

PROVÁDĚCÍ KONCEPT SW ŘEŠENÍ (PK)

projektu

Národní informační systém integrovaného záchranného systému (NIS IZS)

část

H. Testování

Dokument obsahuje: Metodiku testování NIS-IZS.

Verze: 6.1

Schválil za Dodavatele: RNDr. Vladimír Příbramský

Datum aktualizace: 10/10/2014

Obsah

1	O dokumentu	4
2	Definice základních pojmů	5
2.1	Požadavek	5
2.2	Funkční specifikace (analýza)	5
2.3	Testovací scénář	5
2.4	Vada	5
2.5	Akceptační procedura	6
2.6	Akceptační výrok	7
2.7	Akceptační metrika	7
2.8	Akceptační kritérium	7
2.9	Změna	8
3	Typy testů	9
3.1	Testování ve vazbě na životní cyklus IS	9
3.2	Typy testů IS dle jejich specializace	9
3.2.1	Unit testy	9
3.2.2	Funkční a integrační testy	10
3.2.3	Systémové testy a E2E testy	10
3.2.4	Zátěžové testy	10
3.2.5	Testy bezpečnosti	11
3.3	Typy testů IS dle jejich účelu	12
3.3.1	Interní testy dodavatele	12
3.3.2	Akceptační testy (UAT) zadavatele	12
4	Role	13
4.1	Role ČPOZ	13
4.2	Role zadavatele	14
5	Fáze testování	16
5.1	Plánování testů	16
5.1.1	Délka a počet testovacích kol	17
5.2	Příprava testů	18
5.2.1	Schvalování detailní testovací dokumentace zadavatelem	19
5.3	Provedení testů	21
5.3.1	Předávání produktu do interního testování ČPOZ	21
5.3.2	Provedení interního testování ČPOZ	21

	5.3.3	Předávání produktu nebo jeho části do akceptace zadavateli.....	23
	5.3.4	Akceptační testy.....	23
6		Evidence a řešení vad a změn.....	25
	6.1	Evidence a řešení vad.....	25
	6.1.1	Záznam o vadě.....	25
	6.1.2	Řešení vady.....	25
	6.2	Řízení změn (change management).....	28
7		Artefakty procesu testování.....	29
	7.1	Vstupní dokumentace.....	29
	7.2	Plán testování.....	29
	7.3	Testovací scénář.....	30
	7.4	Testovací data.....	31
	7.5	Protokol o provedení systémových testů.....	31
	7.6	Protokol o provedení akceptačního testování.....	31
8		SW nástroje.....	33
	8.1	Nástroj pro evidenci a řízení životního cyklu vad.....	33
	8.2	Ostatní SW nástroje.....	34
9		Rozsah testování.....	35
10		Přílohy.....	36
	10.1	Seznam obrázků.....	36
	10.2	Seznam tabulek.....	36

1 O dokumentu

Tento dokument popisuje primárně metodiku testování požadovaných funkcionalit realizovaného softwarového díla – IS NIS-IZS (dále také *produktu*).

V dokumentu je kladen maximální důraz na popis procesů plánování a testování dodávaného aplikačního SW, role, kompetence a odpovědnosti jednotlivých členů testovacích týmů dodavatele (dále také *ČPOZ*)/zadavatele, testovací dokumentace a používané elektronické nástroje. Okrajově dokument také řeší akceptační procedury dalších typů dodávek, neboť v praxi není předmětem akceptace produktu pouze aplikační SW, ale také jeho dokumentace, HW, systémový SW (např. operační nebo databázové systémy) nebo poskytované služby.

Dokument neřeší:

- principy projektového řízení,
- obchodní, smluvní či právní problematiku.

2 Definice základních pojmů

2.1 Požadavek

Požadavek definuje požadované konkrétní chování nebo vlastnost produktu. Požadavek resp. seznam požadavků na produkt vychází ze závazné dokumentace projektu NIS-IZS. Požadavky dělíme na 2 základní skupiny:

- Funkční požadavky: jedná se o konkrétní definovanou množinu aktivit a funkcionalit, které má produkt zabezpečovat. Každá aktivita nebo funkcionalita produktu musí být detailně popsána.
- Nefunkční požadavky: zpravidla se jedná o globální vymezení vlastností a omezení služeb, poskytovaných daným produktem. Nefunkční požadavky se zpravidla vztahují na produkt jako celek a výrazným způsobem ovlivňují architekturu celého řešení.

2.2 Funkční specifikace (analýza)

Funkční specifikace je analytickým dokumentem, jehož úkolem je strukturovanou formou definovat detailní chování a vlastnosti produktu a to na základě analýzy jednotlivých funkčních požadavků. Dokument funkční specifikace je jedním z výchozích dokumentů pro tvorbu testovacích scénářů.

2.3 Testovací scénář

Testovací scénář je strukturovaný předpis provedení jednoho konkrétního testu a obsahuje vždy aktivity prováděné Testerem a očekávaný výsledek (reakce produktu na provedenou aktivitu Testerem). Testovací scénáře jsou vždy vytvářeny na základě Funkční specifikace (vždy pokrývají všechny funkční požadavky).

Specializované testovací scénáře jsou dále tvořeny i pro část nefunkčních požadavků (výkon, dostupnost, bezpečnost atd.). Testovací scénář má vždy identifikovanou vazbu na požadavek a akceptační kritérium, přičemž platí vazba m:n (tj. jeden testovací scénář může testovat shodu na n požadavků a m akceptačních kritérií a naopak jeden požadavek nebo kritérium může být zahrnuto v n testovacích scénářů).

2.4 Vada

Vada je takové chování produktu nebo jeho části, které je prokazatelně v přímém rozporu s požadovaným chováním. Vada je v případě aplikačního SW detekována výkonem testů dle testovacích scénářů. Výskyt vad v produktu má přímý dopad na akceptaci produktu zadavatelem. Nejzásadnějším parametrem vady je její kategorie. Kategorie definuje závažnost vady a to jak rychle má být vada odstraněna a přetestována.

Kategorie vady jsou uvedeny v následující tabulce.

Kategorie	Vada	Popis
A	Vysoká	Z důvodu vady produktu (nebo jeho dílčího plnění) některé nebo všechny systémy podporující hlavní procesy selhaly a jsou zcela nefunkční nebo je jejich funkčnost omezena tak, že je kritickým způsobem ovlivněna činnost zadavatele. Typickým příkladem takové vady je například nemožnost spustit aplikaci, výpadek celého modulu, zásadní vady ve funkcionalitě klíčových procesů produktu, které znemožňují jejich používání (nefunkční proces nebo část procesu, vadné výsledky nebo mezivýsledky procesu) apod.
B	Střední	Tato kategorie zahrnuje funkce, které sice zcela selhaly, ale nejsou v daný moment využívány a nemají žádný vliv na řádný chod systému. Vada má zanedbatelný vliv na činnost zadavatele nebo lze dotčený proces dokončit náhradním postupem. Vada se vyskytuje v izolované části produktu (dílčího plnění) – využívání produktu (dílčího plnění) je částečně ztíženo a nemá vliv na ostatní funkce.
C	Nízká	Produkt (dílčí plnění) je plně operativní, vada je pouze kosmetického charakteru. Vada nemá vliv na činnost zadavatele. Vada nekomplikuje využívání produktu (dílčí plnění) nebo jeho části. Jedná se například o vady v grafice či vady, které zneprůjemňují používání produktu (dílčího plnění).

Tabulka 1- Kategorie vad

2.5 Akceptační procedura

Akceptační procedura je procesem, v jehož rámci probíhá posuzování shody aktuálně předávaného produktu nebo jeho části se závaznými požadavky zadavatele. Akceptační procedura je ukončena stanovením akceptačního výroku a podpisem Akceptačního protokolu.

Průběh akceptační procedury se liší v závislosti na typu dodávky (aplikační SW, HW a systémový SW, dokumentace produktu, poskytnutá služba), která je předmětem akceptace ze strany zadavatele.

Akceptační procedura produktu pro jednotlivé typy dodávek:

- **Aplikační SW:** akceptace aplikačního SW je prováděna na základě provedení akceptačního testování. Procesy a aktivity týkající se plánování a průběhu akceptačního testování (dále také UAT) jsou popsány v tomto dokumentu.
- **Dokument:** akceptace dokumentu/dokumentů bude probíhat minimálně ve 2kolové akceptační proceduře. V prvním kole zadavatel specifikuje připomínky k předané dokumentaci a v následujících kolech posuzuje způsob jejich vypořádání dodavatelem.

Zároveň platí, že ve všech kolech následujících po prvním kole může zadavatel vznášet nové připomínky pouze ke způsobu zapracování připomínek z prvního kola nebo k novým kapitolám/textům, které zapracováním připomínek z prvního kola vznikly a tedy je nebylo možné připomínkovat dříve.

- **HW a systémový SW:** Akceptační procedura probíhá na základě předloženého Předávacího nebo Akceptačního protokolu, který obsahuje výčet všech předávaných HW a SW komponent (systémového SW).
- **Služba:** akceptace probíhá po poskytnutí dané služby na základě předloženého Akceptačního protokolu.

2.6 Akceptační výrok

Akceptační výrok je výsledkem akceptační procedury. Akceptační výrok definuje, zda je dodávka produktu nebo jeho části ve shodě se závaznými požadavky zadavatele. Akceptační výrok nabývá následujících hodnot:

- **1 splňuje** – dodávka je zcela ve shodě s požadavky a nebyly zjištěny žádné nedostatky.
- **2 splňuje s výhradou** – dodávka je ve shodě s požadavky, byly zjištěny pouze takové rozpory, které nebrání v akceptaci. Zjištěné rozpory budou odstraněny během další fáze projektu nebo v rámci záručního provozu.
- **3 nesplňuje (odstranitelné vady)** – dodávka vykazuje neakceptovatelné rozpory, které lze odstranit, a musí být odstraněny před zahájením další fáze projektu,
- **4 nesplňuje (neodstranitelné vady)** – dodávka vykazuje neakceptovatelné rozpory, které nelze odstranit. V tomto případě je nutno zastavit projekt a změnit jeho koncept – vyžaduje souhlas řídicího výboru.

Akceptační výrok souvisí s nastavením akceptačních metrik pro konkrétní dodávku.

2.7 Akceptační metrika

Akceptační metrikou definujeme možné výsledky akceptačního testování produktu nebo jeho části, tedy akceptační výrok. Akceptační metriky nastavují akceptovatelné/neakceptovatelné počty zjištěných vad a jsou podkladem pro akceptaci/neakceptaci předávaného produktu nebo jeho části zadavatelem. Akceptační metriky pracují s počty a kategoriemi zjištěných vad (viz kapitola Vada).

2.8 Akceptační kritérium

Akceptační kritérium je definice toho, co má produkt přinést koncovému uživateli. Akceptační kritérium má přímou vazbu na jednotlivé požadavky na produkt, přičemž platí vazba na akceptačních kritérií vůči m požadavkům. Splněním všech akceptačních kritérií prokazujeme shodu výsledného produktu se všemi požadavky na tento produkt. Akceptační kritérium je považováno v případě dodávky aplikačního SW za splněné v případě, že všechny testovací scénáře s vazbou na toto kritérium byly úspěšně otestovány.

2.9 Změna

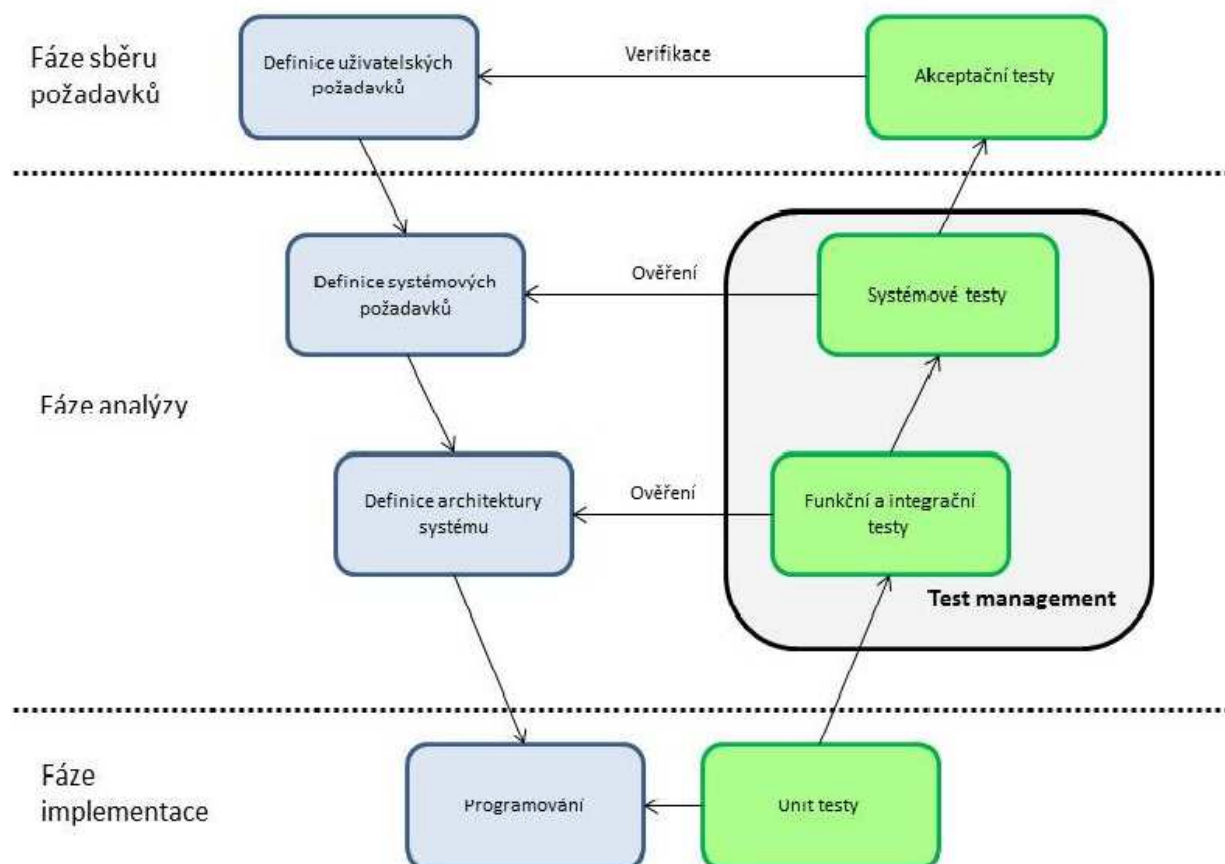
Změna je návrh nebo doporučení vznesené zadavatelem v průběhu trvání projektu. Změna mění původní zdokumentovaný a schválený požadavek. Požadavek na změnu je vždy zaznamenán a jeho vypořádání je řízeno samostatným procesem (change management).

3 Typy testů

3.1 Testování ve vazbě na životní cyklus IS

Každá fáze životního cyklu vývoje IS je svázána s typickou množinou testů. Vazba mezi jednotlivými typy prováděných testů a životním cyklem produktu je znázorněna na obrázku Obrázek 1 - V model testování.

Detailní informace k jednotlivým typům testů jsou uvedeny v kapitole Typy testů IS dle jejich specializace.



Obrázek 1 - V model testování

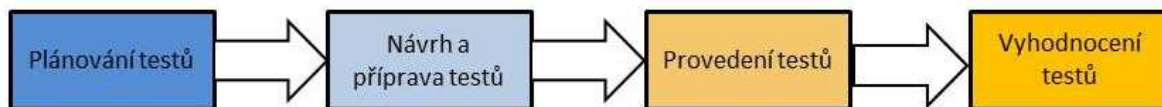
3.2 Typy testů IS dle jejich specializace

3.2.1 Unit testy

Testy jednotlivých aplikačních modulů a jejich vzájemné kooperace na úrovni testování kódu. Jsou realizovány členy vývojového týmu dodavatele a to v každé fázi projektu před předáním do testování testovacímu týmu dodavatele. Unit testy provádí pouze dodavatel a jejich výsledky nejsou předávány zadavateli.

3.2.2 Funkční a integrační testy

Funkční testy slouží k ověření splnění požadavků kladených na funkcionalitu produktu (zpravidla vycházejí z funkčních požadavků na systém). Obrázek 2- Model fází funkčních testů znázorňuje jednotlivé fáze funkčního testování. Funkční testy jsou prováděny na základě připravených nebo zadavatelem schválených testovacích scénářů.



Obrázek 2- Model fází funkčních testů

Integrační testy slouží k ověření integrace (kooperace) mezi jednotlivými moduly aplikačního SW nebo mezi aplikačním SW resp. jeho částí a externím systémem (např. integrace s RUIAN, JSDI, OŘ atd.). Integrační testy jsou prováděny na základě připravených nebo zadavatelem schválených testovacích scénářů. Model fází integračních testů je shodný s modelem pro funkční testy.

3.2.3 Systémové testy a E2E testy

Systémové testy ověřují chování aplikačního SW nebo jeho části z hlediska splnění všech funkčních a nefunkčních požadavků implementovaných v dané fázi projektu. Úkolem systémových testů je ověření **komplexní funkcionality** aplikačního SW nebo jeho části (tedy spojují funkční, integrační, zátěžové a bezpečnostní testy v adekvátním rozsahu odpovídajícím dané fázi projektu) včetně zpětného ověření, že nově integrované funkcionality nenarušily původní funkce aplikačního SW. Ověření, že nebyla narušena integrita a funkčnost původních funkcionalit, je realizováno prostřednictvím smoke a regresních testů, které mohou být automatizovány nebo jsou prováděny manuálně.

- **Smoke testy:** zběžné ujištění, např. že nová verze aplikace (release) není vysloveně rozbitá, než se začne intenzivněji používat (testovat).
- **Regresní testy:** ověření funkcionality aplikačního SW širokou škálou testů a to i méně používaných funkcionalit. Kompletní regresní testy jsou prováděny vždy na finálním release aplikačního SW v dané fázi projektu před předáním zadavateli do akceptačního testování.

Systémové testy realizuje dodavatel vždy před předáním aplikačního SW nebo jeho části zadavateli do akceptačního testování. Tyto testy jsou prováděny na základě zadavatelem schválené sady testovacích scénářů a dat.

3.2.4 Zátěžové testy

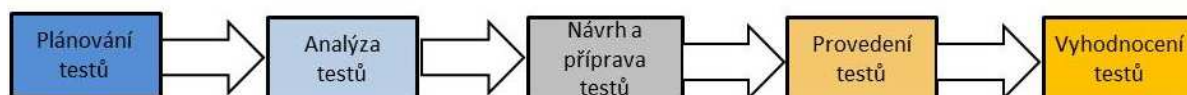
Zátěžové testy slouží k ověření požadovaných výkonnostních parametrů aplikačního SW s cílem najít slabá místa produktu a vyladit výkonnostní parametry produktu jako celku. Na přípravě adekvátní testovací dokumentace, potřebných nástrojů (scriptů atd.) a na provedení testů se zpravidla podílí testovací tým dodavatele ve spolupráci s vývojovým týmem a architekty dodavatele. Zátěžový test může být využit k následujícím účelům:

- ověřování výkonnosti testované aplikace jako součást akceptačního řízení při předávání produktu zadavateli,

- ladění výkonnosti produktu jako součást jejího vývoje,
- zjištění výkonnostních limitů produktu,
- ověření chování produktu při výpadku jednotlivých komponent HW architektury (stress test).

Realizace zátěžových testů může a nemusí být vázána na provedení testů funkčních a integračních/systémových, to je závislé na typu a účelu testu.

Celková koncepce realizace řešení je rozdělena do následujících fází:



Obrázek 3 - Model provedení zátěžových testů

3.2.5 Testy bezpečnosti

Tyto testy ověřují splnění bezpečnostních požadavků na moduly/rozhraní nebo produkt. Celé řešení NIS-IZS bude provozováno uvnitř zabezpečené sítě zadavatele. Bezpečnostní testy proto budou mít zejména charakter ověření autentizace a autorizace k systému tak, aby nemohlo dojít ke zneužití údajů ze strany pracovníků zadavatele. Přestože toto riziko je mnohem menší, než pokud by byl systém veřejně dostupný, nelze ho podceňovat.

Další testy bezpečnosti jsou popsány v příloze F k Prováděcímu konceptu.

3.3 Typy testů IS dle jejich účelu

3.3.1 Interní testy dodavatele

Interní testování produktu provádí testovací tým ČPOZ. Cílem interního testování je průběžné ověřování kvality a správné funkčnosti testovaného aplikačního SW nebo jeho části před tím, než je předán zadavateli k akceptaci. Dodavatel provádí všechny typy testů dle jejich specializace.

3.3.2 Akceptační testy (UAT) zadavatele

Účelem akceptačního testování zadavatelem je formální ověření kvality a funkčnosti aplikačního SW a jeho následná akceptace. Akceptací zadavatel potvrzuje, že akceptovaný aplikační SW nebo jeho část splňují jím definované požadavky dodávky.

Akceptační testování aplikačního SW je prováděno **výhradně** na základě schváleného Plánu testování v **pouze v rozsahu schválené sady** testovacích scénářů a testovacích dat (dále také detailní testovací dokumentace). Schválení testovacích scénářů je popsáno v kapitole Schvalování detailní testovací dokumentace zadavatelem.

Akceptační testování provádí jmenované osoby (testeři) zadavatele **za přímé podpory** Test analytiků v čele s Vedoucím testování (testovací tým ČPOZ je vždy fyzicky přítomen UAT).

Vedoucí testování je oprávněn přizvat k účasti na UAT i další členy implementačního týmu (Analytiky, Vývojáře apod.).

4 Role

Kapitola obsahuje role, které se testování přímo účastní nebo s procesy testování souvisejí. Pro každou roli je uveden popis jejich předpokládaných činností a odpovědností ve vazbě na testování. Další odpovědnosti, povinnosti a práva níže uvedených rolí nesouvisející s testováním a akceptací nejsou touto metodikou řešena.

4.1 Role ČPOZ

Role	Popis činnosti
Vedoucí projektu	<ul style="list-style-type: none"> Účastní se na definici strategie testování, akceptačních metrik a rozsahu testů v každé fázi projektu. Schvaluje testovací dokumentaci před jejím předáním zadavateli. Předává zadavateli ke schválení testovací dokumentaci dané fáze (Testovací plán, testovací scénáře a data). Protokolárně předává produkt nebo jeho dílčí plnění zadavateli do akceptační procedury. Může se účastnit akceptačního testování. Přebírá od zadavatele výsledky akceptačního testování (i průběžné v rámci jednotlivých kol UAT). Odpovídá za vypořádání jednotlivých vad a řeší eskalované problémy. Přebírá o zadavatele akceptační výrok.
Vedoucí testování	<ul style="list-style-type: none"> Ve spolupráci s vedoucím projektu (dodavatele) definuje strategii, metriky a rozsah testů v jednotlivých fázích projektu. Odpovídá za přípravu adekvátní testovací dokumentace (Testovacího plánu, testovacích dat a testovacích scénářů). Organizuje a řídí aktivity testovacího týmu ve všech fázích projektu. Řídí interní testování dodavatele. Účastní se akceptačního testování. Koordinuje aktivity spojené s opravami vad a nasazováním nových verzí aplikačního SW. Eviduje podněty na změny a předává je vedoucímu projektu dodavatele k řešení.
Test analytik	<ul style="list-style-type: none"> Spolupracuje na tvorbě testovacích scénářů. Definuje požadavky na zátěžové testování. Definuje požadavky na testovací data a podílí se na jejich tvorbě. Provádí testování. Eviduje výsledky jednotlivých testů a nalezené vady. Informuje Vedoucího testování o případných problémech. Poskytuje odbornou podporu Testerům. Dává podněty na změny. Může se účastnit akceptačního testování.
Tester	<ul style="list-style-type: none"> Provádí testy podle testovací dokumentace v souladu s platným harmonogramem. Eviduje výsledky jednotlivých testů a nalezené vady. Informuje Vedoucího testování o případných problémech.

Role	Popis činnosti
Administrátor prostředí	<ul style="list-style-type: none"> • Instaluje veškerý SW potřebný pro testování na servery i pracovní stanice testerů. • Instaluje release aplikace do testovacího prostředí. • Provádí zálohy a obnovy prostředí. • Odpovídá za konfiguraci a funkčnost testovacího prostředí. • Může se účastnit akceptačního testování.
Analytik	<ul style="list-style-type: none"> • Poskytuje konzultace Test analytikovi při tvorbě testovacích scénářů a definici testovacích dat. • Spolupodílí se na přípravě testovacích dat. • Analyzuje podněty na změny. • Informuje Vedoucího testování, případně Test analytiky o změnách ve schválené analytické dokumentaci. • Spolupracuje na přípravě seznamu požadavků na dodávku. • Může se účastnit akceptačního testování.
Vedoucí vývoje	<ul style="list-style-type: none"> • Informuje Vedoucího testování o novém release aplikačního SW. • Odpovídá za přípravu release notes. • Řídí aktivity vývojového týmu, koordinuje a plánuje činnosti programátorů při přípravě nového release a při opravách vad. • Předává pracovníkům připravujícím interní, systémové a akceptační testy všechny potřebné informace o testovaném systému. • Organizačně se podílí na přípravě testovacích dat. • Přiděluje řešitele (Programátora) k vadám nalezeným v průběhu všech stádií testů. • Kontroluje řešení chyb v souladu s platným harmonogramem. • Odpovídá za realizaci unit testů. • Může se účastnit akceptačního testování.
Programátor	<ul style="list-style-type: none"> • Vytváří aplikační SW. • Provádí unit testy. • Opravuje detekované chyby. • Podílí se na přípravě automatizovaných testovacích scriptů a scriptů, dat atd. pro zátěžové testování. • Poskytuje konzultace Analytikovi testování při tvorbě testovacích scénářů a spolupracuje na přípravě testovacích dat. • Může se účastnit akceptačního testování.
Architekt	<ul style="list-style-type: none"> • Poskytuje konzultace Test analytikovi při tvorbě testovacích scénářů a definici testovacích dat. • Podílí se na přípravě bezpečnostních testů. • Spolupracuje na přípravě seznamu požadavků na dodávku. • Může se účastnit akceptačního testování.

Tabulka 2- Role ČPOZ

4.2 Role zadavatele

Role	Popis činnosti
Věcný gestor	<ul style="list-style-type: none"> • Je členem akceptační komise. • Stvrzuje Akceptační protokol.

Role	Popis činnosti
Vedoucí projektu	<ul style="list-style-type: none"> • Je členem akceptační komise. • Účastní se akceptačního testování. • Organizuje a řídí přípravu a provedení akceptačních testů zadavatelem. • Spolupracuje s Vedoucím projektu dodavatele na plánování akceptačních testů, definici strategie, rozsahu a akceptačních metrik akceptačního testování. • Schvaluje testovací dokumentaci (Testovací plán, testovací data a testovací scénáře). • Odpovídá za zajištění adekvátního počtu Testerů zadavatele a realizaci UAT dle schváleného harmonogramu a dle podmínek definovaných v této metodice. • Předává Vedoucímu projektu dodavatele konsolidované výsledky akceptačního testování v jednotlivých kolech UAT. • Řeší eskalované problémy a zajišťuje součinnost zadavatele pro řešení hlášených vad.
Věcný gestor složky	<ul style="list-style-type: none"> • Je členem akceptační komise. • Může se účastnit akceptačního testování. • Jmenuje testery pro realizaci UAT.
QA	<ul style="list-style-type: none"> • Je členem akceptační komise.
Tester	<ul style="list-style-type: none"> • Provádí akceptační testy podle schválené testovací dokumentace. • Eviduje zjištěné neshody. • Vyplňuje záznamy o výsledcích akceptačních testů.

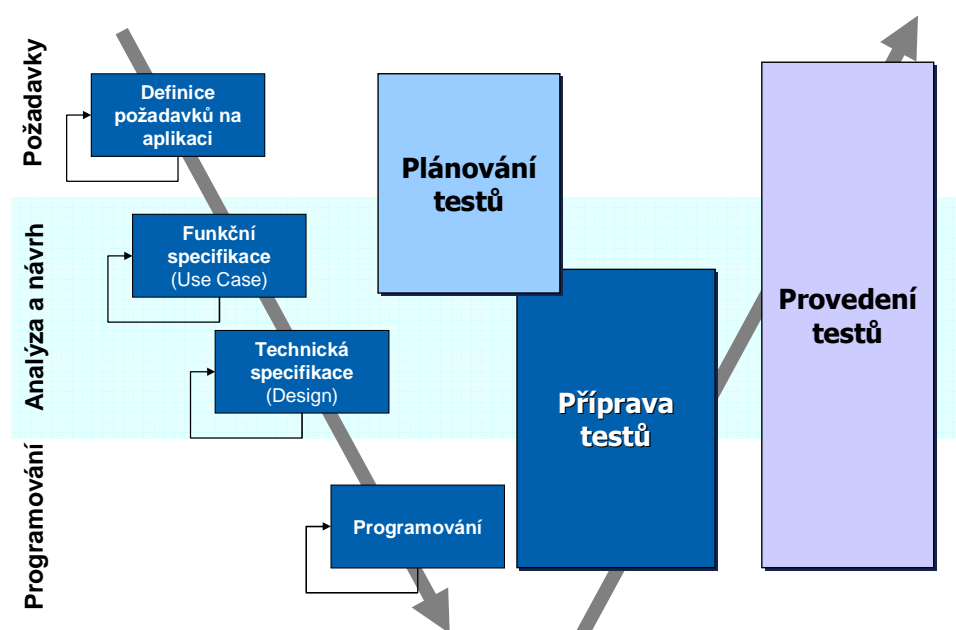
Tabulka 3- Role zadavatele

5 Fáze testování

Proces testování rozdělujeme do 3 fází:

- Plánování testů.
- Příprava testů.
- Provedení testů.

Všechny 3 fáze se periodicky opakují pro každou dílčí nebo celkovou dodávku produktu. Na Obrázek 4 – Fáze testování v průběhu životního cyklu vývoje SW je znázorněn vztah jednotlivých fází testování a životního cyklu vývoje SW.



Obrázek 4 – Fáze testování v průběhu životního cyklu vývoje SW

5.1 Plánování testů

Plánování testů je zahájeno již v úvodní projektové fázi, ve které celý realizační tým ČPOZ spolupracuje na přípravě detailního harmonogramu projektu. V detailním projektovém harmonogramu by měly být zachyceny základní aktivity, jejich časování a subjekty, které budou v průběhu trvání projektu participovat v oblasti testování. Z hlediska odpovědného plánování testování (a nejen testování) musí být provedeny následující aktivity:

- **Detailní analýza smluvního harmonogramu projektu s cílem:**
 - identifikovat projektové milníky, ve kterých je zadavatelem schvalována analytická a technická dokumentace. Schválená analytická a technická dokumentace společně s harmonogramem projektu tvoří základní vstupní sadu dokumentů, se kterou testovací tým pracuje.

- identifikovat projektové milníky, ve kterých je produkt nebo jeho část předávána zadavateli do akceptačního testování (do dílčích nebo komplexních akceptačních testů). Toto tvoří základní rámec pro naplánování testování.
- **Specifikace faktického obsahu produktu nebo jeho části**, který je předmětem předání a akceptace zadavatelem v jednotlivých projektových milnících. Cílem této aktivity je přiřadit jednotlivé požadavky na aplikační SW k jednotlivým projektovým milníkům a jasně identifikovat akceptační kritéria, která se k těmto požadavkům váží. Při mapování požadavků na jednotlivé milníky dodávek spolupracuje Vedoucí testování zejména s Analytiky (ale může spolupracovat také s Architektury a Programátory resp. s jejich vedoucími a to zejména při plánování stress nebo zátěžových testů). Realizací tohoto kroku je ke každému milníku jasně identifikován:
 - předpokládaný objem testovacích prací, které je třeba vykonat a z toho vyplývající čas, který je potřebný na realizaci testů.
 - seznam akceptačních kritérií, které lze po úspěšné akceptaci produktu nebo jeho části považovat za splněné.
- **Identifikace projektových rizik s dopadem na testování.**

Všechny výše uvedené body jsou zcela zásadní, neboť určují nejen objem a pracnost všech aktivit týkajících se testování produktu při každé jeho dodávce, ale také stanovují časový rámec (počet testovacích kol, jejich délku a termíny) těchto aktivit a tím i předurčují strategii testování (které typy testů a pro jakou množinu požadavků budou zařazeny k danému projektovému milníku), která může být v rámci jednotlivých projektových milníků aplikována.

5.1.1 Délka a počet testovacích kol

Délka a počet kol testování (interního i UAT) závisí na harmonogramu projektu. Harmonogram projektu musí být sestaven tak, aby bylo vždy zajištěno **minimálně** dvoukolové testování (a to jak interní tak akceptační) s časovou rezervou pro opravu zjištěných vad.

Výstupním dokumentem této fáze je: aktualizace projektového plánu a Plán testování. Plán testování se zhotovuje pro každou fázi projektu. Struktura Plánu testování je popsána v kapitole Plán testování.

5.2 Příprava testů

Příprava testů začíná po schválení analytického výstupu zadavatelem. Kvalita připravených testů a scénářů je závislá na kvalitě analytické dokumentace a na zvládnutí procesu změnového řízení ze strany vedení projektu.

Výstupními dokumenty této fáze jsou:

- testovací scénáře,
- testovací data,
- automatizované testovací skripty.

Struktura testovacích scénářů je popsána v kapitole Testovací scénář.

Testovací scénáře a návrh testovacích dat (dále také „detailní testovací dokumentace“) jsou:

- zpracovávány ke každému milníku předání produktu nebo jeho části,
- schvalovány zadavatelem a to v rozsahu popsaném v kapitole Schvalování detailní testovací dokumentace zadavatelem 5.2.1.

Rozsah zpracování detailní testovací dokumentace je řízen obsahem konkrétní dodávky. Vzhledem k tomu, že cílem ČPOZ je dosáhnout maximální kvality a souladu předávaného produktu nebo jeho části se závaznými požadavky a definovanými akceptačními kritérii, zpracovává testovací tým co nejširší sadu testovacích scénářů tak, aby bylo v různých fázích projektu dle obsahu dodávky minimálně pokryto:

- ověření kompletní business funkcionality (všech funkcionalit a procesů včetně jejich vzájemné integrace),
- ověření správného chování UI prvků,
- ověření hraničních podmínek:
 - délky řetězců,
 - zadání čísla větší než je maximum/minimum atd.,
 - testování vnitřních omezení (např. datové typy),
 - ověření chování aplikace u výchozích, prázdných, nevyplněných a nulových hodnot,
- ověření chování aplikace na zadání neplatných nebo nesmyslných údajů,
- stresové testování – test běhu aplikace při nejhorších možných podmínkách (výpadky komponent atd.)
- zátěžové testování – simulace velkého množství transakcí atd.
- ověření bezpečnosti.

Testovací dokumentace může být ještě doplňována nebo zpřesňována v průběhu provádění interních testů dodavatele.

5.2.1 Schvalování detailní testovací dokumentace zadavatelem

Před zahájením systémových testů dodavatelem je nezbytně nutné zajistit schválení závazné sady detailní testovací dokumentace zadavatelem. Na základě této schválené sady budou provedeny systémové a také akceptační testy.

Zadavatel schvaluje rozsah, komplexnost a obsah navrhovaných testů.

Schvalování dokumentace zadavatelem bude probíhat ve 2. maximálně 3. kolovém řízení¹ se zachováním principů, které jsou popsány níže v příkladu.

Příklad:

- I. kolo: zástupci zadavatele posuzují úplnost a kvalitu předloženého návrhu testovacích scénářů a testovacích dat ve vztahu k aktuálnímu dodávce produktu nebo části produktu. Na konci I. kola předá zadavatel dodavateli konsolidovaný seznam připomínek k zapracování. U každé připomínky je vždy uvedena minimálně následující sada informací:
 - identifikace testovacího scénáře nebo požadavku (v případě, že testovací scénář chybí), případně kroku testu, ke kterému se připomínka váže.
 - autor připomínky.
 - text připomínky.
- Pauza pro analýzu a zapracování připomínek. Vedoucí testování provede ve spolupráci s Test analytiky rozbor předaných připomínek a:
 - relevantní připomínky předá k zapracování příslušnému Test analytikovi.
 - připomínky zadavatele, které byly vyhodnoceny jako nerelevantní, předá dodavatel nejpozději na konci tohoto období zpět zadavateli s adekvátním odůvodněním (nejlépe odkazem na požadavek nebo detailní analýzu).
 - V případě, že u této sady připomínek nedojde ke shodě obou stran (připomínka je zadavatelem protokolárně zrušena nebo naopak dodavatelem protokolárně přijata jako oprávněná), budou tyto neshody řešeny standardními projektovými eskalačními procedurami.
- Na konci tohoto období předává dodavatel zadavateli následující dokumenty:
 - doplněnou sadu testovacích scénářů.

1 Konkrétní počet kol, jejich termíny a délka jsou definovány v závislosti na možnostech Projektu v dokumentu Plán testování.

- seznam připomínek s uvedením způsobu zpracování dané připomínky.
- II. kolo: zadavatel ve II. kole posuzuje pouze způsob zpracování připomínek z I. kola. zadavatel je oprávněn ve II. kole podávat nové připomínky k testovacím scénářům pouze v těch případech, kdy to nebylo možné v kole prvním (např. z důvodu absence daného testu).
- III. kolo - poslední kolo, ve kterém jsou již pouze posouzeny způsoby zpracování připomínek z kola druhého.

Detailní sada testovací dokumentace musí být schválena zadavatelem v souladu s harmonogramem uvedeným v Plánu testování a to nejpozději před zahájením systémových testů dodavatelem.

5.3 Provedení testů

Provedení funkčního a integračního, systémového atd. nebo akceptačního testování probíhá v cyklech dle Plánu testování.

5.3.1 Předávání produktu do interního testování ČPOZ

Vedoucí testování musí být informován o novém release do testování. Součástí předání release je vždy dokument release notes, který musí obsahovat z hlediska testování minimálně následující informace:

- verze release,
- nově implementovaných funkcionalitách nebo změnách ve stávající funkcionalitě, které nevyplývají přímo z funkční specifikace, ale například technického řešení,
- opravených vadách produktu nebo jeho části (minimálně seznam obsahující unikátní identifikátor opravené vady).

Od okamžiku instalace release v testovacím prostředí není povolena instalace dalších release do tohoto prostředí bez schválení Vedoucím testování.

5.3.2 Provedení interního testování ČPOZ

Nejvíce cyklů je realizováno v průběhu interních testů dodavatele, kdy jsou postupně uvolňovány implementované dílčí funkcionality na základě fází dle harmonogramu Projektu. Efektivita testování v tomto stádiu je závislá na organizaci vývojových prací, tak aby mohly být pro testování uvolňovány smysluplné funkční celky a na kvalitě vývojového testování.

Jak bylo řečeno výše, interní testování probíhá na základě připravených nebo schválených testovacích scénářů v závislosti na tom, zda jsou prováděny funkční anebo integrační testy průběžných release produktu anebo zda jsou prováděny systémové testy před předáním produktu nebo jeho části zadavateli do akceptačního řízení.

5.3.2.1 Funkční a integrační testy

Funkční a integrační testy probíhají průběžně dle dohodnutého harmonogramu nebo mohou být prováděny ad hoc na vyžádání Vedoucím testování, Vedoucím projektu dodavatele nebo Vedoucím vývoje a to na základě dohody všech zúčastněných subjektů.

Po funkčních a integračních testech nikdy není aplikace předávána do akceptačního testování zadavatelem, předání aplikace do UAT je možné vždy až po provedení Systémových testů. Toto je jedním z důvodů, proč můžeme v této fázi testování „tolerovat“ i chyby kritické (vady A). Vždy však platí, že u všech evidovaných vad, nejen kritických, musí být jednoznačně stanoven termín opravy.

Stanovení termínu opravy se řídí následujícími pravidly:

- Kritické vady (A) musí být odstraněny okamžitě v těch případech, kdy jejich existence brání pokračování testování (např. aplikaci nelze spustit, do aplikace se nelze přihlásit atd.).
- Ostatní kritické vady (A) musí být odstraněny do zahájení dalšího kola funkčního a integračního testování a nejpozději do zahájení dalších plánovaných funkčních a

integračních testů případně do zahájení prvního kola systémových testů, pokud tyto testy následují po právě prováděném funkčním a integračním testování.

- Ostatní vady (C) musí být odstraněny nejpozději do nejbližšího 1. kola systémových testů.

Výsledky těchto průběžných testů nejsou předávány zhotoviteli.

Vstupní kritéria pro zahájení:

- Moduly předané do interního testování prošly vývojovými testy (unit testy) a zjištěné chyby byly opraveny.
- Je připraveno funkční testovací prostředí.
- Byla potvrzena připravenost okolních systémů k integračnímu testu.
- Je připravena detailní testovací dokumentace.

Výstupní kritéria pro ukončení:

- Byly provedeny všechny testy na základě testovacích scénářů.
- Všechny zjištěné vady jsou zaevidovány a dále řešeny v souladu s touto metodikou.

Výstupní dokumenty:

- Aktualizované testovací scénáře.
- Vyplněné testovací scénáře.
- Seznam vad včetně jejich kategorizace, vazby na testovací scénář a stanovení termínu jejich opravy.

5.3.2.2 Systémové testy

Vstupní kritéria pro zahájení:

- Jednotlivé moduly produktu prošly dílčími funkčními a integračními testy a splnily kritéria pro jejich ukončení.
- Je připraveno funkční testovací prostředí pro systémové testy.
- Byla potvrzena připravenost okolních systémů k integračnímu testu.
- Zadavatelem byla schválena detailní testovací dokumentace.
- Vady zjištěné v průběhu funkčních a integračních testů byly opraveny.

Výstupní kritéria pro ukončení:

- Byly provedeny všechny testy na základě schválených testovacích scénářů.
- Aplikace neobsahuje žádné kritické vady, které ohrožují úspěšnou akceptaci produktu nebo jeho části.

- U zbylých, nekritických, vad jsou stanoveny termíny opravy.

Výstupní dokumenty:

- Vyplněné testovací scénáře.
- Protokol o provedení systémových testů, který obsahuje minimálně:
 - Seznam provedených testů a jejich výsledek.
 - Seznam zjištěných vad včetně jejich kategorizace, vazby na testovací scénář a stanovení termínu jejich opravy.

5.3.3 Předávání produktu nebo jeho části do akceptace zadavateli

Součástí předání produktu nebo jeho části do akceptačního testování zadavatelem je:

- Protokol o provedení systémových testů – předáván před zahájením I. kola UAT.
- Release notes – předáván před zahájením II. případně dalších kol UAT.

Od okamžiku instalace release v testovacím prostředí nebo jiném prostředí dedikovaném pro akceptační testy není povolena instalace dalších release do tohoto prostředí bez schválení Vedoucího projektu zadavatele.

5.3.4 Akceptační testy

Vstupní kritéria pro zahájení:

- Zadavatelem je schválen Plán testování pro danou fázi.
- Jednotlivé moduly produktu prošly systémovými testy a splnily kritéria pro jejich ukončení.
- Je připraveno funkční testovací (nebo jiné dedikované) prostředí pro akceptační testy.
- Byla potvrzena připravenost okolních systémů k integračnímu testu.
- Vady zjištěné v průběhu systémových testů byly opraveny.

Výstupní kritéria pro ukončení:

- Byly provedeny všechny testy na základě schválených testovacích scénářů.
- Aplikace neobsahuje žádné kritické vady, které ohrožují úspěšnou akceptaci produktu nebo jeho části.
- U zbylých, nekritických, vad jsou stanoveny termíny opravy.

Výstupní dokumenty:

- Vyplněné testovací scénáře.
- Protokol o provedení akceptačních testů, který obsahuje minimálně:
 - Seznam provedených testů a jejich výsledek,

- Seznam zjištěných vad včetně jejich kategorizace, vazby na testovací scénář a stanovení termínu jejich opravy.
- Akceptační protokol.

6 Evidence a řešení vad a změn

6.1 Evidence a řešení vad

Každá vada, která byla zjištěna v průběhu testování produktu nebo jeho části musí být evidována.

6.1.1 Záznam o vadě

Záznam o vadě musí obsahovat minimálně následující sadu informací:

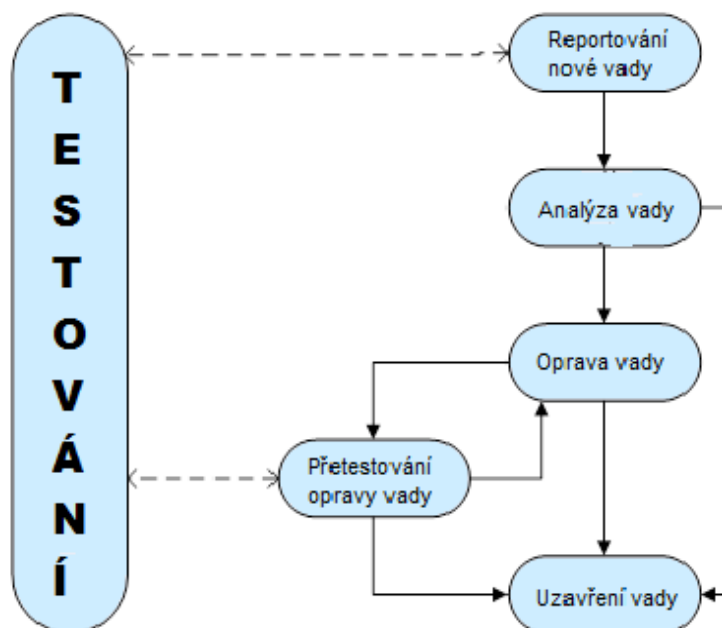
- Název vady: stručný a výstižný text popisující vadu.
- Verze produktu nebo jeho části: identifikuje prostředí, na kterém byl test proveden.
- Datum provedení testu.
- Identifikace subsystému/modulu, ve kterém byla vada nalezena, je-li to možné.
- Kategorie vady. Popis vady: detailní popis vadného chování včetně specifikace testovacích dat a popis postupu, jak vadu zreprodukovat (je-li to možné). Popis může být doplněn URL adresou, otiskem obrazovky nebo výpisem z logu.
- Očekávané chování: popis správného chování.
- Tester.
- Řešitel vady: obvykle jím bývá vedoucí vývoje, který následně vady přiřazuje jednotlivým členům svého týmu k řešení. V případě UAT definuje řešitele vady Vedoucí testování.

Každá vada, která je v průběhu interního testování zjištěna, musí být zaevidována do testovacího scénáře (v případě systémových testů a UAT), při jehož testování byla zjištěna a zaznamenána do příslušného SW nástroje, který podporuje řízení životního cyklu vady.

Za evidenci zjištěné vady v souladu s touto metodikou odpovídá příslušný tester ČPOZ nebo Vedoucí projektu zadavatele v případě, že se jedná o vadu vzniklou v průběhu akceptačního testování.

6.1.2 Řešení vady

Řešení vady je řízeno prostřednictvím workflow, které je naznačeno na Obrázek 5 - Workflow řízení vady.



Obrázek 5 - Workflow řízení vady

Z obrázku je patrné, že řešení vady prochází celou řadou disciplín od analýzy, implementace opravy až po nasazení opravy do testovacího prostředí, její přetestování a uzavření nebo vrácení zpět k opravě.

Pokud je vada předávána vývojovým týmem zpět k internímu přetestování jako opravená, musí být změněn stav vady (např. vyřešena, k testování apod.) a záznam o vadě doplněn/aktualizován o následující údaje:

- Skutečný řešitel vady.
- Způsob vyřešení vady.
- Identifikace verze produktu (release), ve kterém je vada předávána zpět k přetestování.
- Tester, jehož úkolem je vadu přetestovat. Zpravidla je zadáván Vedoucí testování, jehož úkolem je následně dedikovat otestování vady na jednotlivé Testery. Pokud je vada vrácena po provedení retestu, může řešitel této vady buď vadu po opravě přiřadit na posledního testera, který vadu ověřoval anebo zpět na Vedoucího testování, který její ověření Testerem zajistí.

Pokud je vada předávána dodavatelem zadavateli k přetestování, odpovídá za aktualizaci výše uvedených informací Vedoucí testování.

Tester odpovídá za ověření opravy vady v daném release a doplnění záznamu o vadě o následující informace:

- Datum a identifikace release, ve kterém bylo Testerem ověření provedeno.
- Výsledek testu.

- Pokud je retest úspěšný, Tester tuto skutečnost zaznamená a uzavře vadu změnou jejího stavu (například otestováno, opraveno apod.).
- Pokud je retest neúspěšný, Tester tuto skutečnost zaznamená a změní její stav na (vrácena apod.). Tester v tomto případě přiřadí jako řešitele vady posledního řešitele (Programátora), který vadu opravil a předal ji zpět k retestům.

Pokud je vada vrácena zpět jako neopodstatněná, musí být její záznam doplněn o následující údaje:

- Skutečný řešitel vady.
- Vysvětlení odmítnutí vady, které musí mít oporu buď v analytickém, technickém anebo jiném závazném projektovém dokumentu.

V případě, že je jako neopodstatněná vada vrácena zpět vada z UAT, rozhoduje Vedoucí projektu zadavatele o dalších krocích (akceptuje vysvětlení, odmítne vysvětlení a vadu vrátí zpět k opravě). Pokud v těchto případech nedojde ke shodě, je situace dále řešena eskalačními procedurami.

Uzavření vady jsou oprávněny provádět:

- Vedoucí testování v případě, že se jedná o neopodstatněné hlášení vady z interního testování ČPOZ a to pouze na základě písemného vysvětlení a argumentů podaných řešitelem této vady.
- Tester ČPOZ v případě, že se jedná o vadu z interního testování a tato vada byla opravena a přetestována a výsledek retestu byl řádně zdokumentován.
- Vedoucí projektu zadavatele nebo jiná, jím pověřená osoba, jedná-li se o neprokázanou nebo neopodstatněnou vadu hlášenou v průběhu akceptačního testování a toto bylo ze strany ČPOZ písemně prokázáno a zadavatelem akceptováno anebo v případě, že hlášená vada z akceptačního testování byla opravena, testerem zadavatele úspěšně přetestována a výsledek retestu byl řádně zdokumentován.

Změnu kategorie vady je oprávněn provádět:

- Tester ČPOZ v případě, že se jedná o vadu z interního testování a dopad opravy této vady po přetestování, jehož výsledek byl Testerem řádně zdokumentován, umožňuje kategorii vady zvýšit/snížit.
- Vedoucí testování může u vady z interního testování měnit kategorii vady v závislosti na době, po kterou je vada buď neřešena anebo neúspěšně opravována v plánovaných releasech (obvykle se jedná o zvýšení kategorie vady).
- Vedoucí projektu zadavatele nebo jím pověřená osoba u vady z UAT, ale pouze v případě, že dodavatel s touto změnou souhlasí. Případný nesoulad obou stran je řešen standardními eskalačními procedurami.

6.2 Řízení změn (change management)

Change management se zabývá řízením změn, které nastanou v průběhu projektu. V této metodice se zabýváme pouze těmi změnami, které mají přímý dopad na testování produktu nebo jeho částí (tj. změny s dopadem do definice požadavků na aplikační SW a tedy s dopadem do analytické dokumentace). Samotný proces řízení změn není součástí této metodiky.

V případě, že v průběhu projektu dojde ke změně, odpovídá Vedoucí testování, po jejím schválení a zařazení do implementačního plánu, za aktualizaci testovací dokumentace:

- aktualizaci Plánu testování v souladu s aktualizovaným harmonogramem projektu.
- aktualizaci detailní testovací dokumentace.
- exekuci testů pokrývajících implementovanou změnu v souladu s plánem.

Aktualizovaná testovací dokumentace musí být přeschválena zadavatelem.

7 Artefakty procesu testování

7.1 Vstupní dokumentace

Název dokumentu	Popis obsahu
Projektová dokumentace	<p>Zadavatelem schválené dokumenty, které jsou na projektu postupně vytvářeny:</p> <ul style="list-style-type: none"> • Katalog uživatelských požadavků, • Katalog systémových požadavků, • Analytická dokumentace (Funkční specifikace), • Technická specifikace, • Popis architektury celého systému, • Uživatelská příručka (je-li k dispozici), • Instalační příručka (je-li k dispozici).
Harmonogram projektu	<ul style="list-style-type: none"> • Harmonogram celého projektu, s rozpracováním fáze vývoje jednotlivých subdodávek.
Akceptační kritéria	<ul style="list-style-type: none"> • Požadavky zadavatele na funkčnost, výkonnost, bezpečnost systému. • Požadavky zadavatele na způsob ověřování systému – zadavatelem požadované typy testů apod. • Požadavky zadavatele na způsob převzetí systému (požadavky na akceptační testování apod.).

Tabulka 4- Vstupní dokumentace

7.2 Plán testování

Plán testování je taktickým dokumentem, který vychází z principů definovaných touto metodikou a rozpracovává je v detailu pro potřeby plánování jednotlivých dodávek a jejich testování v souladu s klíčovými milníky projektu.

Účelem dokumentu Plán testování je stanovit:

- Cíl testování: definuje, čeho by mělo být testováním dosaženo. Cílem může být ověření všech funkcností systému a podobně.
- Testovací strategie: popisuje, jaké typy testů budou provedeny. Jednotlivé testy pak zařazuje do konkrétních fází dodávek produktu nebo jejich částí. Dále se zde popisují techniky, kterými budou prováděny jednotlivé typy testů.
- Harmonogram testování. Harmonogram testování obsahuje vždy plánování termínů a délky interních testů ČPOZ a akceptačních testů zadavatele.
- Rozsah testování. Musí být jasně vyspecifikováno, co, jak a s jakými testovacími daty bude v rámci dané dodávky produktu nebo jeho části testováno. Podkladem pro definici rozsahu testování je tabulka požadavků na aplikační SW včetně jejich mapování na akceptační kritéria a jednotlivé milníky dodávek produktu nebo jeho částí. Pro část UAT bude specifikován závazný seznam testovacích scénářů, na jejich základě budou UAT realizovány.

- Definice akceptačních metrik.
- Požadavky na zdroje a součinnosti: slouží k definování požadavků, které musí být splněny ČPOZ nebo zadavatelem, aby bylo možné plánované testy realizovat. Sem se řadí například požadavky technického rázu (např. zpřístupnění a instalace prostředí), dále požadavky na lidské zdroje, kancelářskou techniku atd.
- Definice rizik: slouží k vymezení situací, které mohou ohrozit úspěšné testování. U těchto rizik se stanoví míra jejich závažnosti a návrh protipatření.

Za zpracování Plánu testování odpovídá Vedoucí testování, který jej prostřednictvím Vedoucího projektu dodavatele předává zadavateli ke schválení. Schválený Plán testování je závazný pro ČPOZ i zadavatele a jsou dle něj koordinovány a realizovány jednotlivé etapy testování.

7.3 Testovací scénář

Testovací scénář je předpis, na jehož základě je Testerem prováděn test konkrétní definované funkcionality nebo procesu.

Předpis testovacího případu musí obsahovat minimálně následující sadu informací:

- Unikátní identifikátor testovacího scénáře a krátký název.
- Stručný popis cíle testu.
- Identifikaci požadavku.
- Identifikaci souvisejících akceptačních kritérií, jejichž splnění nebo částečné splnění bude testem prokazováno.
- Identifikaci use case nebo jiného analytického předpisu, na jehož základě byl test zpracován.
- Výchozí podmínky, které musí být splněny před realizací testu (včetně testovacích dat).
- Strukturovaný popis kroků, které jsou prováděny testerem včetně očekávaného chování testovaného produktu nebo jeho části.

Při realizaci testů musí být do testovacího scénáře zaznamenána minimálně následující sada informací:

- jméno Testera, který test provedl,
- datum provedení testu,
- verze prostředí a SW, na kterém byl test proveden,
- záznam o vadě, byla-li zjištěna,
- konečný stav testu. Konečný stav testu identifikuje, zda byl test úspěšně nebo neúspěšně proveden a může nabývat následujících stavů:
 - vyhověl – při testu nebyly zjištěny žádné vady,

- vyhověl s výhradou – při testu nebyly zjištěny žádné kritické vady (kategorie A nebo B),
- nevyhověl – při testu byla zjištěna alespoň jedna kritická vada (kategorie A nebo B).

Konečný stav testu není rozhodující pro akceptační výrok, ten je vždy řízen akceptační metrikou platnou (počtem a kategoriemi hlášených vad) pro danou dodávku.

7.4 Testovací data

Specifikace testovacích dat probíhá pod vedením Vedoucího testování za součinnosti dalších členů implementačního týmu (Analytik, Programátor apod.).

Testovací data mohou být:

- přímo součástí testovaného produktu,
- přímo součástí testovacích scénářů nebo
- v samostatných souborech.

7.5 Protokol o provedení systémových testů

Dokument se vytváří při ukončení stádia systémových testů a předání aplikace k akceptačním testům. Může být rozdělen do částí pro funkční, integrační a jiné typy testů. Jednotlivé části obsahují:

- Název a verzi testovaného subsystému.
- Identifikátor testovacího prostředí, na němž byly testy prováděny.
- Datum provedení testů.
- Identifikátory jednotlivých testovacích scénářů a záznam jejich výsledku.
- Statistické přehledy o počtu detekovaných a neodstraněných vad (tříděné dle kategorie), s nimiž je produkt nebo jeho část předáván k akceptačnímu testu.
- Závěr – obsahuje dohodu Vedoucích projektů o termínech a verzích, v nichž budou odstraněny chyby nebo realizovány případné změnové požadavky.

7.6 Protokol o provedení akceptačního testování

Závěrečný protokol pro vyhodnocení a ukončení stádia akceptačních testů. Dokument obsahuje:

- Název a verzi testovaného subsystému.
- Shrnutí průběhu akceptačních testů.
- Datum provedení testů.
- Identifikátor testovacího prostředí, na němž byly testy prováděny.

- Identifikátory jednotlivých testovacích scénářů a záznam jejich výsledku.
- Statistické přehledy o počtu detekovaných a neodstraněných vad (tříděné dle kategorie), s nimiž bylo ukončeno UAT. U těchto vad je definován termín/fáze, do kdy musí být vady odstraněny.
- Závěr obsahující akceptační výrok.

Protokol podepisuje Vedoucí projektu dodavatele, Vedoucí testování a Vedoucí projektu zadavatele. Protokol o provedení akceptačního testování tvoří přílohu Akceptačního protokolu.

8 SW nástroje

8.1 Nástroj pro evidenci a řízení životního cyklu vad

Pro evidenci a řízení vad bude použit podpůrný nástroj Youtrack. Tento systém:

- Umožňuje správu více projektů – bude využito pro oddělení evidence interních vad a vad z UAT.
- Umožňuje správu uživatelských práv – s ohledem na správu více projektů je nutné, aby systém umožňoval správně nastavit práva a role v rámci těchto projektů tak, informace týkající se daného Projektu byly zpřístupňovány pouze oprávněným osobám, které mají na informace o daném projektu nárok.
- Umožňuje správu uživatelských rolí - každý člen realizačního týmu na projektu přidělenou určitou roli a každá role má stanovené činnosti, které může v systému provádět.
- Umožňuje správu workflow řešení vady. Životní cyklus vady se skládá buď přímo ze stavů, kterými vada prochází (např. nová, opravená, schválená, atd.) nebo z přechodů mezi těmito stavy (např. analyzovat, opravit, otestovat, schválit, vrátit, zrušit atd.). Ke každému stavu nebo změně má přístup jiná role (např. Programátor může vadu opravit, ale nemůže ji schválit, tester ji může nahlásit i schválit, ale nemůže ji opravit atd.). Nástroj by měl umožňovat zadání jednotlivých stavů a jejich přechodů a kromě toho také určení, které role mohou s chybou v daném stavu pracovat.
- Umožňuje evidenci minimálně těch informací, které jsou uvedeny v kapitole Evidence a řešení vad.
- Umožňuje realizovat statistické výstupy - z pohledu vedení projektu jde o klíčovou vlastnost, z pohledu testera se jedná o pomůcku při přípravě reportů výsledků testování pro vedení Projektu.

Tento nástroj spravuje odpovědná osoba, která je jmenována Vedoucím vývojového týmu.

Vedoucí jednotlivých realizačních týmů jsou odpovědní za nominaci těch členů týmu, kterým musí být přidělen přístup a adekvátní role a oprávnění v tomto systému.

Do tohoto nástroje musí mít přístup všechny zainteresované subjekty resp. všichni členové realizačních týmů zadavatele, ČPOZ i externího subdodavatele a to v následujícím rozsahu:

- Členové testovacího a vývojového týmu ČPOZ mají k dispozici přehled a detaily všech vad, které byly zjištěny v průběhu interních nebo akceptačních testů.
- Zástupci zadavatele budou mít k dispozici pouze ty záznamy vad, které vznikly a byly hlášeny v průběhu akceptačního testování.

Do bug-trackingového systému mohou být evidovány i nové úlohy související s rozvojem daného produktu (tedy nikoliv vady, ale také zadání nových funkcionalit).

8.2 Ostatní SW nástroje

Ostatní SW nástroje mohou být používány pro:

- Přípravu testovacích dat (různé generátory).
- Přípravu automatizovaných testů.
- Pro přípravu testovacích skriptů např. pro realizaci zátěžového testování.
- Pro realizaci testování SOAP WS a REST služeb IPL a GIS.
- Pro realizaci zátěžových testů. Nástroje jsou určeny pro tvorbu a řízení zátěžových testů včetně testování výkonnosti webových aplikací. Nástroje vytváří konzistentní, měřitelnou a opakovatelnou zátěž odpovídající reálné zátěži v provozu při maximálním zatížení všech částí systému. Dokáží měřit časové odezvy.

9 Rozsah testování

V této kapitole je uveden stručný výčet oblastí, které budou v průběhu trvání projektu testovány. Rozpad těchto oblastí na nižší celky a identifikace typů testů, které budou v rámci testování každého z nich realizovány, bude pro každou fázi projektu specifikován v Plánu testování.

Oblast testování	Stručný popis
Admin GUI	Testy funkcionality administrátorského rozhraní (správa WL/BL, správa společných číselníků, správa rajonizace, reporting a statistiky, audit, správa synonym, importy VTA apod.).
IPL	<ul style="list-style-type: none"> Příjem DV jejich zpracování (validační a syntaktické kontroly, uložení dat do databáze, odpovědní DV odesílateli s výsledky validačních a syntaktických kontrol). Týká se zejména datových vět pro událost, SaP a společné operační situace. Odesílání DV (generování DV, správné určení adresátů, doručení DV na rozhraní adresátů).
GIS	<ul style="list-style-type: none"> Geokódovací služby pro OŘ – MST helper a reversní geocoding, Replikace uzavírek, Služby liniového místopisu, Služby pro vizualizaci operační situace – SaP a Společná operační situace, Import dat do centrálního GIS z následujících zdrojů: <ul style="list-style-type: none"> Lázně Bohdaneč, JSDI (dopravní informace), ČHMÚ (meteo data), Replikace dat z centrálního GIS na krajský GIS, Úprava GIS dat v centrálním GIS, Odeslání dat ke konsolidaci, Import dat JSDI a ČHMÚ. Ostatní služby GIS dle katalogu služeb GIS.
Autorizace a autentizace uživatelů	a Autorizace a autentizace uživatelů do Admin GUI.
Ostrovní režim	Distribuce zpráv (DV) při výpadku konektivity SKDC (výpadek konektivity mezi KDC a SKDC).
Ostatní	Chování systému při výpadcích některých komponent na KDC nebo SKDC (ověření redundance prvků a služeb na KDC a SKDC). Zátěžové testy systému v rozsahu, který bude zadavatelem požadován pro hybridní variantu.

Tabulka 5- Oblasti testování

10 Přílohy

10.1 Seznam obrázků

Obrázek 1 - V model testování.....	9
Obrázek 2- Model fází funkčních testů.....	10
Obrázek 3 - Model provedení zátěžových testů	11
Obrázek 4 – Fáze testování v průběhu životního cyklu vývoje SW	16
Obrázek 5 - Workflow řízení vady	26

10.2 Seznam tabulek

Tabulka 1- Kategorie vad.....	6
Tabulka 2- Role ČPOZ.....	14
Tabulka 3- Role zadavatele	15
Tabulka 4- Vstupní dokumentace	29
Tabulka 5- Oblasti testování.....	35