



**Slovenská technická univerzita v Bratislave**  
**FAKULTA INFORMATIKY**  
**A INFORMAČNÝCH TECHNOLOGIÍ**

---

# **Podpora vzdelávania v predmete Opis digitálnych systémov**

**(Tímový projekt)**

**Akademický rok:** 2009/2010  
**Študijný program:** PKSS  
**Vedúci projektu:** Ing. Elena Tomalová

Bc. Michal Majzlík  
Bc. Juraj Bleho  
Bc. Martin Boroš  
Bc. Štefan Olejník  
Bc. Jozef Szadvári  
Bc. Slavomír Žiak

# Obsah

---

Obsah .....	2
Zadanie.....	3
Podpora vzdelávania v predmete Opis digitálnych systémov.....	3
Úvod .....	4
Motivácia.....	4
Ciele práce.....	4
Štruktúra dokumentu .....	5
Použité skratky .....	5
Analýza.....	7
Podpora vzdelávania v predmete Špecifikačné a opisné jazyky .....	7
Podpora vzdelávania v predmete Špecifikačné a opisné jazyky .....	10
Podpora dištančného vzdelávania v predmete Opis digitálnych systémov .....	11
Podpora vzdelávania v predmete špecifické a opisné jazyky .....	13
Podpora dištančného vzdelávania v predmete Opis digitálnych systémov .....	15
Zhodnotenie analýz .....	17
Špecifikácia .....	18
Špecifikácia systému.....	18
Privilégia pre entitu Učiteľ .....	19
Privilégia pre entitu Študent .....	20
Hrubý návrh.....	22
Rozšírenie modulu Test.....	22
Zdroje: .....	24

# Zadanie

---

## Podpora vzdelávania v predmete Opis digitálnych systémov

Analyzujte existujúce aplikácie a moduly, vytvorené pre podporu predmetu Špecifikačné a opisné jazyky. Analyzujte tiež dostupné vzdelávacie aplikácie s podobným zameraním. Pri analýze sa zamerajte najmä na podporu správa zadaní (prideľovanie zadaní, zber, overovanie správnosti ich vypracovania a automatické ohodnotenie) a overovania praktických zručností študentov.

Na základe analýzy navrhňte a implementujte e-learningové moduly (prípadne externé aplikácie) pre správu výučby predmetu Opis digitálnych systémov (náhrada predmetu Špecifikačné a opisné jazyky), ktoré budú podporovať správu a vyhodnocovanie zadaní a zabezpečovať overovanie získaných praktických zručností študentov v rámci predmetu.

Odporúčaná literatúra:

- Matej Mayer, Bc., Podpora dištančného vzdelávania v predmete Opis digitálnych systémov, Diplomová práca, FIIT STU Bratislava, máj 2009

- Andrej Letkovský, Bc., Podpora dištančného vzdelávania v predmete Opis digitálnych systémov, Diplomová práca, FIIT STU Bratislava, máj 2009

- Izsák Peter, Automatické vyhodnocovanie programov vo VHDL, Záverečný projekt, FEI STU Bratislava, máj 2000

- Tím 3PSS, Podpora vzdelávania v predmete Špecifikačné a opisné jazyky, Tímový projekt, FIIT STU Bratislava, máj 2008

- Tím 4PSS, Podpora vzdelávania v predmete Špecifikačné a opisné jazyky, Tímový projekt, FIIT STU Bratislava, máj 2008

# Úvod

---

Účelom tejto práce je zjednodušiť a zatriktívniť, výučbu a následne aj testovanie znalostí študentov z predmetu Opis digitálnych systémov, ktorý sa vyučuje na Fakulte Informatiky a Informačných technológií STU.

## Motivácia

Zo všetkých zverejnených tém nás najviac oslovila téma s názvom „Podpora vzdelávania v predmete Opis digitálnych systémov“. Nakoľko veľká časť tímu už v minulosti pracovala s touto problematikou či už pri práci na bakalárskom projekte, alebo na iných projektoch v rámci zadaní v predmetoch bakalárskeho štúdia. Na základe doteraz nadobudnutých poznatkov a z dôvodu silnej motivácie pre ďalšiu prácu v oblasti elektronického vzdelávania má náš tím podnet ďalej rozvíjať a riešiť túto problematiku využitím vedomostí a zručnosti. Naším cieľom je zefektívniť a zjednodušiť proces vzdelávania či už ako v predmete Špecifické a opisné jazyky alebo ako elektronického vzdelávania vo všeobecnosti. Vzhľadom na tieto skutočnosti si myslíme, že bol výber témy opodstatnený a naša motivácia je väčšia z presvedčenia, že naša práca bude prínosom v elektronickom vzdelávaní.

## Ciele práce

Cieľom práce je oboznámiť sa s rôznymi systémami elektronického vzdelávania a zistiť, ktorý sa pre náš projekt najviac prospešný (zatiaľ ako cieľ vývoja predpokladáme systém Moodle, pre jeho relatívne ľahkú rozšíriteľnosť a jeho nasadenia v učebnom procese na našej fakulte).

Ďalším cieľom je analyzovať existujúce riešenia, s dôrazom na správu zadaní, ktoré študenti vypracúvajú a odovzdávajú, následne na to vytvoriť systém, ktorý by mal umožňovať plánovanie a správu rôznych akcií pre študentov a to:

- termíny odovzdania priebežných častí projektov,
- termíny prezentácií hotových projektov,
- elektronické odovzdávanie v stanovený čas
- automatizované vyhodnocovanie zadaní

- zobrazit' úlohy v prehľadnom harmonograme

Cieľom je vytvoriť modul schopný automatizovane hodnotiť študentské projekty, taktiež uľahčiť prístup ku kvalitným študijným materiálom, umožniť študentom pridávať vlastné poznámky, riešenia príkladov a zostavenie dostatočného počtu testovacích otázok na generovanie testov s možnosťou okamžitého vyhodnotenia otázky, potvrdenia zadanej možnosti, alebo zamietnutia s príslušným vysvetlením prečo je odpoveď nesprávna.

## Štruktúra dokumentu

Tento dokument je štruktúrovaný nasledovne:

### Analýza

V tejto stati sa náš tím sústredil na analýzu už existujúcich riešení. Analyzované boli diplomové práce a tímové projekty z predošlých akademických rokov, ktoré boli riešené na fakulte Informatiky a informačných technológií STU.

### Špecifikácia

V tejto stati je bližšie špecifikované, ktoré vlastnosti a požiadavky by mal spĺňať náš výsledný produkt.

### Hrubý návrh

V tejto stati je načrtnutý výsledný systém.

## Použité skratky

**PKSS** – Počítačové a komunikačné systémy a siete

**PSS** – študijný odbor Počítačové systémy a siete

**VHDL** – (VHSIC Hardware Description Language) programovací opisný jazyk slúžiaci pre opis hardvéru. Používa sa pre návrh a simuláciu digitálnych integrovaných obvodov

**HTML** –Hypertextový značkový jazyk (HyperText Markup Language; HTML) je značkový jazyk určený na vytváranie webových stránok a iných informácií zobraziteľných vo webovom

prehliadači. HTML kladie dôraz skôr na prezentáciu informácií (odseky, fonty, váha písma, tabuľky atď.) ako na sémantiku (význam slov).

**PHP** – [2]PHP: Hypertext Preprocessor) je populárny open source skriptovací programovací jazyk, ktorý sa používa najmä na programovanie klient-server aplikácií (na strane servera) a pre vývoj dynamických webových stránok..

**SQL** – (Structured Query Language) – jazyk na manipuláciu s relačnou databázou

**MySQL** – [3]MySQL je slobodný a otvorený viacvláknový, viacúčítateľský SQL relačný databázový server. MySQL je podporovaný na viacerých platformách (ako Linux, Windows či Solaris a je implementovaný vo viacerých programovacích jazykoch ako PHP, C++ či Perl. Databázový systém je relačný typu DBMS (database management system).

# Analýza

---

Pri analýze prác a projektov, ktoré nám boli odporúčané v zadaní projektu sa kládol dôraz sa hlavne na zadania s témou „Podpora vzdelávania v predmete Špecifické a opisné jazyky“ a zadania „Podpora dištančného vzdelávania v predmete Opis digitálnych systémov“. Krátkou analýzou prešli aj iné projekty, s podobnou tematikou, ktoré sa však odlišujú od nášho zadania, preto neboli podrobnejšie študované a nie sú uvedené v tomto dokumente.

## Podpora vzdelávania v predmete Špecifikačné a opisné jazyky

(Tím 4 PSS, Tímový projekt 2009)

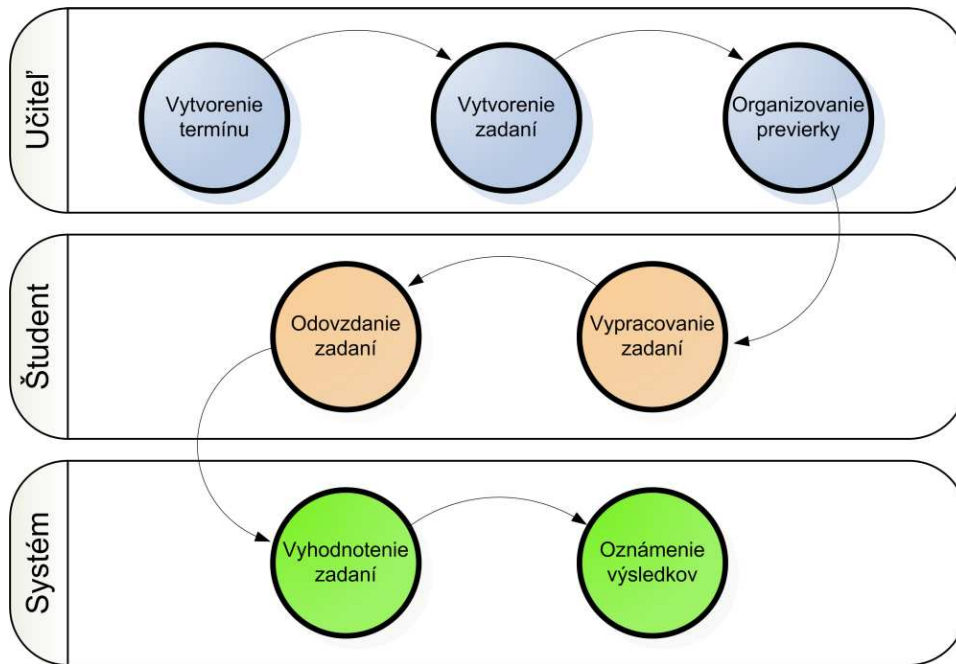
### Ciele projektu

Tím sa zameril na overovanie praktických znalostí v predmete Špecifické a opisné jazyky v oblasti problematiky Petriho sietí. Ako cieľ projektu si tím stanovil vytvorenie aplikácie na modelovanie, simuláciu a analýzu Petriho sietí, ktorá bude schopná vyhodnocovať správnosť študentmi namodelovanej siete a tým poskytovať spätnú väzbu na ich vedomosti. Cieľom vytvorenia aplikácie, bolo aj použiteľnosť pri testovaní študentov a tým odľahčenie náročného kontrolovania správnosti vyučujúcim a tým aj uľahčenie práce a zníženie časovej náročnosti.

### Špecifikácia požiadaviek

Na základe analyzovaných dostupných zdrojov podobných projektov sa tím rozhodol špecifikovať nasledujúce požiadavky.

Vytvorenie aplikácie schopnej automaticky vyhodnocovať zadania z tematiky Petriho sietí. Zameranie na teoretické a praktické vedomosti študentov a tomu prispôsobenie aplikácie. Čiže aplikácia má podporovať tieto dva typy zadaní. Zjednodušený priebeh fungovania možno nájsť na nasledujúcom obrázku, ktorý je prevzatý z tejto práce.



Obr. č.1 Priebeh testovania a opravy

- Systém má podporovať tieto činnosti:
  - vytvorenie termínu
  - vytvorenie zadanií
  - organizovanie previerky
  - vypracovanie zadanií
  - odovzdanie zadanií
  - vyhodnotenie zadanií
  - oznámenie výsledkov
- V systéme vystupujú dve entity:
  - Učiteľ
  - Študent

Entita učiteľ má právo na nasledujúce činnosti uvedené na obrázku.





Obr. č.2 prípady použitia pre používateľa Učiteľ

- Entita Študent ma právo na nasledujúce činnosti:
  - výber termínu
  - výber príkladu
  - vypracovanie príkladu
  - odovzdanie testu
  - zobrazenie výsledkov hodnotenia

## Návrh, implementácia a testovanie

Autori navrhli architektúru systému, ako aplikáciu Klient-Server. Jednotlivé údaje boli ukladané do databázy s použitím technológie MySQL. Hlavná časť systému bola implementovaná v jazyku JAVA. Dizajn aplikácie je postavený na princípe návrhového vzoru MVC (Model-View-Controll) – podrobnejšie opísaný v dokumente. Testovaním tejto aplikácie tím odhalil, pomerne veľké množstvo chýb, ktoré boli následne doladené.

## Záver

Výsledkom projektu je funkčná aplikácia na preverovanie teoretických a praktických znalostí v modelovaní, simulácie a analýze Petriho sietí. Základnú myšlienku a jadro aplikácie by sme mohli využiť pri realizácii nášho tímového projektu. Táto práca obsahuje návrh vhodnej architektúry systému, ktorá by mohla byť použitá pri implementácii našej aplikácie.

## Podpora vzdelávania v predmete Špecifikačné a opisné jazyky

(Tím 4 PSS, Tímový projekt 2008)

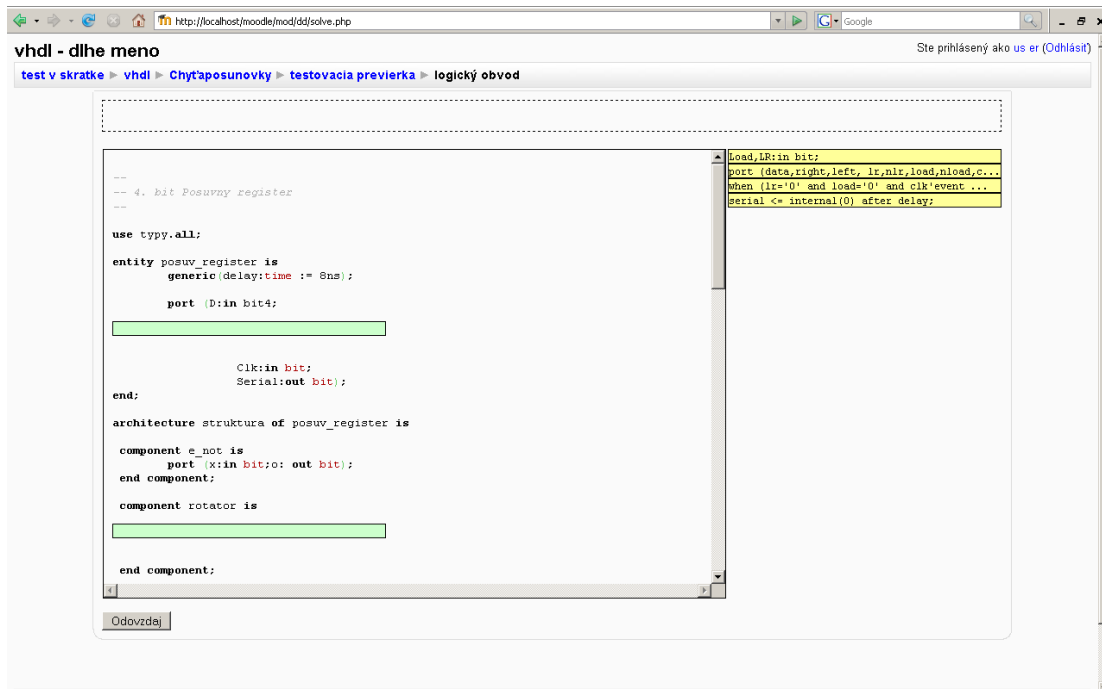
## Analýza

Po analýze niekoľkých riešení sa tím rozhodol vytvoriť samostatný modul do výučbového modulu Moodle.

Tím vytvoril modul, ktorý pedagógom umožňuje zostavenie, plánovanie a organizovanie previerok pre študentov. Previerky sa skladajú z niekoľkých zadaní týkajúcich sa opisu digitálnych obvodov v jazyku VHDL. Vyučujúci zadanie pripraví, má možnosť ho zmeniť, alebo priradiť k previerkam. Študent v stanovenom čase zadanie vyplní a odošle na spracovanie. Systém zadanie ohodnotí. Zadania a previerky sú automaticky skopírované do archívnej časti modulu.

Osobitnou časťou je samotné zadanie - **Chyt'aposunovka**. Ide o zaujímavý spôsob ako otestovať vedomosti študentov, formou dopĺňania chýbajúcich kusov VHDL kódu do existujúcej kostry programu. V pravej časti obrazovky sa nachádza kostra VHDL programu a v ľavej časti sú fragmenty kódu, ktoré je možné systémom drag&drop (chyt' a posuň)

umiestniť na svoje miesto v kostre programu. Jednotlivé fragmenty majú svoju váhu a na základe tejto váhy a počtu správne umiestnených fragmentov dostane študent ohodnotenie. Na obrázku 3. je obrazovka s rozpracovanou Chyťaposunovkou.



Obr. č.3 Ukážka aplikácie Chyťaposunovka

## Záver

Autori tejto práce identifikovali niekoľko návrhov na zlepšenie, náš tím sa rozhodol preskúmať možnosti prvého návrhu - zakomponovať Chyťaposunovku do testovacieho modulu systému Moodle - Test.

## Podpora dištančného vzdelávania v predmete Opis digitálnych systémov

(Bc. Matej Mayer, Diplomový projekt 2009)

## Analýza

Diplomová práca Bc. Mateja Majera sa zaoberá vytvorením modelu na podporu výučbu vzdelávania jazyku VHDL, ktorý sa vyučuje na predmete opis digitálnych systémov. V práci analyzuje viacero druhov vzdelávacích systémov, z ktorých si nakoniec na implementáciu vybral systém Moodle. Ten sa totiž už v danom čase využíval na výučbu v danom predmete. Do systému Moodle implementoval modul, ktorý sa delí na tri časti,

konkrétne časť študent, časť učiteľ a časť systém. Každá z týchto častí podporuje rôzne funkcie zabezpečujúce požiadavky na výučbu VHDL jazyka.

Časť študent slúži na odskúšanie študentov, ktorý študujú predmet Opis digitálnych systémov. Študentom je umožnené editovať, kompilovať a spustiť simuláciu kódu VHDL. Keď sa študent rozhodne, môže odovzdať zadanie ako finálne, toto sa porovná s referenčnou vzorkou a zároveň sa automaticky vyhodnotí. Po finálnom odovzdaní si budú študenti môcť automaticky prezrieť výsledok skúšky.

Časť učiteľ slúži vyučujúcim na vytváranie zadaní, úpravu výsledkov študentov a zároveň prezeranie zadaní. Pri vytváraní zadaní môže učiteľ stanoviť termín prístupnosti modulu pre študentov, dĺžku testu, maximálny počet bodov, bodovú zrážku pri nezhode a základný počet bodov pri úspešnom skompilovaní. Ak sa kód nepodarí skompilovať zadanie sa musí ohodnotiť vyučujúcim manuálne. Táto časť ešte podporuje vkladanie zadaní, obrázkov, kostry kódu, referenčnej testovacej vzorky a vzorového riešenia. Ďalej má učiteľ k dispozícii časť prezeranie zadaní, ktoré mu umožňuje export zadaní do cvs a zip formátov a prezeranie odovzdaných zadaní a výsledkov automatického vyhodnotenia.

Poslednou časťou vytváraného modulu je časť systém, ktorá zabezpečuje chod a funkčnosť vytváraného modulu. Táto časť sa delí na podčasti obsluhy VHDL kódu, odpočítavania času, uloženia a ohodnotenia odoslaných kódov. Pri ohodnocovaní kódov študent vytvoril ohodnotenie na základe porovnania referenčnej vzorky a vzorky odovzdanej daným študentom. Porovnanie zdrojových kódov sa mu implementovať nepodarilo. Podčasť obsluhy VHDL kódu slúži na kompiláciu a simuláciu kódu a nahradenie testovacích vzoriek.

## Záver

Na kompiláciu a simuláciu študent použil dve časti Modelsimu a to kompilátor VCOM a simulátor VSIM, ktoré integroval do systému Moodle. Toto riešenie je celkom jednoduché a prehľadné, a preto sme sa ho rozhodli použiť aj v našej práci. Komunikácia VCOM-u a VSIM-u sa deje prostredníctvom konzoly. Porovnanie vzoriek sa deje tak, že sa odsimuluje odovzdaná vzorka, ďalej sa odsimuluje referenčná vzorka, ktoré sú následne uložené vo wlf súboroch. Simulátor VSIM podporuje porovnanie dvoch takýchto súborov, pričom výsledok porovnania je text. Problémom, s ktorým sa študent nevedel popasovať je grafické znázornenie daného porovnania, čo by mohlo byť zjednodušenie práce pre učiteľa

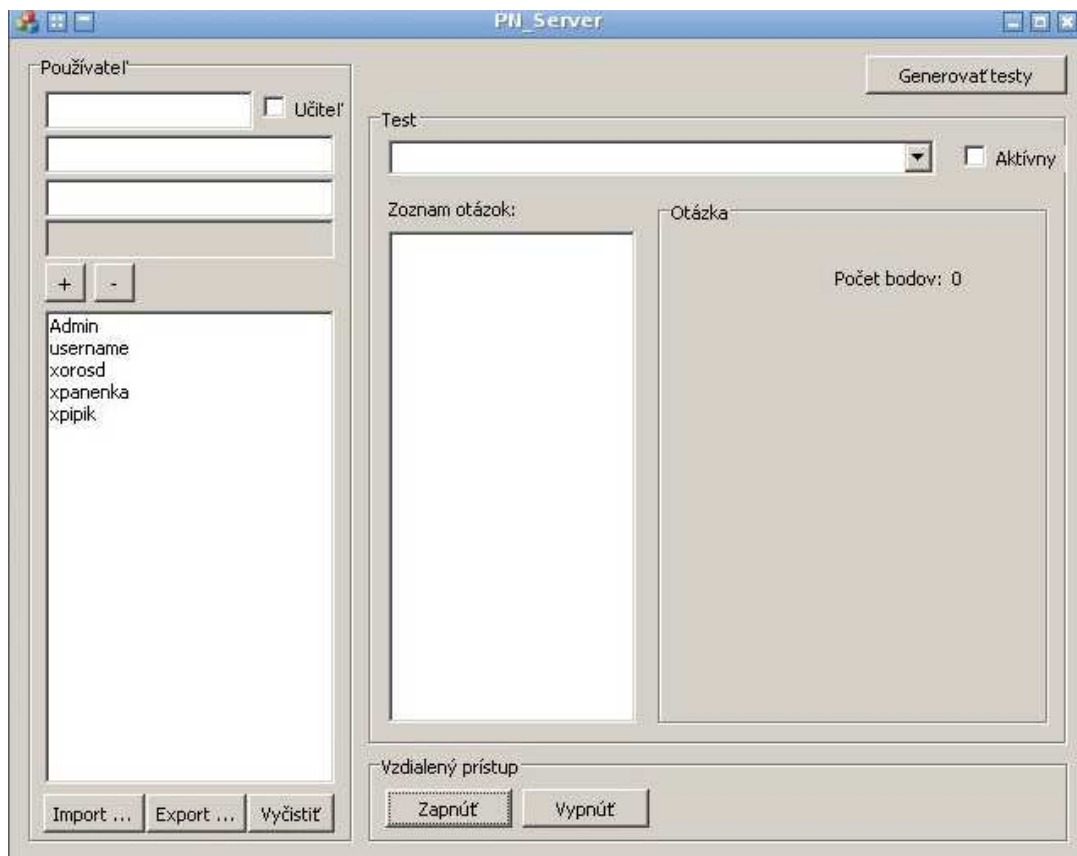
kontrolujúceho výsledky skúšky. Preto by sa o riešenie grafického zobrazenia mohol pokúsiť náš tím.

## Podpora vzdelávania v predmete špecifické a opisné jazyky (Tím Surfin'birds, Tímový projekt 2009)

### Analýza

V rámci predmetu Tímový projekt sa tím SURFIN'BIRDS, v zložení Bc. Peter Koine, Bc. Šimon Hupka, Bc. Ján Turoň, Bc. Michal Choleva, Bc. Emil Moťovský pod vedením Ing Tomalovej, zaoberal Podporou vzdelávania v predmete Špecifické a opisné jazyky. Vo svojom projekte sa venovali Petriho sieťam. Rozhodli sa vytvoriť externú aplikáciu v jazyku C++. Aplikácia je typ klient-server.

Na serveri sa nachádza databáza študentov a vyučujúcich. K serveru majú priamy prístup len vyučujúci. Vyučujúci tu môže vytvárať testové úlohy, z ktorých sa počas testu náhodne niektoré vyberú a zašlú študentom na riešenie. Zasielanie zadaní počas skúšky je zašifrované keďže spolu so zadaním sa zasielajú aj odpovede - testové úlohy teda vyhodnocuje klientska aplikácia.



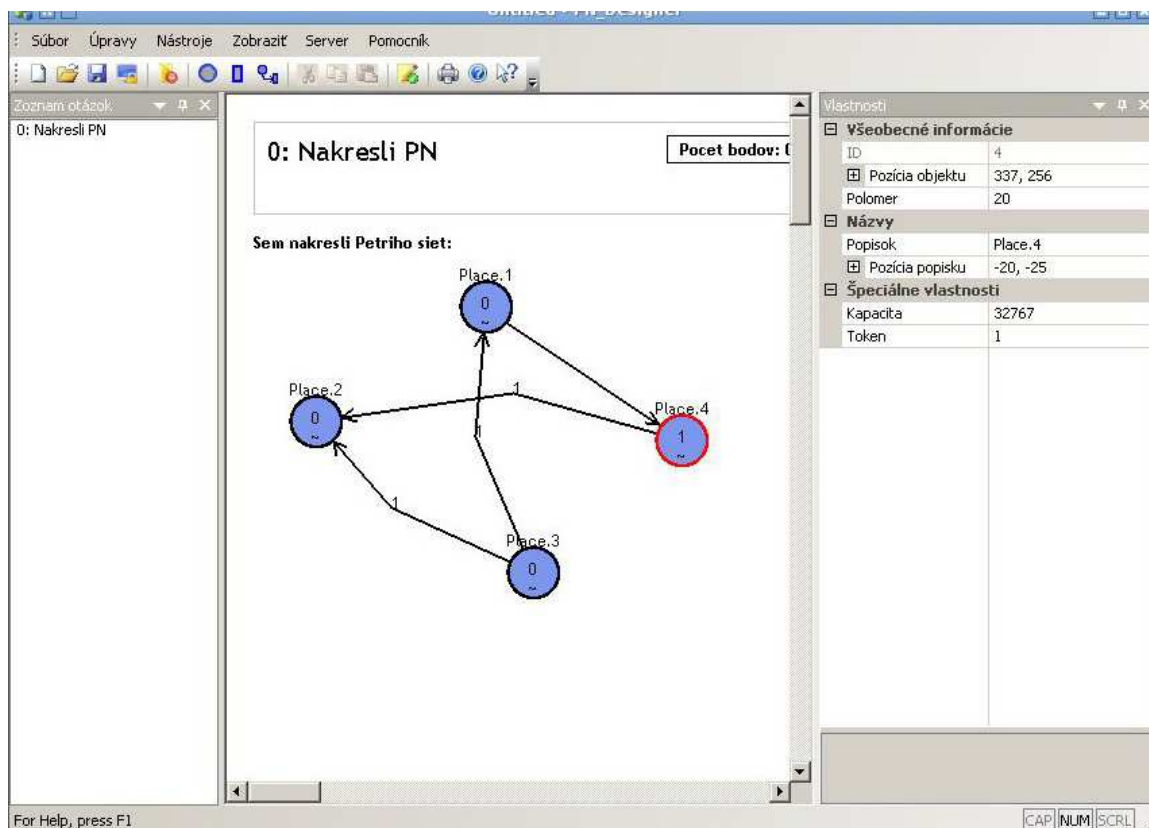
Obr. č.4 ukážka aplikácie PN.Server

Klient má dve odlišné úlohy. Prvou je jeho použitie pri skúškach. Študenti sa pomocou neho prihlásia na server, dostanú úlohy na riešenie. Tie vyriešia v klientskej aplikácii. Spolu so zadaním úloh aplikácia prijme aj ich riešenia, ktoré potom následne vyhodnotí a výsledok odošle na server. Z dôvodu bezpečnosti, aplikácia sleduje spustené procesy aby sa dala odhaliť prípadná komunikácia medzi študentmi, snaha o zistenie odpovedí zo zašifrovanej komunikácie klientskej aplikácie so serverom ale podvádzanie pomocou internetu. Počas testu je teda povolené mať spustenú len klientsku aplikáciu.

Druhou úlohou aplikácie je jej možnosť bežať samostatne bez servera. Spolu s aplikáciou sa distribuuje kolekcia úloh na riešenie doma pre štúdium a precvičovanie sa.

V projekte sú 2 typy úloh. Prvý je doplnenie neúplnej Petriho siete podľa zadania. Druhým je kompletný návrh Petriho siete. Pri prvom type je riešenie len jedno a jednoznačné. Pri druhom type aplikácia vyhodnocuje riešenie podľa vlastností študentom navrhutej siete, ktoré sa dajú overiť algoritmom.

Pre vyhodnocovanie Petriho sietí prebrali algoritmus na vyhodnotenie vlastností Petriho sietí z diplomovej práce Bc. Ľuboša Heribana - Diagnostika porúch založená na PS modeloch. Algoritmus na porovnávanie Petriho sietí pomocou incidenčnej matice.



Obr. č.5 Ukážka modelu Petriho siete

Ovládanie programu má bežné pracovné prostredie, hlavnou časťou je pracovná plocha kde sa kreslia Petriho siete. Tá je rozdelená do mriežky, pričom v každom bode mriežky sa môže nachádzať jeden objekt - miesto alebo prechod siete. Aplikácia sa ovláda pomocou ponuky akcií a panelom nástrojov. Je potrebné len jedno tlačítko myši aby tým zabezpečili bezproblémovosť použitia aj na všetkých Tablet PC zariadenia. Tým sa dostávame k ďalšej úlohe, ktorú Tím č.3 riešil a to je podpora stylusu pre Tablet PC. Okrem gest zabudovaných v operačnom systéme sa rozhodli študenti implementovať aj vlastné riešenie. Aplikácia pozná tri typy akcií stylusom (prípadne myšou). Prvou je používateľ nakreslí akýkoľvek súvislý obrázok, ktorý sa pretne v jednom bode, vyhodnotí sa to ako miesto v Petriho sieti, ak sa tento obrázok nepretne tak sa jedná o prechod. V prípade spojenie miest a prechodov je potrebné kliknúť na jeden aj druhý objekt a tým sa medzi nimi spojenie - hrana.

## **Podpora dištančného vzdelávania v predmete Opis digitálnych systémov**

(Diplomový projekt)

### **Analýza**

Cieľom tohto projektu, bolo vytvorenie multimedialneho nástroja pre podporu výučby v predmete Opis digitálnych systémov. Vytvorený nástroj, by mal umožňovať získavanie a overovanie praktických zručností z jazyka SystemC. Ing. Letkovský sa rozhodol použiť systém moodle a rozšíriť ho o ďalšie moduly. Rozhodol sa tak z nasledovných dôvodov: Je širiteľný pod licenciou GNU, čo znamená, že jeho zdrojový kód môže byť ľubovoľne upravovaný. S tým súvisí aj fakt, že je modulárny a dobre zdokumentovaný. Ďalšou výhodou je to, že spĺňa štandardy e-learningu a štandardy W3C. Dôvodom je aj jeho používanie na Fakulte Informatiky a Informačných technológií. Ako web server bol zvolený Apache HTTP Server a ako databázový server MySQL. Ide teda o aplikáciu typu server-klient. Na strane servera je systém moodle s pripojeným modulom a na strane klienta webový prehliadač.

Vytvorené moduly slúžia dvom skupinám: učiteľom a študentom. Študentom umožňuje:

- Tvorbu zdrojového kódu so zvýrazňovaním syntaxe
- Overenie syntaktickej správnosti vytvoreného zdrojového kódu
- Simuláciu návrhu
- Odovzdanie vytvoreného zadania

- Priebežné uloženie

Učiteľovi systém umožňuje:

- Tvorbu zadaní
- Nastavenie časového limitu pre vypracovanie
- Prezeranie odovzdaných zadaní
- Úprava automatického hodnotenia
- Archivácia odovzdaných riešení

Na overenie syntaktickej správnosti bol použitý externý kompilátor ModelSim. Výhodou ModelSimu je možnosť volania jeho súčastí, kompilátora VCOM a simulátora VSIM z príkazového riadku. Pre kompiláciu zdrojového kódu v jazyku SystemC je však potrebné použiť kompilátor SSCOM.

Systém je tvorený dvomi modulmi: skuskasc a vysledkysc. Modul skuskasc je navrhnutý tak, aby študentovi neumožňoval žiadne neoprávnené zásahy. Študent je pred začatím riešenia informovaný o tom ako test vykonať a aké zadanie má riešiť. Študent programuje svoje riešenie v textovom editore, ktorý umožňuje priebežné uloženie, kompiláciu, odovzdanie a simuláciu. Výsledky kompilácie a simulácie sa zobrazujú pod textovým editorom. Editor umožňuje spustenia zvýrazňovania syntaxe a číslovania riadkov. Modul poskytuje učiteľovi formulár, v ktorom do modulu vloží všetky potrebné informácie ohľadom testu. Modul je implementovaný tak, že jedna jeho inštancia obsahuje iba jedno zadanie. Pre iné zadanie je potrebné vytvorenie novej inštancie. Toto riešenie prináša učiteľovi uľahčenie práce s tvorbou skupín.

Modul vysledkysc je navrhnutý tak aby pomohol učiteľovi vyhľadávať výsledky konkrétneho študenta alebo zálohovať riešenia všetkých študentov, ktorí vypracovávali vybraný test. Tento modul poskytuje učiteľovi dva formuláre. Prvý formulár Vyhľadávanie riešení slúži na zobrazenie riešenia konkrétneho študenta pre vybraný test. V prípade potreby má učiteľ možnosť upraviť hodnotenie. Druhý formulár Archivácia riešení ponúka možnosť uloženia všetkých výsledkov alebo riešení v dvoch formátoch: zip archív, csv.

## Záver

Tento diplomový projekt rozdeľuje privilégia na ovládanie systému, medzi role: učiteľ študent. Podobné rozdelenie práv by sme mohli zohľadniť pri návrhu a implementácie nášho diplomového projektu.



## Zhodnotenie analýz

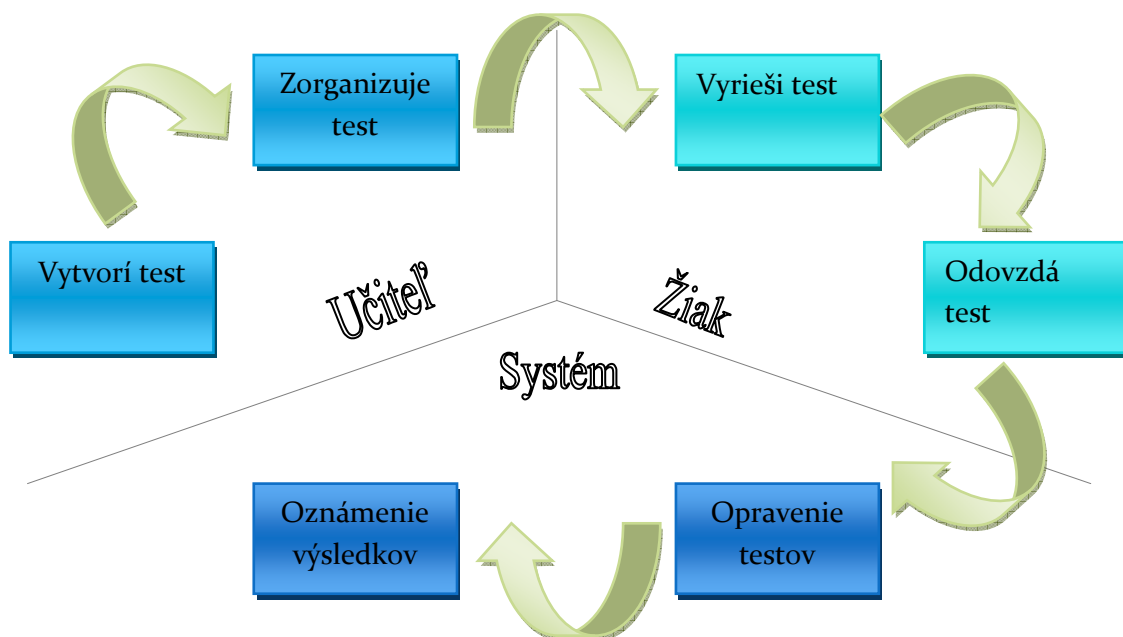
Každý jeden analyzovaný projekt obsahuje niekoľko hlavných bodov, ktorými by sme sa mohli inšpirovať pri riešení nášho projektu. Nad hlavnou kostrou systému sa stále pracuje. Na základe analyzovaných projektov, by náš tím mohol uvažovať nad zostavením aplikácie klient-server, využiť rozdelenie privilégií medzi entity študent-učiteľ, alebo prípadne navrhnúť architektúru systému podobnú už predtým analyzovaných a implementovaných projektov. Pri realizácii nášho projektu tím uvažuje nad využitím a vylepšením už vytvoreného zdrojového kódu z práce: „Podpora vzdelávania v predmete špecifické a opisné jazyky“, ktorú naimplementoval tím 4. z odboru PSS v akademickom roku 2008/2009.

# Špecifikácia

V tejto stati náš tím vytvoril špecifikáciu požiadaviek pre výslednú aplikáciu. Ďalej je bližšie určené aké funkcie bude náš systém obsahovať. V špecifikácií sú rozpracované jednotlivé entity a ich privilégia v e-learningovanom systéme.

## Špecifikácia systému

Systém pre podporu vzdelávania v predmete Opis digitálnych systémov by mal zabezpečovať automatizované vytváranie a vyhodnocovanie testov. Tieto testy sú hlavne z oboru Petriho sietí a z oboru VHDL kódu. Pri probléme Petriho sietí ako aj problému VHDL kódu sme vychádzali z riešenia už existujúcich projektov. Naším cieľom je hlavne ich vylepšenie a zintegrovanie týchto viacerých projektov do jednej aplikácie. Systém má napomáhať pri vytváraní a vyhodnocovaní otázok testovej formy, vytváraní zadaní Petriho sietí a vytvoriť čo možno najlepšie vyhodnotenie týchto zadaní. Ďalej náš tím vychádza z programu „Chyť a posuň“, pri ktorom sa dopĺňajú časti kódu a následne po odovzdaní ho systém vyhodnocuje. Táto časť je zameraná na problém VHDL kódov. Naším hlavným cieľom je vylepšiť tieto už existujúce riešenie, doladiť, prípadne opraviť nefungujúce časti a pokúsiť sa zhrnúť tieto časti do jednej aplikácie. Predpokladanú interakciu systému medzi učiteľmi a žiakmi môžeme vidieť na nasledujúcom obrázku.



Obr. č.6 Interakcia v systéme

Z obrázka môžeme vidieť jednotlivú interakciu. Systém by mal pracovať tak, že učiteľ si bude môcť v ňom vytvoriť zadania, následne bude môcť určiť termín testu. Na to ho žiak vyplní a odovzdá. Systém to vyhodnotí a odovzdá výsledky žiakovi a učiteľovi.

## **Priviléžia pre entitu Učiteľ**

### **Správa testov**

Táto funkcia zahŕňa všetky možnosti, ktoré sa týkajú testov. Obsahuje všetky dostupné funkcie a to vytvorenie, mazanie a modifikáciu testov. Príklady môžu byť tvorené buď len pre jeden problém ako napríklad Petrího siete, alebo všeobecný test v rámci celého predmetu.

### **Správa študentov**

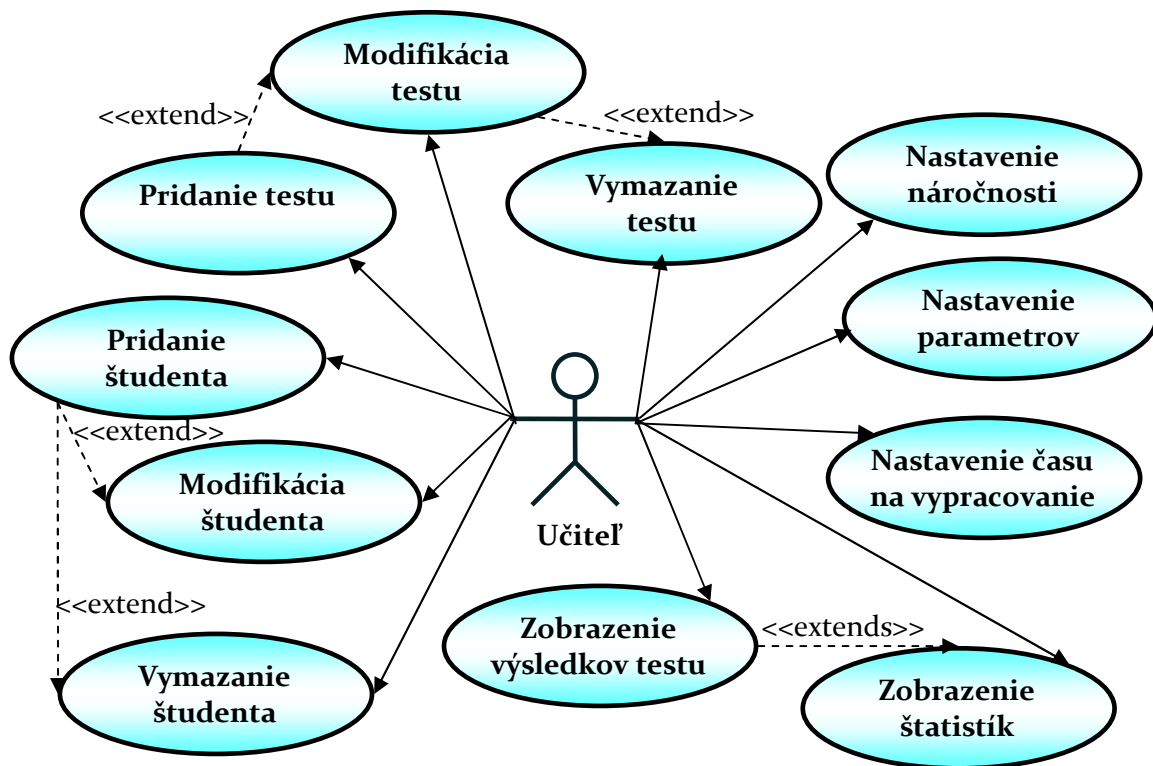
Táto funkcia slúži na vytvorenie účtov pre jednotlivých študentov, ich vytvorenie, mazanie, úprava, pridelenie jednotlivých študentov do skupín a podobne.

### **Nastavenie parametrov testovania**

Učiteľ bude mať možnosť určiť a nastaviť spôsob vyberania testov. Bude môcť sprístupniť prípadne zablokovať testy pre vybranú skupinu študentov. Taktiež by systém mal obsahovať možnosť generovania testov podľa oboru, alebo podľa náročnosti. V systéme by taktiež nemalo odpočítanie a stanovenie časových limitov pre test.

### **Zobrazenie výsledkov testov a štatistika**

Učiteľ bude mať právo zobraziť informácie o celkovej ako aj o úspešnosti každého z jednotlivcov. Učiteľ by mal mať taktiež právo modifikovať a upravovať výsledky testov ako aj nazrieť do vyriešených príkladov.



Obr. č.7 Prípady použitia pre používateľa Učiteľ

## Priviléžia pre entitu Študent

### Výber a vypracovanie príkladu

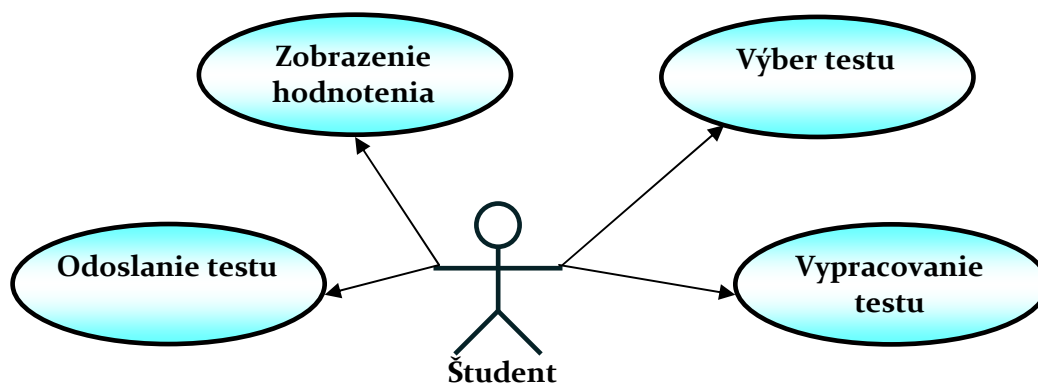
Na vypracovanie učiteľ povolí vyberanie testov, žiak si bude môcť vybrať test z danej problematiky, ktorú považuje za lepšiu voľbu, prípadne si bude môcť vybrať z dvoch podobných testov. Po začatí testovania systém ponúkne študentovi test na vyplnenie, ktorý obsahuje viacero zadaní rôznej náročnosti. Každý študent dostane jedno z niekoľkých zadaní patriacich do jednej skupiny náročnosti čoho cieľom je zabrániť študentom opisovaniu pri skúšaní. K dispozícii bude mať študent informáciu o časovom limite.

## Odovzdanie testu

Študent má možnosť kedykoľvek odovzdať test. V prípade vypršania časového limitu určeného učiteľom sa test odovzdá automaticky.

## Zobrazenie výsledkov a hodnotenia

Po odovzdaní testu a jeho vyhodnotení systémom bude študent informovaný o bodovom ohodnotení svojho testu. Nebude mať však prístup k informáciám o hodnotení testov ostatných študentov. V prípade, že učiteľ nastaví funkciu zobrazovanie a porovnávania správnych riešení, študent bude mať právo pozrieť si kde urobil chybu a správne riešenie, v prípade že nezíska maximálny počet bodov.



Obr. č.8 Prípady použitia pre používateľa Študent

# Hrubý návrh

---

Táto kapitola sa podrobnejšie zaoberá návrhom modulov na preskúšanie praktických vedomostí.

## Rozšírenie modulu Test

Tento modulu slúži na vytváranie a správu nových testov. Je súčasťou jadra výučbového systému Moodle.

Má veľa možností pridelovania vytvorených testov študentom. Ten, kto test vytvára, môže vyhradiť čas na riešenie v hodinách alebo v dňoch. Zadá začiatok spustenia testu, čo znamená čas, kedy bude test sprístupnený pre študentov a môže zadať konečný čas, dokedy si študenti daný test môžu otvoriť.

Učiteľ môže test zostaviť tak, že tam bude len jedna úloha, alebo viacero otázok. Má na výber z veľa rôznych typov otázok:

- výpočtová
- vpísanie textu podľa inštrukcií
- porovnávaciacia
- výber z možností / viac výberová
- áno/nie

Tieto typy otázok nie sú pre náš projekt kľúčové. Vytvoríme dva nové, ktoré implementujeme do modulu Test. Tieto nové typy budú z oboru VHDL programov a Petriho siete. Pri ich implementácii budeme vychádzať z už existujúcich projektov a ich aplikácií:

### 1. Chyťaposunovka

Budeme vychádzať z projektu **Podpora vzdelávania v predmete Špecifikačné a opisné jazyky**, ktorý bol vypracovaný tímom č.4 v roku 2008. V tejto práci sa zaoberali vytvorením nového modulu Chyťaposunovka, ktorý slúžil podobne ako modul Zadanie. Pri ich module však vložili funkcionality a rozhrania pre ľahké vytvorenie úlohy pre VHDL kód.

My zoberieme zdrojové kódy tohto modulu a prispôbíme ho tak, aby sme vytvorili nový modul, ktorý zaradíme medzi moduly typov otázok, ktoré ovláda modul Test.

## 2. Petriho siete

Tu budeme vychádzať z dvoch prác: **Podpora vzdelávania v predmete Špecifické a opisné jazyky**, vypracovaný tímom PSS3 a tímom PSS4 z roku 2008.

Tím **PSS3** vypracoval samostatný program v C++ na riešenie a testovanie Petriho sietí. Z tohto chceme použiť algoritmus na porovnávanie Petri sietí - vyhodnotenie samotných otázok typu Petriho sieť. V projekte tímu **PSS4** bola vytvorená Java aplikácia na kreslenie Petriho sietí. Túto prepracujeme na Java Applet a doplníme algoritmus na porovnávanie Petriho sietí z projektu tímu PSS3. Študent tak počas testu dostane zadanie úlohy kde bude musieť vytvoriť Petriho sieť, ktorá bude spĺňať určité požiadavky podľa zadania. Túto vytvorenú sieť odovzdá, Applet sieť vyhodnotí a určí či je navrhnutá správne alebo nie a výsledok odošle do databázy.

Pridaním týchto dvoch otázok do modulu Test sa značne rozšíria možnosti používania systému Moodle pre predmet Opis digitálnych systémov. Otázky bude možné kombinovať s klasickými testovými otázkami a takto bude možné vytvoriť semestrálne zadania (ak učiteľ nastaví požadovaný čas na riešenie niekoľko týždňov), priebežný, alebo záverečný test (skúšku).

## Zdroje:

---

1. Matej Mayer, Bc., Podpora dištančného vzdelávania v predmete Opis digitálnych systémov, Diplomová práca, FIIT STU Bratislava, máj 2009
2. Andrej Letkovský, Bc., Podpora dištančného vzdelávania v predmete Opis digitálnych systémov, Diplomová práca, FIIT STU Bratislava, máj 2009
3. Izsák Peter, Automatické vyhodnocovanie programov vo VHDL, Záverečný projekt, FEI STU Bratislava, máj 2000
4. Tím 3PSS, Podpora vzdelávania v predmete Špecifikačné a opisné jazyky, Tímový projekt, FIIT STU Bratislava, máj 2008
5. Tím 4PSS, Podpora vzdelávania v predmete Špecifikačné a opisné jazyky, Tímový projekt, FIIT STU Bratislava, máj 2008