

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava 4

Simulated Car Racing Competition 2011

Dokumentácia k riadeniu projektu



Tím číslo:	20	Bc. Maroš Bednár
Vedúci projektu:	Ing. Peter Vilhan	Bc. Adam Brček
Predmet:	Tvorba softvérového systému v tíme	Bc. Marek Briš
Študijný program:	Informačné systémy	Bc. Marián Florek
	Softvérové inžinierstvo	Bc. Vojtech Juhász
Akademický rok:	2010/2011	Bc. Juraj Kosmeľ
		Bc. Ivan Valenčík

A) Obsah

A)	Obsah	A-1
B)	Úvod (Juhász)	B-1
B.1	Účel dokumentu	B-1
B.2	Prehľad dokumentu	B-1
C)	Ponuka	C-1
1.	Členovia tímu (Bednár)	1
2.	Simulated Car Racing Competition 2011 (Valenčík).....	3
Konceptia riešenia (Brček, Valenčík).....	4	
3.	Model používateľa pre jeho identifikáciu (Florek).....	5
Konceptia riešenia (Briš, Kosmeľ)	6	
4.	Portál pre časopis (Juhász)	7
Návrh riešenia.....	7	
5.	Príloha A: Poradie tém podľa preferencií (Bednár).....	10
6.	Príloha B: Aktuálny rozvrh členov tímu (Bednár, Valenčík).....	11
D)	Plány projektu (Juhász)	D-1
D.1	Tvorba plánov	D-1
D.2	Externé termíny.....	D-1
D.3	Hrubý plán projektu na zimný semester	D-2
D.4	Revízia plánu v siedmom týždni.....	D-3
D.5	Revízia plánu v jedenástom týždni.....	D-4
E)	Manažment rizík.....	E-1
E.1	Analýza rizík	E-1
E.2	Identifikované riziká	E-2
F)	Úlohy členov tímu (Juhász)	F-1
F.1	Dlhodobé roly členov tímu	F-1
F.2	Krátkodobé úlohy členov tímu	F-1
G)	Podporné prostriedky a metodiky práce	G-1

G.1	Práca s SVN (Briš).....	G-1
G.2	Komunikácia (Kosmeľ).....	G-3
G.3	Pravidlá písania a komentovania zdrojového kódu (Valenčík).....	G-5
G.4	Manažovanie chýb (Juhász).....	G-9
G.5	Reportovanie úlohy (Florek).....	G-10
G.6	Tvorba plánov (Juhász).....	G-13
G.7	Konvencie písania dokumentácie (Bednár).....	G-14
H)	Zápisnice zo stretnutí.....	H-1
	Zápis 1. stretnutia tímu č. 20 (Brček).....	H-2
	Zápis 2. stretnutia tímu č. 20 (Brček).....	H-3
	Zápis 3. stretnutia tímu č. 20 (Bednár).....	H-5
	Zápis 4. stretnutia tímu č. 20 (Briš).....	H-7
	Zápis 5. stretnutia tímu č. 20 (Kosmeľ).....	H-9
	Zápis 6. stretnutia tímu č. 20 (Florek).....	H-11
	Zápis 7. stretnutia tímu č. 20 (Juhász).....	H-13
	Zápis 8. stretnutia tímu č. 20 (Valenčík).....	H-15
	Zápis 9. stretnutia tímu č. 20 (Bednár).....	H-17
	Zápis 10. stretnutia tímu č. 20 (Briš).....	H-19
I)	Autori jednotlivých častí dokumentácií (Bednár).....	I-1
	I.1 Autori dokumentácie k projektu.....	I-1
	I.2 Autori dokumentácie k riadeniu.....	I-2
J)	Preberacie protokoly (Brček).....	J-1

B) Úvod (Juhász)

B.1 Účel dokumentu

Predložená dokumentácia obsahuje všetky dokumenty súvisiace s riadením projektu v predmete Tvorba softvérového systému v tíme na tému Simulated Car Racing Competition 2011. Dokumenty boli vytvárané priebežne s cieľom sprehľadniť prácu a zdokumentovať vynaložené úsilie pri práci na projekte.

B.2 Prehľad dokumentu

Tento dokument obsahuje nasledujúce časti:

- **Ponuka:** je to dokument, ktorým sme sa uchádzali o pridelenie témy na riešenie v rámci tímového projektu na tému: *Simulated Car Racing Competition 2011*. Ponuka obsahuje predstavenie jednotlivých členov tímu, motiváciu pre výber danej témy a návrh možného riešenia.
- **Plán projektu:** v tejto sekcii uvádzame plán projektu vypracovaný na začiatku zimného semestra, tzv. hrubý plán na zimný a letný semester. Do tejto sekcie sa postupne dopĺňajú jednotlivé zjemnené krátkodobé plány v chronologickom poradí.
- **Úlohy členov tímu:** v tejto časti uvádzame rozdelenie manažérskych funkcií jednotlivých členov tímu ako aj krátkodobé úlohy členov tímu s dátumom predpokladaného ukončenia.
- **Podporné prostriedky a metodiky práce:** táto časť obsahuje popis podporných prostriedkov pre manažment projektu, ako aj metódy, postupy a šablóny na tvorbu dokumentov na ktorých sa tím dohodol.
- **Zápisnice zo stretnutí:** obsahuje zápisy zo spoločných tímových stretnutí. Každý jeden zápis obsahuje vyhodnotenie úloh z predchádzajúceho stretnutia ako aj opis priebehu stretnutia a pridelenie úloh na ďalší týždeň pre jednotlivých členov tímu.

C) Ponuka

Táto kapitola obsahuje dokument vypracovaný za účelom získania projektu.

Slovenská technická univerzita

Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava 4

Ponuka Tímový projekt

Autori: Bednár Maroš, Bc.
Brček Adam, Bc.
Briš Marek, Bc.
Florek Marian, Bc.
Juhász Vojtech, Bc.
Kosmeľ Juraj, Bc.
Valenčík Ivan, Bc.

Kontakt: brcek.a@gmail.com

Predmet: Tímový projekt

Dátum: 24.9.2010

Akad. rok: 2010/11



1. Členovia tímu (Bednár)



Bednár Maroš, Bc. Absolvent bakalárskeho študijného programu Informatika na FIIT STU, ktorý zavŕšil bakalárskou prácou na tému Multimediálny výučbový systém, v ktorom boli využité technológie ASP.NET, MsSQL, JavaScript, Flash, Ajax. Na škole získal vedomosti o objektovom programovaní (C++, JAVA, C#) a znalosti vo všetkých fázach vývoja softvérových systémov. Vo firme zastáva pozíciu Web Developera využívajúci jazyky PHP, SQL, jQuery, Ajax, kde vývoj je zameraný na webové aplikácie pre firemný CMS systém.



Brček Adam, Bc. Absolvent bakalárskeho programu Informatika na FIIT STU. Počas prvostupňového vysokoškolského štúdia si zlepšil svoje vedomosti v procedurálnom programovaní, po naučení objektovo-orientovanej paradigmy však pri riešení úloh v prevažnej miere využíva jazyk JAVA. Počas inžinierskeho štúdia sa okrem softvérového inžinierstva plánuje venovať najmä umelej inteligencii (Neurónové siete, Evolučné algoritmy, Kognitívna veda), ktorou sa zaoberal aj vo svojej bakalárskej práci Konštrukcia diagnózy chybného správania sa zložitých systémov.



Briš Marek, Bc. Absolvent bakalárskeho študijného programu Informatika na FIIT STU, počas ktorého získal skúsenosti s programovacími jazykmi C, JAVA, s relačnými databázovými systémami ako MySQL, s webovými technológiami HTML, CSS a modelovacím jazykom UML. V bakalárskej práci Vytváranie prezentácii si osvojil pravidlá interakcie človeka s počítačom a uplatnil objektovo-orientované programovanie. V rámci rôznych študijných projektov sa podieľal na všetkých fázach životného cyklu vývoja softvéru.



Florek Marian, Bc. Absolvent bakalárskeho programu Informatika na FIIT STU. Počas štúdia využíval v prevažnej miere objektovo orientovaný prístup programovania s preferovaným programovacím jazykom JAVA. Okrem jazyka JAVA sa oboznámil s prácou aj v iných jazykoch ako C/C++, HTML, PHP, JavaScript, značkovacom jazyku XML, modelovacím jazyku UML a relačnými databázami. Štúdium zavŕšil bakalárskou prácou s názvom "Media server pre VoIP komunikáciu" v ktorej preukázal znalosti potrebné počas vývoja softwaru, od analýzy, cez návrh, k samotnej implementácii a testovaniu.



Juhász Vojtech, Bc. Prvý stupeň vysokoškolského vzdelania absolvoval na FIIT STU kde sa oboznámil s programovacími jazykmi, ako sú Java, C/C++, databázové systémy a jazyk SQL. Počas štúdia získal poznatky z vytvárania softvérových systémov zastrešujúc analýzu, návrh, dokumentáciu, implementáciu, testovanie softvéru a jeho prezentáciu. Oboznámil sa s nástrojmi podporujúcimi rôzne fázy vývoja softvéru, napríklad jazyk UML a

rôzne podporné nástroje (IBM Rational Software Architect, Magic Draw UML) i rôzne vývojové prostredia (Eclipse, NetBeans, MS Visual Studio). V posledných rokoch pôsobí vo firme ktorá je lídrom na trhu informačných technológií v Španielsku a v latinskej Amerike. Pri svojej práci najviac používa jazyky Java, SQL, HTML, JavaScript, XML. V praxi sa podieľal na vývoje viacerých softvérov a webových stránok či portálov.



Kosmeľ Juraj, Bc. Je absolventom bakalárskeho štúdia na FIIT STU. Počas štúdia na tejto fakulte si rozšíril svoje vedomosti z procedurálneho programovania o jazyk C a nadobudol nové vedomosti v oblasti objektovo orientovanej paradigmy v jazyku Java, ktorý využil pri riešení väčšiny zadaní. Vo svojej bakalárskej práci sa venoval návrhu a implementácii „Podporného prostriedku pre hodnotenie a overovanie vedomostí“ pre predmet Dátové štruktúry a algoritmy, ktorý realizoval v skriptovacom jazyku PHP v kombinácii s HTML, CSS, JavaScript a databázovým serverom MySQL. Na rôznej úrovni ovláda aj Pascal, Assembler či notáciu UML (softvér IBM Rational Architect).



Valenčík Ivan, Bc. Je absolventom bakalárskeho štúdia v odbore Informatika na FIIT STU, ktoré ukončil s diplomom Magna Cum Laude. Vo svojej bakalárskej práci, za ktorú dostal pochvalu od dekana, vypracovanej na tému Vytvorenie inteligentného agenta subsymbolickým prístupom umelej inteligencie sa zaoberal použitím umelej neurónovej siete na riadenie robota v simulovanom prostredí Robocode. V priebehu štúdia a v doterajšej praxi sa oboznámil s širokým spektrom technológií, ale hlavne s programovacími jazykmi Java, C# a C/C++, počítačovými sieťami, PKI a postupmi softvérového inžinierstva. Okrem doteraz nadobudnutých vedomostí pri práci určite zúžitkuje aj vedomosti z pri projekte paralelne študovaných predmetov, medzi ktorými sú napr. neurónové siete alebo grafy.

2. Simulated Car Racing Competition 2011 (Valenčík)

Inteligentné agenty majú široké využitie na plnenie úloh v rôznych oblastiach. Z tohto dôvodu je užitočné čo najlepšie pochopiť a zlepšiť ich správanie. Na tento účel je vhodné využiť simulované prostredie, pričom za zaujímavé možno považovať hlavne také prostredia, ktoré majú snahu svojím fyzikálnym modelom simulovať reálny svet. Jedným z nich je The Open Racing Car Simulator (TORCS), ktorého doménou sú automobilové preteky.

TORCS framework je výnimočný tým, že poskytuje premyslený fyzikálny model zahŕňajúci prostredie pre interakciu plne autonómneho autopilota s okolím, sieť senzorov a takmer neobmedzené možnosti návrhu autopilota, od jednoduchého, staticky naprogramovaného s uzavretou množinou schopností, cez autopilota so schopnosťou natréňovať sa, až po dynamicky sa zdokonaľujúceho v reálnom čase.

Pri vytváraní inteligentného agenta si je teda možné vyskúšať rôzne prístupy umelej inteligencie na veľmi praktickom probléme, akým riadenie vozidla určite je. Mnohé podproblémy budú zrejme efektívne riešiteľné klasickým symbolickým prístupom, avšak pre zložitejšie situácie bude určite vhodné uvažovať aj nad použitím subsymbolického prístupu k UI, hlavne neurónových sietí. Takto vytvorená problémová oblasť pokrýva záujmy celého tímu a sľubuje zaujímavú prácu na celú dobu trvania tímového projektu. V etape rozvrhovania zastávok pilota v boxoch a nastavenia auta môžeme využiť dlhoročný záujem niektorých členov tímu o motoristický šport po teoretickej a taktickej stránke.

Na danej úlohe je lákavá aj možnosť overiť a porovnať kvalitu vytvoreného riešenia na súťaži s účasťou tímov z prostredia mimo našej univerzity, čo je motiváciou k dosiahnutiu čo najlepšieho výsledku a dobrej reprezentácie svojej fakulty. Vedomie, že človek je pri práci porovnávaný s inou relevantnou prácou je totiž často veľmi dôležité.

Koncepcia riešenia (Brček, Valenčík)

Cieľom projektu je vytvorenie čo najúspešnejšieho autopilota, táto vlastnosť však v sebe zahŕňa pomerne odlišné aspekty v závislosti od výberu súťaže, ktorej sa náš robot zúčastní. V krátkosti preto prezentujeme našu koncepciu pre obidve podujatia.

Championship

V prvej etape je potrebné pilota naučiť jazdiť na neznámej trati. Tu je potrebné zvládnuť základné činnosti ako akceleráciu, brzdenie, radenie a riadenie. Vznikajúce problémy ako pretáčavosť auta či prešmykovanie kolies je možné odstrániť zapracovaním fyzikálnych zákonov (odstredivá sila a iné) do pravidiel. V tejto fáze by pilot určitou konštantnou rýchlosťou, dostatočne nízkou na absolvovanie všetkých zákrut, prešiel celý okruh a ukladal si získané informácie o trati.

V druhej časti bude snahou nájsť ideálnu stopu, čiže takú trajektóriu, ktorá by umožnila dosiahnuť najvyššiu priemernú rýchlosť na jeden okruh. Po jej nájdení je možné ľahko priradiť optimálne rýchlosti pre jednotlivé úseky trate.

Tretie fáza predstavuje preteky, pri ktorých musíme zvládnuť hlavne vyhybanie sa kolíziám a predbiehaniu. Toto chceme zabezpečiť spočiatku základnými úkonmi ako korekcia smeru jazdy a zníženie rýchlosti, neskôr práve tu je možné zapracovať dynamické učenie pomocou konfrontácie s väčším počtom súperov slúžiacej na rozpoznanie vhodných situácií na uskutočnenie predbiehačích manévrov.

Demolition Derby

Pre tento typ súťaže sa na prvý pohľad javí byť vhodná kombinácia event-driven architektúry s prednastaveným správaním. Vozidlo sa bude na trati pohybovať podľa vopred určených pravidiel, napr. nebude zastavovať, aby sa nestalo ľahkým cieľom, bude sa vyhýbať stredu plochy, aby nepútalo zbytočne pozornosť, ale nebude sa ani pohybovať príliš pri kraji, aby malo možnosť uniknúť protivníkovi atď.

Pritom bude rozoznávať výskyt určených situácií, ktoré môžu byť rozdelené do troch skupín: útočné, obranné a strategické. V prípade, že je rozoznaný výskyt viacerých situácií, tak sa vozidlo bude správať podľa tej s najvyššou prioritou (tie bude zrejme vhodné určiť experimentálne, vysokú hodnotu budú mať asi už prebiehajúce útoky alebo úhybné manévry). Význam útočných a obranných akcií je samovysvetľujúci, pri strategických akciách ide o presun do vhodnej pozície alebo o vyhnutie sa nebezpečnej pozícii.

Na ovládanie agenta v tomto druhu súťaže bude pravdepodobne možné aplikovať aj viacero prístupov známych z prostredia Robocode, s ktorým má skúsenosť Ivan Valenčík, ktorý sa v ňom zaoberal súbojom inteligentných agentov.

3. Model používateľa pre jeho identifikáciu (Florek)

V súčasnosti je vo všetkých sférach spoločnosti a najmä v oblasti informačných technológií sústredený obrovské množstvo údajov. Tieto údaje sú často veľmi citlivé a preto sa do popredia dostávajú otázky ich ochrany proti zneužitiu. Ochrana a zabezpečenie údajov je dnes diskutovanou témou na rôznych fórach a konferenciách. Doteraz používané spôsoby zabezpečenia údajov sú často nepostačujúce a je potrebné hľadať nové techniky a možnosti zabezpečenia.

Každý používateľ je jedinečný, a to jednak svojimi biometrickými charakteristikami, ako aj jedinečným správaním sa a používaním informačných systémov. Tieto charakteristiky iniciovali vývoj softvérových a hardvérových prostriedkov umožňujúcich ich rozlišovanie, získavanie a následné využitie pri identifikácii používateľa. Na druhej strane informácie o správaní sa používateľa v systéme umožňujú vytvoriť model používateľa, a na jeho základe predvídať správanie a prispôbiť systém predpokladaným požiadavkám používateľa resp. ponúknuť používateľovi adekvátne návrhy.

Spôsoby zabezpečenia využívajúce na identifikáciu používateľa biometrické charakteristiky sú ešte málo rozšírené a bežný používateľ s nimi príde do kontaktu len veľmi zriedkavo. Radi by sme sa s naším tímom podieľali na rozvoji tejto oblasti, čo by nám realizácia tohto zadania umožnila. Je pre nás výzvou zúžitkovať svoje doteraz získané vedomosti pri riešení takéhoto projektu a zároveň ich obohatiť o nové schopnosti a zručnosti.

Členovia tímu ovládajú na rôznej úrovni viaceré programovacie jazyky a nástroje využívané v rámci životného cyklu softvéru. Svoje vedomosti využili pri riešení zadaní počas štúdia, ale aj v rámci pôsobenia na trhu práce. Toto dáva nášmu tímu veľmi dobré predpoklady na úspešné zvládnutie tohto zadania.

Koncepcia riešenia (Briš, Kosmel)

Funkcionalita

Na základe zadania a doposiaľ získaných informácií z tejto oblasti je možné realizovať riešenie viacerými spôsobmi. Chceli by sme vytvoriť podporný prostriedok, ktorý by umožňoval identifikovať používateľa a mohol by byť použitý v kombinácii so súčasným spôsobom zabezpečenia prístupu k službám a aplikáciám prostredníctvom prihlasovacieho mena a hesla. Ďalšiu možnosť identifikácie vidíme v aplikácii na monitorovanie správania sa používateľa. Táto aplikácia by umožňovala zozbierať údaje o práci a správaní sa používateľa v systéme a ich následné využitie pri tvorbe analýz, štatistík resp. pri tvorbe modelu používateľa.

API

Základom by bolo vytvorenie API na zisťovanie dynamiky používania klávesnice a myši, t.j. čas stlačenia klávesy, interval medzi stlačením dvoch kláves, medzi dvojklikom myši a pod. Toto API by umožňovalo aj ďalším tímom v budúcnosti nadviazať a pokračovať vo vývoji systému, prípadne využiť ho pri vývoji vlastných riešení.

Architektúra

V rámci nášho konceptu by sme realizáciu rozdelili do dvoch častí, a to modul identifikujúci používateľa na základe dynamiky písania na klávesnici pri prihlasovaní a modul na monitorovanie správania používateľa v systéme a vytvorenie modelu používateľa.

Databáza

Tieto dva moduly by boli podporované relačnou databázou, ktorá by uchovávala údaje o jednotlivých používateľoch. Tieto údaje by boli východiskom pre tvorbu modelov používateľov.

4. Portál pre časopis (Juhász)

Náš tím pozostáva z členov s rôznymi zameraniami, ale veríme, že na každú úlohu v rámci tímového projektu vieme nájsť toho najlepšieho. Keďže vývoj webového portálu sa nezačína a nekončí pri HTML – PHP – MySQL, myslíme si, že máme všetko k dispozícii na zvládnutie tohto projektu. V našom rozsiahlom tíme môžeme nájsť web dizajnéra a programátora i členov zameraných viac na analýzu alebo dokumentáciu softvéru.

Dôležitým faktorom je aj skúsenosť, ktorú tento projekt so sebou prináša. Domnievame sa, že ak tento projekt úspešne vyriešime, odborníci využívajúci náš systém budú tou najlepšou referenciou v budúcnosti. Nezanedbateľnou motivačnou silou je aj hrdosť, že sa budeme môcť podieľať na ďalšom vývoji ACM Slovakia Chapter.

Návrh riešenia

Existuje viacero spôsobov na riešenie projektu takéhoto typu, ale vzhľadom na naše skúsenosti s technológiami sme zvolili DHTML podporovanú zo servera pomocou PHP a MySQL. Ďalším dôvodom na voľbu spomenutých technológií je ich dostupnosť.

Systém bude poskytovať nasledujúce funkcionality:

1. web prezentácia časopisu
 - a. zobrazenie jednotlivých článkov
 - b. zobrazenie najviac čítaných a najnovších článkov
 - c. možnosť vyjadriť sa k jednotlivým článkom formou pridávania príspevkov
 - d. možnosť hodnotenia článkov na základe rôznych kritérií (či pomohol používateľovi, či bol aktuálny, atď.)
 - e. možnosť nahlásenia plagiátorstva
2. podpora odovzdania článkov
 - a. vloženie článkov ako súborov rôzneho formátu (pdf, doc, html)
 - b. podpora pridania abstraktu vo viacerých jazykoch
 - c. podpora uvádzania zdrojov v rôznych formátoch
 - d. možnosť posielania správ autorovi o posudkoch alebo príspevkoch pridaných k článku
 - e. možnosť písania článku alebo príspevkov priamo na portáli v textovom WYSWYG editore.
3. podpora posudzovania článkov
 - a. možnosť zadávania posudkov
 - b. možnosť písania posudkov priamo na portáli v textovom WYSWYG editore
 - c. možnosť vedenia diskusie s autorom alebo autormi k nejakému článku
4. elektronická knižnica
 - a. možnosť prehľadávania v publikáciách na základe viacerých kritérií (autor, názov, kľúčové slová, atď.)

- b. zoradovanie výsledkov vyhľadávania na základe relevantnosti voči dotazu používateľa alebo na základe iných kritérií (dátum vloženia, autor, názov atď.)
- c. zobrazenie publikácie on-line, bez stiahnutia samotných súborov

Pri návrhu systému sa budeme snažiť vytvoriť aplikáciu čo najmodulárnejšiu tak, aby sa jednoducho dalo pridať ďalšie moduly alebo nahradiť existujúce novšími alebo efektívnejšími.

Používatelia systému

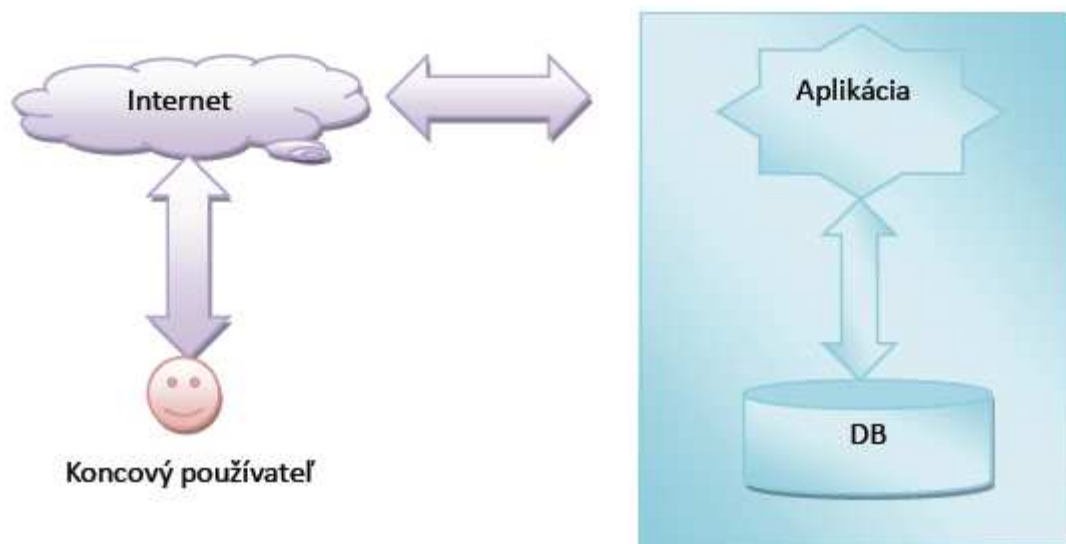
Systém je vytvorený pre širšiu verejnosť, pričom musíme rozlišovať rôzne typy používateľov. Presné určenie práv používateľov bude vyžadovať ďalšie analýzy a konzultácie s vedúcim projektu. Predbežne sme identifikovali nasledujúce skupiny používateľov:

1. Anonymný používateľ – portál bude využívať ako zdroj zverejnených dát
2. Autor – používateľ, ktorý môže odovzdať články a môže ich aj posudzovať.
3. Administrátori – používatelia, ktorí kontrolujú pribúdajúce články, príspevky alebo posudky.

Architektúra systému

Systém pozostáva z troch hlavných častí:

1. Prehliadač používateľa – na jeho počítači vzdialeného od servera
2. Komunikačný kanál – Internet
3. Server – v sebe zahŕňa dátovú (databáza) a aplikačnú vrstvu, je jadrom celého systému.



Používateľské rozhranie

Aj z vlastných skúseností vieme, že zobrazenie nadbytočných prvkov (neaktívne položky v menu, deaktivované polia vo formulároch) môže frustrovať používateľov, ako aj to, keď jednotlivé operácie sú hlboko vnárané v rôznych menu. Práve preto bude používateľské rozhranie pozostávať zo spoločných položiek a z položiek špecifických pre rôzne typy používateľov.

Z hľadiska dostupností informácií, stránka bude dvojúrovňová:

1. verejná úroveň
2. privátna úroveň – vyžaduje autentifikáciu používateľov

5. Príloha A: Poradie tém podľa preferencií (Bednár)

1. Simulated Car Racing Competition 2011 (Car Racing)
2. Model používateľa pre jeho identifikáciu (UserModel)
3. Portál pre časopis (Časopis)
4. RoboCup_ tretí rozmer (RoboCup 3D)
5. Tréner mentálnych schopností (Tréner)
6. Správa študentských projektov na fakulte (Projekty)
7. Crowdsourcing verejných dát (CrowdPublic)
8. Evolučný simulátor umelého života založený na heuristických pravidlách (HERBAL)
9. Platforma pre realizovanie transakcií prostredníctvom mobilných zariadení (Mobily)
10. Objektové úložisko dát (Úložisko)
11. Virtuálna FIIT (VFIIT)
12. Prispôsobiteľný Widget (Widget)
13. Vyhľadávanie a sprístupnenie citácií_(Portál)
14. Dizajn s použitím obohatenej reality (Dizajn)
15. Adaptívny proxy server (Proxy-plugins)
16. 3D grafická podpora vyhľadávania znalostí v dokumentoch (3D-Znalosti)
17. Tvorba rozvrhov (Rozvrhy)
18. Interaktívna vizualizácia grafových štruktúr v 3D priestore (Vizualizácia)
19. Imagine Cup 2011: Game Design (ICup2011)

6. Príloha B: Aktuálny rozvrh členov tímu (Bednár, Valeník)

		7:00	8:00	9:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	20:00	21:00
Pondelok	Bednár Maroš								Pokr. databáz. tech.		Tímový projekt I		Wyskum soft. syst.			
	Brček Adam										Tímový projekt I		Wyskum soft. syst.			
	Bríš Marek								Pokr. databáz. tech.	Zákl. kryptografie	Tímový projekt I		Wyskum info. syst.			
	Flórek Marian								Pokr. databáz. tech.	Zákl. kryptografie	Tímový projekt I		Wyskum info. syst.			
	Juhász Vojtech									Pokr. databáz. tech.	Tímový projekt I		Wyskum info. syst.			
	Kosmeľ Juraj									Pokr. databáz. tech.	Zákl. kryptografie	Tímový projekt I		Wyskum soft. syst.		
Utorok	Bednár Maroš															
	Brček Adam															
	Bríš Marek															
	Flórek Marian															
	Juhász Vojtech															
	Kosmeľ Juraj															
Streda	Bednár Maroš															
	Brček Adam															
	Bríš Marek															
	Flórek Marian															
	Juhász Vojtech															
	Kosmeľ Juraj															
Štvrtok	Bednár Maroš															
	Brček Adam															
	Bríš Marek															
	Flórek Marian															
	Juhász Vojtech															
	Kosmeľ Juraj															
Piatok	Bednár Maroš															
	Brček Adam															
	Bríš Marek															
	Flórek Marian															
	Juhász Vojtech															
	Kosmeľ Juraj															

* cvičenia z predmetu Neurónové siete by som v prípade potreby mohol presunúť o dve hodiny skôr

Belasou farbou sú vyznačené školské povinnosti.

Zlatou farbou sú vyznačené časy mimoškolských aktivít.

Zelenou farbou sú vyznačené najvhodnejšie časy na stretnutia.

D) Plány projektu (Juhász)

D.1 Tvorba plánov

Hrubý plán projektu sa vytvára od tretieho týždňa zimného semestra, po získaní konkrétnej témy tímového projektu. Po vytvorení hrubého plánu sú každý týždeň pridelené úlohy jednotlivým členom tímu a zároveň je kontrolované splnenie úloh z predchádzajúceho týždňa. Plnenie plánu je kontrolované každý štvrtý týždeň, t.j. v siedmom a v jedenástom týždni zimného semestra.

D.2 Externé termíny

Externé termíny sú všetky obmedzujúce záväzné termíny, ktoré sú dané organizáciou predmetu. Externé termíny sú zhrnuté v tabuľke č. 1.

<i>Semester</i>	<i>Udalosť</i>	<i>Termín</i>
<i>Zimný</i>	odovzdanie dokumentácie analýzy problému, špecifikácie požiadaviek a návrhu riešenia	7. týždeň
<i>Zimný</i>	odovzdanie prototypu vybraných častí systému spolu s dokumentáciou a používateľská prezentácia prototypu	12. týždeň / do 14.12.2010
<i>Zimný</i>	používateľská prezentácia prototypu	15 – 21.12.2010
<i>Letný</i>	odovzdanie produktu a dokumentácie k produktu (potrebnej pre používanie produktu)	9. týždeň
<i>Letný</i>	odovzdanie celkového výsledku projektu	12. týždeň
<i>Letný</i>	prezentácia projektov na IIT.SRC - TP CUP 2011	Apríl 2011
<i>Letný</i>	odovzdanie celkového výsledku projektu	10.5. 2011 (utorok), 13.00

Tabuľka č. 1 Externé záväzné termíny.

D.3 Hrubý plán projektu na zimný semester

3. týždeň:

- Analýza problémovej oblasti. Podrobná analýza TORCS framework-u a oponentov.
- Vytvorenie web prezentácie tímového projektu.
- Vytvorenie plagátu a loga tímu.

4. týždeň:

- Analýza tímov z predchádzajúcich rokov.
- Analýza alternatívnych riešení na určenie optimálnej jazdnej dráhy.

5. týždeň:

- Vyhodnotenie analyzovaných riešení.
- Návrh architektúry autopilota.
- Určenie funkcionality jednotlivých navrhnutých modulov.

6. týždeň:

- Implementácia základného autopilota.
- Integrácia modulu obnovenia do základného pilota.
- Návrh modulu na vytvorenie modelu trate.

7. týždeň:

- Implementácia základného modulu na vytvorenie modelu trate.
- *Odovzdanie dokumentácie k riadeniu.*
- *Odovzdanie dokumentácie obsahujúcej analýzu problému, špecifikáciu požiadaviek a návrh riešenia.*
- Kontrola splnenia plánu zostaveného v treťom týždni, zostavenie modifikovaného plánu.

8. týždeň:

- Vypracovanie spôsobu určenia zákrut a rovných úsekov na základe modelu trate.
- Kontrola integrácie a prípadné doladovanie modulu obnovenia.
- Návrh modulu detekcie oponentov.
- Revízia plánu a vytvorenie zjemneného plánu na nasledujúce štyri týždne.

9. týždeň:

- Implementácia modulu detekcie oponentov.

- Návrh modulu Defense na základe navrhnutého modulu detekcie oponentov.
- Kategorizácia zákrut a určenie stratégie prechádzania cez ne.

10. týždeň:

- Návrh modulu predbiehania na základe navrhnutého modulu detekcie oponentov.
- Implementácia modulu asistenta trate (Line Assist) .
- Implementácia modulu Defense.

11. týždeň:

- Integrácia a prípadné ladenie asistenta trate.
- Integrácia a prípadné ladenie modulu Defense.
- Kontrola splnenia plánu zostaveného v siedmom týždni, zostavenie modifikovaného plánu.

12. týždeň:

- Testovanie riešenia: asistent trate, modul detekcie oponentov, modul Defense, modul predbiehania.
- Revízia plánu a vytvorenie hrubého plánu na letný semester.
- Zhodnotenie stavu vývoja auto-pilota.

13. týždeň:

- odovzdanie prototypu vybraných častí systému spolu s dokumentáciou.

D.4 Revízia plánu v siedmom týždni

V tejto kapitole si zosumarizujeme splnenie úloh z prvých štyroch týždňov. Následne je navrhnutý plán na nasledujúce štyri týždne semestra (Tabuľka č. 2).

Počas predchádzajúcich štyroch týždňov sme úspešne analyzovali existujúce riešenia a navrhli sme architektúru vlastného riešenia. Po identifikácii jednotlivých modulov sa určili ich priority a boli porozdeľované medzi jednotlivých členov tímu. Následne sme na základe navrhnutej architektúry vytvorili základného autopilota.

<i>Termín</i>	<i>Úloha</i>
9. týždeň	Implementácia ABS, ASR.
9. týždeň	Kategorizácia zákrut.

10. týždeň	Implementácia plynulého zatáčania.
10. týždeň	Vytvorenie sofistikovaného modelu trate.
11. týždeň	Vypracovanie a implementácia stratégie prechádzania zákrut – LINE ASSIST.
11. týždeň	Ladenie modulu zotavenia – RECOVERY.
11. týždeň	Implementácia modulu obrany – DEFENCE: <ul style="list-style-type: none"> • detekcia oponentov • blokovanie oponentov • predbiehanie
11. týždeň	Zdokumentovať implementované moduly.

Tabuľka č. 2 Plán pre nasledujúce štyri týždne.

D.5 Revízia plánu v jedenástom týždni

V tabuľke č. 3 je zobrazené zhodnotenie úloh od deviateho týždňa. Následne je navrhnutý plán na posledné dva týždne zimného semestra (Tabuľka č. 4).

Úloha	Stav
<i>Implementácia ABS, ASR</i>	splnená na čas
<i>Kategorizácia zákrut</i>	rozpracované
<i>Implementácia plynulého zatáčania</i>	splnená na čas
<i>Vytvorenie sofistikovaného modelu trate</i>	zamietnuté – metóda sa po pridaní šumu ukázala byť nepoužiteľná
<i>Vypracovanie a implementácia stratégie prechádzania zákrut – LINE ASSIST</i>	splnené na čas

<i>Ladenie modulu zotavenia RECOVERY</i>	– rozpracované
<i>Implementácia modulu obrany DEFENCE:</i>	– čiastočne splnené – bola implementovaná detekcia oponentov a spustenie blokovania alebo predbiehania oponentov. Blokovanie je implementované reakčne, nezohľadňujúc informácie o trase. Implementácia predbiehania sa odložila na letný semester.
<i>Zdokumentovať moduly</i>	<i>implementované</i> nespĺnené

Tabuľka č. 3 Zhodnotenie úloh od deviateho týždňa

<i>Termín</i>	<i>Úloha</i>
<i>12. týždeň</i>	vytvorenie technických dokumentácií jednotlivých modulov
<i>13. týždeň</i>	doladiť chyby modulu RECOVERY
<i>13. týždeň</i>	optimalizácia prechádzania zákrut – LINE ASSIST
<i>13. týždeň</i>	zhodnotenie stavu projektu
<i>13. týždeň</i>	vytvorenie plánu na letný semester
<i>13. týždeň</i>	pripraviť používateľskú prezentáciu projektu

Tabuľka č. 4 Plán na posledné dva týždne zimného semestra

E) Manažment rizík

E.1 Analýza rizík

Dátum poslednej revízie analýzy rizík: 6. decembra 2010

Identifikované riziká sú rozdelené do 3 troch pravdepodobnostných skupín:

- malá pravdepodobnosť (M) – neočakáva sa vznik situácie, pri ktorej by došlo k naplneniu rizika. Riziko je však reálne prítomné a jeho naplnenie nie je vylúčené.
- stredná pravdepodobnosť (S) – riziko je v projekte reálne prítomné a jeho naplnenie sa je možné.
- vysoká pravdepodobnosť (V) – naplnenie rizika je veľmi pravdepodobné.

Dôsledky identifikovaných rizík sú rozdelené podľa ich dopadu na čas a výstup. Obidve tieto oblasti sú rozdelené na 3 kategórie. Rozdelenie dôsledkov identifikovaných rizík podľa dopadu na čas:

- malý dopad (T1) – naplnenie rizika spôsobí malé predĺženie plánovaných prác na projekte. Predĺženie nebude vyššie ako 10% pôvodne plánovaného trvania.
- stredný dopad (T2) – naplnenie rizika spôsobí stredné predĺženie plánovaných prác na projekte. Predĺženie nebude vyššie ako 25% pôvodne plánovaného trvania.
- veľký dopad (T3) – naplnenie rizika spôsobí veľké predĺženie plánovaných prác na projekte. Predĺženie bude vyššie ako 25% pôvodne plánovaného trvania.

Všetky vyššie uvedené kategórie môžu byť označené symbolom plus (+) alebo mínus (-), ktoré predstavujú trend vývoja pravdepodobnosti naplnenia daného rizika.

Rozdelenie identifikovaných rizík podľa dopadu na výstup projektu:

- malý dopad (P1) – naplnenie rizika spôsobí malé nedostatky vo výstupe projektu. Spôsobené nedostatky nebudú mať vplyv na celkový výstup projektu.

- stredný dopad (P2) – naplnenie rizika spôsobí stredné nedostatky vo výstupe projektu. Spôsobené nedostatky môžu mať vplyv na funkčnosť vytvoreného diela alebo na kvalitu iných výstupov.
- veľký dopad (P3) – naplnenie rizika spôsobí veľké nedostatky vo výstupe projektu. Spôsobené nedostatky budú mať vplyv na celkový výstup projektu a citelne ovplyvnia funkčnosť vytvoreného diela alebo kvality iných výstupov.

Zodpovedné osoby za riziká majú povinnosť dohliadať na aplikovanie zvoleného postupu manažovanie rizika a sledovať ďalší vývoj rizika. V prípade potreby je ich povinnosťou iniciovať preradenie rizika do inej kategórie, prípadne odstránenie rizika zo zoznam, pokiaľ sa mu už ďalej nie je potrebné špeciálne venovať.

Priorita riešenia rizík vyplýva z kombinácie pravdepodobnosti naplnenia rizika a závažnosti jeho dôsledkov. Všetky riziká uvedené v tabuľke je potrebné riadiť.

E.2 Identifikované riziká

Pri analýze rizík boli identifikované a kategorizované nasledujúce riziká:

Číslo rizika	Popis rizika	Pravdepo – dobnosť	Dôsledok	Zodpovedná osoba	Stav
1	Riziko podcenenia vplyvu šumu	M	P2	Adam Brček	sledované
	Do senzorov, ktoré auto využíva na riadenie, je vnášaný šum. Rizikom je podcenenie jeho vplyvu na správanie sa auta a nesprávne fungovanie zvolených techník jazdy.				
2	Zvolenie nevykonateľného prístupu	S	T3	Ivan Valenčík	sledované
	Rizikom je zvolenie techniky riadenia auta, ktorú nie je možné implementovať. Dôvodom môže byť teoretická nemožnosť použiť zamýšľaný prístup, ktorá môže byť odhalená až pri implementácii alebo prílišná náročnosť implementácie.				
3	Nezvládnutie komunikácie	M+	T2	Ivan Valenčík	sledované
	Nepochopenie svojich úloh v kontexte úloh ďalších členov tímu, zdvojenie				

	práce na úlohe, nezvládnutie koordinácie pri spoločne vypracováanej úlohe a pod. Riziko sa stáva aktuálnejšie pri potrebe komunikovať s niektorým(-i) len na diaľku.				
4	Pozlátenie systému – vytváranie nepotrebných funkcií	S+	T2	Ivan Valenčík	sledované
	Práca na funkcionalite, ktorá neprispieva k naplneniu cieľov projektu. Nie všetko, čo nie je súčasťou vytváraných ovládaní, je zbytočné. Niektoré nástroje môžu mať význam pre vývoj.				
5	Potreba zmeny základného princípu ovládača jazdy	M	T3	Ivan Valenčík	sledované
	Zmeniť základný princíp ovládania by mohlo mať veľmi výrazné následky, viaceré vytvorené komponenty by sa mohli stať nepoužiteľnými.				
6	Potreba zmeny už rozpracovaných komponentov	S-	T2	Ivan Valenčík	sledované
	Potreba zmeny rozpracovaných komponentov znehodnocuje doteraz vykonanú prácu a môže si vyžiadať zmeny aj v iných častiach systému.				
7	Nerealistický rozvrh	S	T1-T3, P2	Vojtech Juhász	sledované
	Nerealistické odhady dĺžky trvania napĺňania úloh môžu mať za následok neschopnosť členov tímu splniť svoje v úlohy v požadovanom rozsahu a kvalite.				
8	Nevhodné obsadenie pozícií v tíme	S	T1-T2, P1-P2	Adam Brček	sledované
	Pridelenie nevhodných riešiteľov k úlohám má za následok neoptimálne riešenie. Plnenie úloh trvá dlhšie a výsledný produkt môže byť nižšej kvality, ako by mohol byť od iného riešiteľa.				

F) Úlohy členov tímu (Juhász)

F.1 Dlhodobé roly členov tímu

Na spoločnom stretnutí sme si rozdelili jednotlivé manažérske pozície v rámci tímu a to nasledovne:

- **Maroš Bednár:** zástupca vedúceho tímu, vývoj a správa webu, manažér testovania
- **Adam Brček:** vedúci tímu, pridelovanie úloh a kontrola ich splnenia
- **Marek Briš:** manažér podporných činností
- **Marian Florek:** manažér vývoja
- **Vojtech Juhász:** manažér plánovania, tvorba dlhodobých a krátkodobých plánov
- **Juraj Kosmeľ:** manažér kvality, definovanie štandardov
- **Ivan Valenčík:** manažér rizík

F.2 Krátkodobé úlohy členov tímu

Meno	Úloha
Maroš Bednár	tvorba web prezentácie tímu
	aktualizácia web prezentácie tímu
	analýza pilotov z predchádzajúcich rokov: Luigi a Onieva&Pelta
	návrh modulov ABS a ASR
	dokumentácia návrhu riešenia
	implementácia modulu ABS
	implementácia modulu ASR
	dokumentácia modulov ABS a ASR
	úprava zápisníc
	dokumentácia k riadeniu verzia 2 – zimný semester
návrh korekcie riadenia po skoku	
Adam Brček	návrh sofistikovaného modelu trate
	analýza možností využitia setup-u v championship-e
	analýza a návrh metódy na zistenie najvyššej bezpečnej rýchlosti
	dokumentácia analýzy TORCS
	návrh telemetrie
	implementácia telemetrie
	dokumentácia telemetrie
	dokumentácia modulu trate - časť
Marek Briš	analýza pilotov z predchádzajúcich rokov: AUTOPIA a Cardamone
	implementácia modulu obnovenia
	vytvorenie loga tímu

	metodika SVN
	dokumentácia modulu obnovenia
	kontaktovať organizátorov
	návrh korekcie riadenia po skoku
Marián Florek	analýza pilotov z predchádzajúcich rokov: COBOSTAR a Mr Racer
	návrh modulu obnovenia
	dokumentácia analýzy minuloročných tímov
	návrh a implementácia plynulého zatáčania
	návrh a implementácia určenia hodnoty frikcie
	vytvorenie riadenia pre prejazd prvého kola
	návrh funkcie na odhad maximálnej rýchlosti vstupu do zákruty
	dokumentácia modulu štartu
	dokumentácia modulu riadenia a radenia
Vojtech Juhász	dokumentácia k riadeniu
	analýza pilotov z predchádzajúcich rokov: Joseph Alton a Jorge Munoz
	metodika plánovania
	metodika manažovania chýb
	návrh modulu detegovania oponentov
	implementácia modulu detegovania oponentov
	dokumentácia modulu detegovania oponentov
Juraj Kosmeľ	analýza pilotov z predchádzajúcich rokov: Tim Alford a Neil Clarke
	dokumentácia špecifikácie riešenia
	metodika komunikácie
	navrhnuť modul line assist
	implementácia modulu line assist
	dokumentácia modulu line assist
Ivan Valenčík	analýza súťaže demolition derby
	návrh modulu trate
	metodika písania zdrojového kódu
	implementácia modulu trate
	vytvorenie prezentácie
	Z integrovania modulu trate s riadením pre prvé kolo
	dokumentácia modulu trate - časť

G) Podporné prostriedky a metodiky práce

G.1 Práca s SVN (Briš)

Manažment verzií

Počas vývoja projektu používame rôzne dokumenty a je potrebné, aby boli dostupné pre každého člena tímu a aby k nim mohol každý jednoducho pristupovať, preto sú situované na SVN.

Ide o dokumenty typu:

- Analýzy účastníkov súťaží z predchádzajúcich rokov
- Manuály k prostrediu nasadenia
- Výsledky čiastkových úloh (návrhy architektúr pilota)
- Projektová dokumentácia a dokumentácia k riadeniu projektu (vrátane samostatných častí od rôznych členov tímu)

Hlavnou podstatou je úložisko pre zdrojové úbory. Zdrojové kódy, ktoré sú predmetom implementácie potrebujeme mať neustále k dispozícii a samozrejme v aktuálnej podobe. Nasledujúca metodika popisuje prácu so zvoleným klientom.

TortoiseSVN

Z dostupných riešení sme sa rozhodli použiť jeden z klientov *Subversion* a to konkrétne *TortoiseSVN*. Používame nasledovnú funkcionality:

- Správu súborov a adresárov uložených v centralizovanom úložisku (v našom konkrétnom prípade sa centrálnym úložiskom stal server `code.google.com`)
 - Aktualizácia lokálneho úložiska (angl. Update)
 - Odovzdanie modifikovaných súborov centrálnemu úložisku (angl. Commit)
 - Zamykanie a odomykanie súborov
- Pamätanie si každej zmeny (vhodné pre detekovanie dobrých a horších zmien)
- Návrat k predchádzajúcim verziám (v prípade, že sa neuberáme správnou cestou)

Centralizované úložisko

Server: <https://carsimulationraceteam20.googlecode.com/svn/>

Štruktúra úložiska:

Nasledovná štruktúra predstavuje úložisko, základných 5 adresárov, v ktorých sú ďalšie súbory či podadresáre

- src (súbory .java)
 - champ2010client (prevzatá kostra)
 - controller (naša implementácia)
- bin (súbory .class)
- Doc
 - JavaDoc (vygenerovaný)
- Documents
 - .pdf
 - .doc
 - .png
 - ...
- OthersSolutions (ostatné riešenia)
 - ChinHiongTan
 - DiegoPerez
 - LeonardKinnaird-Heether
 - ...

Aktualizácia lokálneho (pracovného) úložiska – Update

Prečo nepodceniť aktualizáciu lokálneho úložiska:

- Cieľom je pracovať vždy s aktuálnou verziou centrálného úložiska
 - Čiastočne predídeme kolíziám
 - Neaktuálne lokálne úložisko je hrozbou, pretože nami vykonané zmeny často úzko súvisia so zmenami, o ktorých žiaľ nevieme

Odovzdanie modifikovaných súborov centrálnemu úložisku – Commit

- Zmeny menšieho charakteru je potrebné odovzdávať maximálne na konci dňa
- Odovzdanie vykonať po ucelenej zmene

Samotné odovzdanie zmenených súborov

V prípade dokončenia úprav je potrebné odovzdať ich centrálnemu úložisku zvolením *Commit*. Každé odovzdanie si vyžaduje aj určitú správu o tom, aké zmeny nastali a čoho sa dotýkajú.

V správe uviesť:

- Stručné zhrnutie zmien uvedené ako „názov“ správy (text - tučné)
 - maximálne jeden riadok
- Podrobnejšie zhrnutie zmien
 - podrobnejšie o zmenách (ak je to potrebné)
 - v odrážkach
- Ktorý problém rieši (angl. Issue) (text - kurzíva)
 - napr.: Toto rieši problém #55.

Neuvádzať: (explicitne uvedie TortoiseSVN)

- Autora
- Dátum odovzdávania
- Číslo revízie
- Súbor, ktorých sa zmeny dotýkajú

Na formátovanie textu použiť:

- **text** -> pre **tučné**
- *^text^* -> pre *kurzívu*

Zamykanie a odomykanie súborov

- Nepoužívať!
- Server nepodporuje túto funkcionality

G.2 Komunikácia (Kosmel)

Pri realizácii nášho projektu je potrebné diskutovať a riešiť rôzne problémy, zdieľať a archivovať zdrojové kódy ako aj rôzne materiály použité na štúdium a analýzu danej problematiky a zároveň plánovať a kontrolovať vykonávanie úloh členmi tímu. Okrem osobnej komunikácie na tímových stretnutiach využívame v našom tíme viaceré podporné prostriedky:

- E-mail
 - najmä komunikácia väčšieho rozsahu
 - notifikácia o pridelených úlohách a ich stave

- ICQ
 - interaktívna komunikácia
 - zväčša medzi dvoma členmi
 - okamžité riešenie problémov a získanie informácií
- Google Groups
 - diskusné fórum
 - zdieľanie menších súborov
 - spoločný tímový mail
- Google Code
 - riadenie tímu
 - pridelovanie úloh
 - verziovanie zdrojových kódov
 - wiki
- Google Calendar
 - prehľad dôležitých termínov

Voľba emailu, ako v súčasnosti najrozšírenejšej formy komunikácie, bola samozrejmosťou. Ako podporu emailovej komunikácie sme zvolili klienta ICQ, ktorý používali a poznali všetci členovia tímu. Tieto prostriedky využívame na neformálnu komunikáciu, email aj na komunikáciu s vedúcim a riešiteľmi podobných projektov zo zahraničia, naopak ICQ najmä na interaktívnu komunikáciu, keď bolo potrebné okamžité vyjadrenie a podanie informácie.

Služba Google Groups je využívaná najmä ako diskusné fórum, keď je potrebné vyjadrenie všetkých členov tímu na nejakú tému ohľadom riešenia projektu. Taktiež je výhodou tejto služby spoločný mail, ktorý slúži na komunikáciu tímu ako celku.

Google Code slúži na plánovanie a riadenie činnosti nášho tímu a taktiež na správu, verziovanie a archiváciu vytvorených zdrojových kódov. Poskytuje prehľad o vykonanej práci tímu ako aj plánovaných aktivitách.

G.3 Pravidlá písania a komentovania zdrojového kódu (Valenčík)

Účelom tejto metodiky je zdefinovanie pravidiel pre komentovanie zdrojového kódu za účelom generovania technickej dokumentácie pomocou nástroja Javadoc. Jej obsahom sú pravidlá a odporúčania, ktoré by mali byť dodržané vo všetkých vytvorených súboroch obsahujúcich java zdrojový kód. Metodika predpokladá, že pri vývoji je používané prostredie Eclipse, avšak toto nie je podmienkou.

Dodržanie metodík, ktoré sú označené ako pravidlá alebo sú uvedené slovesom "musieť", je považované za nevyhnutné. Metodiky, ktoré sú označené ako odporúčania alebo sú uvedené slovesom "odporúča sa" alebo "malo by byť", nie je nutné naplniť. Rozhodnutie o tom, kedy odporúčanie nasledovať, je zodpovednosťou programátora zodpovedného za danú časť kódu. Účelom odporúčaní je zvýšenie prehľadnosti kódu pri práci s ním, nie tvorba dokumentácie.

Použité pojmy

- Eclipse – vývojové prostredie
- Java - programovací jazyk navrhnutý spoločnosťou Sun Microsystems (v súčasnosti vlastnená spoločnosťou Oracle Corporation)
- Javadoc - generátor dokumentácie v HTML z java zdrojového kódu od spoločnosti Sun Microsystems

Zodpovednosť za kód

Každý súbor s java kódom musí mať svojho vlastníka, ktorý je zodpovedný za udržiavanie kódu. Pod udržiavaním sa myslí hlavne udržiavanie úpravy kódu, jeho čitateľnosti, konzistentnosti a správnej formy dokumentácie. Zodpovednosť za kód nemusí byť nutne formálne zaznamenaná. Pokiaľ súbor nie je označený, tak zodpovednou osobou je autor prvej verzie súboru, resp. programátor, ktorý kód prvý prebral z externého zdroja. Pri prenose zodpovednosti by mal byť upovedomený celý tím. Vedúci tímu a programátori, medzi ktorými je zodpovednosť prenášaná, musia byť s touto skutočnosťou oboznámený nepodmienečne.

Zodpovednosť za kód môže byť zaznačená v komentári na prvom riadku súboru. Tento riadok obsahuje názov súboru za ktorým nasleduje meno zodpovedného programátora. Príklad: // source.java Ferdinand Carrot

Tvorba dokumentácie

Dokumentácia je tvorená autorom zdrojového kódu, ktorý ju musí priebežne vytvárať. Pokiaľ sú v kóde vykonávané zmeny, tak úpravu dotknutej dokumentácie musí spraviť autor zmien. Zmeny v dokumentácii by mali byť skontrolované aj autorom pôvodnej verzie kódu.

Dokumentácia môže byť vygenerovaná s rôznym stupňom viditeľnosti. Nezávisle od tejto skutočnosti stále platí pravidlo, že zdokumentované musia byť všetky prvky, nezávisle od použitého stupňa viditeľnosti.

Názvy

Všetky názvy tried, metód a atribútov musia byť v anglickom jazyku a čo najvýstižnejšie pomenovávať daný objekt. Na zápis názvov je použitý prístup camelCase, čiže názvy zložené z viacerých slov sa oddeľujú veľkým písmenom každého nasledujúceho slova a medzi slovami nie sú medzery. Skratka je považovaná za jedno slovo.

Názvy tried

Názov triedy vždy začína veľkým písmenom a musí byť podstatné meno vhodne popisujúce obsah danej triedy. Príklad: SampleClass

Názvy rozhraní

Názov rozhrania vždy začína veľkým písmenom a musí byť prídavné meno vhodne popisujúce účel rozhrania. Príklad: Exemplary

Názvy metód

Názov metódy sa musí začínať malým písmenom a prvé slovo metódy je sloveso. Názov metódy vhodne popisuje účel metódy. Je odporúčané, aby názvy metód s podobným účelom odrážali ich príbuznosť, napr. všetky metódy slúžiace na výpočet údaju budú začínať "compute", nikdy nie synonymom, napr. "calculate". Pri nasledujúcich typoch metód je použitie určitého slovesa na začiatku metódy pravidlom:

- metódy vracajúce typ boolean, ktorý vraví, či objekt má istú vlastnosť začínajú "is"
- metódy nastavujúce hodnotu zapuzdreného atribútu začínajú "set"
- metódy slúžiace na získanie hodnoty zapuzdreného atribútu začínajú "get"

Názvy atribútov

Názov atribútu sa musí začínať malým písmenom a mal by byť dostatočne popisný. Názov popisuje na čo slúži daný atribút, pričom je možné použiť aj skratky. Názov konštanty sa skladá iba z veľkých písmen. V prípade viacslovného názvu sú slová oddelené podtržníkom. Príklad: counter, i, CONST_ARG

Názvy balíkov

Názov balíku sa musí skladať iba z malých písmen. Názov zodpovedá triedam, ktoré obsahuje. V prípade, že je to možné, by mal názov obsahovať doménu príslušiacu autorovi kódu, aby sa predišlo možným konfliktom. Príklad: sk.stuba.fiit.project

Komentáre

Okomentované musia byť všetky java zdrojové kódy. Zdokumentované musia byť všetky triedy, rozhrania a metódy, a to pomocou štýlu komentárov určeného pre Javadoc. Všetky komentáre sú písané v slovenskom jazyku bez použitia diakritiky.

Komentáre tried

V komentároch k triedam musí byť zdokumentovaný účel danej triedy. Ďalej musí byť uvedený spôsob práce s triedou, pokiaľ nie je zrejmý z jej metód alebo uvedeného účelu. Silno odporúčané je uviesť aj jej špeciálne vlastnosti, napr. či je immutable.

V komentári tried môže byť uvedený autor, resp. autori, a to použitím alebo opakovaným použitím tagu "@author". Využitie tagu "@version" sa vo všeobecnosti neodporúča, správu verzií má na starosti externý program. Použitie tohto tagu môže požadovať programátor zodpovedný za daný kód. V tomto prípade je verziovanie povinné a jeho pravidlá určí zmienený zodpovedný programátor.

Pri komentovaní triedy, ktorá je implementáciou rozhrania alebo abstraktnej triedy vytvorenej v projekte je možné použiť dedenie komentáru zo spomenutého rozhrania alebo abstraktnej triedy. Nový komentár je nutné poskytnúť, iba ak sa funkcionality odkláňa od pôvodne uvedenej.

Komentáre rozhraní

V komentároch k rozhraniám musí byť zdokumentovaný účel daného rozhrania triedy. V komentári rozhrania môže byť uvedený autor, resp. autori, a to použitím alebo opakovaným použitím tagu "@author". Využitie tagu "@version" sa vo všeobecnosti neodporúča, správu verzií má na starosti externý program. Použitie

tohto tagu môže požadovať programátor zodpovedný za daný kód. V tomto prípade je verziovanie povinné a jeho pravidlá určí zmienený zodpovedný programátor.

Komentáre metód

V komentároch metód musí byť opis činnosti, ktorú metóda vykonáva. Opísané musia byť všetky jej argumenty (pomocou tagu "@param"), návratová hodnota (pomocou tagu "@return") a výnimky, ktoré metóda vyhadzuje (pomocou tagu "@throws"). Pre argumenty musí byť zrejmé, čo má byť obsahom vstupných dát. Z popisu návratovej hodnoty musí byť jasné čo predstavuje a aká je v prípade možného neúspechu metódy. Komentár metódy obsahuje aj popis okolností, za ktorých dôjde k vyhodneniu možných výnimiek.

Komentáre atribútov

Účel atribútov triedy by mal byť zrejmý z ich názvov, takže ich komentovanie nie je v tomto prípade nutné. Komentovať atribút prostredníctvom javadoc komentára sa odporúča len v prípade, že je jeho použitie nejasné.

Konštanty, ktoré boli určené experimentálne alebo je ich pôvod nejasný (napr. fyzikálne konštanty, ale nie hodnota Pi), musia byť riadne okomentované.

Iné komentáre

Okrem technickej dokumentácie vygenerovanej prostredníctvom Javadoc má dokumentačnú hodnotu aj samotný kód. Okrem komentárov využívajúcich javadoc štýl komentovania sa od programátora očakáva, že bude komentovať aj jednotlivé bloky kódu pomocou jednoriadkových, resp. viacriadkových komentárov.

Je odporúčané doplniť každý úsek kódu o krátke vysvetlenie, čo je jeho účelom. Ďalej je silno odporúčané pri blokoch kódu, v ktorých sa nachádzajú matematické výrazy, goniometria, fyzikálne výpočty a pod., podrobnejšie uviesť čo je vstupom a výstupom danej operácie a pokúsiť sa stručne opísať jej priebeh.

V časti kódu, na ktorej je potrebná ďalšia práca alebo nie je zatiaľ implementovaná, je vhodné v popise uviesť označenie TODO nasledované krátkym popisom nedostatku. Príklad: // TODO rozšíriť funkcionálnu

V časti kódu, kde bola zistená chyba, je vhodné v popise uviesť označenie FIXME nasledované krátkym popisom nedostatku. Príklad: // FIXME nezvláda veľké vstupy

G.4 Manažovanie chýb (Juhász)

Pri tvorbe jednoduchého programu v rámci zložitého softvérového systému si môžeme byť istí, že počas vývoja sa budeme stretávať s chybami. Tieto chyby je potrebné správne manažovať. Rozlišujeme dva typy chýb, ktoré sa nahlasujú nasledovne:

1. *Defect report from user*: chyby zistené pri testovaní, ktoré vedú k neočakávaným výsledkom, ale neindikujú chybový stav (t.j. padanie programu)
2. *Defect report from developer*: chyby zistené pri vývoji pri ktorých dochádza k chybovému stavu

Jednotlivé chybové správy musia obsahovať nasledujúce informácie:

- a. Popis chyby: čo sa očakávalo (stav objektu alebo hodnota výstupu) a čo nastalo (chybová správa, logy programu)
- b. Presný popis miesta výskytu chyby: názov súboru resp. objektu, názov funkcie, číslo riadku
- c. Popis okolností: vstupné parametre funkcie resp. stav objektu
- d. Ako problém reprodukovat'

Každá chyba sa nahlasuje cez ticketovací systém tímu na *code.google.com* vedúcemu tímu. Vedúci tímu následne túto správu vyhodnotí a vzniknutú úlohu rozdelí medzi členov tímu.

Životný cyklus spracovanie vzniknutých chýb v ticketovacom systéme:

1. Hlásenie chyby vedúcemu tímu
 - a. stav správy: *New*
 - b. štítky správy (labels): *Type-Defect*, priorita podľa dôležitosti
2. Vedúci tímu vyhodnotí chybu
 - a. stav správy: *Accepted*
3. Vedúci tímu rozdeľuje úlohy na riešenie chyby
 - a. stav správy: *Started*
 - b. stav nových úloh: *New*
 - c. štítky nových úloh: *Type-Defect*, priorita podľa dôležitosti
4. Členovia tímu začnú pracovať na riešení chýb
 - a. stav nových úloh: *Accepted*
5. Členovia tímu dokončili svoju úlohu

- a. stav nových úloh: *Done*
- 6. Každá čiastková úloha je dokončená
 - a. stav správy: *Fixed*

Vedúci tímu sa môže rozhodnúť, že nahlásenú chybu nie je potrebné riešiť. V tomto prípade sa životný cyklus chyby v ticketovacom systéme skončí krokom 3, pričom nie sú vytvorené nové úlohy pre členov a stav správy je modifikovaný na *WontFix*.

G.5 Reportovanie úlohy (Florek)

Kapitola rieši presný postup procesu reportovanie úlohy, ktorého výstupom je report o napredovaní úlohy ako nový tiket, ktorý sa po schválení pripojí k tiketu úlohy. Taktiež rieši formát reportu a veci, ktoré nevyhnutne musí obsahovať. Reporty sa píšú:

- 1) V každej tretine času vyhradeného na prácu na úlohe.
- 2) Ak sa pri realizácii úlohy narazí na potrebu zmeny.
- 3) Ak sa pri realizácii úlohy narazí na neodhalenú chybu softvéru.
- 4) Ak sa úloha počas jej realizácie zruší z externého prostredia.
- 5) Ak nie je možné pokračovať v úlohe z iného dôvodu.

V bodoch 2, 3, 4 a 5 vykonávateľ úlohy postupuje podľa metodík riešiacich situácie v jednotlivých bodoch.

Progres úloh sa musí reportovať minimálne trikrát počas trvania úlohy. Reporty pre úlohu vytvára vykonávateľ úlohy, a predstavujú ich tikety v issue trackeri na code.google.com s presne určenou formou vid' Obr. 01. Jednotlivé riadky sú popísané v podkapitole Formát reportu.

V prípade, že manažér oddelenia nájde v reporte formálne alebo neformálne nedostatky, posiela report na prepracovanie/dopracovanie.

Vytvorenie nového reportu

1. Vytvorenie reportu vo forme nového tiketu na webe projektu na

code.google.com. (vyžaduje si prihlásenie).

- a. Načítanie stránky projektu na code.google.com. V záložke Issues kliknúť na linku s názvom "New issue".
2. Zobrazí sa stránka na vytvorenie tiketu.
 - a. Ako šablónu (Template) zvoliť "Progress report".
 - b. Do poľa Summary napísať číslo reportu podľa kapitoly Formát reportu riadok 1.
 - c. Údaje v textovom poli Description vyplniť podľa kapitoly Formát reportu.
 - d. Nakoniec do poľa Cc navoliť zadávateľa úlohy, čiže manažéra oddelenia, a odoslať report cez "Submit issue".

Dopracovanie vráteného reportu

1. Otvorenie tiketu s reportom na webe projektu code.google.com (vyžaduje si prihlásenie).
 - a. Načítať stránku projektu a v záložke Issues vyhľadať report alebo postupovať podľa možnosti b.
 - b. Kliknúť na adresu v emaili, notifikujúcom o komentári v reporte.
2. Spustiť editor komentára.
 - a. Kliknúť do textového poľa pre komentár alebo postupovať podľa možnosti b.
 - b. Kliknúť na odkaz "Add a comment and make changes below" v ľavom paneli tiketu.
3. Do textového poľa pre komentár urobiť zmeny podľa dotazu od manažéra oddelenia. Treba dodržať formu textu podľa podkapitoly Formát reportu.
4. Odoslať zmeny tlačidlom "Save changes".

Formát reportu

Táto kapitola popisuje jednotlivé riadky vo forme reportu (Obr. 01):

1. **Číslo reportu** – poradové číslo reportu v rámci úlohy v tvare

PR-<číslo tiketu úlohy>-< číslo reportu {1,2,...,n}>

2. **Názov úlohy** – obsahuje názov úlohy a číslo tiketu, na ktorý sa report viaže
<názov úlohy>-< číslo tiketu>
3. **Report vypracoval** – autor reportu jeden z vykonávateľov úloh, ktorý report vytváral
4. **Dátum vytvorenia** – dátum, kedy bol report vytvorený
6. **Stav úlohy(%)** – na koľko percent je úloha splnená
7. **Odhadovaný termín ukončenia** – objektívny názor v akom termíne, by úloha mala byť ukončená, vyplývajúci z doterajšej práce na úlohe
9. **Opis práce** – stručný opis doposiaľ urobenej práce
16. **Splnené časti úlohy** – zoznam častí úlohy, ktoré sú už spravené, pričom sú číslované od 1. po N.
20. **Časti čakajúce na splnenie** – časti úlohy, ktorými sa v rámci plnenia úlohy vykonávateľ bude zaoberať, číslované od N. po M.
24. **Problémy** – zoznam problémov, ktoré sa vyskytli počas doposiaľ vykonanej práce. Pomlčkou (-) označené problémy sú problémy, ktoré riešil alebo bude riešiť vykonávateľ v rámci úlohy. Hviezdičkou (*) označené problémy, sú problémy, ktoré z úlohou súvisia ale v rámci úlohy sa nedajú riešiť. K problémom označených hviezdikou je potrebné na začiatku v hranatej zátvorke uviesť aj číslo tiketu [*číslo tiketu vytvoreného k problému*], ktorý vykonávateľ na základe problému vytvoril.
28. **Detekované chyby** – chyby softvéru, ktoré vykonávateľ zistil pri riešení úloh. Každá chyba musí obsahovať na začiatku v hranatej zátvorke číslo tiketu k nej vytvorené [*číslo tiketu vytvoreného k chybe*].
32. **Navrhované zmeny** – zmeny softvéru, ktoré vykonávateľ navrhuje ako riešenie problémov/chýb označených hviezdikou. Na začiatku problému je v hranatej zátvorke uvedené číslo vytvoreného tiketu [*číslo tiketu vytvoreného k problému*].

```
1  Číslo reportu:
2  Názov úlohy:
3  Report vypracoval:
4  Dátum vytvorenia:
5
6  Stav úlohy(%):
7  Odhadovaný termín ukončenia:
8
9  Opis práce:
10
11
12
13
14
15
16  Splnené časti úlohy:
17     1.
18     2.
19
20  Časti čakajúce na splnenie:
21     3.
22     4.
23
24  Problémy:
25     -
26     *
27
28  Detekované chyby:
29     *
30     *
31
32  Navrhované zmeny (nepovinné):
33     *
34     *
```

Obr. 01 Predpísaná forma reportu

G.6 Tvorba plánov (Juhász)

Plánovanie je dôležitou súčasťou každého projektu. Plány slúžia ako smernice pri organizovaní práce v tíme. Zostavenie plánu sa robí spoločne zohľadnením názorov a odhadov jednotlivých členov tímu. Pri zostavovaní plánov sa postupuje nasledovne:

1. vyhodnotenie splnenia plánu za uplynulé obdobie
2. zistenie dôvodov prečo plán nebol splnený
3. odhad zostávajúceho času pre jednotlivé nesplnené úlohy
4. určenie nových úloh s odhadom ich náročnosti
5. preplánovanie
 - a. naplánovanie nesplnených úloh na ďalšie obdobie

b. naplánovanie nových úloh na ďalšie obdobie

G.7 Konvencie písania dokumentácie (Bednár)

Na dosiahnutie vysokej úrovne dokumentu po formálnej stránke je potrebné pri písaní dodržiavať zadané typy písma a formátovanie.

Obyčajný test

Štýl je tvorený typom písma Palatino Linotype, veľkosť 12 a zarovnanie textu podľa okraja.

Zvýraznenie textu

Pre zvýraznenie textu je možné použiť Tučné písmo alebo Kurzíva.

Nadpis kapitol

Štýl sa nazýva Nadpis 1 a je tvorený z písma Palatino Linotype, veľkosť 18, Tučné písmo, zarovnanie textu podľa okraja, čísloje sa a nachádza sa vždy na novej strane. Patrí do obsahu dokumentu.

Nadpis podkapitol prvej úrovne

Štýl sa nazýva Nadpis 2 a je tvorený z písma Palatino Linotype, veľkosť 14, Tučné písmo, zarovnanie textu podľa okraja a čísloje sa. Patrí do obsahu dokumentu.

Nadpis podkapitol druhej úrovne

Štýl sa nazýva Nadpis 3 a je tvorený z písma Palatino Linotype, veľkosť 12, Tučné písmo, zarovnanie textu podľa okraja a nečísloje sa. Nepatrí do obsahu dokumentu.

Odkazy na literatúru, obrázok, tabuľku

Odkazy na literatúru sa zapisujú do hranatých zátvoriek, v ktorých sa nachádza číslo použitej literatúry. Odkaz na obrázok alebo tabuľku sa píše do guľatých zátvoriek. Príklad (Obr. 1).

Popis obrázkov

Štýl je tvorený typom písma Palatino Linotype, veľkosť 12 a zarovnanie textu je na stred. Popis, že ide o obrázok spoločne s číslom musí byť zvýraznené Tučným písmom.

Popis tabuliek

Štýl je tvorený typom písma Palatino Linotype, veľkosť 12 a zarovnanie textu je na stred. Popis, že ide o tabuľku spoločne s číslom musí byť zvýraznené Tučným písmom.

Formát dokumentu

Všetky okraje musia mať veľkosť 0.98". Pre odrážky sa používajú preddefinované „bodky“. Na číslovanie sa používajú čísla s bodkou alebo malé písmena s bodkou.

H) Zázpisnice zo stretnutí

V tejto kapitole sú priložené všetky zápisy z oficiálnych stretnutiach tímu v softvérovom štúdiu. V každom zápise je možné nájsť nasledujúce informácie:

- Dátum, miesto a čas stretnutia
- Vedúci tímu
- Prítomní a neprítomní členovia tímu
- Informácie o téme stretnutia
- Vyhodnotenie úloh z predchádzajúceho stretnutia
- Opis stretnutia
- Úlohy do ďalšieho stretnutia

Zápis 1. stretnutia tímu č. 20 (Brček)

Vedúci pedagóg: Ing. Peter Vilhan	
Pritomní členovia tímu: Bc. Maroš Bednár Bc. Adam Brček Bc. Marek Briš Bc. Marian Florek Bc. Vojtech Juhász Bc. Juraj Kosmeľ Bc. Ivan Valenčík	Dátum: 29. 9. 2010 Miestnosť: Softvérové štúdio Čas: 13:00 – 13:30 hod.
Chýbajúci:	Zápis vypracoval: Bc. Adam Brček

Téma stretnutia:

Úvod, zoznámenie členov tímu, bližšie oboznámenie sa s témou

Vyhodnotenie úloh z predchádzajúceho stretnutia

1. Neexistujú žiadne úlohy, prvé stretnutie

Opis stretnutia

1. Zoznámenie členov tímu s vedúcim
2. Zhrnutie informácií o súťažiach
3. Oboznámenie sa s formálnymi požiadavkami
4. Rozdelenie úloh na ďalšie stretnutie

Úlohy do ďalšieho stretnutia

Číslo úlohy	Popis úlohy	Kto	Termín ukončenia
1.1	Spraviť návrhy pre logo a názov tímu	Všetci	1.10.2010
1.2	Spraviť návrhy pre plagát	Všetci	3.10.2010
1.3	Rozdelenie tímových rolí	Všetci	6.10.2010
1.4	Sprevádzkovanie vývojového prostredia	Všetci	6.10.2010
1.5	Návrh webovej prezentácie	Maroš, Vojtech	6.10.2010
1.6	Sprevádzkovanie podporných prostriedkov (SVN, tickety)	Maroš	6.10.2010

Zápis 2. stretnutia tímu č. 20 (Brček)

Vedúci pedagóg: Ing. Peter Vilhan	
Pritomní členovia tímu: Bc. Maroš Bednár Bc. Adam Brček Bc. Marek Briš Bc. Marian Florek Bc. Vojtech Juhász Bc. Juraj Kosmeľ Bc. Ivan Valenčík	Dátum: 6. 10. 2010 Miestnosť: Softvérové štúdio Čas: 13:00 – 14:30 hod.
Chýbajúci:	Zápis vypracoval: Bc. Adam Brček

Téma stretnutia:

Diskusia k analýze existujúcich riešení z minulých rokov.

Vyhodnotenie úloh z predchádzajúceho stretnutia

Číslo úlohy	Popis úlohy	Kto	Stav
1.1	Spraviť návrhy pre logo a názov tímu	Všetci	Splnené
1.2	Spraviť návrhy pre plagát	Všetci	Splnené
1.3	Rozdelenie tímových rolí	Všetci	Splnené
1.4	Sprevádzkovanie vývojového prostredia	Všetci	Splnené
1.5	Návrh webovej prezentácie	Maroš, Vojtech	Splnené
1.6	Sprevádzkovanie podporných prostriedkov (SVN, tickety)	Maroš	Splnené

Opis stretnutia

1. Prezentácia vytvoreného logo a plagátu.
2. Rozdelenie rolí v tíme:
 - Maroš – zástupca vedúceho a manažér testovania
 - Adam – vedúci tímu
 - Marek – manažér podporných činností
 - Marian – manažér vývoja
 - Vojtech – manažér plánovania
 - Juraj – manažér kvality
 - Ivan – manažér rizík

3. Všetkým členom sa podarilo bez problémov nainštalovať vývojové prostredie TORCS aj minuloročný patch pre súťaž Championship.
4. Budeme vytvárať autopilotov pre obidve súťaže – Championship aj Demolition Derby.
5. Dohodli sme sa na ďalšom postupe – prvoradá je analýza minuloročných riešení, porovnanie ich výhod/nevýhod, na jej základe zvolíme koncept nášho riešenia.
6. Vyskytol sa problém ohľadne informácií o kategórií Demolition Derby, z vlaňajšieho prvého ročníka nie sú známe žiadne informácie o účastníkoch.
7. Dohodli sme sa na vytvorení modulárneho riešenia – pilot bude rozdelený do modulov s cieľom využitia niektorých z nich v oboch súťažiach.
8. Na záver sme si rozdelili minuloročných účastníkov na analýzu.

Úlohy do ďalšieho stretnutia

Číslo úlohy	Popis úlohy	Kto	Termín ukončenia
2.1	Analýza doterajších riešení	Maroš, Marek, Marian, Vojtech, Juraj	13.10.2010
2.2	Analýza možností nastavenia auta a prenos informácií medzi jednotlivými časťami preteku	Adam	13.10.2010
2.3	Získanie informácií o Demolition Derby	Ivan	13.10.2010

Zápis 3. stretnutia tímu č. 20 (Bednár)

Vedúci pedagóg: Ing. Peter Vilhan	
Pritomní členovia tímu: Bc. Maroš Bednár Bc. Adam Brček Bc. Marek Briš Bc. Marian Florek Bc. Vojtech Juhász Bc. Juraj Kosmeľ Bc. Ivan Valenčík	Dátum: 13. 10. 2010 Miestnosť: Softvérové štúdio Čas: 13:00 – 14:30 hod.
Chýbajúci:	Zápis vypracoval: Bc. Maroš Bednár

Téma stretnutia:

Diskusia k analýze existujúcich riešení z minulých rokov.

Vyhodnotenie úloh z predchádzajúceho stretnutia

Číslo úlohy	Popis úlohy	Kto	Stav
2.1	Analýza doterajších riešení	Maroš, Marek, Marian, Vojtech, Juraj	Splnené
2.2	Analýza možností nastavenia auta a prenos informácií medzi jednotlivými časťami preteku	Adam	Splnené
2.3	Získanie informácií o Demolition Derby	Ivan	Splnené

Opis stretnutia

1. Všetci členovia od prezentovali analýzy minuloročných účastníkov.
2. Analyzovanie nastavenie auta spoločne s voľbou ideálnej stopy.
3. Potvrdilo sa, že kategória Demolition Derby sa neuskutočnila minulý rok, čo malo byť po prvý raz.
4. Predbežne sme sa dohodli na určitej voľbe časti riešenia, riadenie spoločne s predbiehaním využiť od tímu COBOSTAR.

Úlohy do ďalšieho stretnutia

Číslo úlohy	Popis úlohy	Kto	Termín ukončenia
3.1	Hĺbková analýza doterajších riešení, zdrojové kódy pilotov.	Maroš, Marek, Marian, Vojtech, Juraj	20.10.2010
3.2	Analyzovať výpočet ideálnej stopy	Adam	20.10.2010
3.3	Získanie informácií o fyzickom modeli	Ivan	20.10.2010
3.4	Návrh architektúry pilota	Všetci	20.10.2010

Zápis 4. stretnutia tímu č. 20 (Briš)

Vedúci pedagóg: Ing. Peter Vilhan	
Prítomní členovia tímu: Bc. Maroš Bednár Bc. Adam Brček Bc. Marek Briš Bc. Marian Florek Bc. Vojtech Juhász Bc. Juraj Kosmeľ Bc. Ivan Valenčík	Dátum: 20. 10. 2010 Miestnosť: D 109 Čas: 13:00 – 16:00 hod.
Chýbajúci:	Zápis vypracoval: Bc. Marek Briš

Téma stretnutia:

Diskusia ohľadom modulárnej architektúry a návrh jednotlivých modulov.

Vyhodnotenie úloh z predchádzajúceho stretnutia

Číslo úlohy	Popis úlohy	Kto	Stav
3.1	Hĺbková analýza doterajších riešení, zdrojové kódy pilotov.	Maroš, Marek, Marian, Vojtech, Juraj	Splnené
3.2	Analyzovať výpočet ideálnej stopy	Adam	Splnené
3.3	Získanie informácií o fyzickom modeli	Ivan	Splnené
3.4	Návrh architektúry pilota	Všetci	Splnené

Opis stretnutia

- Optimálna trajektória analyzovaná z dostupného dokumentu, vyšla ako nepoužiteľná v našom konkrétnom prípade.
- Potvrdené, že API a pravidlá sa budú modifikovať len minimálne.
- Všetci členovia prezentovali svoje návrhy architektúr.
- Následne sme aplikovali „brainstorming“ pri vytváraní modulárnej architektúry.
- Špecifikovali sme niekoľko základných modulov a to (posledný je na najvyššej úrovni):
 - Recovery – zabezpečuje návrat na trať po jej opustení
 - Driving – zabezpečuje základné akcie ako plyn, brzda, radenie, riadenie, spojka
 - Line-assist – uchováva informácie o segmentoch trate, slúži na optimalizovanie trajektórie
 - Overtake – modul, ktorý zabezpečuje stratégiu predbiehania

- Defense – tzv. obranný modul, ktorý sa snaží eliminovať kolízie pri návrate na trať či predbiehaní
 - Strategic – rozhoduje o tom, či je potrebné ísť do boxov, či je kritické poškodenie alebo málo paliva
6. Riadenie sa bude zakladať na hodnotách najdlhšieho senzora.
 7. Na záver sme si rozdelili nové úlohy.

Úlohy do ďalšieho stretnutia

Číslo úlohy	Popis úlohy	Kto	Termín ukončenia
4.1	Implementovať riadenie, pamätanie si trate	Ivan	27.10.2010
4.2	Analyzovať a navrhnúť „recovery“ modul	Marian, Juraj	27.10.2010
4.3	Implementovať „recovery“ modul	Marek	27.10.2010
4.4	Navrhnúť ABS a ASR	Maroš	27.10.2010
4.5	Špecifikovať max. povolenú rýchlosť	Adam	27.10.2010
4.6	Vytvoriť sofistikovaný model trate (mapa)	Adam	27.10.2010
4.7	Napísať dokumentácia k riadeniu projektu	Vojtech	27.10.2010

Zápis 5. stretnutia tímu č. 20 (Kosmeľ)

Vedúci pedagóg: Ing. Peter Vilhan	
Prítomní členovia tímu: Bc. Maroš Bednár Bc. Adam Brček Bc. Marek Briš Bc. Marian Florek Bc. Vojtech Juhász Bc. Juraj Kosmeľ Bc. Ivan Valenčík	Dátum: 27. 10. 2010 Miestnosť: Softvérové laboratórium Čas: 13:00 – 16:00 hod.
Chýbajúci:	Zápis vypracoval: Bc. Juraj Kosmeľ

Téma stretnutia:

Diskusia k splneným úlohám (riadenie, pamätanie trate, recovery modul) – možnosti ďalšieho vylepšenia. Riešenie problematiky a algoritmu vytvárania modelu trate. Diskusia k vypracovaniu dokumentácie - analýzy, špecifikácie, návrhu a riadenia.

Vyhodnotenie úloh z predchádzajúceho stretnutia

Číslo úlohy	Popis úlohy	Kto	Stav
4.1	Implementovať riadenie, pamätanie si trate	Ivan	Splnené
4.2	Analyzovať a navrhnuť „recovery“ modul	Marian, Juraj	Splnené
4.3	Implementovať „recovery“ modul	Marek	Rozpracované
4.4	Navrhnuť ABS a ASR	Maroš	Splnené
4.5	Špecifikovať max. povolenú rýchlosť	Adam	Splnené
4.6	Vytvoriť sofistikovaný model trate (mapa)	Adam	Rozpracované
4.7	Napísať dokumentáciu k riadeniu projektu	Vojtech	Rozpracované

Opis stretnutia

1. Zhodnotenie úloh z posledného stretnutia, prezentácia výsledkov.
2. Návrh a prototyp recovery modulu hotový – potrebné doladiť detaily, optimalizovať rýchlosť.
3. Riadenie, maximálna rýchlosť, „učenie“ zákrut - implementovaný prototyp – dohodli sme sa na určení bodov vjazdu a výjazdu zo zákrut, možnosť spojenia za sebou idúcich zákrut – optimalizácia prejazdu zákrutami.

4. Model trate – rozpracovaný. Ďalej sme pracovali na algoritme vytvárania modelu trate, ktorý nereprezentoval trať správne.
5. Dokumentácia – dohodli sme sa dopracovať doteraz vykonané analýzy, vytvoriť špecifikáciu a dopracovať predbežný návrh riešenia, dokončiť dokumentáciu k riadeniu – skompletizovať dokumentáciu na odovzdanie.
6. Rozdelili sme si úlohy na ďalší týždeň.

Úlohy do ďalšieho stretnutia

Číslo úlohy	Popis úlohy	Kto	Termín ukončenia
4.3	Implementovať „recovery“ modul	Marek	03.10.2010
4.6	Vytvoriť sofistikovaný model trate (mapa)	Adam	03.10.2010
4.7	Napísať dokumentáciu k riadeniu projektu	Vojtech	03.10.2010
5.1	Zdokumentovať analýzu TORCS (Championship, Demolition derby).	Adam	03.10.2010
5.2	Zdokumentovať analýzu minuloročných účastníkov súťaže.	Marián	03.10.2010
5.3	Dopracovať dokumentáciu k riadeniu.	Vojtech	03.10.2010
5.4	Vypracovať metodiku plánovania.	Vojtech	03.10.2010
5.5	Vypracovať metodiku komunikácie.	Juraj	03.10.2010
5.6	Vypracovať metodiku používania SVN.	Marek	03.10.2010
5.7	Vypracovať metodiku písania zdrojového kódu.	Ivan	03.10.2010
5.8	Vypracovať špecifikáciu riešenia.	Juraj	03.10.2010
5.9	Zdokumentovať návrh riešenia.	Maroš	03.10.2010
5.10	Optimalizácia prechádzania zákrut.	Ivan	03.10.2010

Zápis 6. stretnutia tímu č. 20 (Florek)

Vedúci pedagóg: Ing. Peter Vilhan	
Pritomní členovia tímu: Bc. Maroš Bednár Bc. Adam Brček Bc. Marek Briš Bc. Marian Florek Bc. Juraj Kosmeľ Bc. Ivan Valenčík	Dátum: 03. 11. 2010 Miestnosť: Softvérové laboratórium Čas: 13:00 – 16:00 hod.
Chýbajúci: Bc. Vojtech Juhász	Zápis vypracoval: Bc. Marian Florek

Téma stretnutia:

Odovzdanie dokumentácie k riadeniu a projektovej dokumentácii. Diskusia k splneným úlohám. Hlavnou úlohou dohodnúť sa na reprezentácii trate. Riešenie problematiky a algoritmu vytvárania modelu trate.

Vyhodnotenie úloh z predchádzajúceho stretnutia

Číslo úlohy	Popis úlohy	Kto	Stav
4.3	Implementovať „recovery“ modul	Marek	Rozpracované
4.6	Vytvoriť sofistikovaný model trate (mapa)	Adam	Rozpracované
4.7	Napísať dokumentáciu k riadeniu projektu	Vojtech	Splnené
5.1	Zdokumentovať analýzu TORCS (Championship, Demolition derby).	Adam	Splnené
5.2	Zdokumentovať analýzu minuloročných účastníkov súťaže.	Marián	Splnené
5.3	Dopracovať dokumentáciu k riadeniu.	Vojtech	Splnené
5.4	Vypracovať metodiku plánovania.	Vojtech	Splnené
5.5	Vypracovať metodiku komunikácie.	Juraj	Splnené
5.6	Vypracovať metodiku používania SVN.	Marek	Splnené
5.7	Vypracovať metodiku písania zdrojového kódu.	Ivan	Splnené
5.8	Vypracovať špecifikáciu riešenia.	Juraj	Splnené
5.9	Zdokumentovať návrh riešenia.	Maroš	Splnené
5.10	Optimalizácia prechádzania zákrut.	Ivan	Rozpracované

Opis stretnutia

1. Odovzdanie dokumentácie, vylúčenie modulu stratégie.
2. Zhodnotenie úloh z posledného stretnutia, prezentácia výsledkov.
3. Brainstorming ohľadne reprezentácie trate.
4. Diskusia o dvoch typoch modelov: presný 2D model; zoznam zákrut, ako po sebe nasledujú.
5. 2D model – jeho optimalizácia veľmi náročná na čas (cca 18x4000 matica), predpoklad úplne neznámej trate, môže spôsobiť že nestihne optimalizáciu do stanoveného “wamp up” času (30000 tikov)
6. zoznam zákrut, obsahuje údaj o prejení zákruty (vzdialenosť začiatku od cieľa, priemerná rýchlosť, priemerný uhol, jednotlivé uhly, jednotlivé rýchlosti, ...), optimalizácia na základe fuzzy logiky => množina pravidiel na optimalizáciu
7. doplnenie údajov do záznamu o zákrute (miesto kde sa nachádza v rámci šírky trate, polomer zákruty)
8. úvahy o zjednodušení (urýchlení) optimalizačného algoritmu 2D modelu
9. zistenie, že na polomer zákruty sa nedá aplikovať všeobecný vzorec
10. polomer zákruty sa dá približne odvodiť z natočenia volantu (lineárna funkcia)
11. Rozdelili sme si úlohy na ďalší týždeň.

Úlohy do ďalšieho stretnutia

Číslo úlohy	Popis úlohy	Kto	Termín ukončenia
4.3	Implementovať „recovery“ modul	Marek	10.11.2010
4.6	Vytvoriť sofistikovaný model trate (mapa)	Adam	10.11.2010
5.10	Optimalizácia prechádzania zákrut.	Ivan	10.11.2010
6.1	Implementovať abs, asr pre všetky povrchy	Maroš	10.11.2010
6.2	Modul detekcie oponentov	Marián, Vojtech	10.11.2010
6.3	Vytvoriť kategorizáciu zákrut	Juraj	10.11.2010
6.4	Výpočet maximálnej rýchlosti z polomeru zákruty	Adam	10.11.2010

Zápis 7. stretnutia tímu č. 20 (Juhász)

Vedúci pedagóg: Ing. Peter Vilhan	
Prítomní členovia tímu: Bc. Maroš Bednár Bc. Adam Brček Bc. Marek Briš Bc. Marian Florek Bc. Vojtech Juhász Bc. Juraj Kosmeľ Bc. Ivan Valenčík	Dátum: 10. 11. 2010 Miestnosť: Softvérové laboratórium Čas: 13:00 – 16:00 hod.
Chýbajúci:	Zápis vypracoval: Bc. Vojtech Juhász

Téma stretnutia:

Diskusia k splneným a rozpracovaným úlohám (ABS, ASR, 2D model trate) – možnosti ďalšieho vylepšenia.

Vyhodnotenie úloh z predchádzajúceho stretnutia

Číslo úlohy	Popis úlohy	Kto	Stav
4.3	Implementovať „recovery“ modul	Marek	Rozpracované
4.6	Vytvoriť sofistikovaný model trate (mapa)	Adam	Rozpracované
5.10	Optimalizácia prechádzania zákrut.	Ivan	Rozpracované
6.1	Implementovať abs, asr pre všetky povrchy	Maroš	Rozpracované
6.2	Modul detekcie oponentov	Marián, Vojtech	Rozpracované
6.3	Vytvoriť kategorizáciu zákrut	Juraj	Splnené
6.4	Výpočet maximálnej rýchlosti z polomeru zákruty	Adam	Splnené

Opis stretnutia

1. Zhodnotenie úloh z posledného stretnutia, prezentácia výsledkov.
2. Prezentovali sme aktuálny stav 2D modelu trate a rozvíjali sme diskusie ohľadom možností vylepšenia modelu.
3. Prezentovali sme spôsob implementácie ABS a ASR.
4. Diskutovali sme o možných spôsoboch bránenia sa proti oponentom počas jazdy i počas zotavenia.
5. Diskutovali sme o spôsobe optimalizácie prechádzania zákrut – výber vhodnej pozície pri vchádzaní do zákrut
6. Rozdelili sme si úlohy na ďalší týždeň.

Úlohy do ďalšieho stretnutia

Číslo úlohy	Popis úlohy	Kto	Termín ukončenia
4.3	Implementovať „recovery“ modul	Marek	17.11.2010
4.6	Vytvoriť sofistikovaný model trate (mapa)	Adam	17.11.2010
5.10	Optimalizácia prechádzania zákrut.	Ivan	17.11.2010
6.1	Implementovať abs, asr pre všetky povrchy	Maroš	17.11.2010
6.2	Modul detekcia oponentov	Vojtech	17.11.2010
7.1	Určenie trenia (friction) podľa povrchu	Marián	17.11.2010
7.2	Line Assist – pozicionovanie sa pred zákrut	Juraj	17.11.2010

Zápis 8. stretnutia tímu č. 20 (Valenčík)

Vedúci pedagóg: Ing. Peter Vilhan	
Prítomní členovia tímu: Bc. Maroš Bednár Bc. Adam Brček Bc. Marek Briš Bc. Marian Florek Bc. Juraj Kosmeľ Bc. Ivan Valenčík	Dátum: 24. 11. 2010 Miestnosť: Softvérové laboratórium Čas: 14:30 – 16:00 hod.
Chýbajúci: Bc. Vojtech Juhász	Zápis vypracoval: Bc. Ivan Valenčík

Téma stretnutia:

Zhrnutie poznatkov o možnostiach vytvárania modelu trate na základe informácií o okrajoch trate, prezentácia a diskusia o použití rozoznávania zákrut, diskusia o prepojení jednotlivých modulov.

Vyhodnotenie úloh z predchádzajúceho stretnutia

Číslo úlohy	Popis úlohy	Kto	Stav
4.3	Implementovať „recovery“ modul	Marek	Rozpracované
4.6	Vytvoriť sofistikovaný model trate (mapa)	Adam	Zrušené
5.10	Optimalizácia prechádzania zákrut.	Ivan	Rozdelená
6.1	Implementovať abs, asr pre všetky povrchy	Maroš	Splnené
6.2	Modul detekcia oponentov	Vojtech	Rozpracované
7.1	Určenie trenia (friction) podľa povrchu	Marián	Splnené
7.2	Line Assist – pozicionovanie sa pred zákrutou	Juraj	Splnené

Opis stretnutia

1. Stretnutie prebehlo v neprítomnosti vedúceho tímu
2. Zhodnotenie úloh z posledného stretnutia a prezentácia výsledkov
3. Zamietnutie rozoznávania zákrut pomocou senzorov sledujúcich okraj trate z dôvodu príliš veľkého vplyvu šumu na túto metódu
4. Predvedenie rozoznávania zákrut a diskusia o praktickosti jeho využitia
5. Zmena vstupných údajov pre rozoznávanie zákrut z laterálnej rýchlosti na natočenie volantom

6. Testovanie rozoznávania trate na rôznych riadeniach a tratiach
7. Diskusia o možnosti prepojenia modulu Line assist s ostatnými modulmi
8. Diskusia o stratégii v zahrievacom kole súťaže Simulated Car Racing Championship
9. Ujasnenie ďalšieho postupu – vznik ucelenej predstavy o plánovanom výsledku zimného semestra
10. Rozdelenie úloh na ďalšie stretnutie

Úlohy do ďalšieho stretnutia

Číslo úlohy	Popis úlohy	Kto	Termín ukončenia
4.3	Implementovať „recovery“ modul	Marek	1.12.2010
6.2	Modul detekcia oponentov	Vojtech	1.12.2010
8.1	Telemetrický analyzátor s grafickým používateľským rozhraním	Adam	1.12.2010
8.2	Vytvorenie riadenia pre prejazd prvého kola	Marián	26.11.2010
8.3	Zintegrovať metódu plynulého prechodu zákrutou s riadením Prototype	Marián	1.12.2010
8.4	Vytvoriť funkciu na odhad maximálnej rýchlosti vstupu do zákruty	Marián	1.12.2010
8.5	Korekcia riadenia pri skoku	Marek	1.12.2010
8.6	Zintegrovať presun na žiadanú stranu trate s riadením Prototype	Juraj	1.12.2010
8.7	Vypracovať analýzu rizík	Ivan	30.11.2010
8.8	Zintegrovať rozoznávanie zákrut s riadením na prvé kolo a s riadením Prototype	Ivan	1.12.2010
8.9	Overiť použiteľnosť vytvoreného rozoznávania zákrut na rozličných povrchoch a tratiach	Ivan	1.12.2010
8.10	Zdokumentovať ABS a ASR	Maroš	1.12.2010
8.11	Opraviť všetky zápisnice	Maroš	1.12.2010
8.12	Doplnenie dokumentácie k riadeniu	Maroš	1.12.2010

Zápis 9. stretnutia tímu č. 20 (Bednár)

Vedúci pedagóg: Ing. Peter Vilhan	
Prítomní členovia tímu: Bc. Maroš Bednár Bc. Adam Brček Bc. Marek Briš Bc. Marian Florek Bc. Vojtech Juhász Bc. Juraj Kosmeľ Bc. Ivan Valenčík	Dátum: 1. 12. 2010 Miestnosť: Softvérové laboratórium Čas: 13:00 – 16:00 hod.
Chýbajúci:	Zápis vypracoval: Bc. Maroš Bednár

Téma stretnutia:

Zhrnutie doterajších modul a ich prezentácia. Diskusia o výpočte maximálnej rýchlosti pri prechode zákruty. Diskusia o telemetrii a dodefinovanie ďalšej funkcionality.

Vyhodnotenie úloh z predchádzajúceho stretnutia

Číslo úlohy	Popis úlohy	Kto	Stav
4.3	Implementovať „recovery“ modul	Marek	Splnené
6.2	Modul detekcia oponentov	Vojtech	Splnené
8.1	Telemetrický analyzátor s grafickým používateľským rozhraním	Adam	Splnené
8.2	Vytvorenie riadenia pre prejazd prvého kola	Marián	Splnené
8.3	Zintegrovat' metódu plynulého prechodu zákrutou s riadením Prototype	Marián	Splnené
8.4	Vytvorit' funkciu na odhad maximálnej rýchlosti vstupu do zákruty	Marián	Rozpracované
8.5	Korekcia riadenia pri skoku	Marek	Rozpracované
8.6	Zintegrovat' presun na žiadanú stranu trate s riadením Prototype	Juraj	Rozpracované
8.7	Vypracovať analýzu rizík	Ivan	Rozpracované
8.8	Zintegrovat' rozoznávanie zákrut s riadením na prvé kolo a s riadením Prototype	Ivan	Splnené
8.9	Overit' použiteľnosť vytvoreného rozoznávania zákrut na rozličných povrchoch a tratiach	Ivan	Splnené
8.10	Zdokumentovať ABS a ASR	Maroš	Splnené
8.11	Opraviť všetky zápisnice	Maroš	Splnené
8.12	Doplnenie dokumentácie k riadeniu	Maroš	Splnené

Opis stretnutia

1. Zhodnotenie úloh z posledného stretnutia a prezentácia výsledkov vedúcemu tímu
2. Diskusia o nemožnom aplikovaní zoznamu zákrut na výpočet maximálnej možnej rýchlosti vniknutia do zákruty a jej celý prechod
3. Prispôsobenie jazdy v prvom kole čo najviac na stred trate
4. Zmena výpočtu začiatku a konca zákruty
5. Diskusia o doplnení novej funkcionality telemetrie
6. Diskusia o možných situáciách pri dopade zo skoku
7. Rozdelenie úloh na ďalšie stretnutie

Úlohy do ďalšieho stretnutia

Číslo úlohy	Popis úlohy	Kto	Termín ukončenia
8.4	Vytvoriť funkciu na odhad maximálnej rýchlosti vstupu do zákruty	Marián	8.12.2010
8.5	Korekcia riadenia pri skoku	Marek, Maroš	8.12.2010
8.6	Zintegrovať presun na žiadanú stranu trate s riadením Prototype	Juraj	8.12.2010
8.7	Vypracovať analýzu rizík	Ivan	8.12.2010
9.1	Pokročilý telemetrický analyzátor s grafickým používateľským rozhraním	Adam	8.12.2010
9.2	Zdokumentovať model trate	Adam	8.12.2010
9.3	Zdokumentovať telemetrický analyzátor	Adam	8.12.2010
9.4	Zdokumentovať Line Assist	Juro	8.12.2010
9.5	Zdokumentovať Recovery	Marek	8.12.2010
9.6	Zdokumentovať Defense	Vojto	8.12.2010
9.7	Zdokumentovať rozoznávanie zákrut	Ivan	8.12.2010
9.8	Detekcia a kategorizácia zákrut	Ivan, Marián	8.12.2010
9.9	Zdokumentovať štart	Marián	8.12.2010
9.10	Zdokumentovať riadenie	Marián	8.12.2010
9.11	Zdokumentovať radenie (gear)	Marián	8.12.2010
9.12	Finálna verzia dokumentu k riadeniu	Maroš	8.12.2010

Zápis 10. stretnutia tímu č. 20 (Briš)

Vedúci pedagóg: Ing. Peter Vilhan	
Prítomní členovia tímu: Bc. Maroš Bednár Bc. Adam Brček Bc. Marek Briš Bc. Marian Florek Bc. Vojtech Juhász Bc. Juraj Kosmeľ Bc. Ivan Valenčík	Dátum: 8. 12. 2010 Miestnosť: Softvérové laboratórium Čas: 13:00 – 16:00 hod.
Chýbajúci:	Zápis vypracoval: Bc. Marek Briš

Téma stretnutia:

Zhrnutie doterajších modulov a ich prezentácia. Diskusia o „line assist“ a jeho integrovaní do riadenia. Diskusia o možnosti zamerať sa na technické nastavenie automobilu.

Vyhodnotenie úloh z predchádzajúceho stretnutia

Číslo úlohy	Popis úlohy	Kto	Stav
8.4	Vytvoriť funkciu na odhad maximálnej rýchlosti vstupu do zákruty	Marián	Rozpracované
8.5	Korekcia riadenia pri skoku	Marek, Maroš	Rozpracované
8.6	Zintegrovať presun na žiadanú stranu trate s riadením Prototype	Juraj	Rozpracované
8.7	Vypracovať analýzu rizík	Ivan	Splnené
9.1	Pokročilý telemetrický analyzátor s grafickým používateľským rozhraním	Adam	Rozpracované
9.2	Zdokumentovať model trate	Adam	Rozpracované
9.3	Zdokumentovať telemetrický analyzátor	Adam	Rozpracované
9.4	Zdokumentovať Line Assist	Juro	Splnené
9.5	Zdokumentovať Recovery	Marek	Splnené
9.6	Zdokumentovať Defense	Vojto	Rozpracované
9.7	Zdokumentovať rozoznávanie zákrut	Ivan	Nesplnené
9.8	Detekcia a kategorizácia zákrut	Ivan, Marián	Rozpracované
9.9	Zdokumentovať štart	Marián	Splnené
9.10	Zdokumentovať riadenie	Marián	Splnené
9.11	Zdokumentovať radenie (gear)	Marián	Splnené
9.12	Finálna verzia dokumentu k riadeniu	Maroš	Splnené

Opis stretnutia

1. Zhodnotenie úloh z posledného stretnutia a prezentácia výsledkov vedúcemu tímu
2. Diskusia o dôvode nedokončenia detekcie zákrut (začiatok, koniec) a následne riešenie
3. Využitie „line assist“ je stále povrchné, stále chýbajúce kategórie zákrut, diskusia o možnej optimalizácii počas letného semestra
4. Diskusia o funkcionalite telemetrie
5. Diskusia o možných situáciách pri dopade zo skoku
6. Odhalenie chaotického správania sa pilota v prípade, že server nedostane odpoveď od klienta
7. Rozdelenie úloh na ďalšie stretnutie

Úlohy do ďalšieho stretnutia

Číslo úlohy	Popis úlohy	Kto	Termín ukončenia
8.4	Vytvoriť funkciu na odhad maximálnej rýchlosti vstupu do zákruty	Marián	15.12.2010
8.5	Korekcia riadenia pri skoku	Marek, Maroš	15.12.2010
8.6	Zintegrovať presun na žiadanú stranu trate s riadením Prototype	Juraj	15.12.2010
9.1	Pokročilý telemetrický analyzátor s grafickým používateľským rozhraním	Adam	15.12.2010
9.2	Zdokumentovať model trate	Adam	15.12.2010
9.3	Zdokumentovať telemetrický analyzátor	Adam	15.12.2010
9.6	Zdokumentovať Defense	Vojto	15.12.2010
9.7	Zdokumentovať rozoznávanie zákrut	Ivan	15.12.2010
9.8	Detekcia a kategorizácia zákrut	Ivan, Marián	15.12.2010
10.1	Kontaktovať organizátorov	Marek	15.12.2010
10.2	Zdokumentovať integráciu „Line assists“ s riadením	Juraj	15.12.2010

I) Autori jednotlivých částí dokumentací (Bednár)

I.1 Autori dokumentácie k projektu

Kapitola	Časť dokumentácie k riadeniu	Autor
	Obsah	Generované - Bednár
0.	Úvod	Juhász
1.1	TORCS – The Open Racing Car Simulator	Brček, Valenčík
1.2.1	LUIGI CARDAMONE	Bednár
1.2.2	AUTOPIA	Briš
1.2.3	COBOSTAR	Florek
2.	Špecifikácia	Kosmeľ
3.	Návrh	Bednár
4.1	Modul riadenia	Florek
4.2	Modul Recovery	Briš
4.3	Modul modelu trate	Brček, Valenčík
4.4	Modul podpory riadenia pred vjazdom do zákrut	Kosmeľ
4.5	Modul detekcie oponentov	Juhász
4.6	Modul anti-blokovacieho systému (ABS)	Bednár
4.7	Modul regulácie preklzovania (ASR)	Bednár
4.8	Modul Telemetria	Brček
5.	Literatúra	Brček
	Finálna úprava	Briš, Brček, Bednár
	Gramatická úprava	Brček

I.2 Autori dokumentácie k riadeniu

Kapitola	Časť dokumentácie k riadeniu	Autor
	Obsah	Generované - Bednár
A	Úvod	Juhász
1.	Členovia tímu	Bednár
2.	Simulated Car Racing Competition 2011	Valenčík
2.1	Koncepcia riešenia	Brček, Valenčík
3.	Model používateľa pre jeho identifikáciu	Florek
3.1	Koncepcia riešenia	Briš, Kosmeľ
4.	Portál pre časopis	Juhász
4.1	Návrh riešenia	Juhász
5.	Poradie tém podľa preferencií	Bednár
6.	Aktuálny rozvrh členov tímu	Bednár, Valenčík
D	Plány projektu	Juhász
D.1	Tvorba plánov	Juhász
D.2	Externé termíny	Juhász
D.3	Hrubý plán projektu na zimný semester	Juhász
D.4	Revízia plánu v siedmom týždni	Juhász
D.4	Revízia plánu v jedenástom týždni	Juhász
E	Manažment rizík	Valenčík
E.1	Analýza rizík	Valenčík
E.2	Identifikované riziká	Valenčík
F	Úlohy členov tímu	Juhász
F.1	Dlhodobé roly členov tímu	Juhász
F.2	Krátkodobé úlohy členov tímu	Juhász
G.1	Práca s SVN	Briš
G.2	Komunikácia	Kosmeľ
G.3	Pravidlá písania a komentovania zdrojového kódu	Valenčík
G.4	Manažovanie chýb	Juhász
G.5	Reportovanie úlohy	Florek
G.6	Tvorba plánov	Juhász

G.7	Konvencia písania dokumentácie	Bednár
H	Zápisnice zo stretnutí	Juhász
	Zápis 1. stretnutia tímu č. 20	Brček
	Zápis 2. Stretnutia tímu č. 20	Brček
	Zápis 3. stretnutia tímu č. 20	Bednár
	Zápis 4. stretnutia tímu č. 20	Briš
	Zápis 5. stretnutia tímu č. 20	Kosmeľ
	Zápis 6. stretnutia tímu č. 20	Florek
	Zápis 7. stretnutia tímu č. 20	Juhász
	Zápis 8. stretnutia tímu č. 20	Valenčík
	Zápis 9. Stretnutia tímu č. 20	Bednár
	Zápis 10. Stretnutia tímu č. 20	Briš
I	Autori jednotlivých častí dokumentácií	Bednár
J	Preberacie protokoly	Brček
	Finálna úprava	Bednár, Brček
	Gramatická úprava	Brček

Tabuľka č.3 Autori kapitol dokumentácie k riadeniu

J) Preberacie protokoly (Brček)

Na nasledujúcej strane sa nachádzajú preberacie protokoly k projektu.

TÍMOVÝ PROJEKT

2010/2011

PREBERACÍ PROTOKOL

Predmet:

Tvorba informačného systému v tíme

Tvorba softvérového systému v tíme

časť:

počet strán:

Dokumentácia k riadeniu

Projektová dokumentácia

Poskytovateľ – za tím č. 20: Ready2Win

Preberajúci

Dolu podpísaný preberajúci týmto potvrdzuje, že prevzal predmet od poskytovateľa v uvedenom rozsahu.

poskytovateľ

preberajúci

TÍMOVÝ PROJEKT

2010/2011

PREBERACÍ PROTOKOL

Predmet:

Tvorba informačného systému v tíme

Tvorba softvérového systému v tíme

časť:

počet strán:

Dokumentácia k riadeniu

Projektová dokumentácia

Poskytovateľ – za tím č. 20: Ready2Win

Preberajúci

Dolu podpísaný preberajúci týmto potvrdzuje, že prevzal predmet od poskytovateľa v uvedenom rozsahu.

poskytovateľ

preberajúci