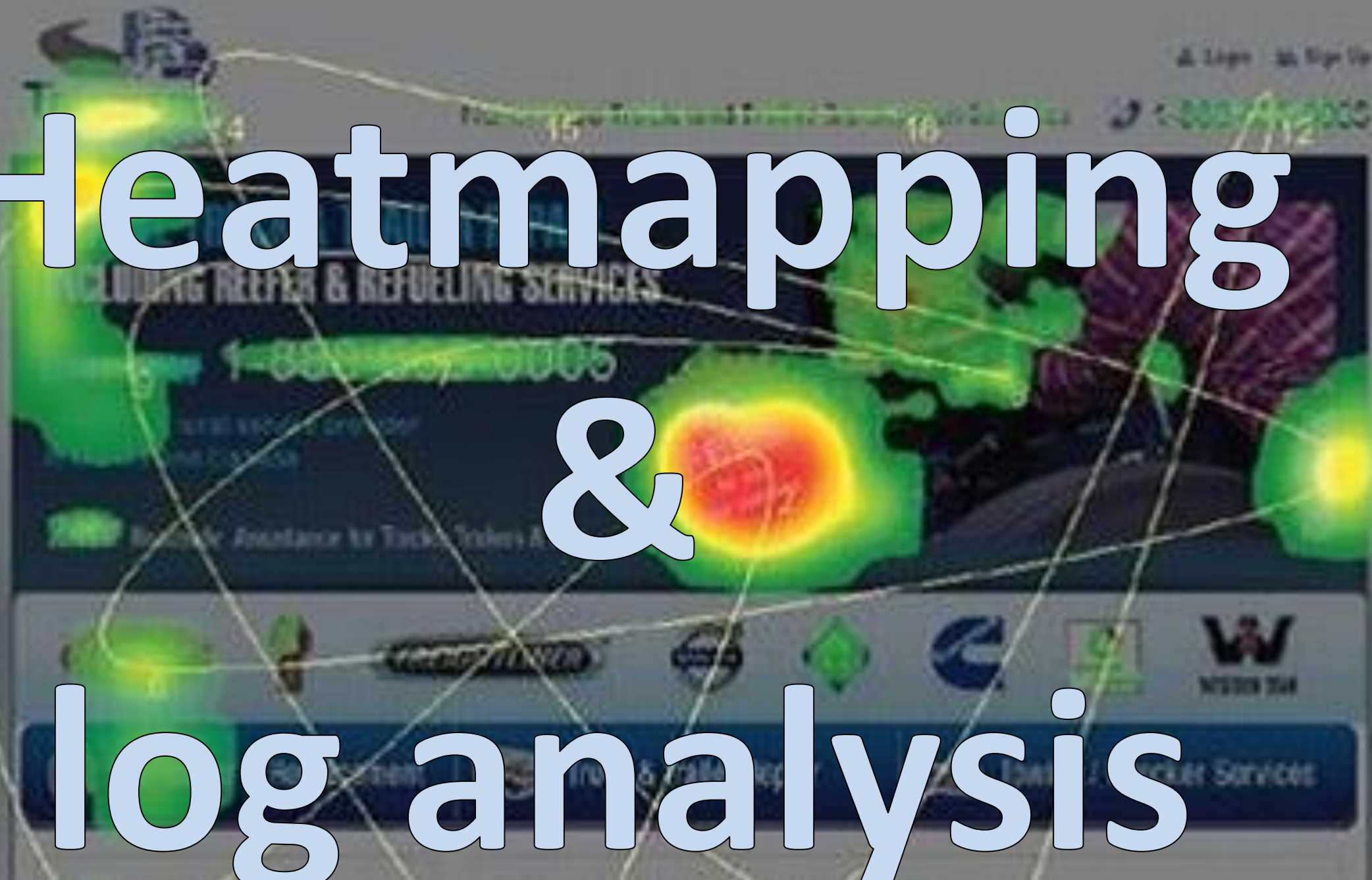
Do security assessments

Verify your operation system monitoring scheme is working

Use KPIs for application change time

Automate your functional checking

# Heatmapping & log analysis



**What gets measured get  
tampered with**

# People react differently to being measured

- Some get competitive
  - Within teams, and between teams
- Some get rebellious
- Some get apatic

Tend to drive behaviour, like when you might still drop by Operations, but have to raise a ticket afterwards

**Don't shy away from  
subjective metrics**

”Ok, that sounds good, but it would never work for us.”

”We are to far from that. I wont take responsibility for that.”