

NÁSKOK
DÍKY
ZNALOSTEM

PROFINIT

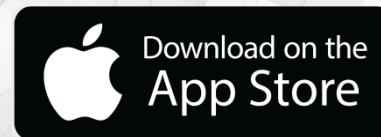
Testování prakticky

Otakar Ertl

17. ledna 2018



Dotazy na
<https://www.sli.do>
event #W485



Agenda

- › Testovací proces a jeho fáze
- › Defekty a jejich životní cyklus
- › Testovací prostředí
- › Reporting
- › Měření a jeho důležitost
- › Automatizace testování

Cíl přednášky:

Cílem je porozumět principům, pojmům, závislostem a problémům testování.

Testovací proces

Fáze testovacích prací

- › Odhad pracnosti
- › Volba a sepsání test strategie
- › Plánování
- › Test analýza
- › Test Design
- › Test exekuce
 - Nová funkcionalita
 - Regresní testy
 - Retesty defektů
- › Akceptace
- › Uzavření testování

The background of the slide is composed of numerous overlapping, translucent, light-gray geometric shapes. These shapes, which include rectangles, trapezoids, and irregular polygons, are arranged in a way that creates a sense of depth and movement, as if they are floating or falling from the top right towards the bottom left. The overall effect is a complex, layered pattern that provides a modern and abstract visual context for the text.

Odhad pracnosti

Odhad pracnosti

- › Různé techniky pro
 - Nový projekt
 - Pokračování stávajícího

- › Nový projekt
 - Volba vhodné metriky – Obrazovky, UC, funkcionality
 - Nad malou reprezentativní částí udělat detailní odhad
 - Nezapomenout započítat čas na testovací kola a retesty defektů
 - Identifikovat rizikové oblasti
 - Kontrola vůči odhadu vývoje (20 – 100 % vývoje dle požadavku na kvalitu)

- › Pokračování stávajícího
 - Použití stejných metrik jako v předchozím releasu s tím, že známe reálné časy získané měřením
 - Vyhradit dostatečný čas na rizikové oblasti
 - Pokud funguje dobře odhad pracnosti vývoje lze dopočíst konstantu pro odhad testů



The background of the slide is a complex, abstract composition of numerous overlapping, translucent geometric shapes. These shapes, which include various polygons and rectangular prisms, are rendered in different shades of light gray. They are arranged in a way that creates a sense of depth and three-dimensional space, with some shapes appearing to float above others. The overall effect is a modern, architectural aesthetic.

Plánování testů

Plánování testů

Test strategie

Vstupní požadavek: Požadovaný nárok na kvalitu

- **Test coverage** (co testovat a co netestujeme)
- **Test methods and tools** (jak testovat, jakými metodami)
- Test environment (na jakých prostředích, datech a zařízeních)
- Test order (pořadí / sequence testů)

Test plan

- › Kdo
- › Kdy (test kola)
- › Co (konkrétní testy)
- › Jak dlouho
- › Kdy a na čem (prostředí, zařízení, data)



Plánování testů

Jak neuspět

- › Zapomenout, že testují lidé
- › Předstírat, že testeři jsou odpovědní za kvalitu, nikoli management
- › Diktovat datum spuštění bez ohledu na reálná omezení projektu
- › Hodnotit testery podle počtu nalezených chyb
- › Nedostatek vzdělávání pro testery
- › Oddělit vývoj a testování



The background of the slide is a complex, abstract composition of numerous overlapping, translucent geometric shapes. These shapes, which include various polygons and rectangular prisms, are rendered in different shades of light gray and white. They are arranged in a way that creates a sense of depth and movement, with some shapes appearing to float above others. The overall effect is a modern, architectural aesthetic.

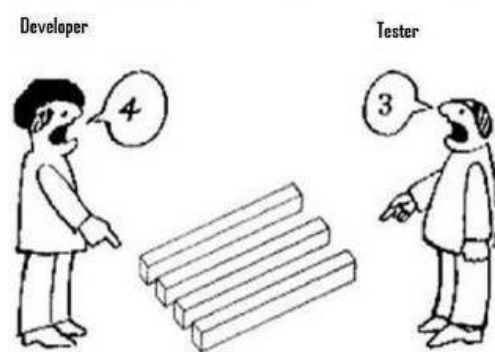
Návrh testovacích případů

Test analýza a design

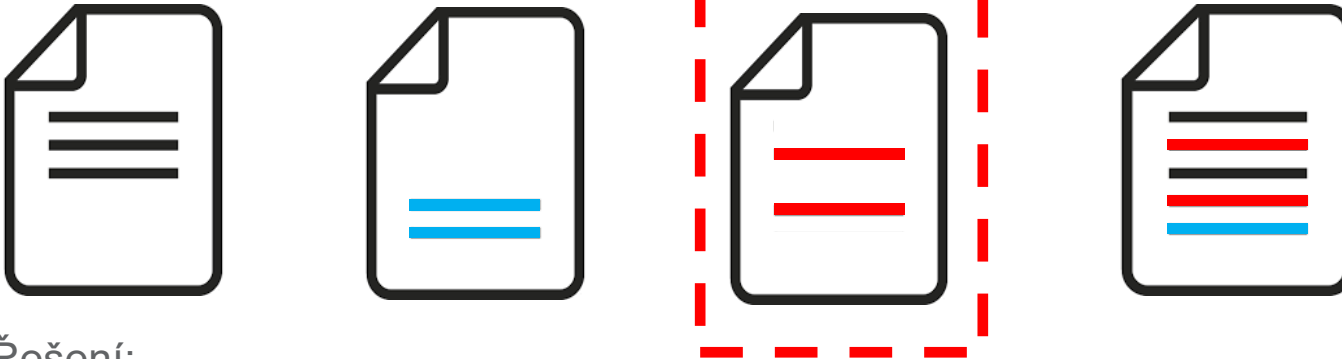
Testovací dokumentace - vstupy

› Specifikace

- Textová forma
 - Jednotlivé požadavky (requirements)
- Grafická forma (UML)
 - Grafy
 - Sekvenční, Activity, Obrázovky.....



Nevýhody inkrementální dokumentace



Řešení:

Místo jediné pravdy – dokument popisující celý současný stav aplikace

Testovací dokumentace – co vytváříme

- › **Test case (testovací případ)** – vytváří Test analytik/Tester
 - Množina instrukcí (kroků), které budou provedeny na testovaném systému s cílem zjistit, zda systém funguje, jak je očekáváno
- › **Test script (automatický test)** – vytváří Test analytik/Tester
 - Množina instrukcí (kroků), které budou automaticky provedeny testovacím nástrojem na testovaném systému s cílem zjistit, zda systém funguje, jak je očekáváno
- › **Test data (testovací data)** – vytváří Test analytik /Tester
 - Data speciálně identifikovaná pro využití v rámci testovacího případu
- › **Test report (výsledky testu)** – vytváří Tester agreguje Test manager
 - Výsledek jednoho či více testů obsahující minimálně identifikaci testu a jednoznačný výsledek společně s komentářem, je-li třeba
- › **Mapa pokrytí** – vytváří Test analytik /Test manager
 - Ukazuje, zda ke každému požadavku existuje minimálně jeden test

Návrh testovacích případů

Jak má vypadat test case?

- › Doporučená délka max. 20 kroků
- › Jednoznačně reprodukovatelný
- › Odpovídající míra detailu
- › Testovací data
- › Standardní struktura

Návrh testovacích případů

Jak neuspět

- › Začít tvořit testy bez finální revidované vstupní dokumentace
- › Malá diverzita použitých technik
 - Pouze specification based testing
 - Pouze function testing
- › Příliš detailní testovací skripty
 - Malá volnost pro kreativitu testera
 - Malý prostor pro „náhodu“
 - Obtížná udržitelnost
- › Exploratory testing bez patřičného vzdělávání
- › Oddělení návrhu a provádění testů
- › Ignorování existujících rizik

Testovací scénář

The screenshot displays the HP Application Lifecycle Management 11.52 web interface. The left sidebar contains a navigation menu with sections: Dashboard, Management, Requirements, Testing, Test Resources, Business Components, Test Plan (selected), Test Lab, Test Runs, and Defects. The main area is titled 'Tests' and shows a tree view of test subjects. The selected subject is 'Individuální kurzy - Aktualizace k', which is expanded to show a list of test steps. The 'Design Steps' tab is active, displaying a table with the following data:

Step Name	Description	Expected
Step 4	Kontrola zobrazených hodnot	Pro měnové páry EUR/CZK a USD/CZK jsou zobrazeny kurzy shodné s individuálními kurzy na Domovské obrazovce (box v levé části pod menu), platnost odpovídá času prvního zobrazení obrazovky Domů po přihlášení uživatele. Pro ostatní měnové páry jsou prázdné.
Step 5	Kliknout na tlačítko Aktualizovat u libovolného měnového páru	U vybraného páru se zobrazí získaný kurz pro nákup i prodej a platnost v čase získání kurzu
Step 6	Kliknout na tlačítko Aktualizovat u libovolného měnového páru	U vybraného páru se zobrazí získaný kurz pro nákup i prodej a platnost v čase získání kurzu
Step 7	Kliknout na tlačítko Zadat FX operaci	Zobrazí se obrazovka Zadání FX Trading Spot/Forward (krok 1 ze 2)

The bottom status bar indicates 'Total Steps: 5' and 'Server Time: 16.10.2013 9:35'.

Kdo píše /reviduje testovací scénáře

Typ testu	Píše	Reviduje	Vykonává
Unity testy	Vývojář	Jiný vývojář	Vývojář
Integrační testy	Vývojář / tester	Jiný vývojář / tester / analytik	Vývojář / tester
Systémové testy	Tester	Jiný tester / vývojář / analytik	Tester
Akceptační testy - uživatelské	Uživatelé (business) / tester	Uživatelé (business)	Uživatelé (business) / tester
Akceptační testy - operační	Provoz / tester	Provoz	Provoz / tester

The background of the slide is composed of numerous overlapping, translucent, light-gray geometric shapes. These shapes, which include rectangles, trapezoids, and other polygons, are arranged in a way that creates a sense of depth and movement, as if they are floating or falling from the top right towards the bottom left. The overall effect is a modern, architectural aesthetic.

Provádění a vyhodnocení testů

Provádění testů

Kdy začít testovat?

- › Plánovaný harmonogram vs. realita
 - zpoždění dodavatele
 - čekat nebo začít dříve ?
- › Dobré časování je zásadní
 - příliš **pozdě** → problém se splněním termínů, málo času na testování
 - příliš **brzo** → nestabilní SW, zbytečně vynaložený čas a práce testerů

Kdy přijmout SW do testů?

- › Funkční integrace
- › Smoke test, Sanity test
- › Předvedení funkčnosti vývojářem



Testovací kola - příklad

- › Smoke Test /Dry Run (nová funkcionalita)
- › Sanity test (regresní smoke test)
- › Integrační testy
- › Systémové integrační testy 1 (SIT 1)
- › Systém integrační testy 2 (SIT 2)
- › Systém integrační testy 3 (SIT 3) - pokud je potřeba
- › Regresní test
- › Core test
- › Core test final

The background of the slide is a complex, abstract composition of numerous overlapping, translucent geometric shapes. These shapes, which include various polygons and rectangular prisms, are rendered in different shades of light gray. They are arranged in a way that creates a sense of depth and three-dimensional space, with some shapes appearing to float above others. The overall effect is a modern, architectural aesthetic.

Řízení testů

Řízení průběhu testů

- › Klasické aktivity jako v případě Project managementu
 - Alokace zdrojů
 - Dynamické přidělování a plánování práce
 - Reakce na problémy
 - Zlepšování procesu testování
 - Snaha optimalizovat
 - ...
- ➔ Musíme vědět, „**jak na tom jsme**“ !

*„Kolik testování jste na projektu už udělali?“
„Jak a podle čeho měřit rozsah testování?“
„Co je to rozsah testování?“*

Řízení průběhu testů

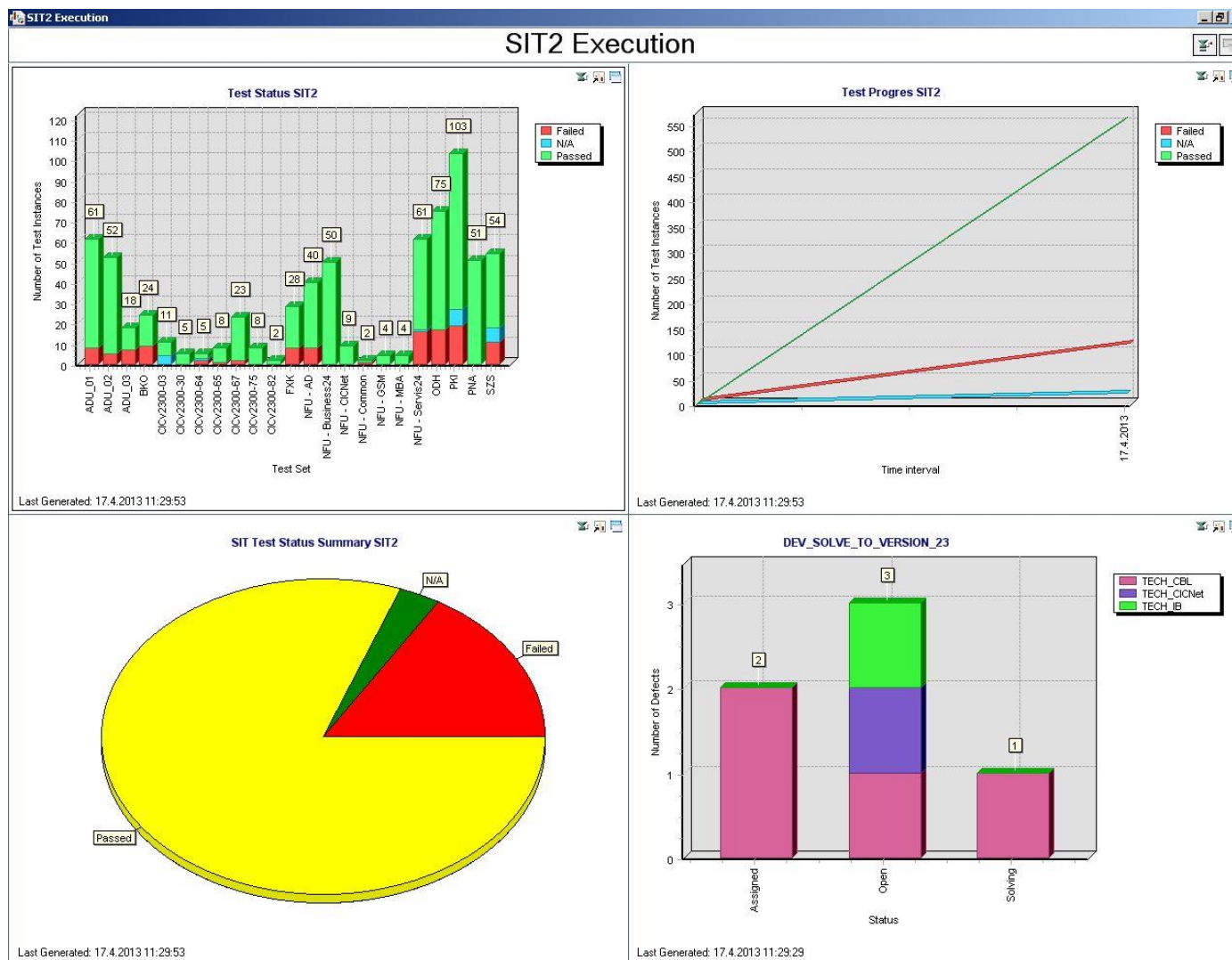
Co je to rozsah testování ?

- › Typicky odpovědi založené na
 - **Produkt:** „Otestovali jsme 70 % řádek kódu“
 - **Plán:** „Provedli jsme 65 % testovacích případů“
 - **Výsledky:** „Našli jsme 753 chyb“
 - **Pracnost:** „Pracovali jsme 3 měsíce 60 hodin týdně, provedli jsme 8956 testů“
 - **Kvalita testování:** „Beta testéři našli 28 chyb, které nám unikly, naše regresní testy se zdají neefektivní“
 - **Rizika:** „Dostáváme spoustu stížností od Beta testerů, stále máme otevřených přes 500 problémů, produkt nebude do 3 dnů připraven ke spuštění“
 - **Projektová historie:** „V tento moment jsme na předchozích projektech měli 12 % nalezených problémů stále otevřených, stejné by to mělo být i teď“

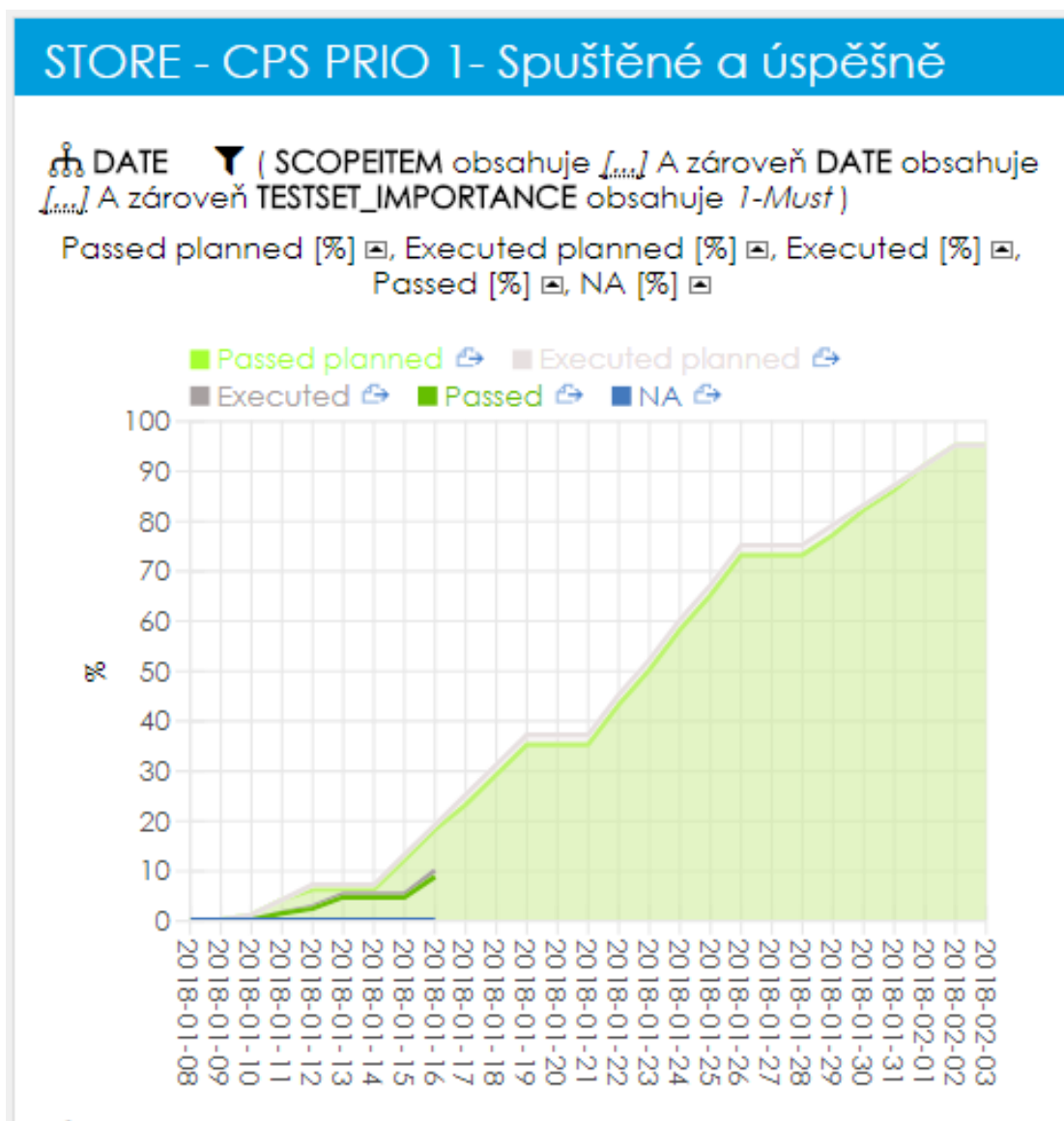
Měření rozsahu testování

- › Žádná metrika není dokonalá
- › Řešením z praxe je ... ?
 - ... kombinace více různých metrik ...
 - pokrytí, pracnost, výsledky, rizika, potíže, ...

Základní reporty měření rozsahu testování



Spuštění a dokončené testy

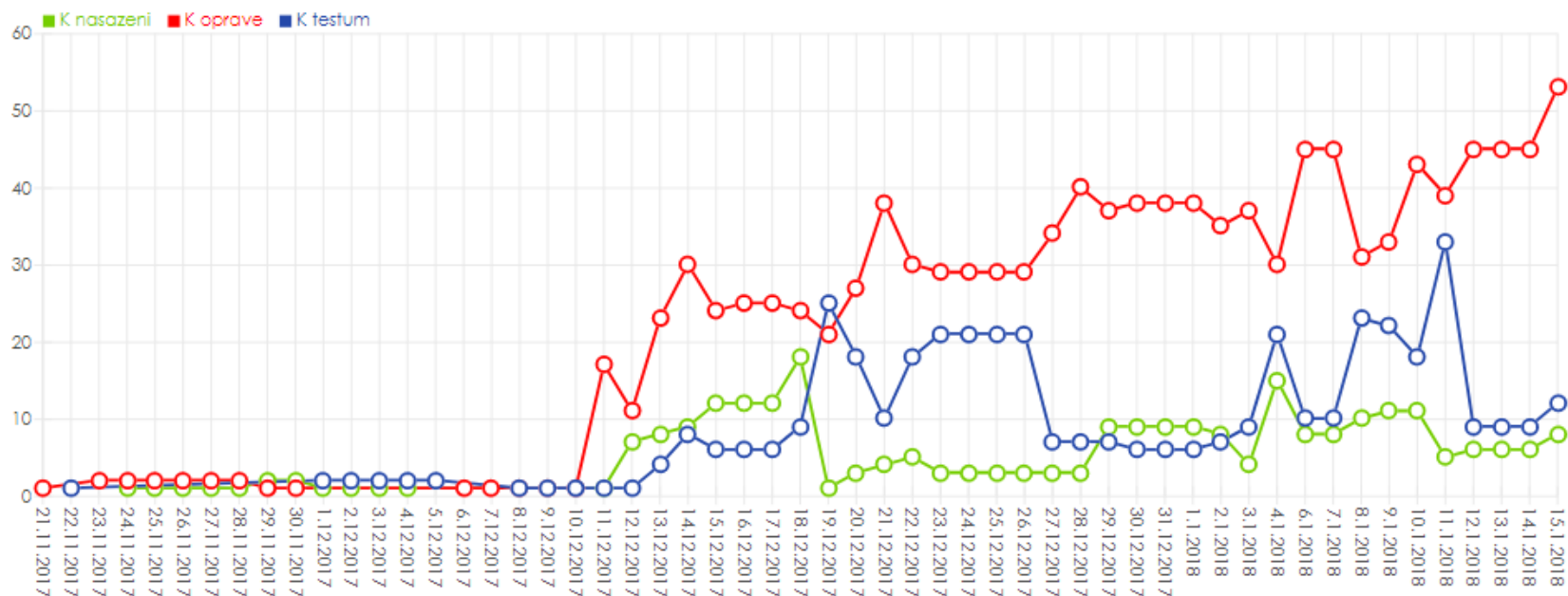


Neuzavřené defekty

STORE - Neuzavřené defekty (všechny severity)

🔗 Date Stamp a Resitelska skupina 📄 (Status neobsahuje Closed, Identified A zároveň Labels obsahuje #PROJ Store)

Count: Unique ID 📄



🔗 TestCenter Library - Stav defektů TC - progres - SANDBOX

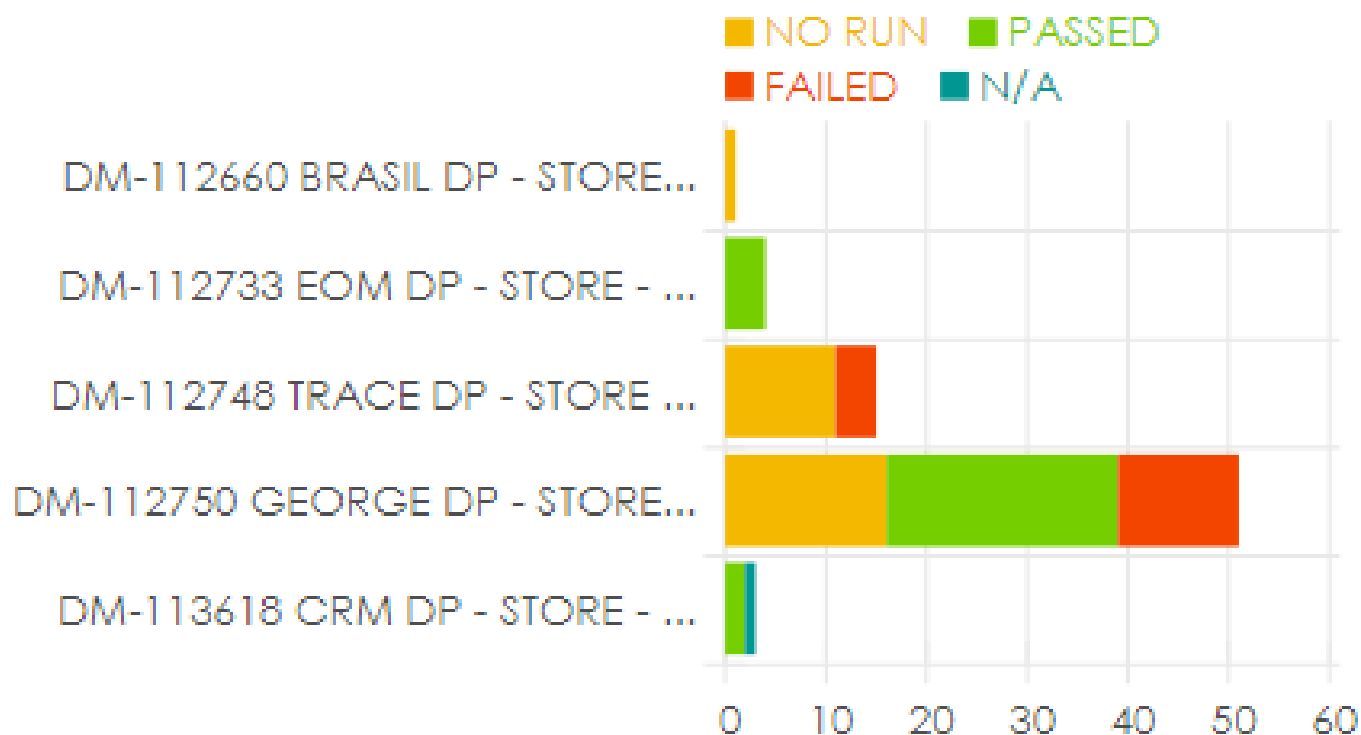
Poslední změna v datech: 15.1.2018 18:02:36

Stav ostatních testů

STORE - stav testů ostatních DP

🔗 DP_NAME ⚑ (APPLICATION neobsahuje CPS_TC A zároveň HP_NAME obsahuje DM-101967 STORE - prodej úvěrových produktů v George Store)

NO RUN 📊, PASSED 📊, FAILED 📊, N/A 📊



Defekty po severitách

STORE - BLOCKERY

Assigned System (Status neobsahuje Closed, Deployed, Identified, New, Rejected A zároveň Labels obsahuje #PROJ Store A zároveň Priority obsahuje 0-Blocker)

Fixed (X), Open (X), Solving (X)



CPS_TC

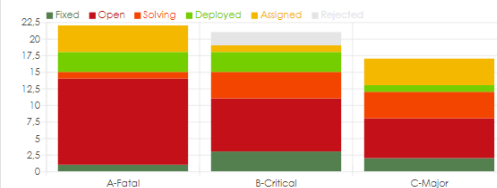
TestCenter Library - Stav defektů TC - SANDBOX

Poslední změna v datech: 15.1.2018 10:40:14

STORE & CPS - neuzavřené defekty

Severity (Severity obsahuje A-Fatal, B-Critical, C-Major, D-Minor A zároveň Status neobsahuje Closed, Identified, New A zároveň Labels obsahuje #PROJ Store)

Fixed (X), Open (X), Solving (X), Deployed (X), Assigned (X), Rejected (X)



A-Fatal

B-Critical

C-Major

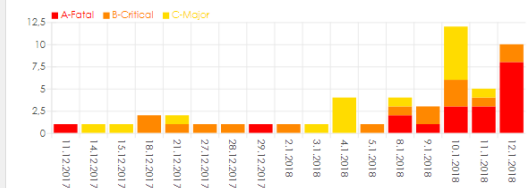
TestCenter Library - Stav defektů TC - SANDBOX

Poslední změna v datech: 15.1.2018 10:40:14

STORE - Neopravené defekty po severitách (X = Detection date)

Detection Date a Severity (Status neobsahuje Closed, Deployed, Identified, Rejected A zároveň Labels obsahuje #PROJ Store)

Count: Unique ID (X)



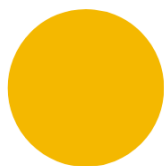
TestCenter Library - Stav defektů TC - SANDBOX

Poslední změna v datech: 15.1.2018 10:40:14

UAT STORE

HP_NAME obsahuje DM-101987 STORE - prodej úvěrových produktů v George Store

PASSED (X), FAILED (X), NO RUN (X), N/A (X)

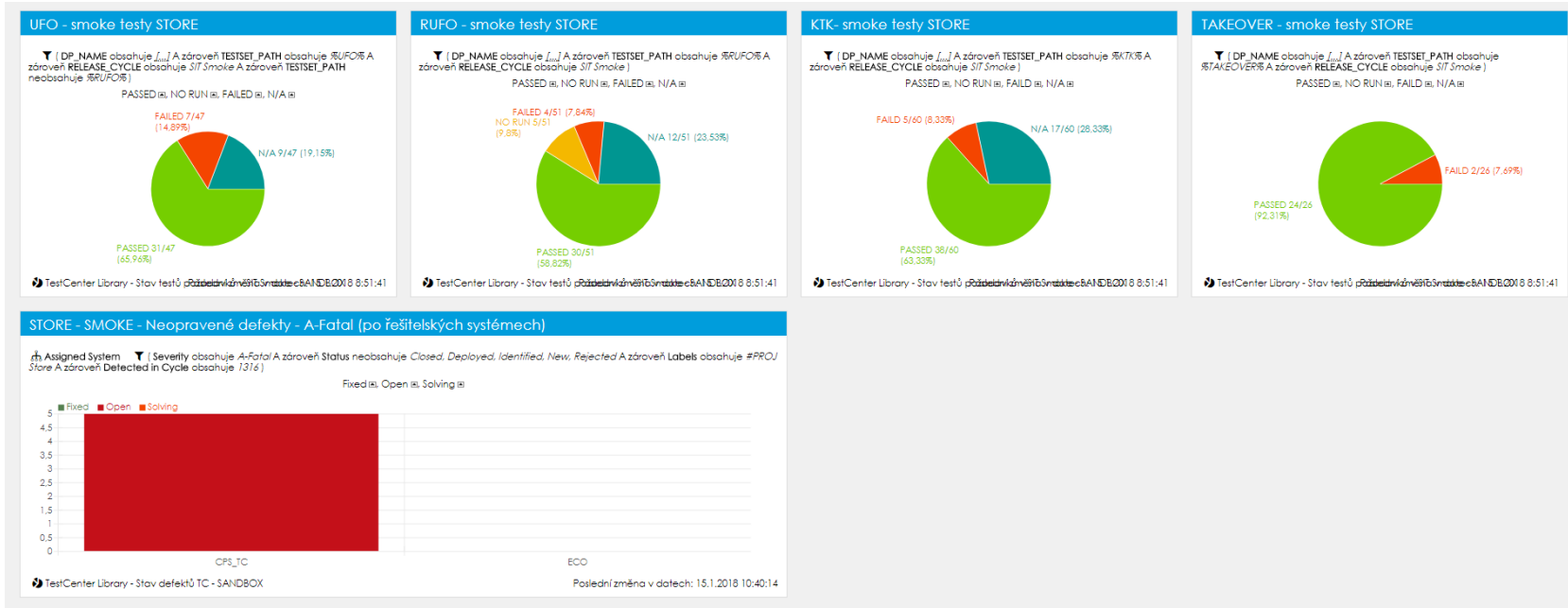


TestCenter Library - Stav testů UAT - SANDBOX

Poslední změna v datech: 15.1.2018 9:10:47

Výsledky smoke testu

- › Rozděleno na jednotlivé produkty
- › Vstup pro Takeover



Reportování výsledků testů

- › Standardizovaný test report
- › Celkové zhodnocení testovaného SW
- › Dopad nedostatků na projekt, systém, ...
- › Detailní výsledky
 - Nalezené problémy
 - Odchytky od testovacích případů
- › Log testů (provedené testy, průběh testů, ...)



Adobe Acrobat
Document

Důvody, proč zastavit/ukončit testování

Seriózní

Střet s realitou

Ideální

Říše snů



Testovací prostředí

Jak se testování dělat nemá

- › Snažit se analyzovat chyby na produkci – aneb za dobrotu na žebrotu

Příklad:

Správce produktu požadoval dočasně změnu čísla klienta na potvrzovací SMS na tel. číslo správce produktu pro ověření jestli SMS chodí či ne.

Databázista zapomněl podmínku „where“ => číslo změněno **všem klientům od Vodafone** operátora.

Výsledek:

Nemožnost práce s IB pro klienty Vodafone na půl dne, pád SMS brány ve Vodafone, článek v novinách.

Testovací prostředí

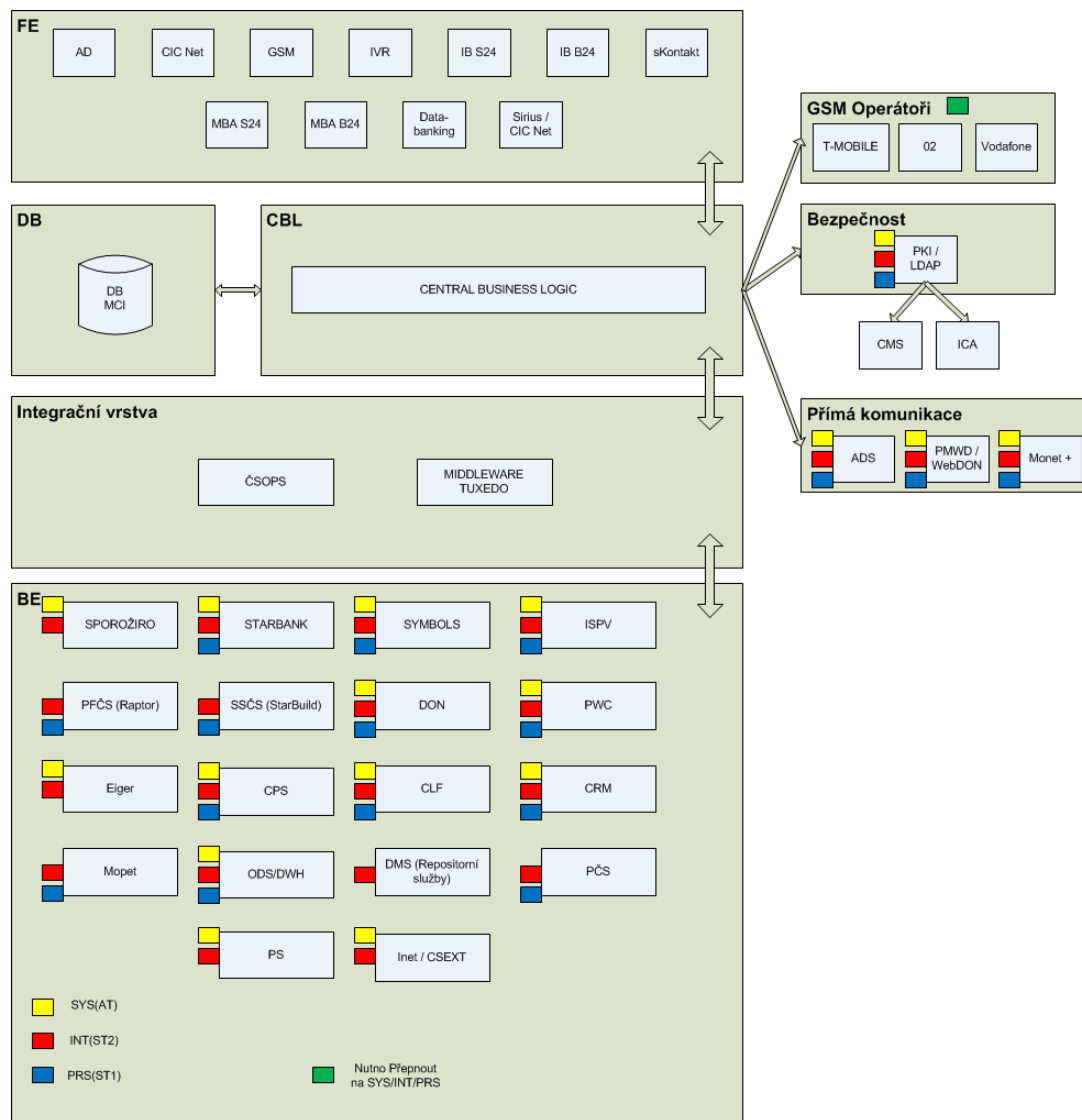
- › Proč testovací prostředí?
 - AXIOM: Na produkci se netestuje!!!
- › Kolik prostředí potřebujeme?
 - Záleží na velikosti projektu, ale u velkých minimálně tři.
 - Vývojové
 - Integrační
 - Podpora produkce
- › Jak má vypadat ideální testovací prostředí?
 - Kopie produkce (v praxi nereálné)
- › Kde můžeme slevit?
 - Výkon
 - Množství dat
 - Náhrada integrací mocky (kde to jde)
- › Testovací data – kde je vzít?
 - Kopie produkce
 - Vytvořena testery

Testovací prostředí

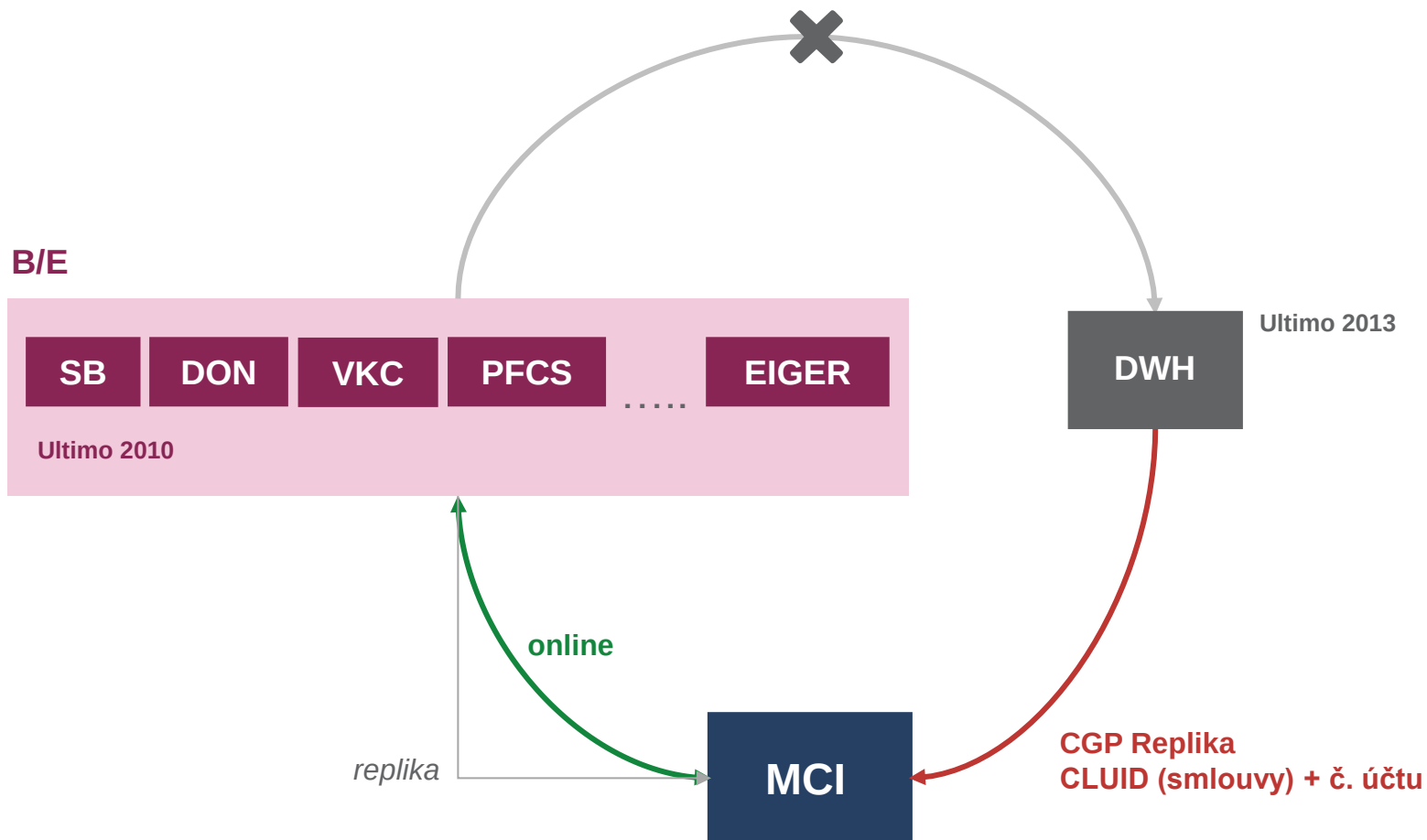
- › SYS (AT)
- › INT (ST2)
- › PRS(ST1)

Komunikace

- › Online
 - Synchronní
 - asynchronní
- › repliky
 - Full
 - inkrement



Testovací prostředí - realita



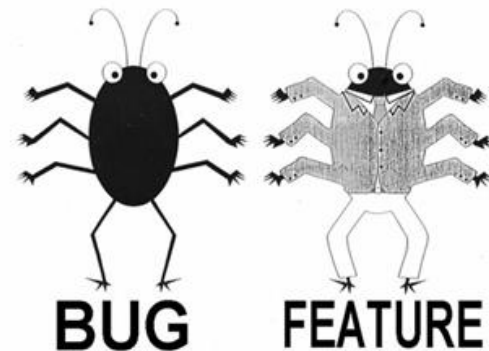
The background of the slide is a dark gray field filled with a complex, overlapping pattern of translucent, light gray polygons. These polygons vary in size and orientation, creating a sense of depth and movement, similar to a low-poly 3D landscape or a crystalline structure. The word "Defekty" is centered in the middle of the image.

Defekty

Trouble ticketing, bug reporting

Základní pravidla

- › Evidence **všech** nalezených **issues**
- › Jediné místo pravdy - specifikace
- › Nejde jen o to, nahlásit issue, důležité je udělat to tak, aby bylo možné jej
 - reprodukovat a
 - opravit
- › **Schopnost odlišit chyby, které „proklouznou“ (produkční)**
- › Trouble ticketing → Bug tracking ?



Náležitosti defektu

- › Defekt musí mít své náležitosti aby byl:
 - REPRODUKOVATELNÝ
 - Reportovatelný / měřitelný

Náležitost		Poznámka
Název defektu		Stručný název vystihující podstatu defektu v několika slovech
Detailní popis defektu		Detailní postup jak chybu vyvolat včetně testovacích dat
Jméno testera		
Jméno vývojáře který chybu opravil		
Stav chyby		
Datum vystavení chyby		
Datum opravení chyby		
Verze SW		Verze SW ve které byla chyba nalezena
Verze SVN revize opravy		Verze například SVN ve které je uložen kód fixující danou chybu
Testovací prostředí		
Logy		
Screenshoty obrazovky		Pouze pokud to usnadní pochopení chyby, či její nasimulování, u chyb UI povinné.
Odkaz na test		Odkaz na test ve kterém byla chyba nalezena
Příčina vzniku chyby		Velmi důležité pro sledování efektivity testování
Příčina neodhalení chyby vývojářem/ testerem		Velmi důležité pro sledování efektivity testování
Severita		Závažnost chyby typicky A -High, B - Major, C -Low
Priorita		Slouží k určení priority opravy defektu vývojem. Tedy pokud z pohledu testera je defekt severity B, ale je blokující pro další testy dáme prioritu A a defekt je přednostně opraven

Defekty – Severita a Priorita

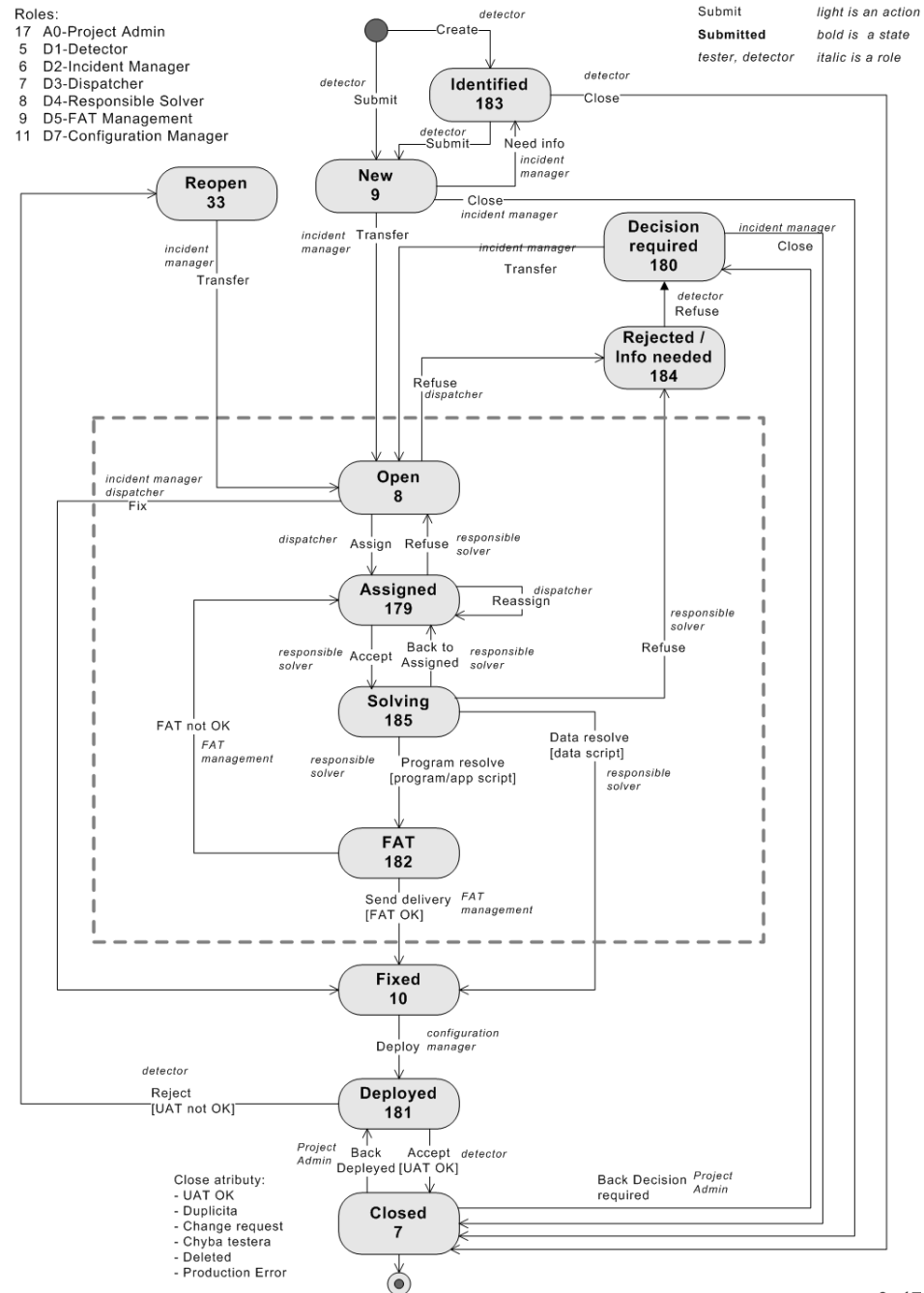
- › Severita A – Fatal
 - Kritická chyba, funkcionalita nefunguje, nelze pokračovat
- › Severita B – Critical
 - Kritická chyba, funkcionalita nefunguje, existuje workaround nebo nefunguje pro specifickou datovou variantu
- › Severita C – Major
 - Některá funkčnost nefunguje, operaci je možné dokončit, chyba není kritická
- › Severita D – Minor
 - Kosmetické chyby, drobné nedostatky

Priorita:

Slouží k urgenci opravy chyby, tj. brzdí-li chyba v testech

- jeden defekt může být severity C a zároveň priority A. (například nefunkční jazyková mutace)

Defect lifecycle



Akceptační kritéria

Akceptační kritéria

- › Stanovují se typicky na základě
 - Protestovanosti testů a Pass rate
 - Počtu zbývajících defektů
- › Je důležité si stanovit:
 - Datum a čas odečtu stavu
 - Dobu na opravu a retest po odečtu
 - Datum, od kdy se již netestuje – obvykle od odečtu
 - Osoby a pravidla, která rozhodnou, zda zbývajcí defekty jsou akceptovatelné



You are a lucky bug. I'm seeing that you'll be shipped with the next five releases.

Akceptační kritéria - příklad

1. Akceptační kritéria FAT (k termínu snapshotu)

- Stav defektů na 100 vývojových MDs 1A, 2B
- Stav otestovanosti Regresních testů – 100 % otestované /85 % OK
- Stav otestovanosti Core regresních testů – 100 % otestované /85 % OK (bez nových funkčností)
- Stav otestovanosti Nových funkčností – 90 % otestované /85 % OK

2. Akceptační kritéria UAT (k termínu snapshotu)

- Stav defektů 0A, 5B, 10C
- Stav otestovanosti Core regresních testů – 100% otestované /95 % OK



Měření

Měření

- › Každou fázi testovacího procesu je vhodné měřit
- › Získáme:
 - Obraz, jak na tom jsme
 - Vstupy pro další zlepšení (Lessons learned)
- › Volba vhodných měřených atributů je zásadní
 - V zásadě platí čím více, tím lépe
 - Je potřeba měřením neúměrně nezvyšovat pracnost
- › Obvykle měříme:
 - Plánovaná délka (pracnost) úkolu vs. reálná délka (pracnost)
 - Zjišťujeme kde máme zpoždění
 - Zjišťujeme (evidujeme) všechny příčiny /důvody zpoždění

Servis24 & Business24

LINKA SERVIS 24 844 11 11 44

SERVIS•24
INTERNETBANKING

ČESKÁ
SPORITELNA

LINKA BUSINESS 24 844 128 128

BUSINESS•24
INTERNETBANKING

726 118 128 (Telefónica O2)
605 661 128 (T-Mobile)
776 991 128 (Vodafone)

ČESKÁ
SPORITELNA

PŘIHLÁŠEN

HESLEM KLIEN

Klient
Heslo

PŘIHLÁŠENÍ BUSINESS 24

English version 

HESLEM KLIENSKÝM CERTIFIKÁTEM

Klientské číslo

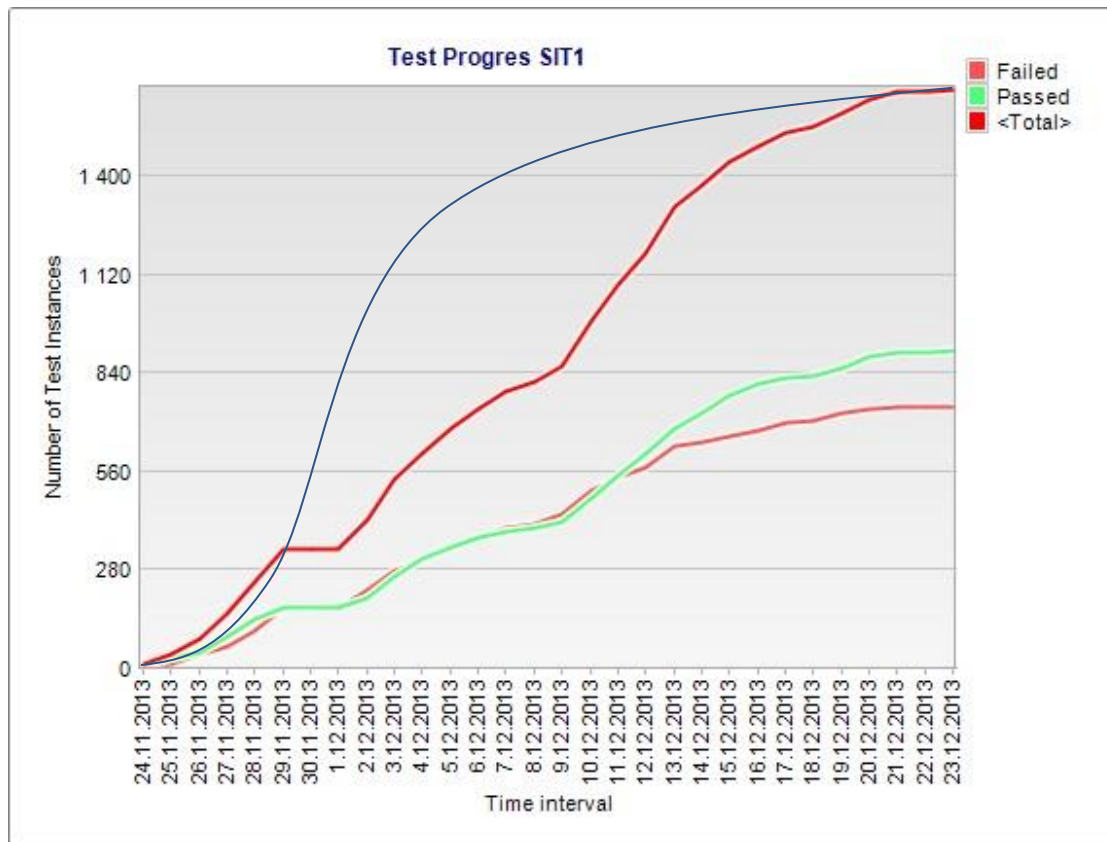
Heslo

ODESLAT

Přihlašování klientským číslem a heslem

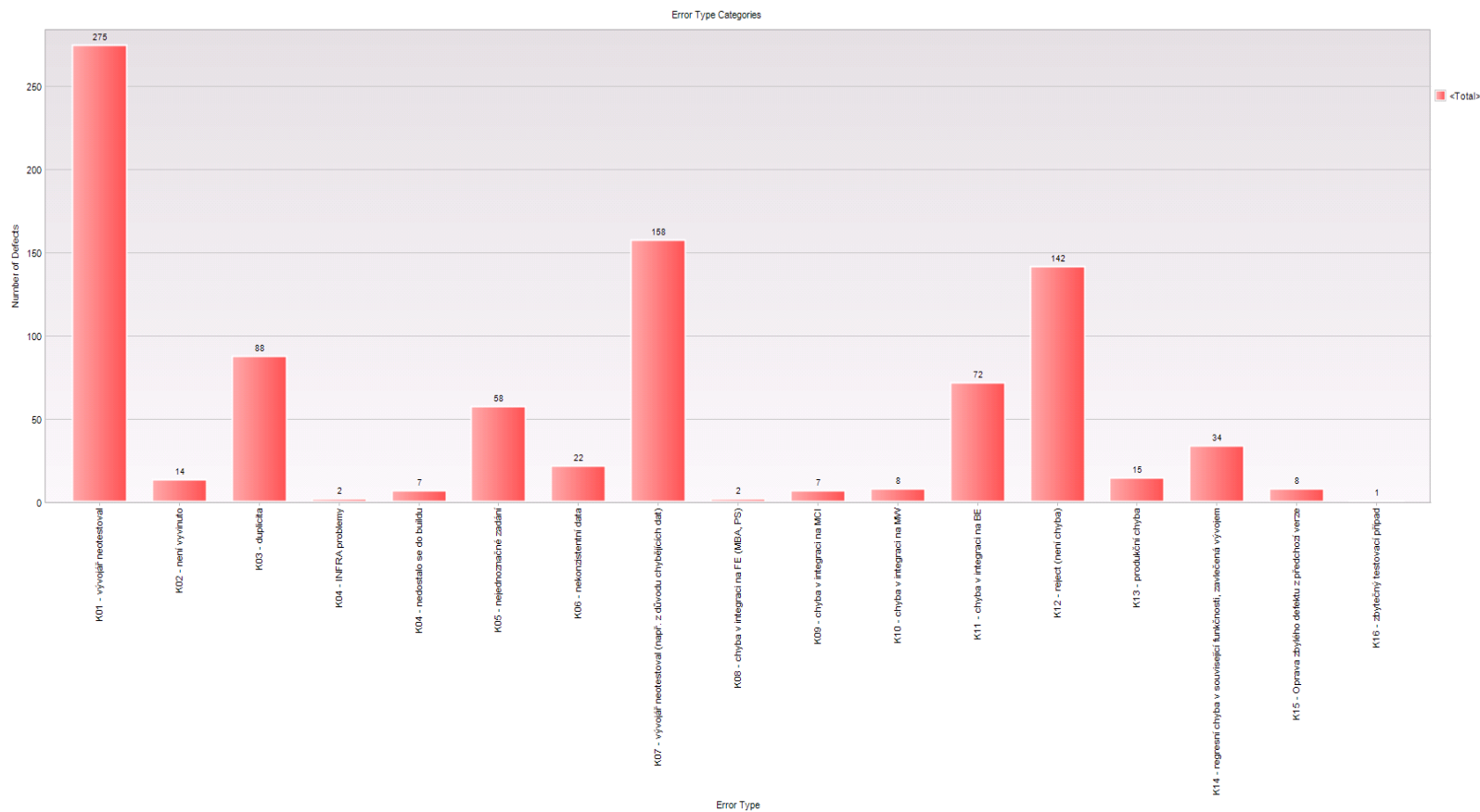
V přihlašovacím dialogu vyplňte, prosím, své **klientské číslo** služby BUSINESS 24 a **heslo** internetového bankovníctví. Po řádném zadání přihlašovacích údajů klikněte na tlačítko **Odeslat**. K prvnímu přihlášení potřebujete znát také **bezpečnostní kód**. Bez tohoto čísla by Vaše první přihlášení nebylo úspěšné.

Příklad - průběh testů - něco je špatně



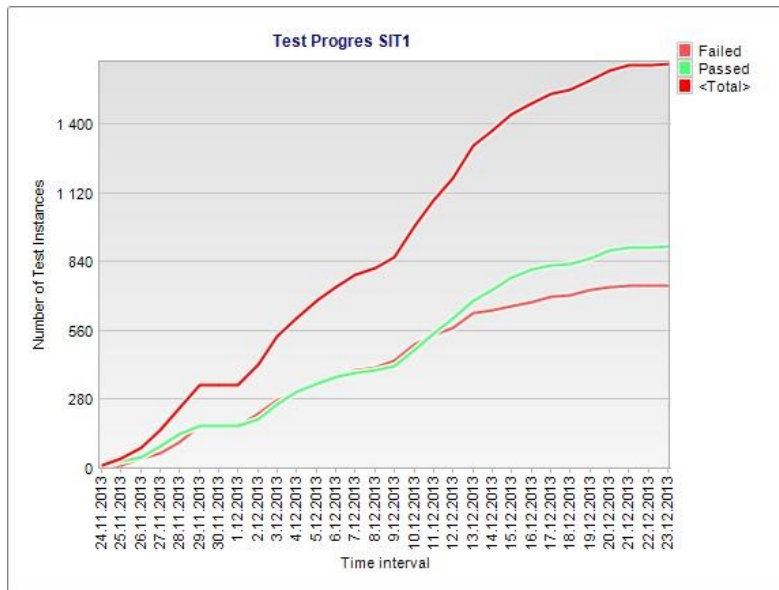
Příklad - Kategorie příčiny neodhalené chyb

Verze 26

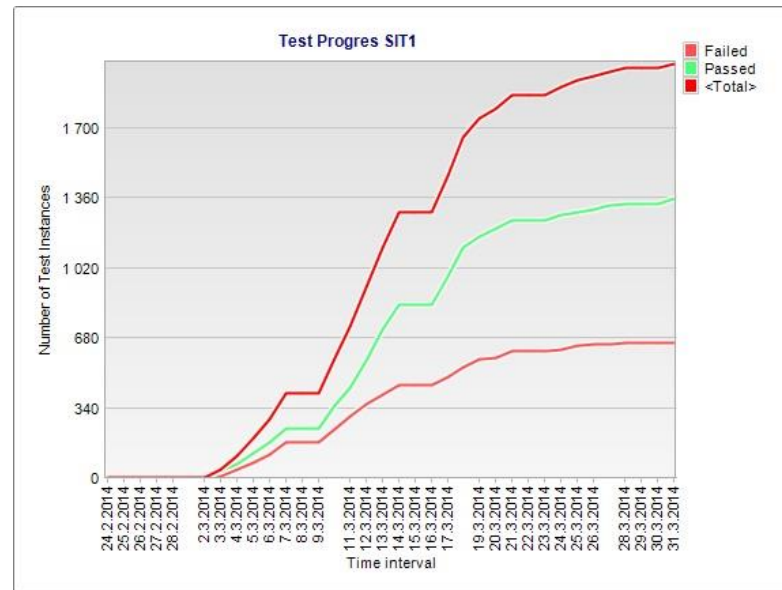


Porovnání průběhu testů

Verze 26

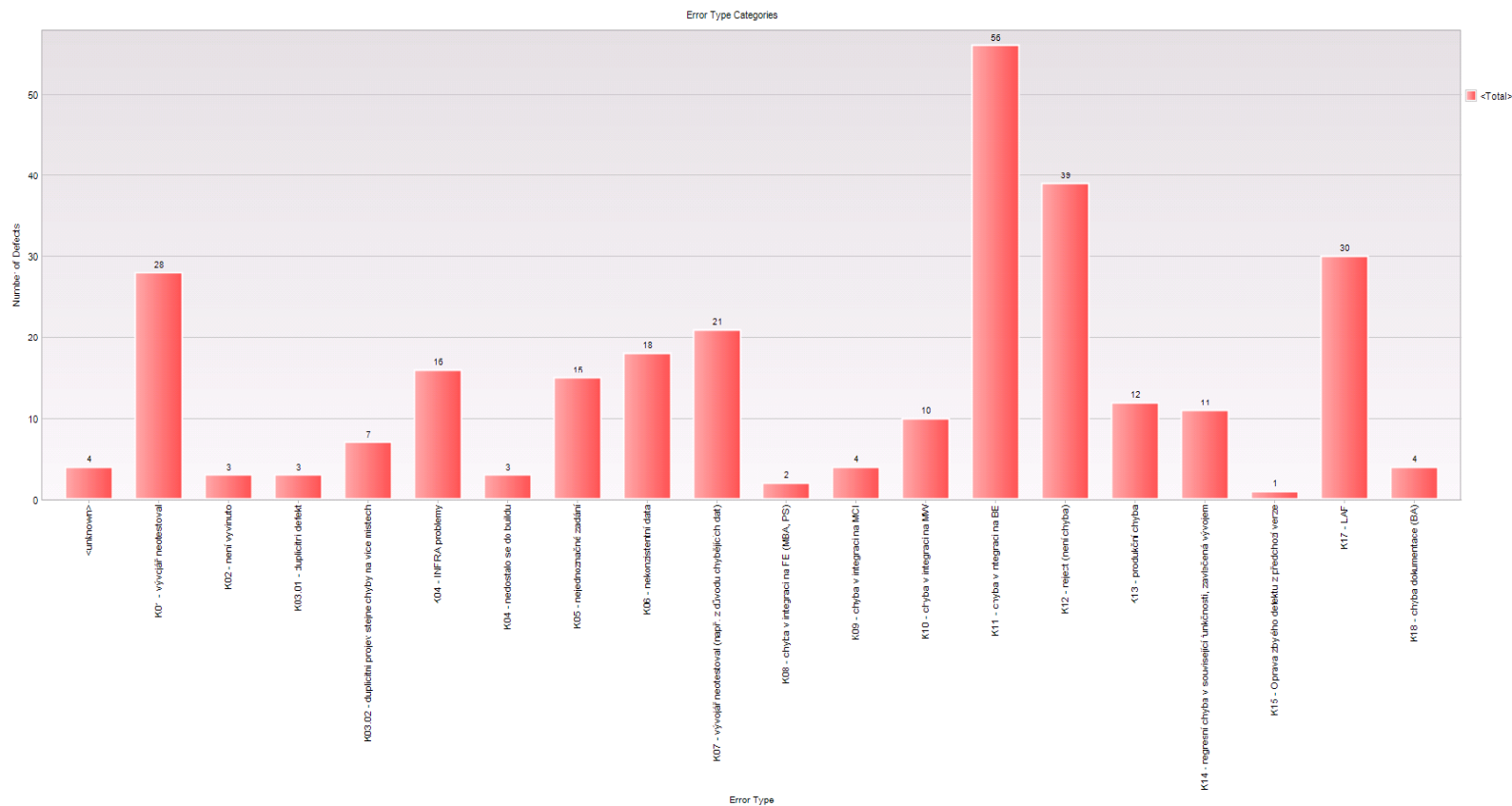


Verze 27

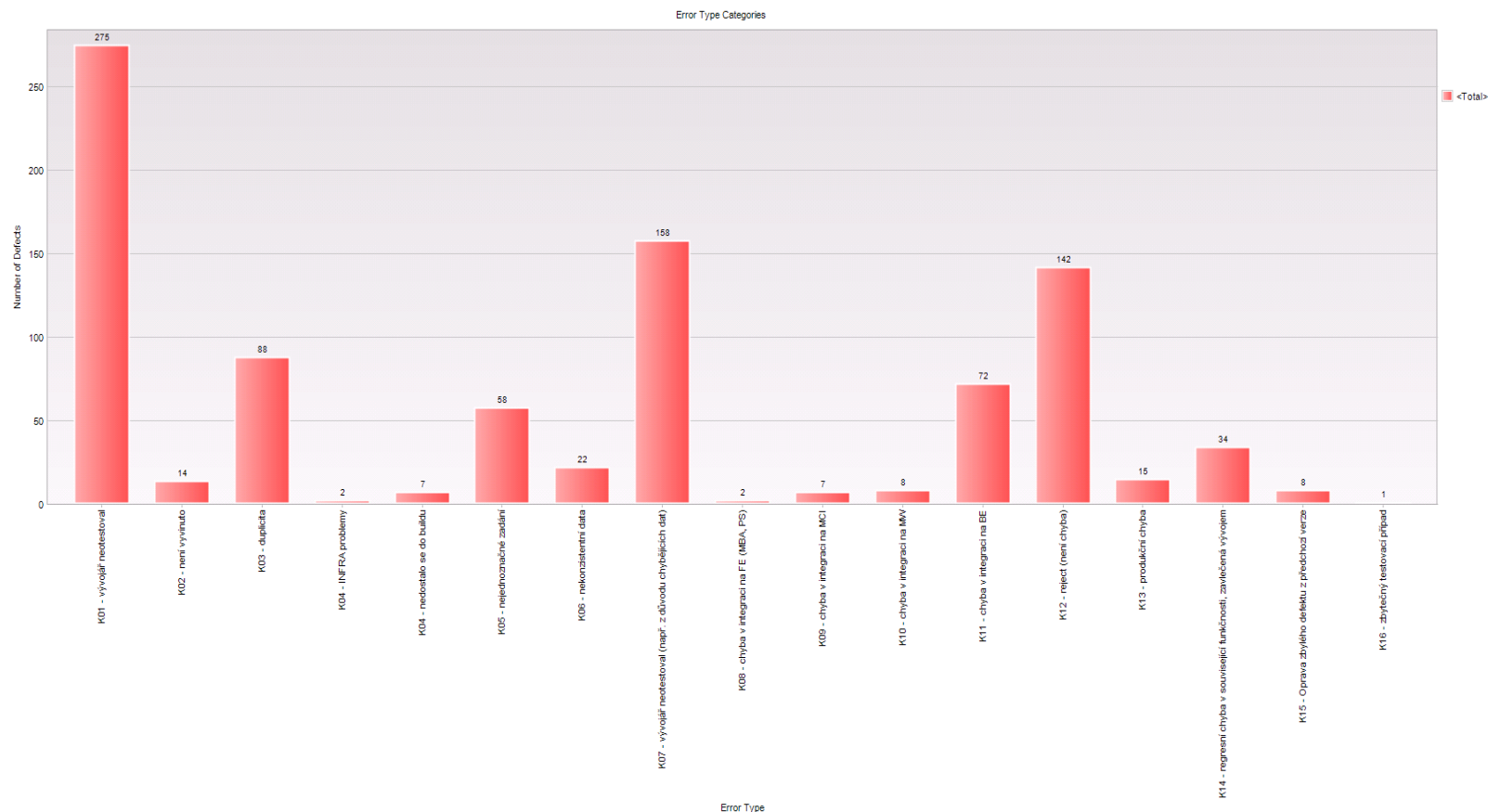


Příklad - Kategorie příčiny neodhalené chyb

Verze 27



Vyhodnocování příčin chyb – root cause analysis



Cena jedné chyby (nalezení, oprava, retest) je 0,5 MD

The background of the slide is a dark gray field filled with numerous overlapping, translucent, light gray geometric shapes. These shapes, which include rectangles, parallelograms, and polygons of various sizes and orientations, create a complex, layered effect that suggests depth and movement. The overall aesthetic is modern and technical.

Automatizace v kontextu testování

Automatizace exekuce testů

- › Snaha automatizovat testy již mnoho desítek let

- › Proč?
 - **Opakovatelnost a konzistence** testů → stejné vstupy a podmínky nezávisle na počtu opakování, odpadá problém s motivací lidí k opakování stejných testů
 - **Praktická znovupoužitelnost** testů → lze opakovat stejný test v různých prostředích, v různých konfiguracích, s mírně modifikovanými vstupními daty, ... a znovuspuštění testu je levné
 - **Praktické baseline testy** → automatizace umožňuje spustit velmi „hutnou“ sadu testů, umožňují efektivně provádět regresní testování

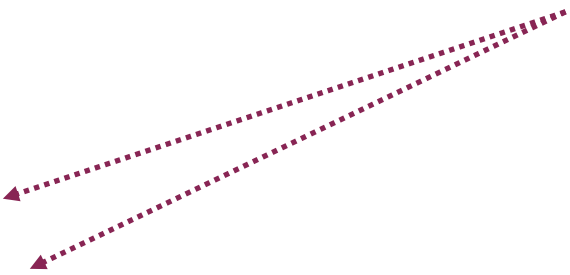
Automatizace regresních testů

- › Velice častý scénář
- › Typický průběh automatizace
 - Vytvořit testovací případ
 - Manuálně jej spustit a ověřit výstup
 - V případě selhání nahlásit chybu
 - V případě úspěchu „uložit“ výsledek
 - Opakovaně spouštět test a výsledky porovnávat s uloženými, hlásit chybové situace
 - Udržovat automatický test

Je to skutečně automatizace?

- › Analýza programu
- › Design testu
- › První spuštění testu
- › Uložení výsledků
- › Dokumentace testu
- › Znovuspuštění testu
- › Vyhodnocení výsledků
- › Údržba testu

**Co z toho vlastně
dělá stroj?**



Automatizace regresních testů

Ne pro automatizaci ...

- › Tvorba testovacích případů je drahá
- › Vyžaduje velmi technicky zkušené členy týmu
- › Vyžaduje dobře definované a stabilní rozhraní
- › Vyplácí se pozdě (výhody automatizace v release N se vrací až v release N+1)
- › Regresní testy mají často menší Power než nové testy
- › ...

Kdy může mít smysl?

- › Smoke testing (**Continuous Integration**)
- › Configuration testing (HW SW compatibility)
- › **Variace**
- › Stress testing
- › Load testing
- › **Příprava testovacího prostředí** (data, ...)
- › ...

Nástroje pro automatizaci testů

› Unit a integrační testy

- [JUnit](#), [TestNG](#), [jMock](#), [EasyMock](#), [DbUnit](#), ...

› Statická analýza kódu

- [Findbugs](#), [PMD](#), [JDepend](#), FoxCop, ...

› Funkční testy – tlustý klient

- [Selenium](#), HP QuickTest, IBM Functional Tester, ...

› Funkční testy – tenký klient

- HP QuickTest, IBM Functional Tester, White, AutoIt, ...

› Výkonové testy

- [JMeter](#), [Dieseltest](#), [QALoad](#), ...

› Komplexní řešení

- HP Test Suite, IBM/Rational Test Suite, ...

› Příprava testovacího prostředí

- IBM Optim, Grid-Tools DataMaker, Oracle Datamasking, ...



Závěr

- › Testovací proces a jeho fáze
- › Defekty a jejich životní cyklus
- › Testovací prostředí
- › Měření a jeho důležitost
- › Automatizace testování

MATERIÁLY SWENG

TESTING

ČLÁNKY

- [What Is Software Testing? And Why Is It So Hard?](#) – absolutní klasika, jejímž autorem je guru v oblasti testování, James A. Whittaker, který řeší problematiku testování pro společnosti jako Google nebo Microsoft
- [What Is a Good Test Case](#) – velmi zdařilý (a krátký) článek, který srozumitelně a současně nezjednodušeně popisuje problematiku vytváření testovacích případů
- [Little Book of Testing – Volume I](#) – jasná a srozumitelná, obsahuje mnoho nahuštěných informací a dobrých rad pro každého, kdo přichází s testováním do kontaktu
- [Little Book of Testing – Volume II](#) – pokračování prvního dílu, obsahuje další pokročilá témata
- [Software Test Planning and Management Guide](#) – průvodce tvorbou a managementem plánů testů vytvořený ve Space and Naval Warfare Systems Center San Diego

CHECKLISTS

- [CxCheck_TestPlan.pdf](#) – checklist firmy Construx vhodný pro vytváření plánu testů

TEMPLATES

- [CxTemp_TestPlan.doc](#) – template firmy Construx pro plán testů
- [IEEE_TestPlanTemplate.pdf](#)
- [MIL-STD-498_SwTestDescriptionTemplate.doc](#)
- [MIL-STD-498_SwTestPlanTemplate.doc](#)
- [MIL-STD-498_SwTestReportTemplate.doc](#)
- [SPAWAR_SwTestPlanTemplate.doc](#)
- [SPAWAR_SwTestReportTemplate.doc](#)

Všechny odkazované materiály jsou poskytnuty výhradně za účelem výuky softwarového inženýrství.

OBLASTI SWENG

- › Project and process
- › Project management
- › Documentation
- › Quality assurance
- › Configuration management
- › Testing
- › Construction
- › Architecture and design
- › Requirements
- › Obecné, přehledové

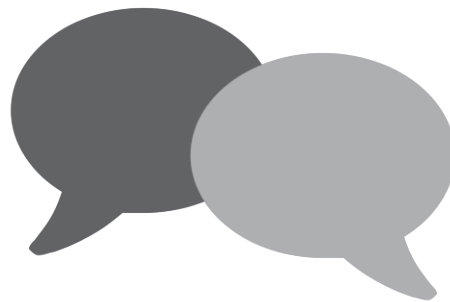


Hodnocení přednášky

<https://www.surveymonkey.com/r/2PSHTKF>

nebo

<https://goo.gl/7onYoG>



Diskuze

Děkujeme za pozornost

PROFINIT

NÁSKOK DÍKY ZNALOSTEM

Profinit EU, s.r.o.

Tychonova 2, 160 00 Praha 6 | Telefon + 420 224 316 016



Web
www.profinit.eu



LinkedIn
linkedin.com/company/profinit



Twitter
twitter.com/Profinit_EU



Facebook
facebook.com/Profinit.EU



Youtube
Profinit EU