

Tímový projekt

REKONŠTRUKCIA 3D SCÉNY

V2

Riadenie

Vedúci projektu:

Ing. Vanda Benešová, PhD.

Členovia tímu:

Bc. Jakub Ginter (SI)
Bc. Miroslav Haščič (SI)
Bc. Mário Hunka (SI)
Bc. Viktor Košťan (IS)
Bc. Richard Pintér (IS)

Názov tímu: 3DRecon (tím č. 11)

Web: <http://team11-16.studenti.fiit.stuba.sk/>

Kontakt: teamfiit11@gmail.com

Akademický rok: 2016/2017

Dátum odovzdania: 13. 12. 2016

Obsah

1	Úvod.....	1
1.1	Prehľad dokumentu	1
2	Role členov tímu a podiel práce	2
2.1	Manažérske činnosti	2
2.2	Podiel práce.....	2
3	Aplikácie manažmentov	3
3.1	Manažment komunikácie	3
3.1.1	Tímové stretnutia	4
3.1.2	Komunikačné nástroje	4
3.2	Manažment vývoja a integrácie.....	4
3.3	Manažment dokumentácie	5
3.4	Manažment plánovania	5
3.5	Manažment riadenia	6
4	Sumarizácia šprintov	7
4.1	Šprint 1	7
4.2	Šprint 2	7
4.3	Šprint 3	8
4.4	Šprint 4	8
5	Globalna retrospektiva	9
6	Používané metodiky	10

1 ÚVOD

Tento dokument obsahuje postupy a metódy, ktoré boli definované naším tímom za účelom riadenia tímového projektu. Projekt rieši segmentáciu 3D scény z rekonštruovaných 3D dát. Hlavnou myšlienkou je umožniť segmentáciu scény z dát v ľubovoľnom formáte pričom výstup by mal byť použiteľný softvérom *AutoCAD*¹.

Štafetu sme prebrali po minuloročnom tíme, ktorý viedol Bc. Lukáš Hudec. Vzhľadom nato, sme pokračovali v používaní niektorých metodík (metodika písania zdrojového kódu), avšak definovali sme si aj vlastné, ktoré sú špecifické pre náš tím.

Projekt je výskumného charakteru. Je teda potrebné umožniť testovanie rôznych segmentačných metód, ktoré po evaluácii môžu skončiť ako prototyp na zahodenie. Samotná práca s 3D dátami je výpočtovo, ale aj pamäťovo náročná. V súvislosti s tým, sme sa spočiatku začali uberať smerom, ktorý by urýchlil prototypovanie (ukladanie medzivýsledkov, ktoré neovplyvňujú výsledky testovania novej metódy). Ako ďalší z našich cieľov, je úplná segmentácia tzv. *outlayerov* (kľučka, svetlo atď.), ktoré vytvárajú nepresnosti pri segmentácii objektov, pri ktorých sa nachádzajú, prípadne sú ich súčasťou (stena, stôl atď.). V neposlednom rade, rozširujeme aj možnosti formátov vstupných dát. Ako ďalšie máme v pláne rekonštruovať 3D dáta nasnímané pomocou zariadenia *Kinect*² z rôznych uhľov pohľadov.

Výsledkom našej práce bude knižnica, ktorá poskytuje rôzne metódy segmentácie 3D dát využiteľná hlavne vo výskumnej sfére. V rámci vývoja tejto knižnice je samozrejme dostupné aj používateľské rozhranie, ktoré umožňuje používateľovi ľahko a efektívne, testovať nové segmentačné metódy.

1.1 PREHĽAD DOKUMENTU

V časti 2 sú opísané manažérske činnosti a zodpovednosti členov nášho tímu. V podkapitolách môžeme nájsť aj podiel práce členov tímu na tomto dokumente. V 3 kapitole sme uviedli aplikácie manažmentov, ktoré

¹ Nástroj určený na dizajn <http://www.autodesk.com/products/autocad/overview>

² Hardware Senzor od Microsoftu <http://www.xbox.com/en-US/xbox-one/accessories/kinect>

používame pri riadení nášho tímu. V kapitole 4 sme sumarizovali výsledky našich 3 šprintov. V rámci vývoja nášho softvéru používame určité metodiky, ktoré sú uvedené v časti 5. V ďalších častiach tohto dokumentu sa nachádzajú exporty z nástroja používaného na manažovanie (TFS), globálna retrospektíva a zápisnice zo stretnutí.

2 ROLE ČLENOV TÍMU A PODIEL PRÁCE

2.1 MANAŽÉRSKE ČINNOSTI

TABUĽKA 1: ROZDELENIE ÚLOH V TÍME

Meno	Rola	Zodpovednosť
Jakub Ginter	vedúci tímu, biznis manažér	analýza a návrh požiadaviek, komunikácia s produkt ownerom
Miroslav Haščič	manažér kvality	vykonávanie testov, riešenie a správa chýb (bug-ov)
Viktor Košťan	manažér vývoja a architektúry	návrh a implementácia architektúry, správa hlavných segmentálnych metód
Mário Hunka	integrátor, manažér propagácie	prepájanie jednotlivých modulov projektu, tvorba a udržiavanie web stránky
Richard Pintér	manažér dokumentácie	udržiavanie dokumentácie a záznamov zo stretnutí

2.2 PODIEL PRÁCE

TABUĽKA 2: PODIEL PRÁCE NA DOKUMENTÁCIÍ RIADENIA

Časť	Člen tímu
Úvod	Mário Hunka
Role členov tímu a podiel práce	Mário Hunka
Manažment komunikácie	Jakub Ginter
Manažment vývoja a integrácie	Jakub Ginter
Manažment plánovania	Mário Hunka

Manažment riadenia	Miroslav Haščič
Manažment dokumentovania	Richard Pintér
Sumarizácia šprintov	Richard Pintér
Metodika zadávania úloh do TFS	Viktor Košťan
Metodika prehliadok kódu	Mário Hunka
Metodika verziovania kódu	Jakub Ginter
Metodika testovania	Miroslav Haščič
Metodika dokumentácie	Richard Pintér
Globálna retrospektíva	Mário Hunka
Export úloh z TFS	Jakub Ginter
Integrácia a finalizácia Kontrolný bod 1	Mário Hunka
Integrácia a finalizácia Kontrolný bod 2	Richard Pintér

TABUĽKA 3: PODIEL PRÁCE NA DOKUMENTÁCIÍ K INŽINIERSKEMU DIELU

Časť	Člen tímu
Úvod	Jakub Ginter
Globálne ciele pre ZS	Jakub Ginter
Celkový pohľad na systém	Viktor Košťan
Moduly systému – úvod	Viktor Košťan
Segmentácia Outlayerov	Mário Hunka
Rekonštrukcia dát z Kinectu	Mário Hunka
Zlepšenie vizualizéra	Miroslav Haščič
Analýza metód na rozpoznávanie a segmentáciu stien	Viktor Košťan
Integrácia a finalizácia	Richard Pintér

3 APLIKÁCIE MANAŽMENTOV

3.1 MANAŽMENT KOMUNIKÁCIE

Naša komunikácia v tíme by sa dala rozdeliť na dva hlavné celky. Prvým sú tímové stretnutia a druhým komunikácia v nástrojoch

3.1.1 TÍMOVÉ STRETNUTIA

Spoločne sme sa stretali podľa rozvrhu každý štvrtok o 13:00 v laboratóriu VGG 4.46. Na týchto stretnutiach sme spolu s vedúcou tímu zhodnocovali čo sa spravilo a čo sa bude robiť. Každý člen vysvetlil na čom pracoval a kde boli problémy, resp. aké by mohli v budúcnosti nastať. Tieto stretnutia prebiehali podľa agilnej metódy SCRUM, ktorá je opísaná v manažmente plánovania. Tam je možné nájsť ako presne tieto stretnutia prebiehali.

Mimo týchto stretnutí sme sa stretávali aj v iných dňoch a riešili problémy, na ktoré sme narazili a bolo potrebné ich vyriešiť čo najskôr alebo jednoducho pracovali. Tieto stretnutia prebiehali zväčša vo VGG alebo v knižnici.

3.1.2 KOMUNIKAČNÉ NÁSTROJE

Na našu komunikáciu sme najčastejšie používali dva nástroje (Slack a Google Drive). V prvotných začiatkoch prác na projekte sme komunikovali prostredníctvom skupinového Facebook chatu. To sa však ukázalo ako neefektívne bolo potrebné nájsť vhodný nástroj.

Slack je aplikácia určená na komunikáciu v tímoch a zvýšenie ich produktivity. Je možné vytvoriť komunikačné kanály čo umožňuje rozdeliť komunikáciu do tematických celkov. V rámci kanálov je možné nastavovať notifikácie pre jednotlivých členov alebo skupinu. To má veľké využitie obzvlášť pri mobilnej aplikácii, ktorá je tiež k dispozícii. Používateľ tak môže nastaviť upozornenia iba na správy kde bol označený a vie, že je potrebné reagovať čo najskôr.

Pomocou tejto aplikácie je možné nahrávať súbory, zdieľať ukážky zdrojového kódu alebo upozorňovať pri commite vetvy.

Google Drive sme používali ako tímové úložisko, kde sme uchovávali dokumentáciu, záznamy zo stretnutí a rôzne iné dokumenty potrebné pre tím. Rovnako sme tu mali uložené datasety pre náš projekt.

Menej často sme používali Skype alebo Gmail, na ktorom sme mali založený tímový alias odkiaľ sa pošta preposielala každému členovi.

3.2 MANAŽMENT VÝVOJA A INTEGRÁCIE

V našom projekte je úloha manažéra vývoja a manažéra integrácie rozdelená medzi dvoch členov. Ich úlohy pozostávajú z manažmentu verzií, integrácie súčastí do jedného veľkého celku ale aj konfigurácia systému a iné. Manažér

integrácie je priamo zodpovedný za repozitáre v TFS a udržiava vetvy projektu. Rieši konflikty a pridáva novú funkcionálnosť do hlavnej vetvy. Viac o manažovaní projektových vetiev a postup ako ich v projekte 3Drecon udržiavame sa nachádza v dokumente Metodika verziovania.

Aby sme dokázali udržiavať prehľadný kód a projekt používame TFS (Team Foundation Server) kde medzi sebou zdieľame vetvy projektu, manažujeme úlohy aj zapisujeme bugy. Tento nástroj je priamo integrovaný do Visual Studia, ktoré používame na vývoj a preto je pre nás TFS najlepšou možnou alternatívou. Nástroj ponúka jednoduchú synchronizáciu a aktualizovanie zmien, ktoré priamo prenáša medzi našim úložiskom aj s prípadnými správami.

Každý z členov tímu je zodpovedný za časť systému, ktorú vyvíja teda prípadné nedostatky a chyby v týchto častiach musí primárne vyriešiť jej tvorca. Tvorca nie je zodpovedný iba za funkcionálnu stránku ale aj za výstup a úpravu, teda za celkové spracovanie (správne okomentovaný a zadokumentovaný kód). Následná integrácia do projektu je vykonaná integrátorom. Tvorca s ním túto skutočnosť odkomunikuje podľa pravidiel manažmentu komunikácie a poskytne integrátorovi potrebnú súčinnosť pri práci.

3.3 MANAŽMENT DOKUMENTÁCIE

Písanie dokumentácie v našom tíme prebieha priebežne. Dokumentácia je vytváraná spôsobom spoločného dokumentu, typu GOOGLE DOC, ktorý máme uložený na spoločnom úložisku GOOGLE DRIVE. Nakoľko dokumentácia sa skladá z viacerých častí, na GOOGLE DRIVE sú postupne pridávané dokumenty, ktoré si následne prejdú viacerí členovia tímu a po spoločnej konzultácii sa dohodnú, či je dokument vyhovujúci alebo je nutné ho prerobiť. V prípade, že je dokument vyhovujúci je následne zaintegrovaný do finálnej dokumentácie. Aby sme predišli rôznym kolíziám typu odlišné riadkovanie alebo font písma, dohodli sme sa dodržiavať pravidlá písania dokumentácie, ktoré sú spísané v dokumente menom Metodika písania dokumentácie.

3.4 MANAŽMENT PLÁNOVANIA

Predpokladom úspešného plánovania je naplnený tzv. product backlog aspoň na 2 šprinty vopred. Product backlog sa vytváral na základe požiadaviek

nášho zadávateľa, ktorý je v tomto prípade zároveň náš vedúci tímu. Vzhľadom na požiadavky, boli hierarchicky vytvorené časti softvéru (Epics, Features), pre ktoré sa vytvárali backlog itemy. Tie sa radia vždy podľa priority – dôležitosť s akou treba daný backlog item spraviť, aby sme naplnili požiadavky čo najefektívnejším spôsobom.

Proces plánovania prebieha na začiatku každého šprintu. Podľa priority sú ku šprintu priradené backlog itemy, na ktorých sa pracuje počas najbližšieho šprintu. Backlog item má určité kritéria akceptácie, ktoré musia byť dodržané, aby bola úloha splnená. Na naplnenie týchto kritérií je väčšinou potrebné viacero úloh (taskov).

Následne sa ohodnotia jednotlivé úlohy vzhľadom na prvú ohodnotenú úlohu, ktorá bola vopred určená (base). Na každom stretnutí sa hrá tzv. *Scrum Poker*, pri ktorom sa odhaduje náročnosť úloh pomocou hlasovania, čo sprevádza diskusia o úlohách o ktorých náročnosti sa hlasuje. Touto aktivitou je zabezpečený čo najlepší odhad k jednotlivým úlohám. Vzhľadom na minulé šprinty prispôbujeme náročnosť ďalšieho šprintu, preto je veľmi dôležité zabezpečiť čo najlepší odhad.

Následne sú zadané úlohy do manažovacieho nástroja TFS, ktoré sú buď priradené jednotlivým členom tímu, alebo si ich členovia tímu vyberajú počas šprintu. Progres v jednotlivých úlohách sa taktiež sleduje v nástroji TFS.

3.5 MANAŽMENT RIADENIA

Riadenie v našom tíme bolo založené na agilnej metóde vývoja SCRUM. V časti manažment plánovania sa môžete dozvedieť ako sme pomocou tejto metódy plánovali postup v našom projekte a ako sme určovali úlohy. Tieto úlohy sme potom evidovali v nástroji TFS. V každom šprinte sme vytvorili definované backlog items a k nim patriace úlohy. Každý člen mal definované svoje úlohy a ich hodnotu (ako sa táto hodnota získava nájdete v časti manažment plánovania). Niektoré úlohy nemali pridelenú zodpovednú osobu nakoľko nebolo jasné kto to bude robiť a nebolo ani potrebné to definovať. Dohodnuté však bolo že sa to musí urobiť a ten kto mal najviac času takúto úlohu pripísal v TFS sebe a urobil ju.

Pre úlohy existujú tri základné stavy: To do, In progress, Done. Logicky je cieľom, každú z definovaných úloh dopracovať do štádia done. Ak by sa však niektoré úlohy v šprinte nestihli urobiť, presúvajú sa do ďalšieho šprintu kde majú najvyššiu prioritu.

Dohľad nad vykonávaním úloh a nad ich rozdeľovaním má vedúci tímu. Ten je zodpovedný za správne rozdelenie úloh, prípadne riešenie problémov v tejto oblasti.

4 SUMARIZÁCIA ŠPRINTOV

4.1 ŠPRINT 1

V prvom šprinte sme sa primárne snažili urobiť všetko podľa odporúčaní odborníkov. Narazili sme však na niekoľko problémov, ktoré sa odzrkadlili aj na jeho kvalite. No práve vďaka týmto chybám teraz vieme omnoho lepšie ako na šprintoch pracovať. Veľkým problémom bolo, že sme sa museli naučiť fungovať ako tím a dokázať tak zvýšiť efektivitu našej práce. Druhým problémom bolo, že pokračujeme v projekte po minuloročnom tíme a na začiatku sme mali minimálne skúsenosti a znalosti v oblasti, ktorej sa projekt venuje. Preto bolo potrebné dlhé analyzovanie, pretože bez neho nebolo možné ďalej implementovať novú funkcionálnosť. Fakt, že celý projekt je výskumného charakteru túto situáciu sťažoval ešte o trochu viac. So SCRUM-om sme v tíme skúsenosti nemali a s jeho použitím na výskumnom projekte už vôbec nie.

Začali sme používať nové nástroje ako SLACK na komunikáciu a TFS na evidenciu úloh. Nejaký čas nám trvalo aj naučiť sa pracovať s týmito nástrojmi, hlavne s TFS.

V prvom šprinte sme definovali 4 backlog items.

1. Zrýchlenie výpočtov ukladaním medzivýsledkov
2. (bug) Slabá viditeľnosť procesov
3. Analýza celkového kódu
4. (bug) Zlyhanie programu pri načítaní konfiguračného súboru (.xml)

4.2 ŠPRINT 2

Pri retrospektíve prvého šprintu sme sa všetci členovia tímu jednohlasne zhodli, že našou najväčšou slabinou bolo TFS a jeho správne používanie, čo sme sa rozhodli do ďalšieho šprintu zmeniť. V druhom šprinte sme sa po dohode s vedúcou projektu a s členom minuloročného tímu Lukášom Hudecom rozhodli pustiť do reštruktulizácie celého programu. Vedeli sme, že tento krok nás spomalí v napredovaní rozširovania funkcionálnosti ale na druhej strane sme vedeli, že ak si to navrhne a prerobíme podľa seba, bude to mať pozitívny účinok do budúcnosti. Najprv sme vytvorili návrh class diagramu nového projektu, ktorý sme následne diskutovali s vedúcou cez Skype hovor. Na konci hovoru, po odsúhlasení štruktúry projektu sme si rozdelili úlohy a postupne sme na nich začali pracovať. Počas šprintu sme narazili na zopár problémov pri rozbehávaní projektu na notebookoch, ktoré sa nám podarilo našťastie vyriešiť. Na konci šprintu sme zistili, že sme v stanovenom čase

nedokázali spraviť všetku prácu, ktorú sme si naplánovali a ,že v tom budeme musieť pokračovať aj v treťom šprinte.

4.3 ŠPRINT 3

Pri retrospektíve druhého šprintu sme zistili, že naša práca s TFS je už na vyššej úrovni ako to bolo pri prvom šprinte no stále sme nerobili niektoré veci tak ako by sme mali respektíve ako by sme si predstavovali. Zistili sme, že pri taskoch nezadáваме správne hodnoty v story pointoch a, že to má výrazný vplyv na nežiadúci výzor nášho Burndown chart diagramu. Taktiež sme zhodnotili že by sme mali v TFS riešiť aj neprogramátorske úlohy a mohli by sme vylepšiť aj spôsob robenia nášho code review. Tieto problémy sme si na stretnutí vydiskutovali a navrhli sme vhodné riešenie na ich odstránenie. Pri vytváraní Backlog Items pre tretí šprint sme takmer všetok čas nášho šprintu investovali do dokončenia úloh z predchádzajúceho šprintu, no taktiež sme už načrtli víziu do budúcnosti a identifikovali sme si ciele ako vyriešiť outliers a taktiež možnosť pracovať na našom projekte pomocou Kinectu. Pri hraní Scrum Pokeru sme si určili vzorový task na základe, ktorého sme pridelovali hodnoty ostatným úlohám. V rámci tretieho šprintu bolo nutné dokončiť dokumentáciu.

4.4 ŠPRINT 4

Plánovanie. Plánovanie pri šprinte 4 prinieslo rozdelenie nášho tímu do viacerých smerov. Narozdiel od predošlých šprintov, kde sme riešili refactoring a novú architektúru softvéru, tieto tasky boli konkrétnejšie a ich úlohou bolo priniesť novú funkcionality. Venovali sme sa segmentácií outlayerov, ktorá je aplikovateľná pri segmentácií stien a môže priniesť lepšie výsledky celkovej segmentácie. Ďalej sme sa venovali problému zalomenia stien, ktorý vzniká pri segmentácií. Jeden člen z nášho tímu hľadal možnosti využitia MS Kinect v našom projekte - rekonštrukcia nasnímaných dát. V neposlednom rade sme sa venovali vylepšovaniu nášho vizualizéra, ktorý je potrebný pri testovaní a samozrejme aj pri prezentácií nášho projektu. Ako posledný task sme si zadali priebežnú dokumentáciu kódu. Určili sme člena, ktorý sa bude venovať DoxyGen-u a bude kontrolovať aké komentáre sa píše a či spĺňajú všetky dohodnuté pravidlá.

Priebeh. V šprinte išlo hlavne o analytické úlohy, ktoré mali priniesť vhodné návrhy riešení, ktoré budú otestované. Analýzy sme predstavili na stretnutí v strede šprintu, kde sme ich bližšie diskutovali s našou vedúcou.

Retrospektíva. Retrospektíva priniesla výraznú zmenu pre náš tím. Zistili sme 2 problémy, ktoré sú kritické. Je potrebné riadiť verziovanie a zmeniť

dĺžky šprintov. V prípade analytických šprintov (ako bol z veľkej časti tento) sme sa rozhodli pre 1 týždňový šprint, ktorá nám dopomôže k tomu, ako naplánovať ďalší 2 týždňový šprint čo najpodrobnejšie. Taktiež je potrebný manažér verziovania, keďže sa stáva, že sme často závislí jeden od druhého a je potrebné, aby niekto kontroloval priebeh a dokončenie novej funkcionality (schválené pull requesty, nové branche a pod.).

5 GLOBALNA RETROSPEKTIVA

V rámci celkovej retrospektívy sme do tohto bodu semestra narazili na viacero problémov, ktoré sme sa snažili riešiť v každom šprinte. Celkovo môžeme vytýčiť niekoľko skutočností, ktoré ovplyvnili vývoj nášho tímu najviac.

1. **Práca s TFS** – jednoznačne jeden z najväčších problémov. Keďže sme doteraz nemali možnosť pracovať v tíme pozostávajúcom z viac ako dvoch ľudí, je to pre nás nová skúsenosť. Nemali sme žiadne skúsenosti s používaním systémov na manažment úloh. Taktiež spadajú pod tento problém aj iné náležitosti, ako napr. tvorba backlogu či zadávanie estimate-u pri taskoch a vlastne celkové poznatky v oblasti manažmentu úloh.

V rámci 3. šprintu vidieť značný progres v tejto časti. Každý pracuje s tfs, interaguje s tabuľou, zadáva bugy a pod. Výsledok je vidieť na burn-down charte, ktorý nabera správny spád a taktiež aj na velocity, ktorá sa postupne zvyšuje.

2. **Plánovanie** – v rámci plánovania šprintu sme učinovali viacero chýb. Nemali sme vytvorený backlog, z ktorého by sme len podsúvali podľa priority item-y do šprintov. Bolo to ovplyvnené hlavne tým, že pokračujeme po minuloročnom tíme a nevedeli sme presne kam sa chceme pohnúť a navyše je projekt výskumného charakteru, čo znamená, že backlog sa nám neustále mení vzhľadom na výsledky jednotlivých prototypov. Planning poker sme nehrali korektne a nezapisovali sme celý jeho opis rovno do description pri jednotlivých taskoch.

V rámci 3. šprintu sme v tejto časti spravili jednoznačne najväčší krok vpred. Aj keď je pravdou, že výskumný projekt nie je možné naplánovať na pol roka dopredu, lebo nikdy nie je isté čo-ako dopadne. Napriek tomu sme sa posnažili naplniť backlog minimálne na najbližšie 3 šprinty. Z týchto item-ov sme podľa priority vybrali

tie s najvyššou a presunuli sme ich do šprintu. Je cítiť oveľa väčší prehľad v tom čo robíme.

Planning poker v 3. šprinte bol jednoznačne prínosnejší ako tie predtým. Hlasovalo sa o všetkých taskoch a diskusia sa písala rovno do ich opisu. Brainstorming zabezpečil lepší odhad a taktiež aj presný opis toho čo je treba v budúcnosti urobiť.

V neposlednom rade môžeme vyzdvihnúť našu komunikáciu. Od začiatku používame *Slack*, štruktúrujeme komunikáciu na jednotlivé témy a všetci sa aktívne zapájame. Taktiež využívame *Google Drive* na zdieľanie jednotlivých dokumentov. Okrem iného, sa stretávame osobne, mimo nášho oficiálneho stretnutia – to hodnotíme kladne a určite v tom chceme pokračovať aj naďalej.

6 POUŽÍVANÉ METODIKY

TABUĽKA 4: POUŽÍVANÉ METODIKY

METODIKA ZADÁVANIA ÚLOH DO TFS	Metodika opisuje spôsob plánovania jednotlivých šprintov a naplnenia backlogu.
METODIKA PREHLIADOK KÓDU	Opisuje spôsob ako vykonávať prehliadky kódu a na čo sa pri nich zamerať.
METODIKA VERZIOVANIA KÓDU	Opisuje spôsob členenia projektu na jednotlivé verzie a ako správne používať Git.
METODIKA TESTOVANIA	Ako správne testovať náš systém.
METODIKA PÍSANIA DOKUMENTÁCIE	Tu sú spísané pravidlá, ktoré je potrebné pri písaní dokumentácie dodržať.