

Metodika manažmentu úloh (Redmine)

verzia 2018-01-20

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá manažovania úloh a používateľských príbehov. Metodika taktiež určuje pravidlá pre zaznamenávanie stavu projektu, pokroku a odpracovanom čase za každého člena. Manažment úloh vychádza z metodológie Scrum. Na manažment úloh sa používa systém Redmine.

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	2
Proces: Zistenie svojich úloh	2
Proces: Zaznamenanie postupu vo svojich úlohách	6
Proces: Plánovanie šprintu	9
Proces: Uzatváranie šprintu	15
Proces: Export evidencie úloh zo systému Redmine	16

3. Znenie metodiky

Proces: Zistenie svojich úloh

Tento proces popisuje postup, ako si v systéme Redmine prezriem úlohy, ktoré mi boli pridelené. Proces prezerania svojich úloh sa dá vykonať dvoma spôsobmi - prostredníctvom sekcie Task Board alebo prostredníctvom sekcie Issues. V sekcii Task Board je možné získať lepší prehľad o šprinte a pridelených úlohách, zatiaľ čo v sekcii Issues sa dajú pridelené úlohy vyhľadať ľahšie.

Spôsob 1: Zistenie úloh cez sekciu Task Board

1. Otvoriť si **Redmine** a prejsť do sekcie **Task Board**
 - 1.1. Poznámka: Ak sekcia Task Board nie je dostupná, je to spôsobené skončením šprintu, preto treba v takomto prípade použiť spôsob B
2. Podľa farby úloh (alebo podľa mena umiestneného pri spodnom okraji každej úlohy) vyhľadať v sekcii **Assigned** svoje úlohy
3. Zamerať sa iba na úlohy, ktorých názov začína na **(X)**
 - 3.1. (X) označuje hlavné úlohy. Úlohy, ktorých názov nezačína na (X) sú iba podúlohami hlavných úloh
 - 3.2. Keďže Task Board nedokáže prehľadne znázorniť rozdiel medzi hlavnými úlohami a ich podúlohami, je použitá (X) notácia
4. Kliknúť na ID vybranej úlohy, ktorá začína na (X)

Story	New	Assigned
<div><p>Valentin Paulen (9 hours) 8916</p><p>[PC] Zaznamenávanie aktivity v GLOBE aplikácii do LRS</p><p>5.0</p></div>		<div><p>8923 (X)[I] Implementovať triedu komunikujúcu Valentin Paulen 9.0</p><p>8925 [I] Vytvoriť implementáciu HTTP access triedy Valentin Paulen 5.0</p><p>8926 [T] Vytvoriť unit testy Valentin Paulen 2.0</p><p>8927 [D] Zdokumentovať implementáciu a testy DB Valentin Paulen 2.0</p></div>
<div><p>Valentin Paulen (16 hours) 8917</p><p>[PC] Získavanie používateľských dát a levelov v GLOBE aplikácii zo vzdialenej databázy</p><p>5.0</p></div>		<div><p>8928 (X)[I] Vytvoriť triedy, ktoré budú reprezentovať Valentin Paulen 3.0</p><p>8929 [I] Implementovať triedy reprezentujúce Valentin Paulen 2.0</p><p>8930 [D] Zdokumentovať vytvorené objekty Valentin Paulen 1.0</p><p>8931 (X)[I] Implementovať získavanie / ukladanie dát Valentin Paulen 13.0</p><p>8932 [I] Implementovať metódy na komunikáciu so Valentin Paulen 8.0</p><p>8933 [T] Vytvoriť unit testy zamerané na testovanie ukladania a Valentin Paulen 3.0</p><p>8934 [D] Zdokumentovať zmenu v DB access triede Valentin Paulen 2.0</p></div>

5. Po kliknutí na ID zvolenej hlavnej úlohy sa zobrazia jej podrobnosti

- 5.1. V popise [A] je zapísaný cieľ úlohy - teda čo má riešiteľ dosiahnuť, aký má úloha zmysel
- 5.2. Estimated Time [B] je súčet zostávajúcich časov všetkých pridelených podúloh
- 5.3. Subtasks [C] sú pridelené podúlohy
 - 5.3.1. Pri každej podúlohe je uvedený aj jej riešiteľ [D]

Task #8931 ~~Update~~ ~~Log time~~ Watch Copy Delete

Feature #8917: [PC] Získavanie používateľských dát a levelov v Globe aplikácii zo vzdialenej databazy
(X)[I] Implementovať získavanie / ukladanie dát z/do databazy

Added by Valentin Paulen 7 days ago. Updated 7 days ago.

Status: Assigned	Start date: 01/14/2018
Priority: Normal	Due date: -
Assignee: Valentin Paulen	% Done: 0% B
Category: -	Estimated time: 13.00 hours
Target version: [A1] Radovan	Spent time: -
Remaining (hours) 13.0 A	

Description Quote

Do DB access triedy vytvorenej v ulohe 8923 pridať metody odosielaajúce data na server API a využívanie server API pre získavanie dát. Zabezpečiť transformáciu medzi datami zo servera a objektami vytvorenými v ulohe 8928

Subtasks Add

Task #8932: [I] Implmentovať metody na komunikáciu so server API do D...	Assigned	Valentin Paulen	
Task #8933: [T] Vytvorit unit testy zamerane na testovanie ukladania ...	Assigned	Valentin Paulen	
Task #8934: [D] Zdokumentovať zmeny v DB access triede	Assigned	Valentin Paulen	

D

Related issues Add

C

Detaily hlavnej úlohy

6. Pri hlavných úlohách označených **(X)** **nepoužívať Update ani Log Time!** [E]
 - 6.1. Zaznamenávanie pokroku (% Done a Estimated Time) sa realizuje cez priradené podúlohy
7. Pri hlavných úlohách je potrebné si prečítať Description [A], aby riešiteľ vedel, čo má v úlohe spraviť
8. Potom si treba prezrieť samotné podúlohy
 - 8.1. Na začiatku názvu je uvedený typ podúlohy
 - 8.1.1. **A** - Analýza
 - 8.1.2. **N** - Návrh
 - 8.1.3. **I** - Implementácia
 - 8.1.4. **T** - Testovanie
 - 8.1.5. **D** - Dokumentácia

8.1.6. R - Revízia (review)

8.1.6.1. **Revíziu** vždy vykonáva **iný člen tímu** a cieľom takejto úlohy je skontrolovať, či je riešenie správne (napr. code review, documentation review)

8.2. Ďalej je uvedený aktuálny stav podúlohy (Assigned, In progress, To review...), riešiteľ podúlohy a % dokončenia

9. Pre zistenie podrobností o danej podúlohe a DOD je potrebné kliknúť na ID podúlohy (Task #XY)

9.1. [A] ukazuje hierarchické zaradenie podúlohy do príslušnej úlohy a user story (Feature)

9.2. **Assignee** [B] pomenúva riešiteľa podúlohy

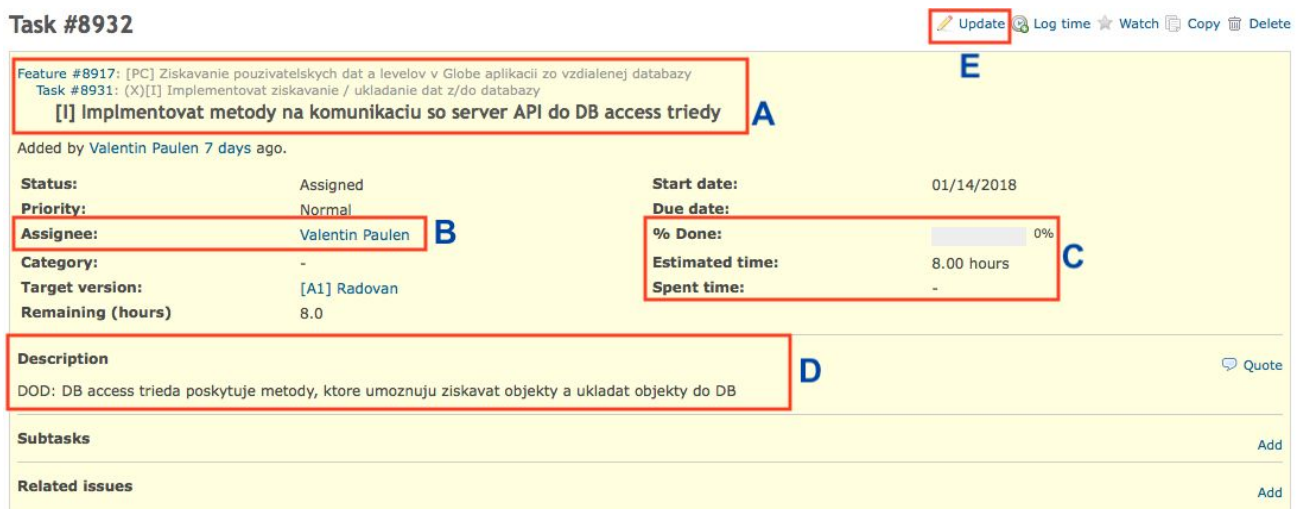
9.3. **% Done, Estimated Time a Spent Time** [C] ukazujú postup v riešení podúlohy

9.4. **Description** [D] určuje DOD - kritérium splnenia podúlohy

9.4.1. Hlavná úloha je považovaná za splnenú, ak sú DOD všetkých jej podúloh splnené

9.5. Postup v podúlohe sa zadáva cez tlačidlo **Update** [E] pri podúlohe (teda pri takej úlohe, ktorej názov sa **NEzačína na (X)**)

Task #8932



Update Log time Watch Copy Delete

Feature #8917: [PC] Získavanie používateľských dát a levelov v Globe aplikácii zo vzdialenej databázy
Task #8931: (X)[I] Implementovať získavanie / ukladanie dát z/do databázy
[I] Implementovať metódy na komunikáciu so server API do DB access triedy **A**

Added by Valentin Paulen 7 days ago.

Status:	Assigned	Start date:	01/14/2018
Priority:	Normal	Due date:	
Assignee:	Valentin Paulen B	% Done:	0% C
Category:	-	Estimated time:	8.00 hours
Target version:	[A1] Radovan	Spent time:	-
Remaining (hours)	8.0		

Description **D** Quote

DOD: DB access trieda poskytuje metódy, ktoré umožňujú získavať objekty a ukladať objekty do DB

Subtasks Add

Related issues Add

Detaily podúlohy

Spôsob 2: Zistenie úloh cez sekciu Issues

1. Otvoriť si **Redmine** a prejsť do sekcie Issues
2. Definovať filter:
 - 2.1. **Status: Open**
 - 2.2. **Tracker: is Task**
 - 2.3. **Assignee: is <<me>>**
 - 2.4. **Target version: is "aktuálny šprint"**
3. potvrdiť filtrovanie kliknutím na **Apply**

Issues

▼ Filters

<input checked="" type="checkbox"/> Status	open	▼	
<input checked="" type="checkbox"/> Tracker	is	▼	Task
<input checked="" type="checkbox"/> Assignee	is	▼	<< me >>
<input checked="" type="checkbox"/> Target version	is	▼	TP1718_Tim4_EduVirtual - [A1] Radovan

► Options

Apply

Issue Filter

4. V zozname úloh nižšie je možné vidieť všetky úlohy pridelené riešiteľovi zvolenému vo filtri (Assignee) [A]
 - 4.1. Na pravej strane je možné vidieť pokrok v jednotlivých úlohách, takže je ľahké sledovať, ktoré úlohy treba dokončiť
5. Úlohy, ktorých názov začína na (X) sú nadradené úlohy
 - 5.1. Úlohy, ktorých názov nezačína na (X) sú podúlohy priradené nadradeným úlohám
6. Kliknutím na názov úlohy (Subject) [B] sa zobrazia podrobnosti danej úlohy

Issues

Filters

- Status: open
- Tracker: is
- Assignee: is
- Target version: is

Task: TP1718_Tim4_EduVirtual - [A1] Radovan

Options

Apply Clear Save

#	Project	Tracker	Status	Priority	Subject	Assignee	Category	Target version	Start date	Due date	% Done
8942	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	[D] Zdokumentoat navrhnutu strukturu ukladania dat	Valentin Paulen		[A1] Radovan	01/14/2018		
8941	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	[N] navrhnut strukturu, v akej budu ulozene data v LRS	Valentin Paulen		[A1] Radovan	01/14/2018		
8939	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	[A] Analyzovat možnosti ukladania dat v systeme LearningLocker	Valentin Paulen		[A1] Radovan	01/14/2018		
8938	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	[A] Analyzovat data, ktore treba ulozit do LRS	Valentin Paulen		[A1] Radovan	01/14/2018		
8935	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	(X)[A] Analyzovat ukladane data a možnosti ich ulozenia do LRS	Valentin Paulen		[A1] Radovan	01/14/2018		
8934	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	[D] Zdokumentovat zmeny v DB access triede	Valentin Paulen		[A1] Radovan	01/14/2018		
8933	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	[T] Vytvorit unit testy zamerane na testovanie ukladania a ziskavania objektov z DB	Valentin Paulen		[A1] Radovan	01/14/2018		
8932	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	[I] Implementovat metody na komunikaciju so server API do DB access triedy	Valentin Paulen		[A1] Radovan	01/14/2018		
8931	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	(X)[I] Implementovat ziskavanie / ukladanie dat z/do databazy	Valentin Paulen		[A1] Radovan	01/14/2018		
8930	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	[D] Zdokumentovat vytvorene objekty	Valentin Paulen		[A1] Radovan	01/13/2018		
8929	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	[I] Implementovat triedy reprezentujuce data ulozene v databaze	Valentin Paulen		[A1] Radovan	01/13/2018		
8928	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	(X)[I] Vytvorit triedy, ktore budu reprezentovat pouzivatske data a levely	Valentin Paulen		[A1] Radovan	01/13/2018		
8927	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	[D] Zdokumentovat implementaciju a testy DB access triedy	Valentin Paulen		[A1] Radovan	01/13/2018		
8926	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	[T] Vytvorit unit testy	Valentin Paulen		[A1] Radovan	01/13/2018		
8925	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	[I] Vytvorit implementaciju HTTP access triedy	Valentin Paulen		[A1] Radovan	01/13/2018		
8923	TP1718_Tim4_EduVirtual	Task	Assigned	Normal	(X)[I] Implementovat triedu komunikujucu so vzdialenym API cez HTTP protokol	Valentin Paulen		[A1] Radovan	01/13/2018		

Issue Board

Proces: Zaznamenanie postupu vo svojich úlohách

Varovanie: Nikdy sa nezaznamenáva pokrok do úloh, ktorých názov začína na (X). Tieto úlohy sú zložené z podúloh, takže Estimated Time a Spent Time sú súčtom rovnomenných veličín zo všetkých priradených podúloh. % Done (X) úlohy je priemerom % Done priradených podúloh.

Poznámka: K vyhľadaniu podúlohy, do ktorej treba zapísať zmenu stavu, je možné využiť postup definovaný v procese Zistenie svojich úloh (uvedený vyššie v tomto dokumente).

1. Otvoriť si **Redmine** a prejsť do sekcie **Task Board**
2. Nájsť požadovanú úlohu (takú, ktorej názov **nezačína na (X)**) a kliknúť na jej ID
3. V pravej hornej časti nad žltým obdĺžnikom kliknúť na **Update**
4. Upraviť požadované hodnoty

Update Update Log time Watch Copy Delete

Change properties

Tracker * Task

Subject * [I] Vytvorit implementáciu HTTP access triedy

Description

Status * Assigned **A**

Priority * Normal

Assignee Valentin Paulen

Target version [A1] Radovan

Parent task 8923

Start date 2018-01-13

Due date

Estimated time 5.0 Hours

% Done 0% **B**

Remaining (hours) 5.0 **C**

Log time

Spent time **D** Hours

Activity **E** --- Please select ---

Comment **F**

Notes

Files

Choose Files no files selected (Maximum size: 24.5 MB)

G Submit Preview

Úprava zvolenej úlohy

- 4.1. V zozname **Status [A]** sa v prípade potreby zmení stav úlohy:
 - 4.1.1. **Assigned** - úloha má prideleného riešiteľa, ale zatiaľ sa na nej nezačalo pracovať, prípadne bola vrátená na prepracovanie
 - 4.1.1.1. Stav sa používa, ak sa na úlohe ešte nič nespravilo alebo ak bola úloha vrátená na prepracovanie, ale riešiteľ ešte úlohu nezačal prepracovávať
 - 4.1.2. **In progress** - úloha, na ktorej sa aktívne pracuje

- 4.1.2.1. Stav sa používa, keď pridelený riešiteľ začal pracovať na úlohe, ale zatiaľ nepovažuje DOD za splnené, prípadne ešte plánuje na úlohe ďalej pracovať
- 4.1.3. **To review** - riešiteľ dokončil prácu na úlohe a teraz je potrebné úlohu skontrolovať určeným členom tímu
 - 4.1.3.1. Stav sa používa, keď riešiteľ považuje DOD za splnené, ale splnenie DOD ešte nebolo potvrdené iným členom tímu
- 4.1.4. **Under Review** - člen tímu kontroluje riešenie úlohy a mieru splnenia DOD
 - 4.1.4.1. Stav sa používa, keď určený člen tímu vyhodnocuje riešenie úlohy
- 4.1.5. **Resolved** - tím považuje úlohu za dokončenú a DOD za splnené
 - 4.1.5.1. Stav sa používa, keď tím už považuje úlohu za hotovú a v šprinte na nej neplánuje ďalej pracovať
- 4.1.6. **Closed** - vedúci a Product Owner potvrdili správnosť riešenia a splnenie DOD
 - 4.1.6.1. Stav sa používa, keď už sa na úlohe nebude ďalej pracovať
- 4.2. Riešiteľ v prípade potreby zmení položku **% Done [B]** tak, aby zodpovedala jeho odhadu, koľko percent úlohy je dokončenej
- 4.3. Riešiteľ upraví položku **Remaining (hours) [C]** tak, aby zodpovedala jeho odhadu, koľko času na úlohe ešte bude pracovať
- 4.4. V časti **Log time** si riešiteľ zaznamená odpracovaný čas
 - 4.4.1. Do políčka **Spent time [D]** zapíše počet hodín, ktoré strávil prácou na úlohe od posledného zaznamenania času
 - 4.4.2. V políčku **Activity [E]** riešiteľ zvolí typ aktivity, ktorú vykonával
 - 4.4.2.1. **Design** - ak riešiteľ analyzoval alebo navrhoval možné riešenia
 - 4.4.2.2. **Development** - ak riešiteľ implementoval riešenie, pripravoval a spúšťal testy alebo opravoval chybu

4.4.2.3. **Documentation** - ak riešiteľ tvoril alebo upravoval dokumentáciu v súvislosti s úlohou

4.4.3. Do políčka **Comment [F]** zapíše, čo za odpracovaný čas riešil

5. Potvrdiť upravené vlastnosti úlohy kliknutím na tlačidlo **Submit [G]**

Proces: Plánovanie šprintu

1. Vybraný člen tímu zabezpečí zobrazenie systému Redmine na projektore alebo obrazovke tak, aby ostatní členovia tímu mohli vidieť priebeh plánovania šprintu a zápisu úloh do systému

TP1718_Tim4_EduVirtual » Master Backlog

Overview Activity Roadmap **Backlogs** Task board Releases Issues New issue Gantt Agile Calendar News Documents Wiki Files Settings

View options (2) Enable Auto-refresh Refresh Multi-line

[03] Cezar	2017-11-02	2017-11-16	61
8857 [WEB] Definovať štruktúru vzdelávacích materiálov	New	5.0	
8863 [PC/VR] Refaktoring zdedeného zdrojového kódu	New	40.0	
8864 [PC/VR] TinCan - sprevádzkovanie serverovej strany a vytvorenie modulu na komunikáciu apliké	New	13.0	
8862 [VUFORIA] Nájsť modely v primeranej cene, ktoré sa použijú (do 100 eur)	New	3.0	

Show Completed Sprints

Product Backlog	Close completed Sprints	1
8836 Namapovanie modelov pre Vuforia na stránke	New	
8803 Implementácia hrateľnosti / gamifikácie	New	
8837 Implementácia úrovni (levelov)	New	
8798 Návrh a implementácia grafických vylepšení	New	
8799 Aplikácia dostupná cez internet (Unity Player)	New	
8801 Zabezpečiť prepínanie AR/VR v rámci jednej aplikácie	New	
8802 Definovanie a vytvorenie konektivity na API LRS za účelom uchovania herných údajov (skóre, čas	New	
8831 Návrh vyberania testov učiteľom	New	
8833 Prihlasovanie do aplikácie	New	
8832 Úvodná obrazovka	New	
8834 Vymazanie nepotrebných súborov	New	
8829 [WEB] Modrá farba vo web playeri (Gamma)	New	1.0
8858 [WEB] Vytvoríť rozhranie pre úpravu vzdelávacích materiálov (CMS a admin rozhranie)	New	

Sekcia Backlogs

2. Upraví sa Product Backlog na základe inštrukcií Product Ownera
 - 2.1. Člen tímu pracujúci so systémom Redmine otvorí sekciu Backlogs
 - 2.2. Postupne sa do Product Backlogu pridávajú nové príbehy, ktoré požaduje Product Owner - pre každý pridávaný príbeh sa vykoná nasledujúci postup:
 - 2.2.1. Pod šípkou naľavo od nápisu "Product Backlog" sa zvolí možnosť "New Story"
 - 2.2.2. Zvolí sa typ príbehu:

- 2.2.2.1. Pre nové vlastnosti systému sa zvolí “Feature”, pre rozšírenie existujúcej vlastnosti systému sa zvolí “Enhancement” a pre opravu existujúcej vlastnosti bez jej rozširovania sa zvolí “Bug”
- 2.2.3. Zvolí sa názov príbehu vo formáte [ČASŤ] Názov:
 - 2.2.3.1. Časť označuje jednu zo 4 častí projektu - WEB, AR, VR a PC
 - 2.2.3.2. Ak sa príbeh týka viacerých častí, je možné ich uviesť za sebou, oddelené znakom /
 - 2.2.3.3. Ak príbeh zahŕňa všetky 4 časti projektu, použije sa označenie ALL
 - 2.2.3.4. Názov je uvedený ako funkcia, ktorú môže používateľ vykonať a má pre neho prínos / význam
 - 2.2.3.5. Názov začína podstatným menom, nie slovesom (napr. Prihlásenie, nie Prihlásiť)
 - 2.2.3.6. Nepoužívať technické termíny, napr. implementovať
 - 2.2.3.7. Príklad:
 - 2.2.3.7.1. [PC] Prihlasovanie do aplikácie
 - 2.2.3.7.2. [AR/VR] Prepínanie AR a VR v rámci jednej aplikácie
 - 2.2.3.7.3. [ALL] Ukladanie aktivity do databázy
- 2.2.4. Stav sa nastaví na “New”, odhad zložitosti zostane nevyplnený
- 2.2.5. Pridávanie príbehu sa ukončí stlačením tlačidla “Save”
- 2.3. (Voliteľné) Z Product Backlogu sa odstránia príbehy, ktoré určí Product Owner - pre každý odstraňovaný príbeh sa vykoná nasledujúci postup:
 - 2.3.1. Kliknutím na ID príbehu sa zobrazia podrobnosti
 - 2.3.2. Vpravo nad podrobnosťami príbehu sa zvolí možnosť “Delete”
- 2.4. Príbehy v Product Backlogu sa usporiadajú podľa priority zostupne - hore sú dôležitejšie príbehy ako dole. Poradie príbehov určuje Product Owner

- 2.4.1. Príbeh, ktorého prioritu je potrebné zmeniť sa presunie na požadované miesto spôsobom Drag-and-Drop (podrží sa názov príbehu)
 - 2.5. Vykoná sa ohodnotenie zložitosti jednotlivých príbehov - zvolí sa moderátor ohodnocovania a pre každý neohodnotený príbeh v Product Backlogu a pre príbehy, ktorých odhad náročnosti sa zmenil od posledného ohodnotenia sa vykoná nasledujúci postup:
 - 2.5.1. Metódou ohodnocovacieho pokru sa určí zložitosť úlohy
 - 2.5.1.1. Moderátor ohodnocovania prečíta názov príbehu a každý člen tímu môže povedať, ako si predstavuje realizáciu príbehu a prípadné komplikácie
 - 2.5.1.2. Všetci si pripravia kartu, ktorá vyjadruje ich odhad zložitosti úlohy
 - 2.5.1.3. Na pokyn moderátora ohodnocovania ukážu všetci členovia tímu svoje odhady
 - 2.5.1.4. Ak sa všetci zhodli v odhadnutej zložitosti, pokračuje sa na krok 2.5.2, inak sa pokračuje ďalším krokom
 - 2.5.1.5. Členovia, ktorí dali najnižší a najvyšší odhad povedia dôvody, ktoré ich viedli k takémuto odhadu
 - 2.5.1.6. Pokračuje sa krokom 2.5.1.2
 - 2.5.2. Člen tímu pracujúci so systémom Redmine klikne na názov ohodnocovaného príbehu a zo zoznamu napravo vyberie požadovanú zložitosť
 - 2.5.3. Ohodnocovanie príbehu sa ukončí kliknutím na tlačidlo "Save"
3. Vytvorí sa nový šprint a pridelia sa do neho príbehy
 - 3.1. Pod šípkou naľavo od nápisu "Product Backlog" sa zvolí možnosť "New Sprint"
 - 3.2. Upraví sa dátum začiatku a konca šprintu
 - 3.3. Zadá sa názov šprintu vo formáte [XY] Meno
 - 3.3.1. XY predstavuje poradové číslo šprintu (01,02,03..10,11...)
 - 3.3.2. Meno je určené podľa aktuálneho mena v kalendári

3.3.2.1. Ak toto meno nevyhovuje, vyberie sa iné meno dohodou členov tímu

3.4. Pred pridávaním príbehov do šprintu je potrebné obnoviť stránku Redmine

3.5. Do šprintu sa postupne presúvajú príbehy z Product Backlogu podľa priority - najprv sa pridávajú príbehy s vyššou prioritou

3.5.1. Pre každý príbeh v Product Backlogu sa vyhodnotí, či pridaním do šprintu nebude presiahnutá maximálna náročnosť šprintu (maximálny počet Story pointov v jednom šprinte) a či pridávaný príbeh nenadväzuje na niektorý už pridaný príbeh

3.5.2. Ak je možné príbeh pridať, pridá sa spôsobom Drag-and-Drop (chytí sa meno príbehu)

Story	New	Assigned	To review	Under review	Resolved	Feedback	Closed	Rejected	In progress		
<p>Sprint Impediments</p>	<p>Jan Jakub Spuro (8 hours) 8857 [WEB] Definovať štruktúru vzdelávacích materiálov 5.0</p>	<p>8860 Vytvoriť prázdnu šablónu na stránke Jan Jakub Spuro 2.0</p>	<p>8859 Analyzovať učebnice a navrhnúť štruktúru Valentín Paulen</p>	<p>8861 Naplniť stránku ukázkovým obsahom Jan Jakub Spuro 6.0</p>	<p>8863 [PC/VR] Refaktoring zdedeného zdrojového kódu 40.0</p>	<p>8866 Formálny refaktoring Kristian Konik 1.0</p>	<p>8870 Presun logiky preč z UI tried Kristian Konik</p> <p>8865 Metodika Dominik Pastierovic</p>	<p>8869 Odstránenie pôvodného UI Dan Pavlenko</p> <p>8868 Úprava výpočtov bodov na Zemi Dominik Pastierovic 6.0</p> <p>8867 Odstránenie game logic Dan Pavlenko 3.0</p>	<p>8864 [PC/VR] TinCan - sprevádzkovanie serverovej strany a vytvorenie modulu na komunikáciu aplikácie s TinCanom 13.0</p>	<p>8872 Vytvoriť modul v PC/VR aplikácii Valentín Paulen 10.0</p>	<p>8871 Vytvorenie serverovej časti Valentín Paulen 10.0</p>
<p>8862 [VUFORIA] Nájsť modely v primeranej cene, ktoré sa použijú (do 100 eur) 3.0</p>	<p>8873 Vytvorenie setu možných modelov Adrian Nagy 16.0</p>										

Sekcia Task Board

4. Príbehy v šprinte sa rozdelia na úlohy, pridelia sa jednotlivým členom tímu a určí sa ich DOD (Definition of Done - kritérium splnenia úlohy)
 - 4.1. Člen tímu pracujúci so systémom Redmine otvorí sekciu Task Board
 - 4.2. Každému príbehu (stĺpec naľavo) sa prideli zodpovedný člen - nasledujúci postup sa opakuje pre každý príbeh
 - 4.2.1. Klikne sa na ID príbehu
 - 4.2.2. Zvolí sa možnosť Update v pravom hornom rohu
 - 4.2.3. Položke Assignee sa nastaví člen tímu, ktorý je zodpovedný za splnenie príbehu
 - 4.2.4. nastavenie sa potvrdí stlačením tlačidla Submit
 - 4.2.5. Klikne sa na sekciu Task Board vo vrchnom paneli
 - 4.3. Každý príbeh sa rozdelí na úlohy - pre každý príbeh sa opakuje nasledujúci postup:
 - 4.3.1. Kliknutím na tlačidlo + pri príbehu sa vytvorí nová úloha
 - 4.3.2. Tím vytvorí názov pre úlohu vo formáte (X)[TYP] Meno
 - 4.3.2.1. (X) označuje, že sa jedná o nadradenú úlohu, ktorá sa skladá z viacerých podúloh
 - 4.3.2.2. Typ určuje, o akú úlohu ide:
 - 4.3.2.2.1. **A : analytická úloha**
 - 4.3.2.2.2. **N : návrhová úloha**
 - 4.3.2.2.3. **I : implementačná úloha**
 - 4.3.2.2.4. **T : testovacia úloha**
 - 4.3.2.2.5. **D : dokumentačná úloha**
 - 4.3.2.2.6. **R : revízia (review)**
 - 4.3.2.3. Meno opisuje, čo má úloha za cieľ dosiahnuť
 - 4.3.2.4. Názov začína slovesom, nie podstatným menom (napr. Implementovať, Nájsť, nie Implementovanie / Implementácia, Návrh)
 - 4.3.3. Vytvorí sa popis úlohy - čo je jej cieľ, čo má riešiť, čo nemá byť súčasťou riešenia a všetky potrebné informácie o úlohe
 - 4.3.3.1. Popis sa zapíše do textového poľa Description
 - 4.3.4. Určí sa člen tímu, ktorý bude danú úlohu riešiť

- 4.3.4.1. V časti Assignee sa zvolí člen tímu
- 4.3.5. Úloha sa presunie do stĺpca Assigned
- 4.3.6. Vytvorí sa podúloha
 - 4.3.6.1. Klikne sa na ID úlohy
 - 4.3.6.2. Pridá sa nová podúloha kliknutím na tlačidlo Add v časti Subtasks. Podúlohy by mali rozdeľovať nadradenú úlohu na časť (alebo viac častí) analýzy / návrhu / implementácie (v závislosti od typu nadradenej úlohy), testovania (ak je to vhodné), dokumentácie a revízie iným členom tímu
 - 4.3.6.2.1. Do poľa Subject sa zadá názov úlohy vo formáte [TYP] Meno (pre tvorbu názvu podúlohy platia pravidlá pre tvorbu názvu úloh - krok 4.3.2)
 - 4.3.6.2.2. Do poľa Description zapísať DOD (Definition of Done)
 - 4.3.6.2.2.1. DOD dokumentačnej podúlohy musí uvádzať, ktoré dokumentácie budú ovplyvnené (technická dokumentácia, dokumentácia inžinierskeho diela...)
 - 4.3.6.2.3. Status nastaviť na Assigned
 - 4.3.6.2.4. V poli Assignee nastaviť, ktorý člen tímu bude podúlohu riešiť
 - 4.3.6.2.5. Target Version nastaviť na aktuálny šprint (šprint, ktorého súčasťou je nadradená úloha)
 - 4.3.6.2.6. Do poľa Estimated Time zapísať odhadovaný čas, ktorý bude potrebné na vyriešenie podúlohy
 - 4.3.6.2.7. Potvrdiť vytvorenie podúlohy stlačením tlačidla Create (alebo Create and Continue, ak je potrebné vložiť ďalšiu podúlohu)
 - 4.3.6.2.8. Klikne sa na Task Board v hornom paneli
- 4.3.7. Ak je potrebné vytvoriť ďalšiu úlohu, pokračuje sa krokom 4.3.1

Proces: Uzatváranie šprintu

1. Vybraný člen tímu zabezpečí zobrazenie systému Redmine na projektore alebo obrazovke tak, aby ostatní členovia tímu mohli vidieť priebeh plánovania šprintu a zápisu úloh do systému.
2. Na základe rozhodnutia vedúceho sa jednotlivé úlohy a príbehy buď uzavrú, alebo sa vrátia na prepracovanie
 - 2.1. Člen pracujúci so systémom Redmine zvolí sekciu Issues
 - 2.2. V tejto sekcii nastaví filter
 - 2.2.1. Status: open
 - 2.2.2. Target version: názov aktuálneho šprintu
 - 2.2.3. Tracker: task
 - 2.2.4. Potvrdiť filtrovanie kliknutím na tlačidlo Apply
 - 2.3. Pre každú úlohu v zozname sa vykoná nasledujúci postup:
 - 2.3.1. Vedúci a Product Owner určia, či je úloha dokončená
 - 2.3.1.1. Ak rozhodli, že je možné úlohu uzatvoriť (považovať ju za dokončenú), člen pracujúci so systémom klikne pravým tlačidlom myši na Tracker danej úlohy, prejde na položku Status a zvolí možnosť Closed
 - 2.3.1.2. Ak rozhodli, že úloha ešte nie je dokončená a je potrebné ju prepracovať, člen pracujúci so systémom klikne pravým tlačidlom myši na Tracker danej úlohy, prejde na položku Status a zvolí možnosť Assigned
 - 2.4. Z filtra sa odstráni pravidlo Tracker a zmena sa potvrdí kliknutím na Apply
 - 2.4.1. Príbehy, ktoré majú všetky priradené úlohy uzavreté (ich stav je nastavený na Closed) sa tiež nastaví ako uzavreté
 - 2.4.1.1. Člen pracujúci so systémom klikne pravým tlačidlom myši na Tracker daného príbehu, prejde na položku Status a zvolí možnosť Closed

3. Uzavrie sa skončený šprint
 - 3.1. Člen pracujúci so systémom Redmine zobrazí sekciu Backlogs
 - 3.2. Všetky príbehy v uzatváranom šprinte, ktoré nie sú uzatvorené (nie sú prečiarknuté) sa presunú do Product Backlogu spôsobom Drag-and-Drop
 - 3.3. Ak Šprint Backlog obsahuje iba prečiarknuté príbehy, člen pracujúci so systémom klikne na šípku naľavo od názvu šprintu, zvolí položku Version
 - 3.4. Klikne na tlačidlo Edit v pravom hornom rohu
 - 3.5. Položku Status zmení na Closed
 - 3.6. Zmenu potvrdí kliknutím na Save

Proces: Export evidencie úloh zo systému Redmine

1. Člen tímu, ktorý vykonáva export evidencie úloh otvorí systém Redmine
2. Zvoliť sekciu Roadmap a urobiť screenshot celej stránky
3. Následne kliknúť na meno šprintu, zvoliť typ diagramu Work burndown a urobiť jeho screenshot
4. Kliknúť na Task Board a urobiť screenshot celej stránky
5. Exportovať zoznam úloh do PDF:
 - 5.1. Zvoliť sekciu Issues
 - 5.2. Exportovať príbehy v aktuálnom šprinte:
 - 5.2.1. Nastaviť filter:
 - 5.2.1.1. Status: open
 - 5.2.1.2. Target version: exportovaný šprint
 - 5.2.1.3. Tracker: is not task
 - 5.2.1.4. Potvrdiť filtrovanie kliknutím na Apply
 - 5.2.2. Kliknúť na tlačidlo PDF v pravej spodnej časti tabuľky
 - 5.3. Exportovať úlohy v aktuálnom šprinte:
 - 5.3.1. Nastaviť filter:
 - 5.3.1.1. Status: open

- 5.3.1.2. Target version: exportovaný šprint
 - 5.3.1.3. Tracker: is task
 - 5.3.1.4. Potvrdiť filtrovanie kliknutím na Apply
 - 5.3.2. Kliknúť na tlačidlo PDF v pravej spodnej časti tabuľky
- 5.4. Exportovať príbehy a úlohy ako celok v aktuálnom šprinte:
 - 5.4.1. Nastaviť filter:
 - 5.4.1.1. Status: open
 - 5.4.1.2. Target version: exportovaný šprint
 - 5.4.1.3. Potvrdiť filtrovanie kliknutím na Apply
 - 5.4.2. Kliknúť na tlačidlo PDF v pravej spodnej časti tabuľky
- 5.5. Exportovať Product Backlog:
 - 5.5.1. Kliknúť na tlačidlo Product Backlog v pravom stĺpci
 - 5.5.2. Kliknúť na tlačidlo PDF v pravej spodnej časti tabuľky

Metodika manažmentu organizácie artefaktov (Google Drive)

verzia 2018-01-21

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá práce so zdrojmi / artefaktmi, spôsob ich pomenovania, ukladania a kategorizovania. K ukladaniu a zdieľaniu artefaktov sa používa priečinok v službe Google Drive, ktorý je zdieľaný so všetkými členmi tímu, s interným aj externým vedúcim. Interný vedúci, externý vedúci a členovia tímu majú prístupové práva na úpravu tohto priečinku a všetkého jeho obsahu.

Súvisiace dokumenty a metodiky:

1. Metodika dokumentovania stretnutí (zápisnice).
2. Metodika manažmentu úloh (Redmine)
3. Metodika dokumentovania retrospektívy šprintu

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	2
Proces: Definovanie hierarchie adresárov	2
Proces: Verziovanie dokumentov	4
Proces: Čítanie a úprava metodík	6
Proces: Ukladanie a získavanie zdrojových súborov produktu	8

Proces: Správa dokumentov stretnutia (zápisníc)	8
Proces: Ukladanie exportovanej evidencie úloh	9
Proces: Správa dokumentov retrospektívy šprintov	10
Proces: Správa dokumentácie projektu	10

3. Znenie metodiky

Proces: Definovanie hierarchie adresárov

Tímové úložisko artefaktov je vytvorené na Google Drive účte teamleadera. Celá hierarchia sa nachádza v priečinku TP_share. K tomuto priečinku a k všetkému jeho obsahu majú prostredníctvom zdieľania plný prístup všetci členovia tímu, vedúci tímu a Product Owner. Hierarchia priečinku je nasledujúca (zvýraznené názvy sú názvy priečinkov, normálnym textom sú písané popisy jednotlivých priečinkov):

Poznámka: uvedené sú iba kľúčové nemenné adresáre

TP_share/

- ❑ **AR Vuforia/** - zdrojové súbory pre Vuforia aplikáciu
- ❑ **Dokumentacia/** - celá dokumentácia projektu
 - ❑ **Dokumentacia analytických a návrhových príbehov/** - jednotlivé dokumenty analytických a návrhových príbehov
 - ❑ **Export evidencie uloh/** - sem sa ukladajú súbory získané z exportu evidencie úloh. Samotný export popisuje *Metodika manažmentu úloh*. Export sa vykonáva po každom stretnutí a príslušné súbory sa ukladajú do podpriečinkov *stretnutie_XY*, kde XY je poradové číslo šprintu (napr. 01, 02, 10, 11), prípadne označenie špeciálneho šprintu

- Backlog-history/** - tu je uložený dokument, ktorý zaznamenáva zmeny v Product Backlogu a archivuje všetky Šprint Backlogy
- Retrospektivy sprintov/** - sem sa ukladajú dokumenty retrospektívy šprintov
- Technická dokumentácia/** - v tomto priečinku je umiestnená celá technická dokumentácia produktu
 - AR/** - doplnkové súbory technickej dokumentácie AR časti produktu
 - Globe/** - doplnkové súbory technickej dokumentácie Globe aplikácie
 - VR/** - doplnkové súbory technickej dokumentácie VR časti produktu
 - Web/** - doplnkové súbory technickej dokumentácie edukačného webu
- Zapisnice/** - zápisnice zo stretnutí
- ZS-kontrolne body/** - súbory odovzdané v kontrolných bodoch v zimnom semestri
- Edukacny web/** - zdrojové súbory pre edukačný web
- Globe aplikacia/** - zdrojové súbory pre Globe aplikáciu
 - Globe from Asset/** - súbory assetu zemegule, ktoré sa nedali vložiť na Github kvôli svojej veľkosti
- Metodiky/** - všetky metodiky, ktorými sa tím musí riadiť
- VR Street View/** - zdrojové súbory VR aplikácie
- ZS/** - archivované artefakty zo zimného semestra
 - Dokumentacie/** - dokumentácie vytvorené v zimnom semestri

Každý súbor alebo skupina súborov musí byť uložená do zodpovedajúceho adresára. Súbor musí byť pomenovaný tak, aby z jeho názvu jasne vyplývalo, čo daný súbor obsahuje.

Pokiaľ sa ukladá skupina súvisiacich súborov, vytvorí sa na určenom mieste adresár, ktorého názov jasne popisuje, čo sa v adresári nachádza. Do tohto adresára sa potom uložia predmetné súbory. Z ich názvu musí jasne vyplývať, čo daný súbor obsahuje. Vo vytvorenom adresári je podľa potreby možné vytvárať ďalšie podadresáre.

Príklad:

- Potrebujeme uložiť class diagramov dokumentujúcich hierarchiu tried v Globe aplikácii. Ide teda o doplnkové súbory technickej dokumentácie Globe aplikácie
- Miesto pre uloženie takýchto artefaktov je (vychádzajúc z tabuľky uvedenej vyššie) *TP_share/Dokumentacia/Technicka dokumentacia/ Globe/*
- Keďže ide o viac súborov, vytvoríme pre nich v *TP_share/Dokumentacia/Technicka dokumentacia/Globe/* adresár *Globe Class diagramy/*
- Do tohoto vytvoreného adresára vložíme súbory
 - Môžeme ich rozdeliť do ďalších podadresárov, napríklad Model, View, Controller
- Každý súbor vhodne pomenujeme, napríklad *Level_view-class_diagram.png*

Proces: Verziovanie dokumentov

Každý Google dokument, ktorý podlieha postupným zmenám, musí byť verziovaný pomocou nástroja **História verzii** v Google Docs.

Postup vytvorenia pomenovanej verzie je nasledovný:

1. Dokument sa upraví podľa potreby
2. Keď sú všetky úpravy vykonané a dokument je kompletný, pomenuje sa jeho aktuálna verzia:
 - 2.1. Predmetný dokument sa otvorí v editore Google Docs
 - 2.2. Zvolí sa možnosť Súbor -> História verzii -> Pomenovať aktuálnu verziu

2.3. Zadá sa názov v tvare IDENTIFIKÁTOR - popis verzie

2.3.1. IDENTIFIKÁTOR vX.Y, prípadne vX.Y.Z

2.3.1.1. X označuje číslo verzie

2.3.1.2. Y označuje úpravu verzie

2.3.1.3. Z označuje gramatickú alebo štylistickú opravu dokumentu bez zmeny jeho obsahu

2.3.1.4. Popis verzie stručne charakterizuje zmeny vykonané v dokumente od poslednej verzie

2.4. Verzia sa potvrdí kliknutím na Uložiť

Ako používať čísla verzií, úprav a opráv:

- Metodiky:
 - Zásadná zmena metodiky, postupu, pridanie nového procesu, odstránenie existujúceho, pridanie zásadnej informácie - zvýšenie čísla verzie (X)
 - Pridanie podrobností, upresnenie postupu tak, že to nezmení jeho vykonávanie, ale zabezpečí jednoznačnosť, pridanie obrázku, príkladu - zvýšenie čísla úpravy (Y)
 - Oprava gramatických alebo štylistických chýb - pridanie alebo zvýšenie čísla opravy (Z)
- Dokumentácia
 - Pridanie podkapitoly, úpravy väčšieho rozsahu - zvýšenie čísla verzie (X)
 - Pridanie podrobností a detailov, úprava malého rozsahu (jeden odstavec, maximálne jedna strana) - zvýšenie čísla úpravy (Y)
 - Oprava gramatických alebo štylistických chýb - pridanie alebo zvýšenie čísla opravy (Z)
- Dokument retrospektívy šprintu
 - Akákoľvek zmena obsahu dokumentu - zvýšenie čísla verzie (X)

- Oprava gramatických alebo štylistických chýb - pridanie alebo zvýšenie čísla opravy (Z)

Príklady:

- Metodiky (aktuálne verzia v2.0):
 - v3.0 - Pridanie procesu X
 - v3.0 - Úprava postupu exportovania evidencie úloh
 - v2.1 - Spresnenie postupu exportovania evidencie úloh
 - v2.0.1 - oprava gramatických chýb v procese X
- Dokumentácia (aktuálne verzia v2.0):
 - v3.0 - Pridanie opisu implementácie DB access layer
 - v2.1 - Spresnenie popisu metód na webovom API
 - v2.0.1 - oprava gramatických chýb v popise implementácie X
- Dokument retrospektívy šprintu (aktuálne verzia v1.0):
 - v2.0 - Pridanie nedostatku X
 - v1.0.1 - oprava gramatických chýb

Proces: Čítanie a úprava metodík

Všetky metodiky, ktorými sa projekt riadi, sú uložené v adresári **Metodiky** v TP_share.

- Pre každú oblasť je vytvorený súbor Google Docs s názvom Metodika X.
 - X formálne označuje oblasť, ktorej sa metodika týka
 - Za formálnym označením môže byť v zátvorke uvedený neformálny názov
 - Príklad: Metodika dokumentovania stretnutí (zápisnice)
- Súbory metodík sa verziuujú podľa procesu *Verziovanie dokumentov* uvedeného v tejto metodike
- Každý dokument metodiky obsahuje nasledujúce časti:
 - **Nadpis** - zhodný s názvom súboru metodiky
 - **verzia YYYY-MM-DD** - dátum poslednej zmeny metodiky

- **Dedikácia** - popisuje, komu je metodika určená a akú oblasť upravuje
- **Obsah metodiky** - obsah metodiky pre ľahšiu orientáciu
- **Znenie metodiky** - skladá sa z **procesov**, ktoré predpisujú postup danej činnosti

Postup pre čítanie metodík:

1. Otvoriť adresár Metodiky v TP_share a podľa názvu nájsť požadovaný dokument metodiky
2. Otvoriť vybraný dokument a podľa obsahu nájsť požadovaný proces
3. Prejsť na požadovaný proces
 - 3.1. V procese sa môžu vyskytovať referencie na iné procesy v aktuálnej metodike alebo v iných metodikách - tieto procesy si treba tiež preštudovať, pretože priamo súvisia s aktuálnym procesom

Postup pre zmenu metodík:

1. Člen, ktorý navrhuje zmenu metodík, vykoná zmenu do príslušného dokumentu metodiky
 - 1.1. Existujúce časti, ktoré navrhuje zmazať, označí **červenou farbou písma (ale nezmaže ich!)**
 - 1.2. Nové časti napíše zelenou farbou fontu
2. Prostredníctvom nástroja **Slack** (v kanáli **general**) informuje o tomto návrhu všetkých členov tímu
3. Prebehne diskusia o zmene a zapracujú sa prípadné úpravy návrhu
4. Ak všetci členovia zmenu schvália, teamleader, upraví dokument metodiky tak, že bude zodpovedať navrhovaným zmenám a všetok text bude čierny
5. Teamleader upraví dátum poslednej zmeny a označí aktuálnu verziu podľa procesu *Verziovanie dokumentov* definovaného v tejto metodike.

Dôležité upozornenie: o každej zmene v metodikách (okrem opravy gramatických chýb) musí teamleader informovať všetkých členov prostredníctvom nástroja Slack (v kanáli general).

Proces: Ukladanie a získavanie zdrojových súborov produktu

Zdrojové súbory produktu, ktoré sa nenachádzajú na Githube, sú ukladané na **TP_share** do príslušných adresárov:

- **TP_share/AR Vuforia/** - úložisko pre zdrojové súbory používané aplikáciou Vuforia (napr. súbory modelov)
- **TP_share/Edukacny web/** - úložisko pre zdrojové súbory používané edukačným webom
- **TP_share/Globe aplikacia/** - úložisko pre zdrojové súbory používané Globe aplikáciou (napr. textúry)
- **TP_share/VR Street View/** - úložisko pre zdrojové súbory používané VR aplikáciou (napr. 360-stupňové videá)

Ukladanie zdrojových súborov:

1. Na základe oblasti použitia súboru / súborov určiť adresár, do ktorého budú súbory uložené (AR Vuforia, Edukacny web, Globe aplikacia, VR Street View)
2. Pokiaľ sa ukladá iba jeden samostatný súbor, je možné ho uložiť priamo do adresára vybraného v kroku 1
3. Pokiaľ sa ukladá viac súvisiacich súborov, je nutné v adresári vybranom v kroku 1 vytvoriť podadresár a súbory umiestniť doňho
4. Názov súborov, ako aj adresárov musí jasne charakterizovať obsah súborov alebo obsah adresárov

Proces: Správa dokumentov stretnutia (zápisníc)

Dokumenty zo stretnutí (zápisnice) sú uložené v adresári **TP_share/Dokumentacia/Zapisnice/** . Ukladanie zápisníc prebieha nasledujúcim spôsobom:

1. Zápisnica sa vytvorí v programe MS Word
 - 1.1. Proces tvorby zápisnice definuje Metodika dokumentovania stretnutí (zápisnice) [1]

2. Výsledná zápisnica sa uloží priamo do adresára **TP_share/Dokumentacia/Zapisnice/**
 - 2.1. Nevytvára sa žiadny podadresár
3. Súbor sa pomenuje **Zapisnica X.docx**
 - 3.1. X označuje poradové číslo stretnutia
4. Dokument sa exportuje do PDF formátu
5. Tento PDF súbor sa tiež uloží priamo do adresára **TP_share/Dokumentacia/Zapisnice/**
 - 5.1. Nevytvára sa žiadny podadresár
6. PDF súbor sa pomenuje **zapisnica_stretnutie_X.pdf**
 - 6.1. X označuje poradové číslo stretnutia

Poznámka: Dokumenty zápisníc (DOCX a PDF) sa neverziujú.

Proces: Ukladanie exportovanej evidencie úloh

Po každom stretnutí tímu sa exportuje aktuálny stav evidencie úloh (export zo systému Redmine). Proces exportu evidencie úloh je popísaný v metodike [2]. Výsledkom exportu sú 3 obrázkové súbory (screenshoty) a 4 PDF súbory. Tieto artefakty sa ukladajú nasledujúcim spôsobom:

1. V adresári **TP_share/Dokumentacia/Export evidencie uloh/** sa vytvorí nový podadresár s názvom **stretnutie_XY**
 - 1.1. XY označuje poradové číslo stretnutia (napr. 01, 02, 10, 11) alebo špeciálne označenie stretnutia (napr. h1)
2. Obrázkové súbory a PDF súbory sa nakopírujú priamo do adresára **stretnutie_XY**
3. Pomenovanie obrázkových súborov nie je presne určené, ale v názve musí byť uvedený dátum vytvorenia screenshotu
4. PDF súbory sú pomenované nasledujúcim spôsobom:
 - 4.1. PDF obsahujúce iba export úloh (tasks) sa pomenuje **sZ_tasks.pdf**
 - 4.1.1. Z reprezentuje poradové číslo šprintu, do ktorého sú úlohy zaradené

- 4.2. PDF obsahujúce iba export príbehov (features) sa pomenuje **sZ_stories.pdf**
 - 4.2.1. Z reprezentuje poradové číslo šprintu, do ktorého sú príbehy zaradené
- 4.3. PDF obsahujúce export príbehov aj úloh sa pomenuje **sZ_stories-tasks.pdf**
 - 4.3.1. Z reprezentuje poradové číslo šprintu, do ktorého sú príbehy / úlohy zaradené
- 4.4. PDF obsahujúce export Product Backlogu sa pomenuje **sZ_pb.pdf**
 - 4.4.1. Z reprezentuje poradové číslo šprintu, ktorý je práve otvorený

Poznámka: Súbory exportu evidencie úloh sa neverziujú

Proces: Správa dokumentov retrospektívy šprintov

Retrospektívy šprintov sa vytvárajú v programe Google Docs. Ukladajú sa do adresára **TP_share/Dokumentacia/Retrospektivy sprintov/** .

Postup ukladania dokumentov retrospektívy šprintov je nasledovný:

1. V adresári **TP_share/Dokumentacia/Retrospektivy sprintov/** sa vytvorí nový Google Docs dokument, ktorý sa pomenuje **Sprint_XY_Review**
 - 1.1. XY označuje poradové číslo hodnoteného šprintu (napr. 01, 02, 10..)
2. Dokument sa naplní obsahom podľa metodiky [3]
3. Dokument sa označí verziou **v1.0** podľa procesu Verziovanie dokumentov definovaného v tejto metodike

Proces: Správa dokumentácie projektu

Všetka dokumentácia projektu je uložená v adresári **TP_share/Dokumentacia/**

Zoznam dokumentov a dokumentových adresárov:

Poznámka: zvýraznené dokumenty sa musia verziovať podľa procesu Verziovanie dokumentov definovaného v tejto metodike

- TP_share/Dokumentacia/**Dokumentacia inžinierskeho diela**
- TP_share/Dokumentacia/**Globalne ciele projektu**
- TP_share/Dokumentacia/**Instalacna prirucka**
- TP_share/Dokumentacia/**Pouzivatelska prirucka**
- TP_share/Dokumentacia/**Riadenie - Big Picture**
- TP_share/Dokumentacia/Dokumentacia analytickych a navrhovych pribehov/
- TP_share/Dokumentacia/Technicka dokumentacia/**Technicka dokumentacia produktu**
- TP_share/Dokumentacia/Technicka dokumentacia/AR/
- TP_share/Dokumentacia/Technicka dokumentacia/Globe/
- TP_share/Dokumentacia/Technicka dokumentacia/VR/
- TP_share/Dokumentacia/Technicka dokumentacia/Web/

Dôležité upozornenie: v adresári TP_share/Dokumentacia/ je možné vytvárať nové súbory a adresáre iba tam, kde to umožňuje výnimka.

Výnimky zákazu tvorby súborov a adresárov:

Poznámka: V nižšie uvedených adresároch v TP_share/Dokumentacia/ je možné vytvárať nové adresáre a ukladať nové súbory:

- Dokumentacia analytickych a navrhovych pribehov/
- Technicka dokumentacia/AR/
- Technicka dokumentacia/Globe/
- Technicka dokumentacia/VR/
- Technicka dokumentacia/Web/
- Export evidencie uloh/
- Retrospektivy sprintov/

- Zapisnice/
- ZS-kontrolne body/

Metodika manažmentu verzií (Github)

verzia 2018-01-21

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá práce so systémom Github, ktorý je určený na správu verzií zdrojových kódov. Metodika obsahuje všeobecné pravidlá, ktoré je nutné dodržiavať pri práci s ktorýmkoľvek repozitárom v rámci projektu. Ďalej metodika určuje pravidlá špecifické pre každý repozitár. Tieto pravidlá je nutné dodržiavať iba pri práci s príslušným repozitárom.

V prípade, že počas práce vznikne dočasný repozitár, je možné metodiku práce s ním umiestniť do jeho wiki stránky. Ak však bude repozitár aktívne používaný viac ako 20 dní, metodika jeho používania musí byť uvedená v tomto dokumente. V prípade, že metodika použitia repozitára je uvedená v tomto dokumente aj na wiki stránke daného repozitára, je nutné riadiť sa pravidlami uvedenými v tomto dokumente (tento dokument je nadriadený wiki stránke repozitára).

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	2
Proces: Definovanie repozitárov pre projekt	2
Proces: Tvorba a práca s vetvami	3
Proces: Vykonanie commit-u	5
Proces: Spájanie vetiev - Pull Request a Code Review	6

3. Znenie metodiky

Dôležité upozornenie: Pri práci s Github-om sa používa anglický jazyk. Po slovensky je možné uvádzať iba vlastné podstatné mená príbehov a úloh. Pomenovanie vetiev a commit správy musia byť písané po anglicky.

Proces: Definovanie repozitárov pre projekt

Projekt je rozdelený do 4 nezávislých častí:

- Edukačný web
- AR aplikácia
- VR aplikácia
- Globe aplikácia

Súčasťou manažmentu projektu je aj webová stránka tímu.

Toto rozdelenie projektu reflektujú aj vytvorené repozitáre. Tých je **5**:

- **eduvirtual-eduweb**
 - Zdrojové kódy edukačnej webovej stránky
- **eduvirtual-vuforia_standalone**
 - Zdrojové kódy AR aplikácie pre Android
- **eduvirtual-vr**
 - Zdrojové kódy VR aplikácie
- **eduvirtual-globe_pc**
 - Zdrojové kódy Globe aplikácie
 - Tento repozitár neobsahuje textúry pre Globe asset, pretože presahovali maximálnu veľkosť jedného súboru. Tieto textúry sú uložené na TP_share
- **eduvirtual-web**
 - Zdrojové kódy tímovej webovej stránky
 - Tento repozitár nepatrí k produktu

Poznámka: nie je možné vytvárať a používať iné repozitáre ako tie, ktoré definuje táto metodika.

K všetkým uvedeným repozitárom má prístup každý člen tímu, vedúci tímu a Product Owner. Administrátorom všetkých repozitárov je teamleader.

Proces: Tvorba a práca s vetvami

Na každý repozitár sa vzťahujú rovnaké pravidlá práce s vetvami. Repozitár musí obsahovať minimálne 2 vetvy:

- master
 - V tejto vetve je funkčná verzia aplikácie / stránky, schválená Product Ownerom, pokrytá testami, otestovaná a zdokumentovaná
 - Táto vetva je má zablokovaný priamy commit - je nutné vykonať Pull Request z devel vetvy a schváliť ho iným členom tímu
 - Verzia, na ktorej prebieha používateľské testovanie musí byť označená ako Release
- devel
 - Do tejto vetvy sa vkladajú ucelené úpravy alebo nové funkcie
 - Kód v tejto vetve musí byť vždy funkčný a správny

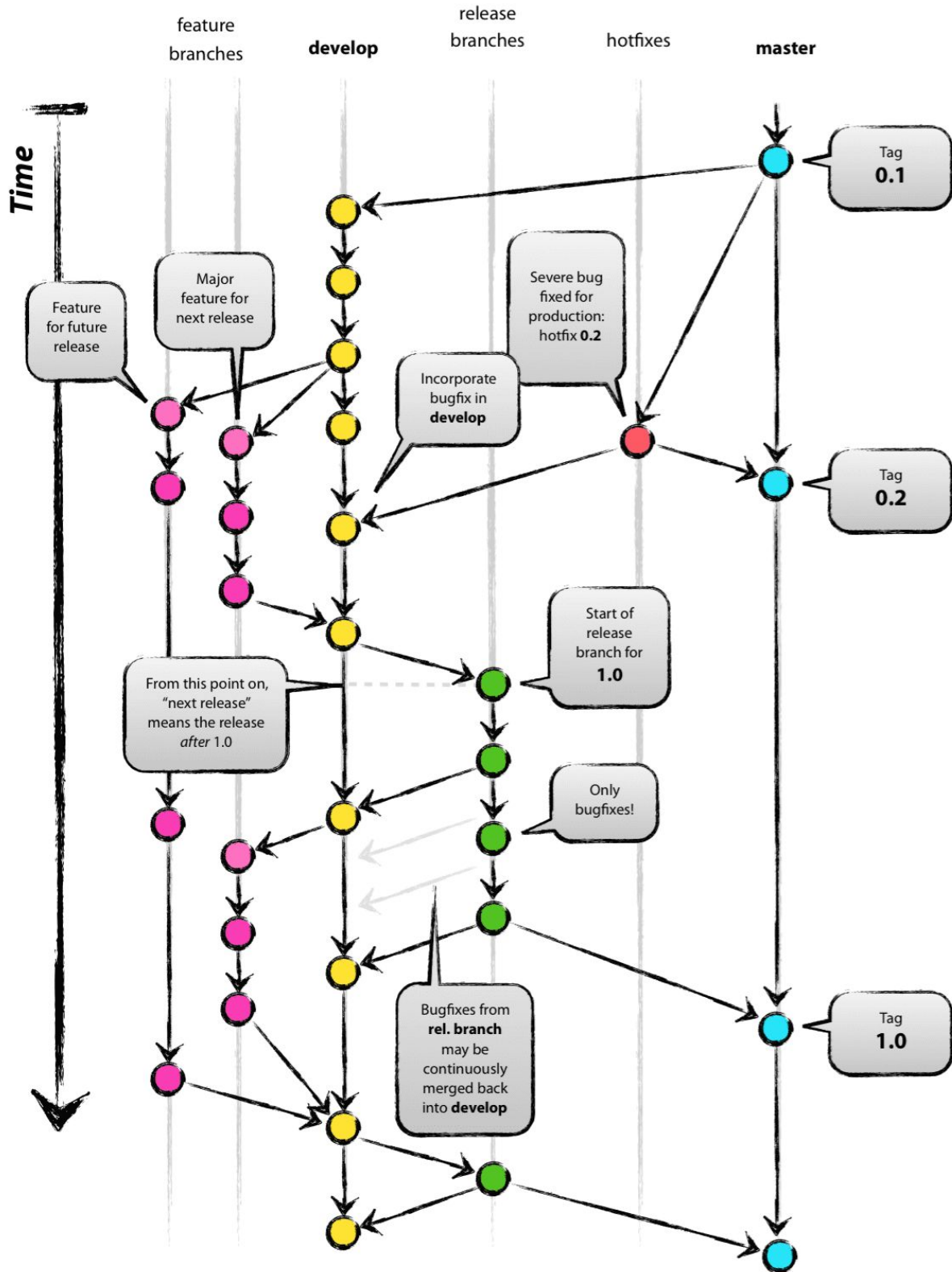
V prípade, že sa začína práca na väčšej časti, vyvíja sa nová funkcionálna alebo sa opravuje rozsiahlejšia chyba, vytvorí sa z vetvy devel nová vetva:

- Táto vetva sa pomenuje tak, aby už z názvu jasne vyplývalo, čo má daná vetva za účel riešiť
 - Názov sa píše po anglicky
 - V názve sa nepoužívajú medzery, diakritika ani špeciálne znaky
 - Názov začína malým písmenom
 - Namiesto medzier sa v názve používajú znaky _
- Pokiaľ je to potrebné alebo vhodné, vytvorí sa z nej ďalšia vetva
 - Z tejto vetvy však nie je možné vykonať Pull Request do devel vetvy

Keď je cieľ vetvy dosiahnutý, spojí sa s devel vetvou pomocou Pull Requestu.

Dôležité upozornenie: Pull Request do master vetvy sa môže vytvárať iba z devel vetvy

Používanie vetiev znázorňuje nasledujúci obrázok:



Proces: Vykonanie commit-u

Každý commit musí obsahovať commit správu. Tá má nasledujúcu štruktúru:

- Názov commitu
 - Stručný popis pridanej hodnoty commitu - čo sa zlepšilo v zdrojových kódoch od posledného commitu
 - Musí byť stručný (maximálne 50 znakov), unikátny a výstižný
- Popis commitu
 - Rozsiahlejší a podrobnejší opis pridanej hodnoty commitu
 - Popisuje podrobne, čo bolo v zdrojových kódoch zmenené od posledného commitu
 - Nie je rozsahovo obmedzený

Postup, ako vykonať commit a zapísať commit správu cez Git Bash:

1. Otvoriť Git Bash a prejsť do koreňového adresára repozitára
2. `git pull`
3. Vyriešiť konflikty, ak nejaké vznikli
4. `git add -A`
5. `git commit`
6. Otvorí sa predvolený textový editor:

Metodika komunikácie (Slack a tímový mail)

verzia 2018-01-21

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá komunikácie medzi členmi tímu navzájom, medzi členmi tímu a vedúcimi a medzi tímom a ďalšími osobami. K vnútrotímovej komunikácii (vrátane komunikácie s interným vedúcim) sa používa systém Slack. Ku komunikácii s externým vedúcim a ďalšími osobami sa používa emailová komunikácia cez tímový e-mail (outlook.com).

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	1
Proces: Komunikácia s Product Ownerom a externými osobami	1
Proces: Komunikácia medzi členmi tímu a komunikácia s vedúcim	2
Proces: Tvorba a spravovanie Slack komunikačných kanálov	3
Proces: Realizovanie standup-ov	4
Proces: Informovanie o neprítomnosti na standup-e	5

3. Znenie metodiky

Proces: Komunikácia s Product Ownerom a externými osobami

- Pre tento druh komunikácie sa používa tímový mail

- fiit_tp1718_team04@outlook.com
- Tento mail slúži na komunikáciu s product ownerom a externými osobami
- Prístup k účtu má každý člen tímu
- Nové maily píše hlavne teamleader po dohode s ostatnými členmi tímu
 - V prípade potreby môže napísať nový mail ktokoľvek s tímu, pričom o tom informuje ostatných členov tímu čo najskôr
- Predmet novej správy má formu [TP] Téma - kde Téma je výstižný názov obsahu mailu
- Podpis na konci má formu Meno Priezvisko, tím 4, kde Meno a Priezvisko je meno a priezvisko autora mailu
- Outlook automaticky zakončuje maily pätičkou:
 - Tim c. 4
 - EduVirtual
 - Timovy projekt FIIT STU

Proces: Komunikácia medzi členmi tímu a komunikácia s vedúcim

Na komunikáciu v rámci tímu sa používa výhradne Slack.

- Účet: tim4eduvirtual.slack.com

Komunikácia môže prebiehať dvoma rôznymi spôsobmi:

- Komunikácia medzi dvoma konkrétnymi členmi tímu prebieha formou priamych správ
 - Takto môže byť vedená iba komunikácia, ktorá sa netýka žiadneho ďalšieho člena tímu
- Komunikácia prostredníctvom kanálov
 - Takto prebieha komunikácia, ktorá sa týka viac ako dvoch členov
 - Každý člen, ktorého sa komunikácia týka, musí mať prístup ku kanálu v ktorom takáto komunikácia prebieha

Spravovanie kanálov popisuje proces Tvorba a spravovanie Slack komunikačných kanálov definovaný v tejto metodike.

Upozornenie: je potrebné, aby každý člen venoval zvýšenú pozornosť nastaveniu upozornení pri svojom Slack účte. Ide hlavne o funkciu Nerušiť,

ktorá môže spôsobiť, že v určitých časoch nebudú používateľovi chodiť upozornenia. Keďže predvolene je táto funkcia zapnutá, je vhodné venovať jej zvýšenú pozornosť.

Proces: Tvorba a spravovanie Slack komunikačných kanálov

Každému členovi tímu a vedúcemu musia byť dostupné minimálne tieto 3 kanály:

- **general** - kanál slúži na komunikáciu o všeobecných veciach, ktoré sa týkajú alebo môžu zaujímať všetkých členov tímu
 - V tomto kanáli sa môže diskutovať o akejkoľvek časti projektu alebo o akejkoľvek vlastnosti produktu
 - Je to preferovaný kanál pre všeobecnú komunikáciu
- **standup** - kanál, v ktorom sa pravidelne konajú tímové standup-y
 - Priebeh standup-ov a používanie tohto kanála určuje proces Realizovanie standup-ov definovaný v tejto metodike
 - Tento kanál je možné použiť iba počas standup-ov a iba predpísaným spôsobom
- **random** - kanál slúžiaci na diskusiu o veciach, ktoré sa netýkajú tímového projektu
 - V tomto kanáli nie je možné diskutovať o projekte

Tvorba ďalších kanálov:

- V prípade potreby je možné vytvoriť ďalšie kanály, ktoré budú slúžiť na komunikáciu týkajúcu sa určitej konkrétne definovanej časti projektu
 - Príklad: databáza, vuforia, globe-backend
 - K týmto kanálom musia mať prístup všetci členovia tímu
 - Tieto kanály môžu existovať neobmedzene dlho

Dôležité upozornenie: Ku každému vytvorenému kanálu musí mať prístup každý člen tímu. Nie je možné vytvoriť komunikačný kanál, do ktorého nebude pozvaný niektorý člen. Z hľadiska viditeľnosti je teda každý kanál definovaný ako **Public**.

Názov kanálu musí vystihovať, k akému účelu bol kanál vytvorený. Každý kanál musí mať zadaný **Purpose**, v ktorom bude presne a úplne popísané, o čom sa bude v danom kanáli komunikovať. Časťou Purpose môžu byť aj pravidlá používania kanálu.

Ak niektorého člena nezaujíma určitý kanál, môže si nastaviť, aby mu nechodili upozornenia.

Proces: Realizovanie standup-ov

Každý člen tímu musí v stanovených časoch ostatných členov informovať o práci, ktorú urobil od posledného standup-u, o úlohách, ktoré práve rieši, o úlohách, ktoré bude riešiť do ďalšieho standup-u a o problémoch, ktoré objavil počas práce na projekte.

Stanovené časy:

- Každý týždeň v
 - deň o čas-čas

Standup prebieha na tímovom Slacku, v kanáli **standup**. Každý člen musí v stanovenom čase ostatných (vrátane vedúceho) informovať o vyššie uvedených veciach.

Priebeh standupu:

1. Teamleader otvorí standup v určenom komunikačnom kanáli
2. Každý člen napíše jednu správu alebo viac správ priamo za sebou
 - 2.1. Informuje ostatných, čo urobil na projekte od posledného standup-u
 - 2.2. Napíše, na akých úlohách pracuje (aké úlohy má rozpracované)
 - 2.3. V skratke napíše, čo plánuje urobiť do najbližšieho standup-u
 - 2.4. Oznámi tímu, aké problémy identifikoval
3. Teamleader zhrnie a uzavrie standup

Komunikácia týkajúca sa problémov pomenovaných v standup-e už neprebieha v kanáli **standup**, ale v príslušnom kanáli (v **general**, ak problém nie je zaradený do iného kanálu). Cieľom standup-u je rýchlo informovať členov tímu o postupe šprintu,

nie o vyriešení jednotlivých problémov. Preto sa o konkrétnom riešení nediskutuje v kanáli **standup**.

Proces: Informovanie o neprítomnosti na standup-e

Vo výnimočných prípadoch, kedy sa niektorý člen nebude môcť zúčastniť standup-u v stanovenom čase, mal by o tom informovať ostatných, aby zbytočne nečakali na jeho príspevok.

Dôležité upozornenie: Tento postup je možné aplikovať iba vo výnimočných prípadoch. Pravidelné nezúčastňovanie sa standup-ov je považované za vážny problém pri spolupráci v tíme.

Postup pri informovaní o neprítomnosti na standup-e:

1. Člen, ktorý predpokladá svoju neúčast' na standup-e o tom **ešte pred začatím standup-u** napíše správu do kanálu **standup**
 - 1.1. Príklad: "Vo štvrtok 18.1. sa nezúčastním standupu"
2. Člen informuje ostatných o svojej práci a identifikovaných problémoch buď:
 - 2.1. Ak je to možné: pred začatím predmetného standup-u do komunikačného kanálu **standup**, čo najbližšie k jeho začiatku
 - 2.2. Inak: do komunikačného kanálu **standup** čo najskôr po skončení predmetného standup-u

Metodika manažmentu globálnych cieľov projektu

verzia 2018-01-12

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá pre tvorbu, aktualizovanie a dokumentovanie globálnych cieľov projektu. Jasne definované ciele projektu sú základný predpoklad pre úspešné dokončenie projektu, pre efektívne napredovanie a pre skvalitnenie a zjednodušenie komunikácie v rámci tímu aj mimo neho. Preto si takto dôležitá časť projektu vyžaduje vlastnú metodiku.

Súvisiace dokumenty a metodiky:

1. Metodika manažmentu organizácie artefaktov (Google Drive).
2. Metodika komunikácie (Slack a tímový mail).

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	2
Proces: Vytvorenie dokumentu s globálnymi cieľmi projektu	2
Proces: Nastavenie štruktúry dokumentu s globálnymi cieľmi projektu	2
Proces: Aktualizácia dokumentu s globálnymi cieľmi projektu	3

3. Znenie metodiky

Proces: Vytvorenie dokumentu s globálnymi cieľmi projektu

- Dokument je uložený v tímovom Google Drive priečinku
 - Miesto uloženia dokumentu určuje metodika [1]
- Dokument je prístupný všetkým členom tímu, vedúcemu a vlastníkovi produktu po celú dobu trvania projektu
- Pri každej zmene cieľov projektu je nutné dokument ihneď aktualizovať
 - Každú zmenu globálnych cieľov musí schváliť vlastník produktu
- Štruktúru a obsah dokumentu určuje proces Nastavenie štruktúry dokumentu s globálnymi cieľmi projektu v tejto metodike

Proces: Nastavenie štruktúry dokumentu s globálnymi cieľmi projektu

Dokument obsahuje tieto časti:

1. Nadpis Globálne ciele projektu EduVirtual
2. Číslo verzie: verzia YYYY-MM-DD
3. Vodorovná čiara oddeľujúca hlavičku a obsah
4. Charakteristika projektu
 - 4.1. Jedna veta, o čom je projekt
 - 4.2. Jeden odstavec podrobnejšie opisujúci zmysel projektu, dôvod jeho vzniku, jeho smerovanie a ciele
5. Kľúčové body projektu
 - 5.1. V odrážkach napísané hlavné ciele, ktoré sa snaží projekt dosiahnuť
 - 5.2. Maximálne 15 bodov
6. Logické členenie projektu
 - 6.1. Rozdelenie projektu na menšie, relatívne nezávislé časti (ak je to možné)
 - 6.2. Stručný popis každej časti

7. Ciele jednotlivých častí projektu
 - 7.1. Pre každú časť napísať zoznam cieľov, ktoré sa snaží daná časť dosiahnuť
8. Infografika
 - 8.1. Jednoduchá infografika popisujúca projekt - je určená pre komunikáciu s osobami priamo nezúčastnenými pre tvorbu projektu
9. Záznam zmien v globálnych cieľoch projektu
 - 9.1. Zoznam vykonaných zmien - každá zmena cieľov tu je stručne opísaná
 - 9.1.1. Dôvod zmeny
 - 9.1.2. Výpis, ktoré ciele sa zmenili

Proces: Aktualizácia dokumentu s globálnymi cieľmi projektu

1. Návrh na zmenu cieľov môže predložiť každý člen tímu
2. Návrh na zmenu prebieha nasledovným spôsobom:
 - 2.1. Člen tímu, ktorý chce vytvoriť návrh na zmenu cieľov, vytvorí novú dočasnú verziu dokumentu s globálnymi cieľmi podľa metodiky [1]
 - 2.2. Návrh na zmenu premietne do dokumentu globálnych cieľov - teda upraví dokument tak, aby zodpovedal jeho predstavám o nových cieľoch
 - 2.3. Informuje celý tím a vedúceho o vytvorení návrhu na zmenu globálnych cieľov podľa metodiky [2]
 - 2.4. TeamLeader informuje o tomto návrhu vlastníka produktu podľa metodiky [2]
 - 2.5. Prebehne diskusia medzi všetkými členmi tímu, vedúcim tímu a vlastníkom produktu
 - 2.6. Ak vlastník produktu zmenu schváli, verzia s návrhom zmeny sa stáva platnou verziou dokumentu s globálnymi cieľmi projektu
 - 2.7. Zmena platnej verzie sa vykoná podľa metodiky [1]

Metodika vývoja C# kódu

verzia 2017-11-02

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá písania programového kódu v jazyku C#. Tieto pravidlá je nutné dodržiavať vo všetkých častiach projektu.

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	1
Proces: Písanie C# kódu	1
Proces: Pomenovávanie premenných, práca s premennými	2
Proces: Funkcionálna dekompozícia	3

3. Znenie metodiky

Proces: Písanie C# kódu

- Najdôležitejšie pravidlo: pokiaľ nerobím grafické výpočty ktoré sa spúšťajú každý frame tak čitateľnosť kódu a jeho pochopiteľnosť sú dôležitejšie ako dobrý výkon. Ak robím kód náročný na výkon, môžem ho napísať horšie pochopiteľne ale treba okomentovať čo a ako kód robí!!!
 - Nesprávne:

```
public static double GetDistance(GpsCoordinates point1, GpsCoordinates point2)
{
    return GlobalConstants.EARTH_RADIUS_METERS * 2 * Math.Atan2(Math.Sqrt(Math.Sin(ConvertToRadians(point2.Latitude - point1.Latitude) / 2) *
        * Math.Sin(ConvertToRadians(point2.Latitude - point1.Latitude) / 2) +
        Math.Cos(ConvertToRadians(point1.Latitude)) * Math.Cos(ConvertToRadians(point2.Latitude)) *
        Math.Sin(ConvertToRadians(point2.Longitude - point1.Longitude) / 2) * Math.Sin(ConvertToRadians(point2.Longitude - point1.Longitude) / 2)),
        Math.Sqrt(1 - Math.Sin(ConvertToRadians(point2.Latitude - point1.Latitude) / 2) * Math.Sin(ConvertToRadians(point2.Latitude - point1.Latitude) / 2) +
        Math.Cos(ConvertToRadians(point1.Latitude)) * Math.Cos(ConvertToRadians(point2.Latitude)) *
        Math.Sin(ConvertToRadians(point2.Longitude - point1.Longitude) / 2) * Math.Sin(ConvertToRadians(point2.Longitude - point1.Longitude) / 2)));
}
```

- Správne
-
-
- Napísať summary funkcie (Visual studio po napísaní /// automaticky doplní snippet s popisom funkcie, existujúcimi argumentmi a návratovou hodnotou.

```
public static double GetDistance(GpsCoordinates point1, GpsCoordinates point2)
{
    float radius = GlobalConstants.EARTH_RADIUS_METERS;

    double latitude1Rad = ConvertToRadians(point1.Latitude);
    double latitude2Rad = ConvertToRadians(point2.Latitude);

    double deltaLatitudeRad = ConvertToRadians(point2.Latitude - point1.Latitude);
    double deltaLongitudeRad = ConvertToRadians(point2.Longitude - point1.Longitude);

    double a = Math.Sin(deltaLatitudeRad / 2) * Math.Sin(deltaLatitudeRad / 2) +
        Math.Cos(latitude1Rad) * Math.Cos(latitude2Rad) *
        Math.Sin(deltaLongitudeRad / 2) * Math.Sin(deltaLongitudeRad / 2);
    double c = 2 * Math.Atan2(Math.Sqrt(a), Math.Sqrt(1 - a));

    return radius * c;
}
```

Toto
summary
sa
následne
zobrazuje
pri

mouseoveraní funkcie/parametrov, vďaka čomu je možné pochopiť čo funkcia vyžaduje a čo spraví bez študovania jej kódu.

```
/// <summary>
/// Calculates flight distance (earth-curved trajectory ignoring hills) between two longitude and latitude defined points on Earth
/// </summary>
/// <param name="point1">Starting point</param>
/// <param name="point2">Point to where we want to calculate distance from starting point</param>
/// <returns>Distance from one point to another in kilometers</returns>
public static double GetDistance(GpsCoordinates point1, GpsCoordinates point2)
```

Proces: Pomenovávanie premenných, práca s premennými

- POUŽÍVAŤ { get; set; } namiesto java getter setter. get a set môžu obsahovať aj implementáciu ale väčšinou ju netreba. Ak ju treba (príklad):
 - Verejný getter, privátny setter, getter obsahuje úpravu údaju
 - double debt { public get { return Math.Round(field, 2); }; private set; }
 - Ak je logika príliš rozsiahla, je možné ju napísať ako funkciu na viac riadkov.
- Názvy:
 - Triedy: čo najkratšie názvy (public class UpperCamelCase{}

- Privátne premenné, lokálne premené: začínajú s malým písmenom, ostatné začiatky slov s veľkým (private int lowerCamelCase;)
- Funkcie: UpperCamelCase
- Interface: I<UpperCamelCase>
- Neinicializovať fieldy v classach, sú automaticky nastavené na 0/null/false. Pokiaľ potrebujeme inú hodnotu, ktorá nezávisí od argumentov konštruktorov, vtedy môžeme použiť inicializáciu pri deklarácii.
- Operátory: oddeľovať medzerou (1 + x) * 3 || y << 2 namiesto (1+2)*3||y<<2
- Blokové zátvorky na samostatný riadok
- Pokiaľ mám príliš dlhý riadok, rozdeliť ho na viac s odsadením jedného tabu:

```
double a = Math.Sin(deltaLatitudeRad / 2) * Math.Sin(deltaLatitudeRad / 2) +
    Math.Cos(latitude1Rad) * Math.Cos(latitude2Rad) *
    Math.Sin(deltaLongitudeRad / 2) * Math.Sin(deltaLongitudeRad / 2);
```

- Pri konvertovaní primitívnych typov (int->double, float->string) použiť namiesto double x = (double)y; class Convert -> var x = Convert.ToDouble(y). Convert je ľahšie debugovateľný pri prípadnej strate dát, ľahšie sa chápe zámer kódu.
- Používať var namiesto explicitného názvu premennej pokiaľ pravá strana výrazu jednoznačne určuje typ výsledku:
 - foreach (var leaf in Tree.Leafs) namiesto foreach (Leaf leaf in Tree.Leafs)
- Garbage collector odstráni objekt len ak na neho neexistuje referencia. Ak pridáte objekt do listu, nezabudnite ho odstrániť.

Proces: Funkcionálna dekompozícia

- C# má JIT compiler (Just in Time), čo znamená, že kompiluje kód za behu programu. Toto umožňuje optimalizáciu strojového kódu podľa hardvérovej špecifikácie systému, na ktorom je program vytvorený. Ak vytvárame veľké bloky kódu, strácame výhody JIT compileru a spôsobujeme pomalší beh programu (musí sa dlho čakať na kompiláciu). JIT kompiluje celý kód funkcie. Funkcie by mali byť čo najkratšie. Ak je vo funkcii veľa if logiky, je lepšie ju

nahradiť funkciami alebo objektami. Ak sa funkcionalita opakuje, treba ju vytiahnuť do samostatnej funkcie, pretože táto funkcia po skompilovaní už bude pripravená pri každom jej volaní, namiesto kompilácie tejto logiky na rôznych miestach ako podčasť inej funkcie.

Metodika vývoja SQL kódu

verzia 2017-11-09

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá pre tvorbu SQL kódu a pomenovávanie dátových entít.

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	1
Proces: Písanie SQL kódu	1

3. Znenie metodiky

Proces: Písanie SQL kódu

- Entity sa pomenúvajú po anglicky, názvom, ktorý čo najjasnejšie vystihuje ich zmysel alebo funkciu
 - Využíva sa PascalCase - názov sa začína veľkým písmenom, pokračuje malými, nové slovo sa napája bez medzery na to predchádzajúce, ale prvé písmeno nového slova je veľké
 - Príklad: User, Course, CourseLevel, CourseUser
- Atribúty v entitách sa píšu malými písmenami, slová sa oddeľujú podtržníkom
 - Príklad: id, user_id, bounding_box

Metodika vývoja HTML_CSS_JS kódu

verzia 2018-01-16

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá písanie HTML, CSS a JavaScript kódu. Metodika tiež stanovuje spôsob organizácie súborov a pravidlá pre tvorbu a zmenu obsahu tímovej stránky.

Súvisiace dokumenty a metodiky:

1. Metodika manažmentu organizácie artefaktov (Google Drive).

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	2
Proces: Pomenovávanie HTML, CSS a JS súborov	2
Proces: Tvorba hierarchickej štruktúry súborov, organizácia súborov	2
Proces: Písanie HTML a CSS kódu	2
Proces: Písanie JavaScript kódu	4
Proces: Publikovanie dokumentu stretnutia (zápisnice) na tímovej stránke	5

3. Znenie metodiky

Proces: Pomenovávanie HTML, CSS a JS súborov

Názvy súborov, priečinkov, tried a identifikátorov uvádzame malým písmom v angličtine a medzery nahrádzame znakom “_” prípadne “-”. Snažíme sa však udržiavať rovnakú konvenciu v názvoch súborov.

Príklad: **example_main.css** alebo **example-main.css**

Proces: Tvorba hierarchickej štruktúry súborov, organizácia súborov

- Písma v priečinku **fonts**
- CSS štýly v priečinku **css**
- JavaScripty v priečinku **js**
- Obrázky v priečinku **images**
- Videá v priečinku **videos**
- v prípade ak sa dajú vyššie spomínané súbory ďalej organizovať do podpriečinkov tak sa je vhodné ich takto organizovať, v prípade iných súborov postupujeme analogicky

Proces: Písanie HTML a CSS kódu

- Pri písaní HTML kódu usilujeme o dodržiavanie atribútov pre HTML5 a zachovanie sémantiky kódu - elementy, ktoré špecifikujú svoj obsah ako je <form>, <table>, <article>, <p>, ... by mali obsahovať len obsah na, ktorý sú určené
- Kaskádové štýly (CSS) ukladať do externých súborov mimo HTML v separátnom priečinku s názvom **css**, rovnako sa vyhýbať inline štýlom ak je to možné, definovanie pozadia (background-image) a podobne sa môžu vyskytovať

Príklad inline štýlu: **<p style="display: block; font-size: 14px;">Text</p>**

- Názvy štýlov by mali byť organizované do hierarchie ako `.parent .child .subchild { /* Some stuff... */ }` namiesto `.parent-child-subchild { /* Some stuff... */ }`, tento spôsob umožňuje jednoduchšie vykonanie prípadných zmien a šetrí miesto v HTML kóde. Príklad je možné vidieť na nasledujúcom obrázku:

NESPRÁVNE	SPRÁVNE
<pre> 1 <html> 2 <style> 3 /* Some CSS */ 4 .parent { 5 display: block; 6 width: 80%; 7 height: auto; 8 } 9 .parent-child { 10 display: inline-block; 11 width: auto; 12 height: auto; 13 } 14 .parent-child-subchild { 15 display: inline-block; 16 width: auto; 17 height: auto; 18 } 19 </style> 20 <body> 21 <div class="parent"> 22 <div class="parent-child"> 23 <div class="parent-child-subchild">Lorem ipsum...</div> 24 </div> 25 </div> 26 </body> 27 </html> </pre>	<pre> 1 /* example.css */ 2 .parent { 3 display: block; 4 width: 80%; 5 height: auto; 6 } 7 .parent .child { 8 display: inline-block; 9 width: auto; 10 height: auto; 11 } 12 .parent .child .subchild { 13 display: inline-block; 14 width: auto; 15 height: auto; 16 } 17 18 <html> 19 <link rel="stylesheet" type="text/css" href="css/example.css"> 20 <body> 21 <div class="parent"> 22 <div class="child"> 23 <div class="subchild">Lorem ipsum...</div> 24 </div> 25 </div> 26 </body> 27 </html> </pre>

- Podľa možnosti využívame CSS atribúty, ktoré sú kompatibilné s viacerými prehliadačmi a vhodné je taktiež definovanie fallback atribútov - napríklad pre písma "font: 12px Roboto-Light, Arial" ak nie je možné použiť písmo **Roboto** bude namiesto neho použité písmo **Arial**
- Veľké množstvo atribútov je možné zapísať v skrátenej tvare do jedného riadku, ak je to možné a nespôsobuje to neprehľadnosť využívame skrátenejší tvar. Príklad:

DOBRÉ	LEPŠIE
<pre> 1 .parent { 2 display: block; 3 width: 80%; 4 height: auto; 5 6 border-width: 1px; 7 border-style: solid; 8 border-color: #f9e9d3; 9 } 10 </pre>	<pre> 1 .parent { 2 display: block; 3 width: 80%; 4 height: auto; 5 6 border: 1px solid #f9e9d3; 7 } 8 9 10 </pre>

- V rámci CSS štýlu oddeľujeme súvislé bloky voľným riadkom pre lepšiu prehľadnosť
- Pri mapovaní štýlu preferujeme triedy
 CSS “.parent” -> HTML “<p class=“parent”></p>” pred identifikátormi
 CSS “#parent” -> HTML “<p id=“parent”></p>”
- Ak je to možné, minimalizujeme počet definovaných tried tak, že uvedieme názov tagu namiesto definovania triedy tak ako je možné vidieť na nasledujúcom obrázku:

DOBRÉ	LEPŠIE
<pre> 2 /* Some CSS */ 3 .parent { 4 display: block; 5 width: 80%; 6 height: auto; 7 } 8 .parent .link { 9 color: red; 10 text-decoration: none; 11 } 12 13 <html> 14 <body> 15 <div class="parent"> 16 Some link 17 </div> 18 </body> 19 </html> </pre>	<pre> 2 /* Some CSS */ 3 .parent { 4 display: block; 5 width: 80%; 6 height: auto; 7 } 8 .parent a { 9 color: red; 10 text-decoration: none; 11 } 12 13 <html> 14 <body> 15 <div class="parent"> 16 Some link 17 </div> 18 </body> 19 </html> </pre>

Proces: Písanie JavaScript kódu

- Názvy súborov, metód, objektov, premenných uvádzame v angličtine.
- Okrem inicializácie objektov / funkcií sa musia všetky kódy nachádzať v externých súboroch v separátnom priečinku s názvom **js** - podporuje to jednoduchšie vykonávanie zmien, kde stačí upraviť jeden JS skript namiesto všetkých HTML súborov, kde sa nachádza daný interný skript - podľa možnosti by sa však aj inicializačné skripty mali nachádzať v externom súbore
- Pri programovaní skriptov využívame **objekty** kde sa dá namiesto procedurálneho prístupu
- Pri názvoch premenných a funkcií sa využíva **camelCase**
- Pri definovaní názvov “tried” využívame **PascalCase**

- Názvy globálnych premenných a konštánt uvádzame VEĽKÝM_PÍSMOM a ak sa jedná o viac slov oddelíme ich znakom “_”
- Medzi operátormi nechávame medzeru: $a = b$; $a = b + 3$; $\text{if } (a > b)$

Proces: Publikovanie dokumentu stretnutia (zápisnice) na tímovej stránke

Hneď po vytvorení dokumentu stretnutia (zápisnice) je potrebné tento dokument publikovať na webovej stránke tímu. Na webovú stránku sa vkladá PDF súbor uložený v priečinku TP_share na Google Drive podľa procesu Správa dokumentov stretnutia (zápisníc) uvedeného v metodike [1]. Názov dokumentu je *zapisnica_stretnutie_X.pdf*, kde X je poradové číslo šprintu. Postup publikovania dokumentu je nasledovný:

1. Na začiatku je potrebné pripojiť sa pomocou SFTP protokolu na virtuálny stroj, na ktorom je umiestnený webový server
 - 1.1. Adresa servera je: *team04-17.studenti.fiit.stuba.sk*
 - 1.2. Protokol je SFTP (port 22)
 - 1.3. Používateľské meno a heslo poskytne teamleader
2. Ako pracovný adresár sa nastaví */var/www/html/*
3. Do adresára *documents/zapisnice/* je potrebné nakopírovať PDF súbor dokumentu pomenovaný *zapisnica_stretnutie_X.pdf*, kde X je poradové číslo šprintu
4. Na záver je potrebné upraviť súbor *index.html* v pracovnom adresári (*/var/www/html/index.html*)
 - 4.1. Je potrebné nájsť element `<h2 class="documents-table-title">Zápisnice zo stretnutí</h2>`
 - 4.2. Pod týmto elementom sa nachádza element `<div class="documents-table">` ktorý sa bude rozširovať:
 - 4.2.1. Na koniec tohto elementu je potrebné nakopírovať element uvedený v tabuľke 1

- 4.2.2. X sa nahradí za poradové číslo šprintu, Y sa nahradí za poradové číslo šprintu vyjadrené slovom - radovou číslovkou
- 4.2.3. Zmeny v súbore index.html sa uložia

Tabuľka 1

```
<div class="documents-table-entry">  
  <div class="documents-table-entry-cell icon"></div>  
  <div class="documents-table-entry-cell name">Zápisnica č.X</div>  
  <div class="documents-table-entry-cell date">07.12.2017</div>  
  <div class="documents-table-entry-cell description">Zápisnica z Y  
stretnutia</div>  
  <div class="documents-table-entry-cell download"><a  
class="documents-table-button"  
href="documents/zapisnice/zapisnica_stretnutie_X.pdf"  
target="_blank">Stiahnúť</a></div>  
</div>
```

Metodika dokumentovania analytických a návrhových príbehov

verzia 2018-01-16

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá pre dokumentovanie analytických a návrhových príbehov. Takéto príbehy je potrebné zdokumentovať minimálne v rozsahu, v akom to predpisuje táto metodika. V nej sú uvedené formálne požiadavky na dokument a obsahové požiadavky. Dokumentácia je použitá na archiváciu postupu a výsledkov analytických a návrhových príbehov. Revízia tejto dokumentácie slúži ako spôsob overenia samotnej analýzy alebo návrhu.

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	2
Proces: Dokumentovanie analytických a návrhových príbehov	2

3. Znenie metodiky

Proces: Dokumentovanie analytických a návrhových príbehov

Každý príbeh v Scrume musí mať výstup, ktorý prinesie zákazníkovi pridanú hodnotu. Na rozdiel od implementačných príbehov, tie analytické a návrhové nepridávajú do produktu žiadnu funkcionality. Analytické príbehy však vytvárajú poznatky a rozhodnutia, návrhové príbehy vytvárajú realizačné postupy. Tieto výstupy treba explicitne formulovať v príslušných dokumentoch. Dokumentácia inžinierskeho diela a technická dokumentácia sa zameriavajú hlavne na realizáciu, preto sa analytické a návrhové príbehy dokumentujú predovšetkým do samostatných dokumentov, ktoré môžu byť verifikované, validované, prezentované a použité na komunikáciu s Product Ownerom.

V priebehu alebo na záver každého analytického a návrhového príbehu je potrebné vytvoriť formálny dokument. Postup tvorby takéhoto dokumentu prebieha nasledovne:

1. Vytvorí sa nový Google dokument v adresári
TP_share/Dokumentácia/Dokumentácia analytických a návrhových príbehov/
2. Pomenuje sa ID_TYP_Názov-príbehu
 - 2.1. ID zodpovedá ID príbehu v systéme Redmine
 - 2.2. TYP môže byť:
 - 2.2.1. A ak ide o analytický príbeh
 - 2.2.2. N ak ide o návrhový príbeh
 - 2.2.3. AN ak ide o analytický a zároveň návrhový príbeh
 - 2.3. Názov-príbehu zodpovedá názvu príbehu v systéme Redmine
 - 2.4. Príklad: 8893_AN_[VR] Analýza a návrh Street View
 - 2.5. Príklad: 8879_A_[PC/VR] Analýza post-processingu
3. Do dokumentu vložíme ID a názov príbehu, meno riešiteľa a dátum poslednej zmeny dokumentu
4. V dokumente sa vytvorí kapitola **Motivácia a cieľ**

- 4.1. V nej sa popíše, prečo robíme túto analýzu / návrh a cieľ, ktorý sa snažíme dosiahnuť vykonaním danej analýzy / návrhu
5. V dokumente sa vytvorí kapitola **Hodnota splnenia príbehu**
 - 5.1. V nej v odrážkach uvedieme, čo nám splnenie príbehu prinesie
 - 5.2. Pri analytických príbehoch sú to najčastejšie poznatky o možnostiach, prípadne poznatky o aktuálnom stave
 - 5.3. Analytický príbeh musí priniesť rozhodnutie - napríklad ako budeme realizovať analyzovanú vlastnosť systému
 - 5.4. Pri návrhových príbehoch je to väčšinou plán alebo spôsob realizácie určitej časti
6. V dokumente sa vytvorí kapitola **Postup**
 - 6.1. Táto kapitola popisuje, akým spôsobom sme vykonávali analýzu
 - 6.2. Píše sa v odrážkach
 - 6.3. Pri návrhových príbehoch nemusí byť táto kapitola uvedená
7. Pri analytických príbehoch sa vloží kapitola **Poznatky**
 - 7.1. V štruktúrovanej forme (v odrážkach) popisuje všetky poznatky, ktoré sme získali pri analýze
8. Pri analytických príbehoch sa uvedie kapitola **Výstup analýzy**
 - 8.1. V nej sa uvedie rozhodnutie, akým spôsobom sa bude daná časť realizovať
9. Pri návrhových príbehoch sa uvedie kapitola **Návrh**
 - 9.1. Táto kapitola podrobne popisuje vytvorený návrh - vrátane diagramov, obrázkov, náčrtov, príkladov
10. Príklad organizovania dokumentu analytického alebo návrhového príbehu:

Analytický príbeh
ID a názov príbehu Meno riešiteľa, dátum poslednej zmeny dokumentu
Motivácia a cieľ Hodnota splnenia príbehu Postup Poznatky

Výstup analýzy

Návrhový príbeh

ID a názov príbehu
Meno riešiteľa, dátum poslednej zmeny dokumentu

Motivácia a cieľ
Hodnota splnenia príbehu
(Voliteľne) Postup
Návrh

Analytický a zároveň návrhový príbeh

ID a názov príbehu
Meno riešiteľa, dátum poslednej zmeny dokumentu

Motivácia a cieľ
Hodnota splnenia príbehu

Analýza:
 Postup
 Poznatky
 Výstup analýzy

Návrh:
 Návrh

Metodika dokumentovania retrospektívy šprintu

verzia 2018-01-15

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá dokumentovania retrospektívy šprintov. Retrospektíva prebieha po každom uzatvorení šprintu. Táto metodika predpisuje obsah dokumentu retrospektívy a aj formálnu stránku dokumentu.

Súvisiace dokumenty a metodiky:

1. Metodika manažmentu organizácie artefaktov (Google Drive).

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	2
Proces: Vytvorenie dokumentu retrospektívy končiaceho šprintu	2
Proces: Priebeh retrospektívy šprintu a jeho dokumentovanie	2
Proces: Dodatočná modifikácia dokumentu retrospektívy šprintu	4

3. Znenie metodiky

Proces: Vytvorenie dokumentu retrospektívy končiaceho šprintu

Retrospektíva šprintu sa vykonáva na stretnutiach tímu hneď po uzatvorení šprintu. Má za cieľ analyzovať priebeh šprintu a identifikovať nedostatky a úspechy v priebehu šprintu týkajúce sa tvorby produktu a riadenia projektu. Pre každý identifikovaný nedostatok sa určí postup, ktorý bude popisovať, ako nedostatok odstrániť aby sa nevyskytoval v ďalších šprintoch. Priebeh vytvorenia dokumentu retrospektívy končiaceho šprintu je definovaný v nasledujúcich bodoch:

1. Vybraný člen tímu zabezpečí zobrazenie webového prehliadača na projektore alebo obrazovke tak, aby ostatní členovia tímu mohli vidieť priebeh retrospektívy šprintu a tvorby dokumentu retrospektívy
2. Vytvorí sa dokument retrospektívy šprintu podľa procesu Správa dokumentov retrospektívy šprintov podľa metodiky [1]
3. Vo vytvorenom dokumentu sa vytvorí definovaná štruktúra častí a podčastí:
 - 3.1. Nadpis "Sprint Review"
 - 3.2. Aktuálny dátum vo formáte DD.MM.YYYY
 - 3.3. Identifikátor tímu: "Tím 4 (EduVirtual)"
 - 3.4. Identifikátor šprintu: "Sprint X", kde X je poradové číslo šprintu
 - 3.5. Kapitola "Nedostatky"
 - 3.6. Kapitola "Úspechy"
4. Začne sa retrospektíva šprintu podľa procesu "Priebeh retrospektívy šprintu a jeho dokumentovanie" definovaného v tejto metodike

Proces: Priebeh retrospektívy šprintu a jeho dokumentovanie

Pred začatím tohto procesu je potrebné vytvoriť dokument retrospektívy šprintu podľa procesu "Vytvorenie dokumentu retrospektívy končiaceho šprintu" definovaného v tejto metodike". Ak je dokument vytvorený a naplnený definovanou štruktúrou, môže sa začať samotná retrospektíva končiaceho šprintu.

- Táto retrospektíva má formu otvorenej diskusie medzi všetkými členmi tímu a vedúcimi projektu
 - Téma diskusie je, ako prebiehal šprint, čo sa podarilo a čo sa nepodarilo. Pre každý bod v časti neúspešných vlastností sa kolektívne hľadá konkrétne a realizovateľné riešenie
- Počas priebehu retrospektívy šprintu vybraný člen tímu (štandardne teamleader, ak sa kolektívne neurčí niekto iný) zobrazí na projektore alebo obrazovke upravovaný dokument retrospektívy a vyplňa ho obsahom, na ktorom sa kolektívne zhodol celý tím
 - Dokument vzniká priamo na stretnutí tak, že ho všetci vidia a môžu do neho prispieť (počas písania je napríklad zobrazený na projektore)
 - Ide o kolektívne dielo vznikajúce na základe diskusie, píše ho jeden člen, ale jeho meno sa v dokumente neuvádza (keďže dokument je kolektívny)

Nasledujúce body definujú postup retrospektívy šprintu a jeho dokumentovanie:

1. Každý člen sa postupne vyjadrí, aké nedostatky sa podľa neho vyskytli v šprinte. Pre každý identifikovaný nedostatok sa vykoná nasledujúci postup:
 - 1.1. Pre nedostatok sa tímovo určí jeho rozsah a závažnosť
 - 1.2. Identifikovaný nedostatok sa zapíše do dokumentu k novej odrážke v kapitole Nedostatky
 - 1.3. Vo forme pododrážok sa uvedie opis nedostatku, jeho závažnosť, identifikované príčiny vzniku, dôsledky a ďalšie podrobnosti, ktoré tím k nedostatku priradí
 - 1.4. Jedna z pododrážok začína na slovo "Riešenie:" a v skratke popisuje, aká akcia bude vykonaná, aby sa nedostatok nevyskytol v žiadnom ďalšom šprinte
 - 1.5. Do pododrážok k odrážke Riešenie sa napíše presný postup, ako sa vykoná táto akcia
2. Každý člen sa postupne vyjadrí, aké úspechy sa podľa neho vyskytli v šprinte. Pre každý identifikovaný úspech sa vykoná nasledujúci postup:
 - 2.1. Identifikovaný úspech sa zapíše do dokumentu k novej odrážke v kapitole Úspechy

- 2.2. Voliteľne je možné do pododrážok zapísať akcie, ktoré je potrebné vykonať, aby sa úspech zopakoval aj v ďalších šprintoch

Proces: Dodatočná modifikácia dokumentu retrospektívy šprintu

Za dodatočné zmeny v dokumentoch retrospektívy šprintu sa považujú také zmeny, ktoré sú vykonané po skončení stretnutia tímu, na ktorom predmetný dokument vznikol. Takéto zmeny v dokumentoch retrospektívy šprintu štandardne nie sú povolené, môžu sa vykonávať iba vo výnimočných prípadoch, a to so súhlasom všetkých členov tímu aj vedúceho. Nasledujúci postup popisuje, ako prebieha dodatočná modifikácia dokumentu retrospektívy šprintu:

1. Člen tímu, ktorý navrhuje zmenu v konkrétnom dokumente retrospektívy šprintu o tejto zmene informuje ostatných členov tímu a vedúceho prostredníctvom tímového komunikačného nástroja.
 - 1.1. Informuje ich, o akú zmenu ide, aký bude výsledok tejto zmeny a o dôvodoch, ktoré ho viedli k návrhu tejto zmeny
 - 1.2. Prebehne diskusia o navrhovanej zmene
2. Ak všetci členovia tímu a vedúci jednohlasne schvália zmenu, navrhovateľ zmeny vykoná úpravu predmetného dokumentu retrospektívy šprintu podľa nasledujúceho postupu:
 - 2.1. Otvorí si predmetný dokument
 - 2.2. Vykoná požadovanú a schválenú úpravu
 - 2.3. Na koniec dokumentu pridá kapitolu Vykonané dodatočné zmeny (ak neexistuje)
 - 2.4. Do tejto kapitoly pridá ako novú odrážku svoje meno a aktuálny dátum
 - 2.5. Do pododrážok napíše, čo konkrétne zmenil, dôvod zmeny

Metodika dokumentovania stretnutí (zápisnice)

verzia 2018-01-15

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá pre dokumentovanie oficiálnych stretnutí (raz týždenne). Metodika stanovuje, aké časti treba dokumentovať a aký má byť obsah každej časti. Metodika tiež určuje, akú formu má mať vytvorený dokument.

Súvisiace dokumenty a metodiky:

1. Metodika manažmentu organizácie artefaktov (Google Drive).
2. Metodika vývoja HTML_CSS_JS kódu.

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	2
Proces: Vytvorenie dokumentu stretnutia (zápisnice zo stretnutia)	2
Proces: Tvorba obsahu dokumentu stretnutia (zápisnice)	3
Proces: Nastavenie štruktúry dokumentu (zápisnice)	4
Proces: Písanie programu stretnutia	4
Proces: Písanie vyhodnotenia úloh z predchádzajúceho stretnutia	4
Proces: Písanie opisu stretnutia	6
Proces: Písanie aktuálneho stavu úloh	6

3. Znenie metodiky

Proces: Vytvorenie dokumentu stretnutia (zápisnice zo stretnutia)

- Zápisnica sa vytvorí čo najskôr po každom stretnutí tímu (ešte v deň stretnutia)
- V ten istý deň sa zápisnica uverejní na webovej stránke tímu vo formáte PDF
- Počas stretnutia sa určí člen tímu zodpovedný za vytvorenie zápisnice a člen zodpovedný za jej revíziu a umiestnenie na webovej stránke tímu
 - Štandardne je za tvorbu zápisnice zodpovedný teamleader, ak sa na stretnutí tímovo neučí iný člen
- Dokument stretnutia sa vytvára v programe MS Word

Vytvorenie dokumentu stretnutia (zápisnice) prebieha podľa nasledujúceho postupu:

1. V deň stretnutia po skončení stretnutia určený člen tímu vytvorí dokument zápisnice podľa procesu *Správa dokumentov stretnutia (zápisníc)* v metodike [1]
2. Do dokumentu sa zapíše určená štruktúra (šablóna) podľa procesu *Nastavenie štruktúry dokumentu (zápisnice)* definovaného v tejto metodike
3. Pripravená štruktúra dokumentu sa naplní obsahom podľa procesu *Tvorba obsahu dokumentu stretnutia (zápisnice)* definovaného v tejto metodike
4. Dokument sa vyexportuje do PDF podľa procesu *Správa dokumentov stretnutia (zápisníc)* v metodike [1]
5. Dokument sa umiestni na webovú stránku tímu podľa procesu *Publikovanie dokumentu stretnutia (zápisnice) na tímovej stránke* podľa metodiky [2]

Proces: Tvorba obsahu dokumentu stretnutia (zápisnice)

Na tento proces sa odvoláva proces *Vytvorenie dokumentu stretnutia (zápisnice zo stretnutia)*, preto je nutné pred začatím vykonávania tohto procesu vykonať kroky 1 a 2 procesu *Vytvorenie dokumentu stretnutia (zápisnice zo stretnutia)*.

Napíňanie vytvorenej štruktúry dokumentu obsahom je realizované nasledujúcimi krokmi:

1. Do kapitoly *Téma stretnutia* sa uvedie hlavná téma stretnutia
2. Do kapitoly *Dátum stretnutia* sa uvedie dátum konania stretnutia vo formáte DD. MM. YYYY
3. Do kapitoly sa uvedie miestnosť, v ktorej prebiehalo stretnutie (zvyčajne FEI STU B402)
4. Do kapitoly *Prítomní* sa uvedú mená prítomných členov tímu a vedúceho, ak je prítomný
5. Do kapitoly *Neprítomní* sa uvedú mená neprítomných členov tímu a vedúceho, ak je neprítomný
6. Do kapitoly *Zapisovateľ* sa uvedie meno člena tímu, ktorý vytvára obsah dokumentu stretnutia (zápisnice)
7. Do kapitoly *Program stretnutia* sa napíše program stretnutia podľa procesu *Písanie programu stretnutia* uvedeného v tejto metodike
8. Do kapitoly *Vyhodnotenie úloh z predchádzajúceho stretnutia* sa napíše vyhodnotenie úloh podľa procesu *Písanie vyhodnotenia úloh z predchádzajúceho stretnutia* uvedeného v tejto metodike
9. Do kapitoly *Opis stretnutia* sa napíše opis priebehu stretnutia podľa procesu *Písanie opisu stretnutia* uvedeného v tejto metodike
10. Do kapitoly *Aktuálny stav úloh v systéme manažmentu projektu* sa napíše aktuálny stav úloh podľa procesu *Písanie aktuálneho stavu úloh* uvedeného v tejto metodike

Proces: Nastavenie štruktúry dokumentu (zápisnice)

V dokumente sa vytvoria nasledujúce časti (nadpis + kapitoly):

- Nadpis: Zápisnica stretnutia tímu EduVirtual (tím číslo 4)
- Téma stretnutia
- Dátum stretnutia
- Miesto stretnutia (zvyčajne FEI STU B402)
- Prítomní
- Neprítomní
- Zapisovateľ
- Program stretnutia
- Vyhodnotenie úloh z predchádzajúceho stretnutia
- Opis stretnutia
- Aktuálny stav úloh v systéme manažmentu projektu

Proces: Písanie programu stretnutia

- V odrážkach napísané témy, okruhy alebo konkrétne otázky, ktoré sa na stretnutí riešili
- Každý bod musí byť pochopiteľný
 - Ak nie je účel témy jasný z názvu, je potrebné ho niekoľkými vetami upresniť
- Voliteľne je možné uviesť aj meno člena, ktorý navrhol daný bod zaradiť do programu

Proces: Písanie vyhodnotenia úloh z predchádzajúceho stretnutia

- Vyhodnotenie úloh z predchádzajúceho stretnutia - Backlog ukončeného šprintu (ak počas stretnutia šprint nekončí, uvedie sa namiesto nižšie popísanej tabuľky informácia "Počas tohto stretnutia šprint ešte prebiehal,

úlohy teda neboli uzavreté. Aktuálny stav úloh a priebežné výsledky sme prezentovali vedúcim a zapísali sme pripomienky.”)

- ide o tabuľku obsahujúcu stories alebo úlohy daného šprintu vo forme:
 - ID - identifikátor story alebo úlohy v nástroji na manažovanie projektu
 - Názov - meno story alebo úlohy
 - Členovia - na prvom mieste je uvedené meno člena, ktorý je za danú story alebo úlohu zodpovedný, za čiarkou môžu voliteľne pokračovať mená členov, ktorí sa na riešení story / úlohy podieľajú
 - Táto bunka tabuľky musí obsahovať minimálne jedno meno: zodpovedného člena
 - V prípade, že na riešení story / úlohy sa podieľajú všetci, členovia, uvedie sa na prvom mieste meno zodpovedného člena a za čiarkou sa uvedie slovo “všetci”
 - Dátum zadania - dátum, kedy bola story / úloha vytvorená
 - Očakávaný dátum ukončenia - dátum, kedy je naplánované dokončenie danej story / úlohy
 - Stav - zhoduje sa so stavom, ktorý je danej úlohe pridelený v nástroji na manažment projektu
 - V zátvorke za názvom stavu sú uvedené percentá vyjadrujúce časť úlohy, ktorá je dokončená
 - V prípade, že je úloha uzavretá a Product Owner potvrdil splnenie Definition of Done, je táto skutočnosť uvedená v zátvorke (DOD splnené)
 - V prípade, že je úloha uzavretá a Product Owner potvrdil jej zrušenie, je v zátvorke uvedené (Zrušená)
 - Za zátvorkou môže byť uvedená krátka poznámka
 - Ak je potrebné upresniť stav, je možné využiť poznámku pod čiarou a do nej umiestniť podrobné vysvetlenie aktuálneho stavu vrátane faktorov, ktoré tento stav zapríčinili

Proces: Písanie opisu stretnutia

- Opis stretnutia - každý bod z časti Program stretnutia sa podrobne rozpíše
 - Uvádza sa sem, čo zapríčinilo potrebu riešiť tento bod, aký je očakávaný cieľ diskusie, samotný priebeh diskusie so všetkými podrobnosťami a jasne zhrnutý záver

Proces: Písanie aktuálneho stavu úloh

- Aktuálny stav úloh v systéme manažmentu projektu
 - Tu sa uvádza Sprint Backlog práve prebiehajúceho alebo novovytvoreného šprintu
 - Štruktúra tabuľky sa zhoduje s tabuľkou v časti Vyhodnotenie úloh z predchádzajúceho stretnutia, ale bol pridaný jeden stĺpec:
 - Story Points / Odhad. čas - počet story pointov predelený story / úlohe a odhadovaný čas na splnenie úlohy
 - v prípade prebiehajúceho šprintu sa ako odhadovaný čas uvádza, koľko času ešte zostáva
 - V prípade, že sa na stretnutí zmení Product Backlog, uvedie sa v tejto kapitole aj tabuľka s kompletným Product Backlogom, jej štruktúra je nasledovná:
 - ID - identifikátor story alebo úlohy v nástroji na manažovanie projektu
 - Názov - meno story alebo úlohy
 - Zodpovedný člen - člen, ktorému bola story alebo úloha priradená
 - bunka môže byť prázdna (úlohy v product backlogu nemusia byť pridelené)
 - Typ - typ story alebo úlohy uvedený v nástroji na manažovanie projektu

Metodika dokumentovania sumarizácie šprintov

verzia 2018-01-15

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá dokumentovania sumarizácie šprintov. Sumarizácia šprintu je opis tvorby šprintu, priebehu šprintu, uzavretia šprintu a zhodnotenie výsledkov vytvorených v priebehu šprintu. Tento dokument predpisuje spôsob dokumentovania tejto sumarizácie po formálnej a obsahovej stránke.

Súvisiace dokumenty a metodiky:

1. Metodika manažmentu organizácie artefaktov (Google Drive).

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	2
Proces: Vytvorenie záznamu v dokumente sumarizácie šprintu	2
Proces: Nastavenie štruktúry príspevku k sumarizácii šprintu	3

3. Znenie metodiky

Proces: Vytvorenie záznamu v dokumente sumarizácie šprintu

Sumarizácia šprintov sa nachádza v kapitole *Sumarizácia šprintov* v dokumente *Riadenie - Big Picture* umiestnenom v tímovom Google Drive priečinku podľa procesu *Správa dokumentácie projektu* definovanom v metodike [1]. Príspevok do sumarizácie šprintov sa tvorí:

- **V deň vytvorenia šprintu** - dokumentuje sa plánovanie šprintu, vybrané príbehy do šprintu, nevybrané príbehy do šprintu aj s uvedením dôvodu ich nevybratia
- **V deň ukončenia šprintu** - dokumentuje sa priebeh šprintu, ukončenie šprintu a zhrnutie výsledkov jednotlivých príbehov

Sumarizácia šprintu má za cieľ popísať plánovanie šprintu, vybrané príbehy do šprintu, nevybrané príbehy do šprintu aj s uvedením dôvodu ich nevybratia, priebeh šprintu, ukončenie šprintu a zhrnutie výsledkov jednotlivých príbehov. Sumarizácia šprintu obsahuje stav Product Backlogu pred naplánovaním šprintu a Šprint Backlog.

Vytvorenie sumarizácie šprintu je definované nasledujúcim postupom:

1. Otvorí sa dokument *Riadenie - Big Picture* z tímového priečinku na Google Drive podľa metodiky [1]
2. Pokiaľ ide o príspevok **v deň vytvorenia šprintu**, vykonajú sa nasledujúce aktivity:
 - 2.1. Definuje sa štruktúra sumarizácie daného šprintu podľa procesu *Nastavenie štruktúry príspevku k sumarizácii šprintu* definovaného v tejto metodike
 - 2.2. Vloží sa obsah tabuľky zachytávajúcej Product Backlog v stave, v akom sa nachádzal pred naplánovaním predmetného šprintu
 - 2.3. Vloží sa dátum začiatku šprintu
 - 2.4. Vloží sa tabuľka zachytávajúca Šprint Backlog predmetného šprintu
 - 2.5. Napíše sa zoznam a popis príbehov, ktoré boli vybrané do šprintu

- 2.6. Uvedie sa dôvod, prečo neboli do šprintu zaradené ostatné príbehy z Product Backlogu
3. Pokiaľ ide o príspevok **v deň ukončenia šprintu**, vykonajú sa nasledujúce aktivity:
 - 3.1. Napíše sa na priebeh šprintu. Tu sa uvedú všetky identifikované problémy, ich riešenie, stručné zhrnutie aktivít, ktoré boli vykonané pri plnení jednotlivých príbehov a všetky dôležité poznatky, ktoré boli počas šprintu identifikované
 - 3.2. Zhodnotenie šprintu, zoznam úspešne zvládnutých príbehov spolu s popisom, akú pridanú hodnotu poskytli zákazníkovi, zoznam nedokončených príbehov s uvedením dôvodov, ktoré viedli k nedokončeniu príbehov
 - 3.3. Vloží sa dátum konca šprintu

Proces: Nastavenie štruktúry príspevku k sumarizácii šprintu

- Pri vkladaní sumarizácie nového šprintu sa na začiatku vytvorí prázdna štruktúra (šablóna) ktorá sa neskôr vyplní obsahom. Táto štruktúra sa tvorí uvedeným postupom:
 - Vloží sa identifikátor šprintu "Šprint X:", kde X je poradové číslo šprintu
 - Vloží sa prázdna tabuľka, ktorá bude zachytávať Product Backlog pred vytvorením backlogu nového šprintu
 - tabuľka má 2 stĺpce: ID, Meno príbehu
 - Vloží sa miesto pre dátum začiatku šprintu
 - Vloží sa prázdna tabuľka, ktorá bude zachytávať Šprint Backlog novovytvoreného šprintu
 - tabuľka má 2 stĺpce: ID, Meno príbehu
 - Vloží sa prázdne miesto pre popis vybraných príbehov do šprintu a priebeh šprintu
 - Vloží sa miesto pre dátum konca šprintu

Metodika tvorby inštalačnej príručky

verzia 2018-01-16

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá pre tvorbu inštalačnej príručky k vytvorenému produktu. Metodika predpisuje formálnu stránku inštalačnej príručky, obsahovú stránku vrátane rozsahu jednotlivých častí a úrovne detailov.

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	2
Proces: Stanovenie cieľov inštalačnej príručky a definovanie štruktúry dokumentu inštalačnej príručky	2
Proces: Písanie inštalačného / konfiguračného postupu	3
Proces: Písanie postupu riešenia problémov	5

3. Znenie metodiky

Proces: Stanovenie cieľov inštalačnej príručky a definovanie štruktúry dokumentu inštalačnej príručky

Inštalačná príručka je dokument popisuje postup inštalácie / nasadenia vytvoreného produktu do produkčného prostredia. Každá časť dokumentu musí dodržať vymedzené ciele tohto dokumentu:

- **Úplnosť** - inštalačná príručka musí obsahovať **všetky** kroky, ktoré je nutné vykonať počas inštalácie produktu. Predpokladá sa, že osoba, ktorá bude inštaláciu vykonávať, nemá žiadne znalosti o fungovaní produktu a nikdy predtým s produktom nepracovala.
- **Jednoznačnosť** - inštalačná príručka je písaná vo forme číslovaných krokov, ktoré je treba vykonať pre nainštalovanie systému. Musí byť jednoznačné, v akom poradí budú kroky vykonávané, aké kroky je potreba vykonať pred aktuálnym krokom a aký má byť výstup z jednotlivých krokov
- **Pochopiteľnosť** - kroky musia byť formulované tak, aby nebol problém pochopiť, aká konkrétna akcia sa má v každom kroku vykonať. Je nevyhnutné písať kroky stručne, jednoducho, bez zbytočných komentárov. V jednom kroku sa očakáva vykonanie práve jednej akcie.
- **Používanie príkladov a ukážok** - jednotlivé kroky je potrebné obohatiť o ukážku, príklad, konkrétny príkaz alebo obrázok vždy keď to bude možné.
- **Uvedenie výstupu** - pri každom kroku, ktorého vykonanie generuje výstup, musí byť uvedený vzorový výstup. Okrem úspešného výstupu je nutné uviesť aj možné chybové výstupy a ich riešenie.

Inštalačná príručka je rozdelená do **2** hlavných častí:

- **Postup inštalácie systému EduVirtual** - v tejto časti je uvedený postup inštalácie **celej** vzdelávacej platformy od začiatku do konca v číslovaných odrážkach. V tejto časti je možné sa odvolávať na inštalačné postupy uvedené v druhej časti dokumentu.

- **Postup inštalácie jednotlivých častí systému EduVirtual** - v tejto kapitole sa budú nachádzať postupy inštalácií jednotlivých častí systému. Každá časť je považovaná za komponent systému a prístup k jeho inštalácii sa môže líšiť od prístupu k inštalácii iných komponentov. Za časť systému EduVirtual môže byť považovaná napríklad MySQL (vrátane potrebnej schémy), LRS (vrátane úplnej konfigurácie), web server, aplikácia pre Android...

Druhá časť dokumentu je zložená z podčastí, kde každá podčasť je označená ako:

- **Inštalácia X** - zoznam číslovaných krokov, ktoré vedú k nainštalovaniu časti systému, ktorá sa nazýva X. *Príklad: Inštalácia MySQL databázy.*
- **Konfigurácia X** - zoznam číslovaných krokov, ktoré vedú k úplnému nakonfigurovaniu časti systému, ktorá sa nazýva X. *Príklad: Konfigurácia MySQL databázy a vytvorenie schémy.* Konfigurácia môže byť súčasťou postupu inštalácie. Odporúčanie:
 - Ak je konfigurácia jednoduchá alebo priamočiara, je vhodné ju zahrnúť do postupu inštalácie (podčasť popísaná vyššie)
 - Ak je konfigurácia zložitá, môže pozostávať z viacerých ciest alebo sa pri konfigurácii často vyskytujú komplikácie, je vhodné ju definovať ako samostatnú podkapitolu, ktorá bude nadväzovať na podkapitolu inštalácie.
- **Riešenie X** - zoznam číslovaných krokov, ktoré vedú k vyriešeniu problému, ktorý sa nazýva X. *Príklad: Riešenie Apache: could not bind to port.*

Proces: Písanie inštalačného / konfiguračného postupu

Tento proces predpisuje, ak sa píše postup inštalácie (celého systému EduVirtual aj jednotlivých častí) a konfigurácie.

1. Zvolí sa názov pre časť, ktorej inštaláciu alebo konfiguráciu treba popísať:
 - 1.1. **Postup inštalácie systému EduVirtual** - prvá kapitola dokumentu, popisuje inštalácie celého systému EduVirtual (všetkých častí)

- 1.2. **Inštalácia X** - zoznam číslovaných krokov, ktoré vedú k nainštalovaniu časti systému, ktorá sa nazýva X. *Príklad: Inštalácia MySQL databázy.*
- 1.3. **Konfigurácia X** - zoznam číslovaných krokov, ktoré vedú k úplnému nakonfigurovaniu časti systému, ktorá sa nazýva X. *Príklad: Konfigurácia MySQL databázy a vytvorenie schémy.*
2. Použiť číslovaný zoznam typu čísla oddelené bodkou (1. / 1.2. / 3.6.1.)
 - 2.1. Pre jednoznačné kroky sa používa iba prvá úroveň číslovania
 - 2.2. Pre vetvenie je možné použiť druhú úroveň číslovania pre podmienku vetvenia a tretiu úroveň pre jednotlivé kroky v danej vetve

1. Krok 1
2. Krok 2
3. Krok 3 sa môže vykonať 2 spôsobmi
 - 3.1. Ak platí podmienka P
 - 3.1.1. Krok A
 - 3.1.2. Krok B
 - 3.2. Ak platí podmienka Q
 - 3.2.1. Krok C
 - 3.2.2. Krok D
 - 3.2.3. Krok E
4. Krok 4
5. Krok 5

- 2.3. Za číslom na prvej úrovni nasleduje úplná inštrukcia pre vykonanie daného kroku
- 2.4. Doplnkové informácie, príklad, obrázok, príkaz, SQL sa uvádza pod hlavnú inštrukciu na rovnakej úrovni odsadenia od ľavého okraja

1. Krok 1
2. Krok 2
 - echo "example" > file.txt**
3. Krok 3
 - 3.1. Podmienka P

3.1.1. Krok A
source .bashrc
3.1.2. Krok B
...

3. Príkazy určené pri príkazový riadok operačného systému sa uvádzajú fontom Roboto Mono
4. Názvy priečinkov a súborov sa uvádzajú kurzívou
5. SQL príkazy sa uvádzajú fontom Roboto Mono
6. Výstupy sa uvádzajú tučným písmom
 - 6.1. Výstupy z príkazového riadku sa uvádzajú tučným písmom fontu Roboto Mono

Proces: Písanie postupu riešenia problémov

Tento proces predpisuje, ak sa píše postup riešenia problémov a situácií, ktoré vzniknú počas inštalácie a konfigurácie systému a jeho častí, ale riešenie týchto problémov nie je priamo uvedené v inštaláčnom / konfiguračnom návode

1. Zvolí sa názov problému / situácie, ktorého / ktorej riešenie treba popísať:
 - 1.1. **Riešenie X** - zoznam číslovaných krokov, ktoré vedú k vyriešeniu problému, ktorý sa nazýva X. *Príklad: Riešenie Apache: could not bind to port.*
 - 1.2. X je písané formátom Y: Z, kde Y je názov (alebo vhodný identifikátor) časti, v ktorej problém nastal a Z je prejav (prípadne popis, ak sa prejav nedá vhodne formulovať) chyby
2. Použiť číslovaný zoznam typu čísla oddelené bodkou (1. / 1.2. / 3.6.1.)
 - 2.1. Pre jednoznačné kroky sa používa iba prvá úroveň číslovania
 - 2.2. Pre vetvenie je možné použiť druhú úroveň číslovania pre podmienku vetvenia a tretiu úroveň pre jednotlivé kroky v danej vetve

6. Krok 1
7. Krok 2
8. Krok 3 sa môže vykonať 2 spôsobmi
 - 8.1. Ak platí podmienka P
 - 8.1.1. Krok A
 - 8.1.2. Krok B
 - 8.2. Ak platí podmienka Q
 - 8.2.1. Krok C
 - 8.2.2. Krok D
 - 8.2.3. Krok E
9. Krok 4
10. Krok 5

- 2.3. Za číslom na prvej úrovni nasleduje úplná inštrukcia pre vykonanie daného kroku
- 2.4. Doplnkové informácie, príklad, obrázok, príkaz, SQL sa uvádza pod hlavnú inštrukciu na rovnakej úrovni odsadenia od ľavého okraja

3. Krok 1
4. Krok 2
 - echo "example" > file.txt**
4. Krok 3
 - 4.1. Podmienka P
 - 4.1.1. Krok A
 - source .bashrc**
 - 4.1.2. Krok B
 - ...

3. Príkazy určené pri príkazový riadok operačného systému sa uvádzajú fontom Roboto Mono
4. Názvy priečinkov a súborov sa uvádzajú kurzívou
5. SQL príkazy sa uvádzajú fontom Roboto Mono
6. Výstupy sa uvádzajú tučným písmom

- 6.1. Výstupy z príkazového riadku sa uvádzajú tučným písmom fontu Roboto Mono

Metodika tvorby používateľskej príručky

verzia 2018-01-16

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá pre tvorbu používateľskej príručky k vytvorenému produktu. Metodika predpisuje formálnu stránku používateľskej príručky, obsahovú stránku vrátane rozsahu jednotlivých častí a úrovne detailov.

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	1
Proces: Definovanie štruktúry používateľskej príručky	1
Proces: Vytvorenie návodu na použitie určitej vlastnosti systému EduVirtual	2

3. Znenie metodiky

Proces: Definovanie štruktúry používateľskej príručky

Používateľská príručka je rozdelená do 4 kapitol:

- Edukačný web

- Rozšírená realita (AR)
- Virtuálna realita (VR)
- Hra Globe

Každá kapitola je rozdelená do podkapitol, pričom každá podkapitola opisuje používanie jednej vlastnosti systému. Opis vlastnosti systému pozostáva z:

- Nadpisu, ktorý jednou vetou objasňuje, čo daná podkapitola opisuje
- Informácia o tom, čo je možné použitím danej vlastnosti dosiahnuť
- Samotný postup použitia opisovanej vlastnosti

Proces: Vytvorenie návodu na použitie určitej vlastnosti systému

EduVirual

Pri tvorbe príspevku do používateľskej príručky sa postupuje podľa týchto krokov:

1. Ohraničí sa vlastnosť, ktorej použitie bude príspevok opisovať
 - 1.1. Vždy musí ísť o ucelenú časť - teda nie je vhodné opisovať viacero samostatných častí v jednej kapitole ani nie je vhodné opisovať použitie jednej časti vo viacerých kapitolách
 - 1.2. **Príklad dobrej kapitoly:** Nájdenie a spustenie vytvoreného levelu v Globe hre
 - 1.3. **Príklad dobrej kapitoly:** Zobrazenie vybraného objektu v rozšírenej realite
 - 1.4. **Príklad zlej kapitoly:** Používanie aplikácie pre rozšírenú realitu (príliš veľká časť pre jednu kapitolu, je potrebné ju rozdeliť na viac častí použitia),
 - 1.5. **Príklad zlej kapitoly:** Odpovedanie na zobrazenú otázku v Globe hre (príliš malá časť, skôr je vhodné sa zamerať na celý priebeh hry, nie len na voľbu odpovede)
2. Určí sa nadradená kapitola, do ktorej bude príspevok zaradený, zoznam nadradených kapitol:
 - 2.1. Edukačný web

- 2.2. Rozšírená realita (AR)
- 2.3. Virtuálna realita (VR)
- 2.4. Hra Globe
3. Vo vybranej nadradenej kapitole sa vytvorí nová podkapitola, ktorej názov jednou vetou objasňuje, čo daná podkapitola opisuje
4. Napíše sa jeden odstavec opisujúci, čo je možné dosiahnuť využitím práve opisovanej vlastnosti
 - 4.1. Používajú sa súvislé vety, ale kvôli prehľadnosti je vhodné na vymenovávanie použiť odrážky
 - 4.2. Táto časť môže byť obohatená obrázkom, schémou alebo príkladom
5. Samotný návod na použitie vybranej vlastnosti
 - 5.1. Tu sa používa hlavne číslovaný zoznam, ktorý jednoznačne, úplne a pochopiteľne uvádza kroky vedúce k získaniu očakávaného výsledku
 - 5.2. Vždy keď je to možné sa pri krokoch používajú príklady, obrázky, screenshoty a schémy
6. Voliteľne je možné na záver vo forme súvislého textu a odrážok opísať výsledok alebo množinu výsledkov, ktoré boli získané vykonaním postupu popísaného vyššie

Metodika tvorby technickej dokumentácie

verzia 2018-01-16

1. Dedikácia

Metodika je určená všetkým členom tímu 4 (EduVirtual). Metodika určuje pravidlá pre tvorbu technickej dokumentácie vytvoreného produktu. Metodika predpisuje formálnu stránku technickej dokumentácie, obsahovú stránku vrátane rozsahu jednotlivých častí a úrovne detailov.

2. Obsah metodiky

Dedikácia	1
Obsah metodiky	1
Znenie metodiky	2
Proces: Pridanie príspevku do technickej dokumentácie	2
Proces: Pridanie automaticky generovanej dokumentácie	3

3. Znenie metodiky

Proces: Pridanie príspevku do technickej dokumentácie

Technická dokumentácia musí popisovať všetky technické aspekty systému tak, aby ďalší vývojári mohli porozumieť fungovaniu systému a aby mohli modifikovať a rozširovať projekt. Pri písaní príspevku je potrebné dodržať nasledujúce vlastnosti:

- Zameriavať sa na podrobný opis technických záležitostí
- Podrobne a úplne opísať implementačné detaily
- Vysvetliť spôsob fungovania funkcií
- Vyjadriť prepojenie a vzájomné fungovanie jednotlivých častí systému

Technická dokumentácia sa skladá z Google dokumentu *Technicka dokumentacia produktu* a z adresárov Web, AR, VR a Globe. Tieto adresáre obsahujú automaticky generovanú dokumentáciu a samostatné dokumenty, napríklad diagramy, SQL, časti zdrojového kódu a podobne.

Tento proces sa zameriava iba na pridanie príspevku do dokumentu *Technicka dokumentacia produktu*.

Nasledujúce kroky popisujú postup pre pridanie príspevku do dokumentu:

1. Dokument *Technicka dokumentacia produktu* je rozdelený na kapitoly a podkapitoly. Jedna kapitola sa zameriava na jeden komponent a podkapitola by mala popisovať implementáciu jednej časti komponentu.
2. Ak už existuje kapitola, do ktorej je vhodné vložiť nový príspevok, vloží sa do tejto kapitoly. Inak sa vytvorí nová kapitola s názvom, z ktorého bude jasne vyplývať, akú časť systému (aký komponent) táto kapitola popisuje
3. Vytvorí sa podkapitola príspevku, ktorý je potrebné pridať - názov podkapitoly by mal jasne ukazovať, akú časť systému daná podkapitola popisuje.
4. Radenie kapitol a podkapitol je zobrazené v nasledujúcej tabuľke:

Kapitola 1: Komponent systému A

Podkapitola 1.1: Opis implementácie časti X komponentu A

Podkapitola 1.2: Opis implementácie inej časti komponentu A

Podkapitola 1.3: Opis fungovania ..

Kapitola 2: Komponent systému B

Podkapitola 2.1: Implementácia časti Y komponentu B

Podkapitola 2.2: Implementácia a fungovanie ďalšej časti komponentu B

Kapitola 3: ...

Podkapitola 3.1: ...

...

5. Napísať popis implementácie alebo fungovania

- a. Je potrebné písať štruktúrovanie, v odrážkach alebo číslovaných zoznamoch
- b. Je potrebné používať ukážky, príklady, časti zdrojového kódu, diagramy, schémy, príkazy
 - i. zdrojový kód a príkazy je potrebné vložiť do tabuľky 1x1 (tak ako je to vidieť pri ukážke kapitol v tomto dokumente) a je potrebné nastaviť font na Roboto Mono

Proces: Pridanie automaticky generovanej dokumentácie

Automaticky generovaná dokumentácia, nezávislé dokumenty technickej dokumentácie, súbory s diagramami, súbory so zdrojovým kódom, SQL súbory a vzorové výstupy sa ukladajú do podadresárov Web, AR, VR, Globe. Postup na pridanie takéhoto typu dokumentácie je nasledovný:

1. Určenie, do ktorého priečinku bude príspevok zaradený
 - 1.1. Ak sa príspevok vzťahuje k edukačnej webovej stránke, k serveru, databáze, LRS, serverovému komponentu na zabezpečovanie komunikácie s databázou a LRS, umiestni sa do adresára Web

- 1.2. Ak sa príspevok týka Android aplikácie AR, Vuforia, práce s 3D objektami, umiestni sa do adresára AR
 - 1.3. Ak sa príspevok týka Android aplikácie VR, Street View, umiestni sa do adresára VR
 - 1.4. Ak sa príspevok týka hry Globe, Unity assetov pre túto hru, umiestni sa do adresára Globe
2. Ak sa vkladá iba jeden súbor, môže sa umiestniť priamo do adresára vybraného v kroku 1.
 - 2.1. Súbor musí byť pomenovaný tak, aby z jeho názvu priamo vyplývalo, na akú časť systému sa zameriava. Názov súboru musí byť unikátny a nezameniteľný
 3. Ak sa vkladá príspevok pozostávajúci z viacerých súborov alebo sa vkladá jeden súbor, ale v budúcnosti sa predpokladá pridanie ďalších súborov k tejto časti, tak sa v adresári určenom v kroku 1 vytvorí nový podadresár
 - 3.1. Adresár musí byť pomenovaný tak, aby z jeho názvu priamo vyplývalo, na akú časť systému sa zameriava. Názov súboru musí byť unikátny a nezameniteľný
 - 3.2. Do tohto adresára sa potom vložia pridávané súbory
 4. Príklad:

Web/

Schema SQL databazy/

Logicky model.png

Fyzicky Model.png

slq_dump.sql

Mapa_stranky.html

Globe/

Globe asset/

Globe asset dokumentacia.html

Class diagram.png

...

5. Každý príspevok v jednom z adresárov Web, AR, VR, Globe musí byť spomenutý v dokumente *Technická dokumentacia produktu*