

Slovenská technická univerzita v Bratislave  
Fakulta elektrotechniky a informatiky  
KATEDRA INFORMATIKY A VÝPOČTOVEJ TECHNIKY  
Študijný odbor: SOFTVÉROVÉ INŽINIERSTVO

Bc. Juraj MIKLUŠ  
Bc. Andrej ORIŠKO  
Bc. Miroslav PANÁK  
Bc. Boris STEINER  
Bc. Martin SZOLGAY

ANIMÁCIA SIMULAČNÝCH EXPERIMENTOV  
S MODELOM DISKRÉTNĚHO SYSTÉMU

Tímový projekt

Vedúci tímového projektu: Ing. Ivan Lališ

Školský rok 1998/99

# OBSAH

## I. Špecifikácia

|  |    |
|--|----|
| 1. Úvod                                      | 2  |
| 1.1 Zadanie projektu                         | 2  |
| 1.2 Ohraničenia                              | 2  |
| 1.3 Riešiteľský kolektív                     | 2  |
| 2. Plán projektu                             | 4  |
| 2.1 Podrobný plán činnosti                   | 4  |
| 3. Model systému                             | 7  |
| 3.1 Funkcie modulov a ich rozhrania          | 7  |
| 3.2 Kontext systému                          | 8  |
| 4. Špecifikácia funkcií systému              | 9  |
| 4.1 Vytvorenie, načítanie a zapísanie modelu | 9  |
| 4.2 Riadenie simulácie                       | 10 |
| 4.3 Dynamické zmeny modelu                   | 10 |
| 4.4 Vizualizácia                             | 10 |
| 4.5 Vyhodnocovanie výsledkov                 | 11 |
| 5. Špecifikácia údajov v systéme             | 13 |
| 5.1 Triedy údajov                            | 13 |
| 5.2 Popis tried údajov                       | 13 |
| 5.2.1 Zákazník                               | 13 |
| 5.2.2 Pokladňa                               | 13 |
| 5.2.3 Regál                                  | 14 |
| 5.2.4 Tovar                                  | 14 |
| 5.2.5 Oddelenie                              | 15 |
| 6. Špecifikácia správania systému            | 16 |
| 6.1 Aktívne komponenty                       | 16 |
| 6.1.1 Simulačný procesor                     | 16 |
| 6.1.2 Simulovaný model                       | 16 |
| 6.2 Pasívne komponenty                       | 17 |
| 6.3 Pomocné komponenty                       | 17 |
| 7. Ďalšie požiadavky a ohraničenia           | 18 |

## II. Uživatelská príručka

|   |    |
|---|----|
| 1. Úvod   | 20 |
| 2. Vytvorenie, otvorenie a uloženie špecifikácie modelu obchodného domu do súboru | 21 |
| 3. Modifikácia vlastností modelu obchodného domu                                  | 22 |
| 4. Riadenie simulácie   | 24 |
| 4.1 Body zastavenia   | 25 |
| 4.1.1 Zastavenie v konkrétnom čase  | 25 |
| 4.1.2 Zastavenie pri zmene objektu  | 25 |
| 4.1.3 Zastavenie pri kúpe   | 25 |
| 5. Vyhodnocovanie výsledkov simulácie   | 27 |
| 5.1 Vlastnosti elementov obchodného domu  | 27 |
| 5.2 Vyhľadávanie  | 28 |
| 5.3 Externé moduly  | 29 |

|  |    |
|--|----|
| 5.3.1 Nastavenie externých modulov . . . . .   | 30 |
| 6. Nastavenie parametrov aplikácie . . . . .   | 31 |
| 6.1 Nastavenie aplikácie . . . . .             | 31 |
| 6.2 Úpravy pracovnej plochy . . . . .          | 31 |
| 7. Často kladené otázky . . . . .              | 32 |
| 8. Inštalácia a požiadavky na systém . . . . . | 33 |
| 8.1 Požiadavky programu na systém . . . . .    | 33 |
| 8.2 Inštalácia . . . . .                       | 33 |
| 8.3 Odstránenie inštalácie . . . . .           | 33 |

### III. Návrh a implementácia

|   |    |
|---|----|
| 1. Úvod . . . . .   | 35 |
| 2. Plán projektu . . . . .                                      | 36 |
| 3. Architektúra systému . . . . .                               | 37 |
| 4. Fyzický model údajov systému . . . . .                       | 38 |
| 5. Ohraničenia, zmeny špecifikácie, priority riešenia . . . . . | 41 |
| 5.1 Zmeny atribútov entít a vyhodnocovaných údajov . . . . .    | 41 |
| 5.1.1. Oddelenie . . . . .                                      | 41 |
| 5.1.2 Zákazník . . . . .  | 41 |
| 5.1.4 Regál . . . . .   | 42 |
| 5.2 Zmeny algoritmu pohybu zákazníkov . . . . .                 | 42 |
| 5.3 Priorita riešení . . . . .                                  | 42 |
| 6. Určenie spôsobu testovania a údajov na testovanie . . . . .  | 44 |
| 6.1 Testovanie transakcií v obchodnom dome . . . . .            | 44 |
| 6.2 Testovanie bodov zastavenia . . . . .                       | 44 |
| 6.3 Testovanie archivácie . . . . .                             | 44 |
| 6.4 Testovanie užívateľského rozhrania a integrácia . . . . .   | 45 |
| 7. Výber implementačného jazyka a prostredia . . . . .          | 46 |
| 8. Štandardy kódovania . . . . .                                | 47 |
| 8.1 Zdrojové súbory . . . . .                                   | 47 |
| 8.2 Objekty a atribúty . . . . .                                | 47 |
| 9. Opis implementácie jednotlivých modulov . . . . .            | 49 |
| 9.1 Modifikácie knižnice Tiny . . . . .                         | 49 |
| 9.2 Vzor Observer . . . . .                                     | 50 |
| 9.3 Body zastavenia . . . . .                                   | 50 |
| 9.4 Vyhodnocovanie výrazov . . . . .                            | 50 |
| 9.5 Entity obchodného domu . . . . .                            | 52 |
| 9.6 Algoritmus pohybu zákazníka . . . . .                       | 53 |
| 9.7 Štatistiky . . . . .  | 54 |
| 10. Testovanie . . . . .  | 55 |
| 10.1 Testovanie transakcií . . . . .                            | 55 |
| 10.2 Testovanie bodov zastavenia . . . . .                      | 55 |
| 10.3 Testovanie archivovania údajov . . . . .                   | 55 |
| 10.4 Testovanie užívateľského rozhrania . . . . .               | 55 |
| 11. Záver . . . . .   | 56 |

## **Prílohy**

|   |     |
|---|-----|
| Príloha A: Ponuka .....   | A-1 |
| Príloha B: Príklad vstupného súboru .....                                 | B-1 |
| Príloha C: Záznamy zo stretnutí .....                                     | C-1 |
| Príloha D: Posudok špecifikácie a hrubého návrhu .....                    | D-1 |
| Príloha E: Preberacie protokoly .....                                     | E-1 |
| Príloha F: Posudok prototypu .....  | F-1 |
| Príloha G: Dopracovanie nedostatkov špecifikácie .....                    | G-1 |
| Príloha H: Testovacie dáta - súbor s definíciou simulovaného modelu ..... | H-1 |
| Príloha I: Údaje na testovanie bodov zastavenia .....                     | I-1 |
| Príloha J: Dopracovanie nedostatkov užívateľskej príručky .....           | J-1 |

# **I. ŠPECIFIKÁCIA**

# 1. ÚVOD

Cieľom projektu je vytvoriť simulačný systém umožňujúci simuláciu bežných procesov prebiehajúcich v obchodnom dome, ich sledovanie a vyhodnocovanie. Na základe výstupu systému môže užívateľ (riaditeľ obchodného domu) rozhodovať o zmene vlastností obchodného domu, napríklad o určení počtu košíkov, počtu pokladní, rozmiestnení tovaru do oddelení. Rozhodnutia systému vychádzajú zo zadaných kategórií zákazníkov, rozloženia ich príchodu do obchodného domu a ich vlastností.

## 1.1 Zadanie projektu

Navrhnuť a implementovať systém, umožňujúci simulovať vývoj systému (obchodného domu) v čase a vhodným spôsobom sprostredkovať užívateľovi. Úlohou projektu je:

- vyšpecifikovať model obchodného domu, jeho atribúty, objekty v systéme, ich atribúty, stavy systému, prechody medzi stavmi a podmienky týchto prechodov
- navrhnuť dostatočne univerzálne rozhranie, ktoré by malo dovoliť pridávanie užívateľských objektov umožňujúcich rozširovanie možností vizualizácie a vyhodnocovania
- navrhnuté riešenie musí pokrývať riadenie toku riadenia (zastavenie, spustenie simulácie), vizualizáciu atribútov objektu a ich modifikáciu; zároveň treba navrhnuť prostriedky pre vyhodnocovanie vývoja systému v závislosti od určitých faktorov (napr. prejavenie reklamy na množstve predaného tovaru, určenie dostatočného množstva košíkov, pokladní a pod. )
- používateľ musí mať možnosť interaktívne vstupovať do simulačného výpočtu a vhodnými prostriedkami modifikovať parametre systému a zasahovať tak do jeho vývoja
- výsledný návrh implementovať.

## 1.2 Ohraničenia

Základným ohraničením je čas určený na projekt, ktorý ovplyvňuje ďalšie rozhodnutia - nebude pravdepodobne možné výrazne rozširovať vedomosti členov tímu. Možnosti tímu vychádzajú zo súčasných skúseností.

Tieto faktory ovplyvňujú aj výber platformy. Systém bude vyžadovať operačný systém Windows 95/98/NT; požiadavky na hardvér vyplývajú z nárokov operačného systému. Pravdepodobné implementačné prostredie bude Microsoft Visual C++ 5.0.

Od pôvodnej požiadavky aby systém mal vymeniteľný vizualizačný front-end sme vzhľadom na komplexnosť riešenia odstúpili. Keďže sa nejedná o vytváranie systému na simuláciu ľubovoľných systémov, ale konkrétne obchodného domu, sústredili sme sa viac na poskytnutie služieb vhodných pri stanovovaní optimálnych vlastností obchodného domu. Systém bude umožňovať rozširovanie vizualizácie a vyhodnocovania pridávaním užívateľom definovaných objektov.

## 1.3 Riešiteľský kolektív

Všetci členovia tímu ukončili bakalárske štúdium v odbore Informatika na FEI STU.

### Juraj Mikluš

Študuje 1. ročník inžinierskeho štúdia na FEI STU odbor softvérové inžinierstvo. Má skúsenosti s programovaním v C/C++ (MS DOS, Windows).

**Andrej Oriško**

Študuje 1. ročník inžinierskeho štúdia na FEI STU odbor softvérové inžinierstvo.  
Má skúsenosti s programovaním v C/C++ (Macintosh, Windows), Java, AppleScript a tvorbou HTML dokumentov.

**Miroslav Panák**

Študuje 1. ročník inžinierskeho štúdia na FEI STU odbor softvérové inžinierstvo.  
Má skúsenosti s programovaním v C/C++ (MS DOS, Windows).

**Boris Steiner**

Študuje 1. ročník inžinierskeho štúdia na FEI STU odbor softvérové inžinierstvo.  
Má skúsenosti s programovaním v C/C++ (MS DOS, Windows 95/NT, Linux), MFC, DAO, Qt, X Windows, KDE.

**Martin Szolgay**

Študuje 1. ročník inžinierskeho štúdia na FEI STU odbor softvérové inžinierstvo.  
Má skúsenosti s programovaním v C/C++ (MS DOS, Windows).

## 2. PLÁN PROJEKTU

Cieľom projektu je vytvoriť nový simulačný systém s využitím súčasných poznatkov získaných počas štúdia, prípadne mimo školy. Ako vzor, či skôr „odrazový mostík“ poslúži systém Tiny, ktorý je známy všetkým členom tímu z predmetu Modelovanie a simulácia.

Činnosť v zimnom semestri bude spočiatku pozostávať so stretnutí so zadávateľom, analýzy a definície požiadaviek. Na základe týchto údajov sa vytvorí projektová dokumentácia. Vypracovanie jednotlivých častí dokumentácie sa pridelí členom tímu a bude sa priebežne upresňovať a kontrolovať ich úplnosť a konzistentnosť. Vzhľadom na časové obmedzenie, nie je možné aby sa členovia tímu začali zoznamovať s novou problematikou, čo bude ovplyvňovať pridelovanie konkrétnych úloh.

Základnou prioritou a cieľom v prvej polovici semestra je vypracovať projektovú dokumentáciu, ktorá je potrebná v druhej polovici semestra pri vytváraní prototypu a nevyhnutná pre úlohy v letnom semestri. Hlavné ciele možno zhrnúť do nasledovných bodov:

1. vypracovať presnú špecifikáciu
2. navrhnuť vhodnú hierarchiu tried
3. postupne implementovať systém.

Priority pri návrhu a implementácii vzhľadom na jednotlivé časti systému možno zoradiť takto:

1. základné triedy modelujúce časti obchodného domu
2. základná funkčnosť simulátora
3. grafické rozhranie a vizualizácia
4. zabezpečenie vstupu modelu obchodného domu zo súboru
5. kompletne riadenie simulácie
6. modifikovanie simulovaného modelu
7. používanie externých modulov.

### 2.1 Podrobný plán činnosti

#### Stretnutie 1 (5. týždeň)

Cieľom prvého stretnutia je formulácia základných požiadaviek na systém zadávateľom. Jedná sa o upresnenie textu témy, určenie základných služieb a ohraničení systému na základe konzultácie so zadávateľom.

Zadanie témy bude potrebné upresniť v nasledujúcich bodoch:

1. Užívateľ musí mať dostatočné možnosti špecifikácie modelu. Je potrebné presne určiť, do akej miery užívateľ presne špecifikuje geometriu obchodného domu, príchod zákazníkov a ich kategórie a správanie zákazníkov v obchodnom dome.
2. Systém musí pravdepodobne umožňovať vizualizáciu vybraných veličín nielen obchodného domu ako celku, ale aj jednotlivých elementárnych objektov (napr. pokladní).
3. Je možné modifikovanie modelu počas simulácie (čo zvyšuje požiadavky na systém)? Ak áno, systém bude musieť okrem samotnej simulácie slúžiť aj ako editor modelu. Modifikovaný model je vhodné uložiť naspäť do súboru.
4. Je potrebné ukladať aj stav simulácie?



Na základe prvého stretnutia vykonať prvý pokus o písomnú formuláciu požiadaviek na systém; identifikovať základné objekty systému a ich vzťah k základným objektom simulácie (transakcia, entita, obslužné miesto). Dokument pripraví vedúci tímu (Andrej Oriško).

Členovia tímu sa priebežne oboznámia s poslednou verziou knižnice Tiny. V učebni D132 sa preskúmajú dostupné vývojárske nástroje. V budúcnosti bude vhodné mať informácie o možnostiach vytvárania COM objektov v prostredí MS Visual C++ 5.0 (Boris Steiner).

### **Stretnutie 2 (6. týždeň, stretnutie členov tímu bez zadávateľa)**

Diskusia na základe pripraveného dokumentu, rôzne pohľady členov tímu na možné riešenie. Analýza základných požiadaviek, návrhy na spôsob špecifikácie správania zákazníka (transakcie) a jeho atribútov. Rozlíšenie atribútov zadaných užívateľom a atribútov generovaných systémom. Určenie základných obmedzení.

### **Stretnutie 3 (6. týždeň)**

Cieľom stretnutia je analýza problému a hrubý návrh riešenia; najmä však určenie ohraničení na vyššej úrovni detailnosti na základe konzultácie so zadávateľom. Prezentácia základnej štruktúry a správania systému podľa predošlého stretnutia (Andrej Oriško).

Presné určenie atribútov entít a miery náhodnosti. Slovná formulácia (v prirodzenom jazyku) akcií systému na príklade príchodu jedného zákazníka do obchodného domu a algoritmu jeho rozhodovania.

Detailnejšie vymedzené užívateľské rozhranie a možnosti užívateľa. Presné určenie termínu „riadenie toku riadenia simulácie“ (spustenie, zastavenie, ladiace body atď. ).

Určenie veličín, ktoré sa budú vyhodnocovať a spôsob ich zobrazenia. Na základe týchto informácií vypracujú členovia tímu pridelené úlohy. Rozdelenie vypracovania projektovej dokumentácie medzi členov tímu:

1. Úvod (Juraj Mikluš)
2. Plán projektu (Andrej Oriško)
3. Kontext systému (zatiaľ neurčené)
4. Špecifikácia funkcií systému (Boris Steiner)
5. Špecifikácia údajov v systéme (Martin Szolgay)
6. Špecifikácia správania systému (Miroslav Panák)
7. Ďalšie požiadavky a ohraničenia (zatiaľ neurčené)

### **Stretnutie 4 (7. týždeň)**

Cieľom stretnutia je analýza problému, hrubý návrh riešenia. Členovia tímu sa oboznámia s celou doteraz vypracovanou dokumentáciou. Kontrola vypracovanej dokumentácie - úplnosť, presnosť, konzistencia, odhad reálnych možností. Diskusia o sporných otázkach.

Predchádzajúce stretnutia sa sústreďovali viac na model a samotnú simuláciu; pravdepodobne bude potrebné vyjasniť možnosti vyhodnocovania a vizualizácie (ktoré veličiny sa budú merať a ako sa budú prezentovať). Taktiež bude potrebné reagovať na požiadavku na vytvorenie univerzálneho rozhrania na vizualizáciu - zhodnotiť a nájsť reálne implementovateľné a zároveň dostatočne flexibilné a použiteľné riešenie.

Spojenie predbežnej verzie dokumentácie. Určenie obsahu nepridelených bodov dokumentácie. Rozdelenie úloh, pokračovanie v pridelených úlohách. Vytvorenie prototypu web stránky tímu (Andrej Oriško).

### **Stretnutie 5 (8. týždeň)**

Príprava finálnej verzie projektovej dokumentácie. Spojenie dokumentácie do jedného celku. Členovia tímu upozornia na prípadné chyby, ktoré našli pri štúdiu dokumentácie.

Vyriešiť ostatné body dokumentácie - Kontext systému a Ďalšie požiadavky a ohraničenia. Zvážiť, v ktorých častiach dokumentácie je vhodné použiť grafické zobrazenie a vytvoriť ho.

Kontrola všetkých častí dokumentu vrátane formálnej stránky (číslovanie strán), príloh, titulnej a záznamov zo stretnutí (vrátane tohto stretnutia).

Na základe možností členov tímu určenie člena tímu, ktorý zabezpečí grafickú úpravu a tlač dokumentu. Iný člen tímu zabezpečí zviazanie dokumentácie. Obnovenie web stránky.

### **Stretnutie 6 (9. týždeň)**

Vypracovanie posudku špecifikácie a hrubého návrhu iného tímu. Všetci členovia tímu by mali byť oboznámení s obsahom dokumentácie iného tímu už pred stretnutím. Diskusia a rozdelenie úloh a vypracovanie posudku, určenie člena tímu, ktorý zabezpečí grafickú úpravu a tlač dokumentu.

### **Stretnutie 7 (10. týždeň)**

Dopracovanie zistených nedostatkov v dokumentácii a návrh prototypu vybraných častí. Príprava tvorby prototypu. Presné určenie, prototyp ktorých modulov sa bude implementovať. Pridelenie úloh členom tímu na základe ich znalostí. Pravdepodobne skúsenejší členovia tímu sa budú plne venovať tvorbe prototypu, ostatní budú dopĺňať dokumentáciu. Definitívny výber implementačného prostredia a použitých knižníc na tvorbu prototypu.

### **Stretnutie 8 (11. týždeň)**

Implementácia prototypu vybraných častí.

### **Stretnutie 9 (12. týždeň)**

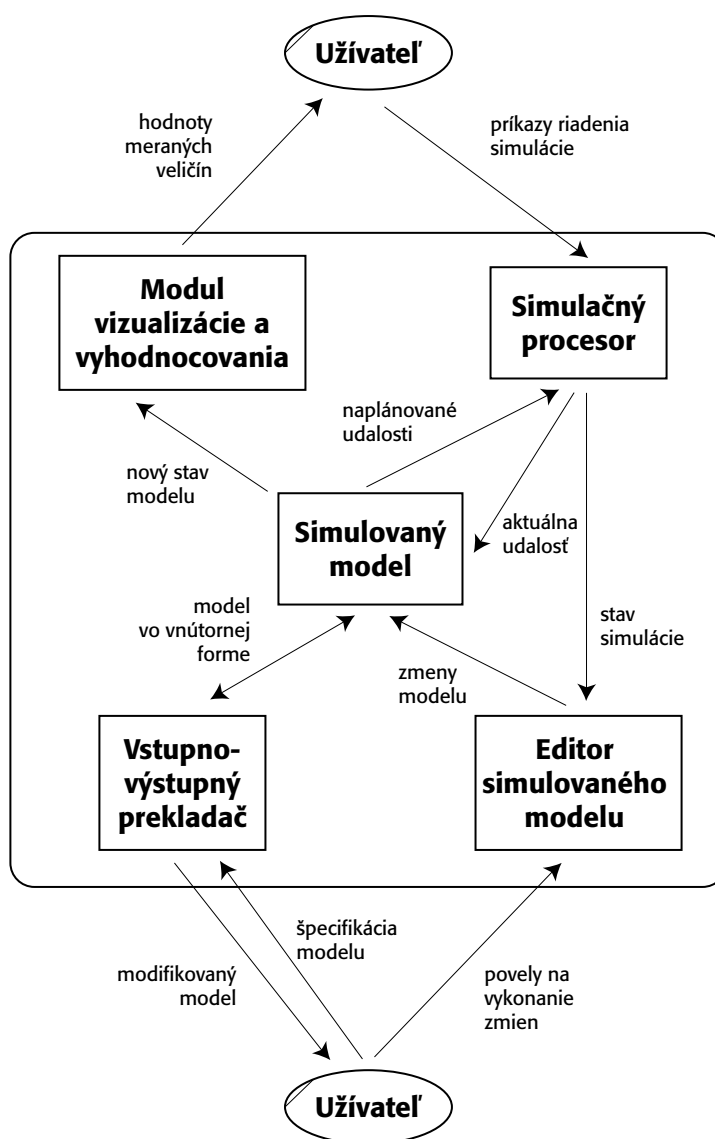
Dokončenie implementácie prototypu vybraných častí a príprava prezentácie. Kontrola doplnenej dokumentácie, tlač, obnova web-stránky.

### 3. MODEL SYSTÉMU

Na základe požiadaviek na systém a ich charakteru možno vytvoriť model systému a vykonať dekompozíciu na prvej úrovni nasledovným spôsobom:

1. simulačný procesor
2. simulovaný model
3. modul vizualizácie a vyhodnocovania
4. editor simulovaného modelu
5. vstupno/výstupný prekladač.

Štruktúra modelu je zrejmá z obr. 1.



Obr. 1. Moduly systému dekomponovaného na 1. úrovni

#### 3.1 Funkcie modulov a ich rozhrania

Jadrom celého systému je modul simulačný procesor. Tento modul riadi chod simulácie, spravuje kalendár udalostí - plánuje a vyberá udalosti a udržiava čas. Má priradený práve

jeden simulačný model. Simulácia je spustená na príkaz užívateľa. Priebeh simulácie sa zastavuje podľa zadaných parametrov (body zastavenia, čas zastavenia). Simulačný procesor je schopný riadiť simuláciu ľubovoľného modelu.

Simulačný procesor úzko súvisí s modulom simulovaný model, ktorý je v prípade tohto systému model obchodného domu. Model určuje pohyb transakcií (správanie zákazníka), a vykonáva elementárne akcie (obsluha, dopĺňanie zásob) v pridelenom čase. Komunikáciu a vzťah týchto dvoch modelov možno označiť ako vzťah master - slave.

Modul vizualizácie a vyhodnocovania je modul určený na získavanie informácií o simulácii. Skladá sa z väčšieho množstva menších modulov, z ktorých každý akumuluje a vyhodnocuje informácie potrebné na vyhodnotenie konkrétnej funkcie. Moduly tohto typu môžu byť interné alebo externé moduly doplnené užívateľom; sú určené predovšetkým na globálne vyhodnocovanie. Lokálne vyhodnocovanie údajov (napr. určovanie využitia jednej pokladne a jeho časový priebeh) síce logicky patrí do modulu vizualizácie ale fyzicky bude pravdepodobne súčasťou samotného modelu, resp. jeho komponentov.

Editor simulovaného modelu odzrkadľuje požiadavku na umožnenie modifikovanie simulovaného modelu užívateľom. Spolupracuje s modulom vstupno/výstupný prekladač, ktorý zabezpečuje preklad definície modelu zo vstupného súboru do vnútornej formy a naopak. Editor priamo modifikuje aktuálny (simulovaný) model; nemôže ho však modifikovať, pokiaľ je simulácia práve spustená, čo vyžaduje povolenie na modifikáciu od modulu Simulačný procesor a tiež upozornenie modulov vyhodnocujúcich správanie systému (zmena štruktúry systému môže ovplyvniť namerané výsledky).

### **3.2 Kontext systému**

Z predchádzajúceho popisu vyplývajú interakcie systému s okolím. Primárnym a vlastne jediným externým zdrojom informácií pre systém je užívateľ. Vytváraný systém je jedno-užívateľský; nerozlišujú sa typy a prístupové práva jednotlivých užívateľov.

Užívateľ špecifikuje model simulovaného systému, ktorý je základným vstupom pre systém. Tento vstup systém prijme pomocou modulu vstupno/výstupný prekladač.

Užívateľ ovplyvňuje činnosť simulačného procesoru - môže ho kedykoľvek zastaviť, spustiť, nastaviť zastavenie na konkrétny čas alebo akciu entity modelu.

Užívateľ prijíma vyhodnotené informácie, vyberá z nich, ktoré práve potrebuje a určuje spôsob ich zobrazenia.

Prostredníctvom editoru modelu užívateľ modifikuje samotný model. Zmenený model je možné opäť pomocou modulu vstupno/výstupný prekladač zapísať do súboru (vyprodukuje sa výstup, ktorý je možné opäť použiť ako vstup).

V neposlednom rade užívateľ dopĺňa vlastné externé moduly, ktoré poskytujú vlastné výstupy s vlastnou štruktúrou a formou.

## 4. ŠPECIFIKÁCIA FUNKCIÍ SYSTÉMU

Táto časť špecifikácie pokrýva užívateľský pohľad na funkcionálnosť simulačného systému.

### 4.1 Vytvorenie, načítanie a zapísanie modelu

Systém bude schopný načítať model systému z textového súboru. Vstupný súbor je možné editovať i mimo prostredia systému. Model bude zaznamenaný v presne definovanom formáte, ktorý bude popisovať jednotlivé vlastnosti komponentov (vrátane rozdelenia náhodnej premennej a jej parametre) pre triedu údajov (viď. definíciu dát pre bližšie informácie o triedach údajov typov):

#### Zákazník

- typ zákazníka
- potreba
- preferencia

#### Pokladňa

- intervaly obsluhy
- počet a trvanie prestávok

#### Regál

- doba spracovania
- množstvo a typ tovaru
- oneskorenie pri doplňovaní tovaru

#### Tovar

- druh, poddruh tovaru
- názov
- cena
- reklamovanosť

Zo vstupného súboru sa načítajú väzby jednotlivých komponentov a hierarchia modelu. Ak sa načítanie podarí bude vytvorený model. V systéme bude existovať práve jeden model. Ak počas načítania súboru dôjde k chybe, nevytvorí sa žiadny model.

Duálnou funkciou k funkcii otvorenia bude funkcia uloženia modelu. Model sa bude ukladať do súboru v rovnakom formáte ako je vstupný formát. Súbor bude možné opätovne načítať.

Pri ukladaní modelu sa uložia len statické atribúty modelu, zákazníka a tovarov t. j. charakteristika komponentov. Uloženie dynamického stavu viď. odsek 4.3 Dynamické zmeny modelu.

Tento prístup umožní jednoduchý zápis veľkého množstva modelov s možnosťami definovať vlastnosti jednotlivých komponentov tak, aby zodpovedali predstave užívateľa. Vstupný súbor modelu bude zapísaný v textovom formáte, čo umožní editovanie popisu modelu aj mimo prostredia simulátora.

Funkcie vytvorenia, načítania a zapísania modelu majú nižšiu prioritu.

## 4.2 Riadenie simulácie

Ak v systéme existuje platný model bude možné spustiť simuláciu tohto modelu. Ak model neexistuje, nebude sa dať simulácia spustiť.

Bežiacu simuláciu bude kedykoľvek možné zastaviť a znovu spustiť. Bude možné si prezerať históriu priebehu simulácie celého modelu ale aj jednotlivých komponentov.

Bude možné prezerať rad naplánovaných udalostí, premiestňovať jednotlivé udalosti v rade.

Bude možné naplánovať zastavenie simulácie, ak stav niektorého komponentu alebo hodnota atribútu dosiahne požadovanú hodnotu (watchpoint), alebo v konkrétnom čase.

Riadenie simulácie má najvyššiu prioritu.

## 4.3 Dynamické zmeny modelu

Ak v systéme existuje platný model, bude do tohto modelu možné pridávať a odoberať z neho komponenty aj počas behu simulácie. Bude možné pridať alebo odobrať novú pokladňu alebo nové oddelenie. V prípade, že model by nebol pri pridaní alebo odobratí v konzistentnom stave - napr. v oddelení sú zákazníci, systém sa bude snažiť vykonať operáciu až v čase, keď bude operácia viesť ku konzistentnému stavu.

V systéme bude existovať funkcia zastavenia činnosti komponentu a jeho opätovné spustenie. Taktiež bude možné naplánovať pozastavenie činnosti komponentu na určitý čas.

Zmeny v štruktúre modelu bude možné zaznamenať do výstupného súboru v rovnakom formáte ako má vstupný súbor. Bude možné zaznamenať aj dynamický stav modelu do zvláštneho súboru. Umožní ukladať viacero verzií toho istého modelu.

Dynamický stav bude zaznamenaný stavom atribútov statických entít. Toto je možné porovnať so situáciou, keď všetci zákazníci odídu z obchodného domu, ale zachová sa množstvo peňazí, ale čakacie rady budú prázdne.

Ukladanie dynamických zmien modelu má nízku prioritu.

## 4.4 Vizualizácia

Počas behu simulácie bude možné sledovať stav, zmeny alebo priebeh zmien atribútov všetkých komponentov. Tieto atribúty alebo ich priebeh budú vizualizované týmito prostriedkami: 2D-graf, histogram, tabuľka.

Zobrazované budú len atribúty statickej zložky modelu t. j. obchodný dom ale nie zákazníci. Zobrazované budú závislosti atribútov komponentov od času napr. využitie pokladne, regálu.

Model bude vizualizovaný stromom, z ktorého bude možné vybrať množinu ľubovoľných atribútov alebo komponentov v každej úrovni modelu a sledovať priebeh ich zmien počas simulácie. Tiež bude možné určiť bod zastavenia (watchpoint) na ľubovoľný atribút ľubovoľného komponentu.

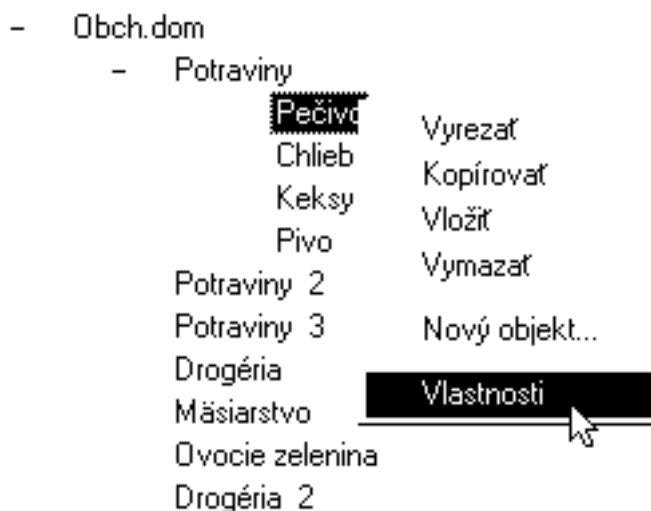
Zobrazené atribúty budú periodicky obnovované počas behu simulácie tak, aby zodpovedali skutočnému stavu modelu.

Užívateľ bude môcť rozširovať funkcie vizualizácie pomocou externých modulov. Externé moduly sú pasívne komponenty, ktoré môžu sledovať stav simulácie a zmeny statických entít. Údaje akumulované externým modulom sú pre ne špecifické ako aj spôsob zobrazenia. Každý modul zobrazuje vyhodnotené výsledky vo vlastnom priestore (okne).

Užívateľ môže priradiť každému komponentu externý modul uvedením CLSID COM rozhrania vo vstupnom súbore. Simulátor bude komunikovať s externým modulom pomocou COM rozhrania.

Systém bude vizualizovaný prostriedkami štandardného GUI MS Windows 95. Užívateľ bude komunikovať so simulátorom užívateľským menu a kontextovými menu. Objekty simulačného modelu budú mať kontextové menu, ktoré budú obsahovať príkazy charakteristické pre dané objekty (pozri obr. 2). Ovládanie bude zjednodušené klávesovými skratkami. Riadenie simulácie bude zjednodušené toolbarmi a bude podobné ako krokovanie programu v prostrediach pre vývoj aplikácií vo vyšších jazykoch (napr. Visual C++)

Vizualizácia má vysokú prioritu. Komunikácia s externými modulmi má najnižšiu prioritu.



Obr.2. Návrh užívateľského rozhrania na modifikáciu štruktúry modelu

## 4.5 Vyhodnocovanie výsledkov

Okrem pozorovania statických častí bude systém obsahovať aj vyhodnocovanie globálnych výsledkov. Globálne budú vyhodnocované položky pre:

### Oddelenie

- Počet ľudí v závislosti od času,
- Počet ľudí v závislosti dennej doby
- Maximálna a minimálna dĺžka vstupného radu
- Množstvo peňazí v pokladni v závislosti od času
- Typ zákazníkov
- Priemerná dĺžka radu na vstupe

### Zákazník

- Priemerná doba čakania
- Čas strávený v oddelení
- Počet odchodov z radov pre ich veľkú dĺžku
- Počet odchodov kvôli netrpezlivosti zákazníkov

Systém bude poskytovať po ukončení alebo zastavení simulácie výber komponentov s atribútmi zodpovedajúcimi kritériám užívateľa.

Výber bude možné uskutočniť z týchto atribútov:

- dĺžka radu
- obsadenie oddelenia
- využitie pokladne
- výpadky pokladne
- počet prestávok
- využitie regálov
- intervaly dopĺňovania tovaru do regálov
- počet zákazníkov ktorí odišli od regálov bez tovaru
- počet netrpezlivých zákazníkov pre jednotlivé oddelenia

Užívateľ bude môcť vyberať komponenty podľa stavu atribútov definovaním kritérií pre atribúty. Kritérium bude tvorené porovnávaním s číselnými hodnotami alebo priemernou hodnotou. Pri tvorbe kritérií budú použité relačné operátory  $<$ ,  $>$ ,  $\leq$  a  $\geq$ . Takto bude užívateľ môcť vybrať komponenty s malou priepustnosťou, s dlhými čakacími radmi alebo regále s veľkým dopytom a dlhým časom doplnenia stavu. Toto umožní užívateľovi nájsť najzaťaženejšie alebo najnevyužitejšie komponenty v modeli. Podľa potreby môže meniť štruktúru modelu, tak aby využitie komponentov bolo optimálne.

Vyhodnocovanie výsledkov má veľmi vysokú prioritu.



## 5. ŠPECIFIKÁCIA ÚDAJOV V SYSTÉME

### 5.1 Triedy údajov

V simulačnom systéme sa zaznamenávajú údaje o entitách patriacich do tohoto systému, prípadne s ním interagujúcich. Entity sa dajú rozdeliť do tried príbuznosti. V nami vytvorenom modeli systému rozoznávame tieto údajové entity:

- Zákazník
- Pokladňa
- Regál
- Tovar
- Oddelenie

### 5.2 Popis tried údajov

#### 5.2.1 Zákazník

Trieda zákazník modeluje zákazníka obchodného domu. Obsahuje údaje o vlastnostiach zákazníka, z ktorých sa dá odvodiť jeho správanie v obchodnom dome, druh kupovaného tovaru a iné činnosti potrebné pre simuláciu systému. Potrebné vlastnosti zákazníka sú tieto:

- kategória zákazníka - definovaná užívateľom, reťazec znakov
- potreba - zoznam pravdepodobností kúpy jednotlivých druhov tovaru podľa potrieb zákazníka (tento tovar zákazník „musí“ kúpiť), v percentách
- preferencia - zoznam pravdepodobností výberu jednotlivých tovaru na základe zhliadnutia reklamy, v percentách
- trpezlivosť - číslo vyjadrujúce maximálnu dĺžku radu, do ktorého je zákazník ochotný postaviť sa (keď už zákazník stojí v rade, neodíde z neho)
- hotovosť - suma peňazí, ktoré má zo sebou, rozdelenie náhodnej premennej
- solventnosť - suma peňazí, ktorú je ochotný minúť navyše, rozdelenie

Sledované údaje:

- čas príchodu a odchodu zákazníka, v sekundách
- suma minútých peňazí za potrebný tovar
- suma minútých peňazí za reklamovaný tovar
- počet tovarov, ktoré zákazník nedostal kúpiť
- počet radov, do ktorých sa nepostavil

Na odvodenie správania sa zákazníka sa použije hodnotová funkcia, ktorá na základe vlastností a daných náhodných rozdelení vytvorí zoznam nakupovaných tovarov. Táto funkcia sa vyhodnotí pred vstupom do obchodného domu.

#### 5.2.2 Pokladňa

Táto trieda modeluje pokladňu s jedným alebo viac obslužnými miestami. Obsahuje rad. Dĺžka radu nie je zhora ohraničená. Dané vlastnosti tejto triedy sú:

- počet obslužných miest
- pravdepodobnosť výpadku obslužného miesta v percentách

Štatisticky sledované údaje:

- maximálna, priemerná a aktuálna dĺžka radu (počet zákazníkov)

- priemerné zaťaženie radu - ako je pokladňa využitá, v percentách
- doba obsluhy zákazníka - ako dlho stál zákazník v rade, v sekundách
- suma peňazí v pokladni

### 5.2.3 Regál

Táto trieda modeluje regál s jedným alebo viac obslužnými miestami. Obsahuje rad. Dĺžka radu nie je zhora ohraničená. Pri nízkom množstve tovaru sa doplní zo skladu s určitým oneskorením. V tejto entite sa zaznamenávajú tieto údaje:

- názov tovaru v regáli (jeden regál obsahuje len jeden tovar)
- množstvo tovaru v regáli
- oneskorenie pri dopĺňaní a minimálne množstvo tovaru

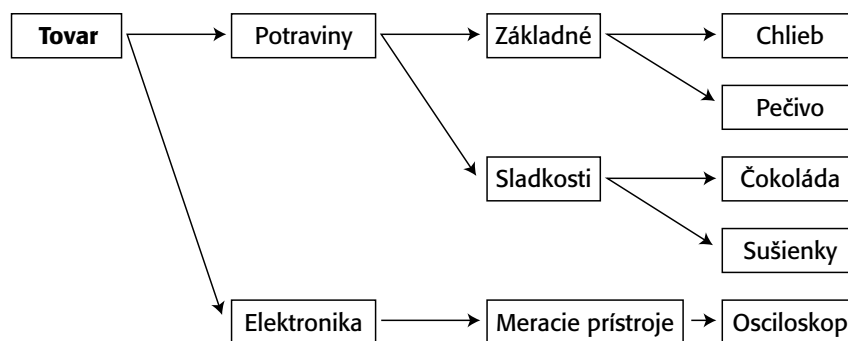
Štatisticky sledované údaje:

- maximálna, priemerná a aktuálna dĺžka radu
- doba obsluhy zákazníka - ako dlho stál zákazník v rade
- počet zákazníkov, ktorí odišli neuspokojení
- celkový počet kúpeného tovaru

### 5.2.4 Tovar

Tovar predstavuje všetky druhy tovaru, vyskytujúce sa v obchodnom dome. Je reprezentovaný hierarchicky - rozdelený na podtriedy (obr. 3). Vytvorili sme tri abstrakcie delenia:

1. druh tovaru - napr. potraviny, elektronika, odevy,...
2. poddruh - napr. potraviny - mliečne výrobky, pekárenské výrobky, mäso,...
3. konkrétny tovar - chlieb, pečivo, mlieko, maslo,...



Obr. 3. Hierarchické usporiadanie tovaru

Vlastnosti evidované pri tovare:

- názov - musí byť jedinečný, jednoznačne určuje entitu
- druh - vyššia úroveň hierarchie, druh, do ktorého tovar patrí
- cena jednotkového množstva tovaru
- reklamovanosť - je/nie je reklamovaný (v zmysle prebieha/neprebieha práve reklamná kampaň na tovar)

Tovar predstavuje hierarchiu, štatistické údaje o tovare sa sledujú v regáloch.

### 5.2.5 Oddelenie

Predstavuje jedno oddelenie obchodného domu. Skladá sa z entít typu pokladňa a regál. Slúži na sledovanie jemu patriacich entít a vzťahov medzi nimi. Umožňuje globálnejší pohľad na parametre a výstupy entít. Oddelenie má vstupný rad. Zadávané vlastnosti:

- názov - musí byť jedinečný, jednoznačne určuje oddelenie
- kapacita oddelenia (počet košíkov)
- zoznam pokladní a regálov patriacich oddeleniu

Sledované údaje:

- maximálny, priemerný a aktuálny počet zákazníkov v oddelení

## 6. ŠPECIFIKÁCIA SPRÁVANIA SYSTÉMU

Výsledkom dekompozície systému na prvej úrovni bolo rozdelenie na tieto moduly:

1. simulačný procesor
2. simulovaný model
3. modul vizualizácie a vyhodnocovania
4. editor simulovaného modelu
5. vstupno/výstupný prekladač

Jednotlivé moduly a ich funkcie (obr. 1) sú popísané v kapitole 3. Moduly navrhovaného systému je možné rozdeliť vzhľadom na ich správanie na niekoľko skupín: aktívne, pasívne a pomocné (doplnkové).

### 6.1 Aktívne komponenty

V systéme sa nachádzajú dva aktívne komponenty - simulačný procesor a simulovaný model. Ich vzťah však nie je rovnocenný - nadradený modul (procesor) riadi model. Simulovaný model vykonáva akcie (spracúva udalosti a generuje nové udalosti) len v čase pridelenom simulačným procesorom.

#### 6.1.1 Simulačný procesor

Prostredníctvom simulačného procesoru užívateľ riadi priebeh simulácie (napr. nastavuje body zastavenia). Túto činnosť možno popísať algoritmom:

1. Načítanie modelu zo vstupného súboru  
Komunikácia s modulom vstupno/výstupný prekladač  
Ak nie je možné spracovať vstup, koniec
2. Inicializácia simulácie  
Vytvorenie prázdneho kalendáru udalostí, nastavenie simulačného času na 0
3. Výber nasledujúcej udalosti z kalendára udalostí
4. Posun simulačného času na čas výskytu udalosti
5. Kontrola, či na aktuálny čas nie je nastavený bod zastavenia  
Ak áno, zastavenie simulácie, kým užívateľ nedá príkaz na pokračovanie
6. Kontrola, či nie je nastavené zastavenie na zmenu stavu entity, súvisiacej s aktuálnou udalosťou  
Ak áno, zastavenie simulácie, kým užívateľ nedá príkaz na pokračovanie
7. Obsluha udalosti modelom  
Udalosť spracuje konkrétna entita modelu a prípadne naplánuje iné udalosti
8. Skok na bod č. 3

#### 6.1.2 Simulovaný model

Simulovaný model (model obchodného domu) pozostáva z troch statických (oddelenie, pokladňa a regál) a dvoch dynamických entít (zákazník a košík). Entita košík je realizovaná obmedzením počtu zákazníkov v oddelení. Pomocou algoritmu správania sa zákazníka (transakcie) možno vysvetliť činnosť statických entít (obr. 4).

Pred vstupom zákazníka do systému sa nastaví jeho atribúty: vytvorí sa zoznam tovarov, ktoré zákazník chce kúpiť (podľa priority), určí sa obsah peňaženky a pod. Na základe zoznamu tovarov sa vytvorí zoznam oddelení, ktoré treba navštíviť.

Pri príchode zákazníka do oddelenia sa najprv skontroluje, či nie je rad na košíky pre zákazníka príliš dlhý. Ak je príliš dlhý, zákazník odíde do ďalšieho oddelenia. Inak sa zaradí do radu na košíky. Po zaplatení pri pokladni zákazník odchádza do ďalšieho oddelenia. Voľný košík sa prideli prvému z radu na košíky. Tento zákazník prichádza ku regálu s tovarom, ktorý má nakúpiť.

Regál skladuje jeden druh tovaru. Zákazník vyberá z regálu tovar. Ak sa tovar v regáli minie, doplní sa o daný čas (atribút regálu). Zákazník potom odchádza od regálu bez tovaru a pokúša sa kúpiť iný tovar z tej istej skupiny. Zákazník postupne prechádza k ďalším regálom s tovarmi, ktorý má nakúpiť. Nakoniec sa zaradí do radu ku pokladni.

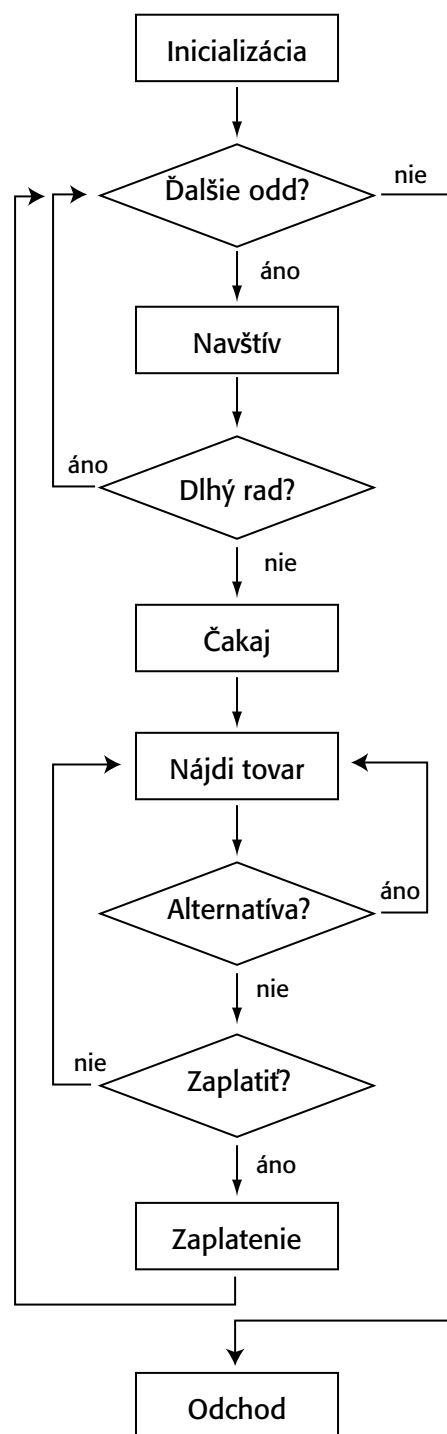
Pokladňa obsluhuje zákazníkov v rade na zaplatenie nakúpeného tovaru. Zákazník zaplatí za tovar, odíde od pokladne a ku pokladni príde nasledujúci z radu na zaplatenie. Pokladňa uchováva sumu zaplatených peňazí.

## 6.2 Pasívne komponenty

Modul vizualizácie a vyhodnocovania je pasívny modul určený na zber informácií o simulácii. Je tvorený množinou menších modulov akumulujúcich a vyhodnocujúcich špecifické informácie potrebných na vyhodnotenie konkrétnej funkcie. Pre činnosť týchto modulov je nevyhnutné, aby mali informácie o zmene stavu konkrétnych entít modelu. Entity modelu majú priradený zoznam objektov, ktoré je potrebné informovať o zmene stavu. Správa o zmene stavu je zaslaná objektu synchrónne.

## 6.3 Pomocné komponenty

Medzi pomocné komponenty sme zaradili moduly vstupno/výstupný prekladač a editor modelu. Tvoria rozhranie medzi systémom a užívateľom. Vstupno/výstupný prekladač prekladá definíciu modelu uloženú v textovom súbore do vnútornej formy a naopak. Príklad vstupného súboru sa nachádza v prílohe B. Okrem prekladu samotnej definície modelu zabezpečuje aj preklad stavu simulácie. Editor modelu umožňuje zmeny v štruktúre aktuálneho modelu; zmeny musia byť povolené simulačným procesorom (simulácia musí byť zastavená).



Obr. 4. Algoritmus správanie sa zákazníka

## 7. ĎALŠIE POŽIADAVKY A OHRANIČENIA

Počas stretnutí so zadávateľom v súvislosti s procesom pochopenia a vyjasnenia požiadaviek na systém vyvstali nasledujúce požiadavky a z nich vyplývajúce obmedzenia:

1. Modifikovanie modelu užívateľom. Medzi dvoma krajnými alternatívami - (1. ) nevytvoriť editor modelu s grafickým rozhraním, čiže všetky zmeny modelu by musel užívateľ vykonať zmenou vstupného súboru systému v nejakom textovom editore alebo (2. ) umožniť meniť model iba so špeciálnym editorom modelu (vstupný súbor by bol binárny) sme zvolili kompromisné riešenie. Vstupný súbor bude modifikovateľný bez špeciálneho editora modelu. Model sa bude dať meniť aj prostredníctvom editoru, ktorý bude súčasťou systému, čo bude výhodné počas vývoja systému, pretože editor bude implementovaný pravdepodobne až v neskorších fázach vývoja.
2. Systém nebude uvažovať geometriu obchodného domu. Obchodný dom si možno predstaviť ako jednoposchodovú budovu so štvorcovým pôdorysom so halou v strede a oddeleniami po obvode. Počet zákazníkov v hale je neobmedzený; tento fakt však možno prehliadnúť vzhľadom na spôsob pohybu zákazníka, ktorý sa v hale nezdržuje - okamžite po východe z oddelenia vchádza do iného alebo opúšťa obchodný dom.
3. Spôsob rozhodovania zákazníka. Zákazník postupne navštevuje oddelenia, do ktorých potrebuje ísť. Ak sa do nejakého oddelenia nedostal (lebo bol príliš dlhý rad na košíky) alebo v ňom nekúpil, čo potreboval, nebude sa to pokúšať spraviť druhý raz. Analogická situácia nastáva pri regáloch. Nerozlišuje sa, či zákazník videl reklamu na konkrétny tovar; zákazník je charakterizovaný jednou hodnotou vyjadrujúcou mieru, akú pozornosť venuje reklame.
4. Sklad. Sklad je neobmedzený. Regále sa dopĺňajú s daným oneskorením ak počet tovaru v nich klesne pod stanovenú hodnotu. Týmto oneskorením možno do určitej miery simulovať aj prípadné objednanie tovaru. Zaujímavé (no časovo nerealizovateľné) by bolo spojenie systému s iným systémom realizujúcim sklad.
5. Ako implementačný jazyk bol zvolený C++, vzhľadom na znalosti a skúsenosti členov tímu. Inak by sa dalo uvažovať o implementácii v Java, čím by sa odstránilo obmedzenie na konkrétnu platformu a v určitých aspektoch zjednodušila implementácia.
6. Pôvodná požiadavka na kompletne vymeniteľný front-end bola zredukovaná na umožnenie rozširovania možností vizualizácie a vyhodnocovania pridávaním užívateľom definovaných objektov (pozri úvod). Prípadná výmena modulu riadenia simulácie v budúcnosti by nemala byť obtiažna vzhľadom na vlastnosti objektovo orientovaného prístupu (nahradenie triedy simulátora inou).

## **II. UŽÍVATELSKÁ PRÍRUČKA**

# 1. ÚVOD

Navrhnutá aplikácia je určená na simulovanie bežných procesov prebiehajúcich v obchodnom dome, ich sledovanie a vyhodnocovanie. Výsledky týchto simulácií môžu pomôcť pri rozhodovaní o optimalizácii vlastností obchodného domu, napríklad určovaní kapacity oddelení, počtu pokladní a množstva tovaru v regáloch. Rozhodnutia systému vychádzajú zo zadaných kategórií zákazníkov, rozloženia ich príchodu do obchodného domu a ich vlastností. Táto príručka poskytuje všetky potrebné informácie potrebné na inštaláciu, nastavenie parametrov a používanie programu.

Typické činnosti pri používaní programu možno rozdeliť do niekoľkých okruhov:

- vytvorenie, otvorenie a uloženie špecifikácie modelu obchodného domu do súboru
- modifikácia vlastností modelu obchodného domu
- riadenie simulácie
- vyhodnocovanie výsledkov simulácie
- nastavenie parametrov aplikácie
- inštalácia a požiadavky na systém.



## 2. VYTVORENIE, OTVORENIE A ULOŽENIE ŠPECIFIKÁCIE MODELU OBCHODNÉHO DOMU DO SÚBORU

Pred začatím simulácie je potrebné špecifikovať simulovaný obchodný dom prostredníctvom simulačného modelu. Model je možné vytvoriť priamo v simulačnom programe, prípadne načítať z disku (vstupný súbor má textový formát).

Na vytvorenie nového modelu sa použije funkcia „Nový“ z ponuky „Súbor“. Vytvorí sa prázdny model. Popis pridávania objektov do modelu je uvedený v kapitole 3. Súčasne môže byť otvorený len jeden simulačný model.

Načítanie objektu z disku sa vykoná funkciou „Otvoriť...“ z ponuky „Súbor“. Zobrazí sa štandardný dialóg na otvorenie súboru. Po vybratí súboru sa načíta model. Súčasne s modelom sa načíta aj dynamický stav modelu, teda stav, v ktorom sa model nachádzal naposledy.

Uloženie modelu na disk pre budúce použitie sa vykoná funkciou „Uložiť“ z ponuky „Súbor“. Súbor sa uloží na disk pod menom, ktoré mu bolo naposledy priradené. V prípade, že súbor bol práve vytvorený a nemá priradené meno, zobrazí sa štandardný dialóg na zadanie mena súboru. Tento dialóg na zvolenie mena súboru sa zobrazí aj pri výbere funkcie „Uložiť ako...“ z ponuky „Súbor“.

Pre jeden model môže byť uložených viacero dynamických stavov, čo poskytuje možnosť prípadného návratu ku konkrétnemu stavu modelu. Stav modelu (okrem základného, ktorý je uložený spolu s definíciou modelu) sa ukladajú do samostatných súborov. Súbory používané programom sú teda dva:

1. súbor obsahujúci definíciu modelu a dynamický stav modelu v čase uloženia tohto súboru („sim“ súbor)
2. súbor obsahujúci iba dynamický stav modelu („dyn“ súbor). Uloženie aktuálneho dynamického stavu modelu sa vykoná zvolením položky „Uložiť dynamický stav...“ v ponuke „Súbor“ a zadaním názvu súboru v nasledujúcom dialógu. Ukladanie je jednorázové, t.j. pri opätovnom výbere položky „Uložiť dynamický stav...“ je potrebné zadať nový názov súboru a pri voľbe „Uložiť“ sa dynamický stav ukladá len do „sim“ súboru.

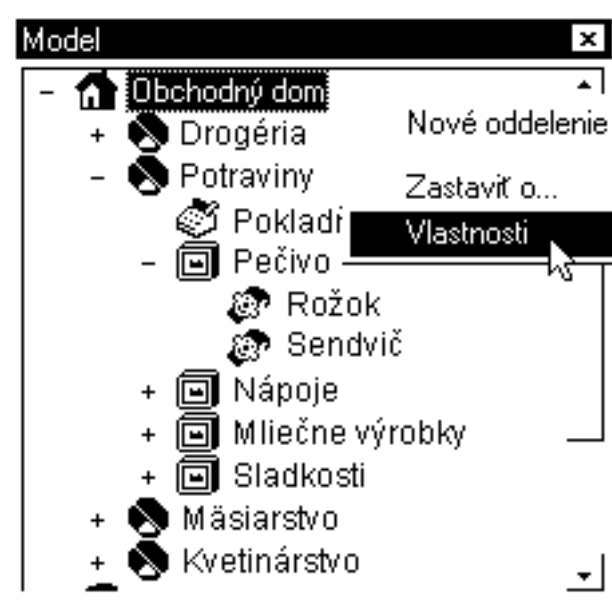
Poznámka: súbory obsahujúce dynamický stav modelu nie je možné použiť s iným modelom ako s tým, ktorého stav obsahuje. Pri použití uloženého dynamického stavu po zmene definície modelu budú mať nové objekty počiatočný stav.

### 3. MODIFIKÁCIA VLASTNOSTÍ MODELU OBCHODNÉHO DOMU

Simulovaný obchodný dom sa špecifikuje pomocou simulačného modelu. Program umožňuje vytvárať štruktúru obchodného domu a modifikovať jeho vlastnosti. Model sa zobrazuje v paneli definície obchodného domu, ktorý sa zobrazí výberom položky „Model“ v ponuke „Okno“.

Panel obchodného domu obsahuje tri hierarchické zoznamy:

1. Štruktúra obchodného domu - obsahuje oddelenia, ich pokladne, regále a tovar nachádzajúci sa v regáloch (obr. 1).
2. Hierarchia tovaru obsahuje kategórie a konkrétne druhy tovarov. Nie všetky definované tovary sa musia nachádzať v regáloch. Tovary sa rozdeľujú do kategórií. Počet, názov a priradenie tovarov do kategórií je definované užívateľom a závisí od jeho potrieb a konkrétnej úlohy.
3. Skupiny zákazníkov. Obsahuje všetky skupiny zákazníkov, ktorí môžu obchodný dom navštíviť.



Obr. 1. Hierarchické usporiadanie elementov obchodného domu

Nový objekt do štruktúry sa pridá zvolením položky „Vložiť nový objekt“ z ponuky „Úpravy“. Nové objekty sa pridávajú na úroveň v hierarchii o jeden stupeň nižšiu ako je úroveň označenej položky. Napríklad, ak je práve zvolený vrchol hierarchie, vytvorí sa nové oddelenie. Alternatívnou možnosťou, ako vytvoriť nový objekt, je položka „Vytvoriť nový objekt“ v kontextovej ponuke každého objektu. Presné znenie položky obsahuje aj typ objektu, ktorý sa vytvorí.

Vlastnosti objektov sa menia tak, že zvolený objekt označíme a vyberieme položku „Vlastnosti“ z ponuky „Úpravy“. Ekvivalentnou možnosťou je výber položky „Vlastnosti“ z kontextovej ponuky objektu. Podrobnejšie informácie o vlastnostiach jednotlivých objektoch sa nachádzajú v kapitole 5.1.

Odstraňovanie objektu sa robí položkou „Vymazať“ z ponuky „Úpravy“. Objekty na najvyššej úrovni (Obchodný dom a Tovar) sa nedajú vymazať. Kontextová ponuka obsahuje väčšinu príkazov z ponuky „Úpravy“ a je to rýchlejší spôsob zmien.

Poznámka: môžete použiť aj drag&drop. Objekty na rovnakej úrovni hierarchie je možné ľubovoľne presúvať a kopírovať; týmto spôsobom je možné aj rozmiestňovať tovar do regálov.

## 4. RIADENIE SIMULÁCIE

Po vytvorení modelu sa môže spustiť simulácia. Z ponuky „Simulácia“ vyberieme funkciu „Štart“. Simulácia prebieha počas otváracích hodín obchodného domu, ktoré sa nastavujú spolu s jeho ostatnými vlastnosťami.

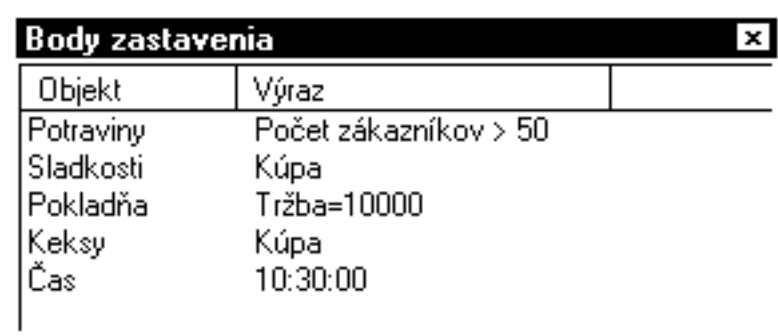
Zastavenie simulácie sa vykoná zvolením funkcie „Zastaviť“ v ponuke „Simulácia“. Opätovným použitím príkazu „Štart“ bude simulácia pokračovať od miesta, kde bola prerušená. Príkaz „Ukončiť“ v ponuke „Simulácia“ má za následok úplné ukončenie simulácie a vynulovanie všetkých zaznamenaných výsledkov. Simuláciu je potom možné spustiť od začiatku.

V prípade potreby sa dá simulácia prezerať po jednotlivých krokoch (všetkých naplánovaných udalostiach) príkazom „Krokovat“. V prípade, že je potrebné zastaviť iba v určitej situácii je omnoho výhodnejšie použiť bod zastavenia. Program poskytuje tri základné druhy bodov zastavenia:

1. Zastavenie v danom čase. Výberom funkcie „Zastaviť o...“ z ponuky „Simulácia“ sa dá zadať časový okamih, v ktorom sa celá simulácia zastaví.
2. Zastavenie pri zmene dynamického stavu objektu. Pomocou tejto funkcie (položka „Zastaviť pri zmene...“ v ponuke) je možné nastaviť špecifickejšie podmienky zastavenia. Funkcia je aktívna len v prípade, keď je v modeli vybraný objekt, ktorému sa nastavuje podmienka (t.j. oddelenie, regál alebo pokladňa). Po jej zvolení sa zobrazí dialóg na nastavenie podmienky.
3. Zastavenie pri kúpe tovaru alebo tovaru z kategórie. Túto funkciu (položka „Zastaviť pri kúpe...“ v ponuke) umožňuje zastaviť simuláciu pri kúpe určitého druhu tovaru v danom množstve. Množstvo sa počíta pre všetky regále obsahujúce daný tovar. Funkcia je aktívna iba ak je zvolený v hierarchii tovaru nejaký tovar alebo kategória.

Funkcie na nastavenie bodov zastavenia sú prístupné aj z kontextovej ponuky každého objektu. Popis parametrov bodov zastavenia je uvedený v samostatnej kapitole.

Všetky druhy bodov zastavenia sú prehľadne zobrazené v paneli bodov zastavenia (obr. 2), kde je možné ich dodatočne modifikovať, vymazať alebo dočasne deaktivovať. Tieto akcie možno vykonať pomocou kontextovej ponuky. Panel bodov zastavenia sa zobrazí výberom položky „Body zastavenia“ v ponuke „Okno“.



| Objekt    | Výraz                 |
|-----------|-----------------------|
| Potraviny | Počet zákazníkov > 50 |
| Sladkosti | Kúpa                  |
| Pokladňa  | Tržba=10000           |
| Keksy     | Kúpa                  |
| Čas       | 10:30:00              |

Obr. 2. Panel so zoznamom bodov zastavenia

## 4.1 Body zastavenia

### 4.1.1 Zastavenie v konkrétnom čase

Výberom funkcie „Zastaviť o...“ z ponuky „Simulácia“ alebo z kontextovej ponuky obchodného domu sa zobrazí dialógové okno, v ktorom sa špecifikuje časový okamih, kedy sa simulácia zastaví. Zadaná hodnota musí byť intervale otváraciej doby obchodného domu, ktorá sa nastavuje spolu s jeho ostatnými vlastnosťami. Čas sa zadáva v 24-hodinovom časovom cykle aj so sekundami. Čas je možné zadať aj len v sekundách (napr. „36127“).

Vytvorený bod zastavenia je možné zrušiť alebo dočasne deaktivovať výberom príslušnej položky v kontextovej ponuke bodu zastavenia v paneli bodov zastavenia.

### 4.1.2 Zastavenie pri zmene objektu

Funkcia slúži na zastavenie simulácie vždy, ak sú splnené zadané podmienky, týkajúce sa zmien dynamických atribútov objektu (oddelenia, regálu alebo pokladne). Užívateľ špecifikuje jednu alebo viac podmienok.

Medzi zadanými podmienkami je vzťah „a súčasne“. Podmienka je tvorená dynamickým atribútom objektu, relačným operátorom a hodnotou, s ktorou sa bude porovnávať. Atribúty a prípustné operátory sa vyberajú zo zoznamu.

Podmienky sa pridávajú na koniec zoznamu pomocou tlačidla „Viac možností“, posledná podmienka sa odstraňuje tlačidlom „Menej možností“.

Funkcia sa aktivuje výberom položky „Zastaviť pri zmene...“ z ponuky „Simulácia“ alebo z kontextovej ponuky oddelenia, pokladne alebo regálu. Funkcia je v ponuke aktívna len v prípade, keď je v paneli definície modelu vybraný objekt.

Spôsob pridávania podmienok a pravidiel, ktoré musia pre podmienky spĺňať sú obdobné ako vyhľadávaní objektov (pozri kapitolu 5.2).

Na rozdiel od vyhľadávania, zoznam operátorov pri číselných atribútoch obsahuje aj operátor „sa zmení o“ a pri logických atribútoch operátor „sa zmení“. Pomocou týchto operátorov je možné sformulovať podmienku zastavenia nezávisle od absolútnej hodnoty atribútu.

Vytvorený bod zastavenia je možné zrušiť alebo dočasne deaktivovať výberom príslušnej položky v kontextovej ponuke bodu zastavenia v paneli bodov zastavenia.

### 4.1.3 Zastavenie pri kúpe

Táto vlastnosť umožňuje zastaviť simuláciu pri kúpe určitého druhu tovaru alebo tovarov z určitej kategórie v danom množstve. Užívateľ špecifikuje jednu alebo viac podmienok; medzi zadanými podmienkami je vzťah „a súčasne“. Podmienka je tvorená atribútom, relačným operátorom a hodnotou, s ktorou sa bude porovnávať. Atribúty a prípustné operátory sa vyberajú zo zoznamu. V zozname atribútov sa nachádzajú tieto atribúty:

1. Počet kusov - je počet všetkých predaných kusov tovaru všetkým zákazníkom
2. Počet kusov (jeden zákazník) je počet kusov tovaru kúpených jedným zákazníkom
3. Kategória zákazníka - je kategória zákazníka vytvorená v paneli definície modelu

Podmienky sa pridávajú na koniec zoznamu pomocou tlačidla „Viac možností“, posledná podmienka sa odstraňuje tlačidlom „Menej možností“.

Funkcia sa aktivuje výberom položky „Zastaviť pri kúpe...“ z ponuky „Simulácia“ alebo z kontextovej ponuky tovaru alebo kategórie tovaru. Funkcia je aktívna len v prípade, keď je v paneli definície modelu vybraný objekt.

Spôsob pridávania podmienok a pravidiel, ktoré musia pre podmienky splňať sú obdobné ako vyhľadávani objektov (pozri kapitolu 5.2).

Na rozdiel od vyhľadávania, zoznam operátorov pri číselných atribútoch obsahuje aj operátor „sa zmení o“ a pri logických atribútoch operátor „sa zmení“. Pomocou týchto operátorov je možné sformulovať podmienku zastavenia nezávisle od absolútnej hodnoty atribútu.

Vhodným rozdelením tovaru do kategórií sa výrazne zjednoduší formulácia podmienky zastavenia pri kúpe, pretože namiesto zdĺhavého vymenúvania všetkých druhov tovaru stačí uviesť jeho kategóriu. Podobné zjednodušenia možno dosiahnuť jednotným pomenovaním kategórií.

Vytvorený bod zastavenia je možné zrušiť alebo dočasne deaktivovať výberom príslušnej položky v kontextovej ponuke bodu zastavenia v paneli bodov zastavenia.

**Parametre zastavenia** [X]

Zastaviť simuláciu pri kúpe tovaru (z kategórie) „Plniace pero“ ak bude platiť.

Počet kusov (jeden zák.) [v] Je viac ako [v] 5

Kategória zákazníka [v] Je [v] študent

Viac možností | Menej možností | Zrušiť | OK

Obr. 3. Nastavenie podmienok zastavenia pri kúpe tovaru

## 5. VYHODNOCOVANIE VÝSLEDKOV SIMULÁCIE

Výsledky simulácie sa vyhodnocujú priebežne. To znamená, že všetky výstupy, či už zobrazené grafy alebo číselné hodnoty odzrkadľujú aktuálny stav. Prínosom tohto prístupu je, že práca s programom nie je rozdelená do dvoch krokov - simulácie a po jej skončení prezeranie výsledkov. Ak sa počas simulácie stav modelu nevyvíja želaným spôsobom, simuláciu je možné zastaviť a zmeniť vlastnosti modelu. Na prehliadanie a vyhodnocovanie stavu modelu program poskytuje niekoľko spôsobov:

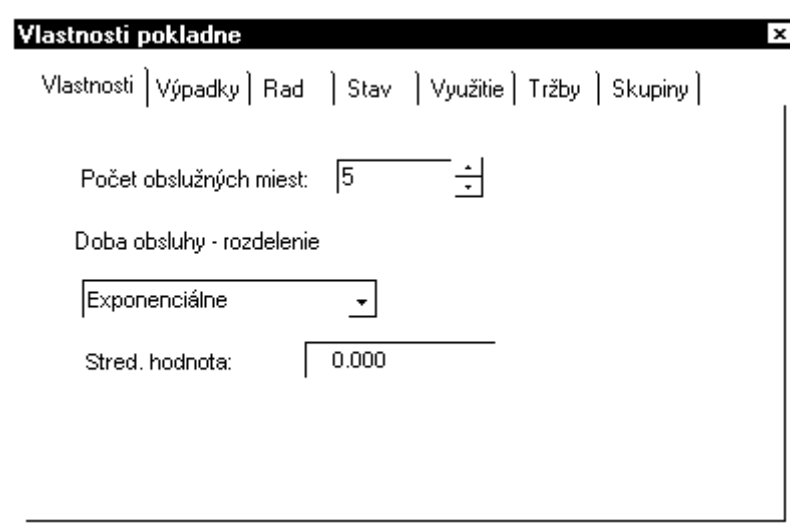
1. Zobrazenie vlastností objektov. V rámci vlastností objektov je zobrazený aktuálny stav, grafické znázornenie stavu objektov v čase a závislosti od iných atribútov
2. Vyhľadávanie objektov so zadanými vlastnosťami
3. Užívateľom doplnené možnosti vyhodnocovania prostredníctvom externých modulov

### 5.1 Vlastnosti elementov obchodného domu

Vlastnosti každého elementu obchodného domu je možné zobraziť dvoma spôsobmi:

1. výberom položky vlastnosti z kontextovej ponuky daného objektu
2. zvolením objektu v paneli s definíciou modelu a spustením funkcie vlastnosti v ponuke „Úpravy“.

Vlastnosti všetkých elementov obchodného domu sa zobrazujú v samostatných oknách obsahujúcich viacero strán (panelov). Vo všeobecnosti obsahuje prvý panel parametre objektu („Vlastnosti“), modifikovateľné užívateľom (obr. 4). Tieto parametre nie sú modifikovateľné ak je spustená simulácia. Panel „Stav“ obsahuje vždy aktuálne hodnoty charakteristických dynamických atribútov objektu, prípadne ich percentuálne vyjadrenie. Nasledujúce panely obsahujú zobrazenie priebehu stavu objektu v čase, prípadne iné závislosti (napr. závislosť počtu predaných kusov tovaru od kategórie zákazníkov). Počet a obsah týchto panelov závisí od typu objektu.

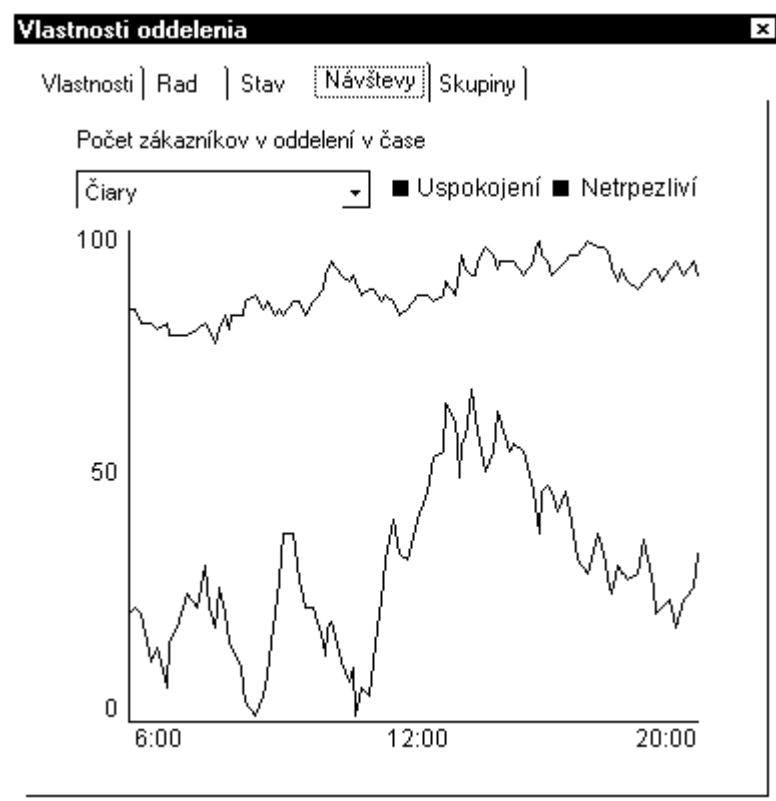


Obr. 4. Nastavenie vlastností pokladne

Spôsob zobrazenia grafov je voliteľný, pri x-y grafoch je možná voľba medzi bodovým, čiarovým, plošným zobrazením alebo histogramom (obr. 5). Zvolené typy zobrazenia sa

uchovávajú. Základné hodnoty pre zobrazenie sa dajú vybrať v nastavení aplikácie, ktoré je uvedené v kapitole 6.

Vhodným rozmiestnením okien s vlastnosťami objektov je možné sledovať priebeh zmien viacerých objektov naraz.



Obr. 5. Príklad vyhodnotenia časovej závislosti veličiny

## 5.2 Vyhľadávanie

Program umožňuje vyhľadávanie objektov v obchodnom dome, ktoré spĺňajú určité vlastnosti. Túto funkciu je výhodné použiť namiesto pracného manuálneho prezerania všetkých objektov, najmä pri rozsiahlejších modeloch.

Hľadané objekty musia byť rovnakého typu, čiže je možné vyhľadávať naraz iba pokladne, regále alebo oddelenia. Okrem typu objektu užívateľ špecifikuje jednu alebo viacero podmienok, ktoré musia pre každý nájdený objekt platiť. Medzi zadanými podmienkami je vzťah „a súčasne“. Podmienka je tvorená atribútom objektu, relačným operátorom a hodnotou, s ktorou sa bude porovnávať. Atribúty a prípustné operátory sa vyberajú zo zoznamu. Atribúty objektov zahŕňajú statické aj dynamické vlastnosti.

Podmienky sa pridávajú na koniec zoznamu pomocou tlačidla „Viac možností“, posledná podmienka sa odstraňuje tlačidlom „Menej možností“.

Rozlišujú sa štyri typy atribútov objektov: text (refazec), celé, reálne čísla a logické hodnoty. Od typu atribútov závisia ponúkané operátory.

Porovnávanie textových atribútov je možné ovplyvniť nastavením potrebnej kombinácie možností „rozlišovať malé a veľké písmená“ a „ignorovať diakritiku“ v nastavení aplikácie. Vyhľadávať možno aj podrefazce, pomocou operátorov „začína na“, „končí na“ alebo „ob-



sahuje“. Vhodným pomenovaním objektov je možné napríklad nájsť napríklad všetky regále obsahujúce televízory, ak pomenujeme konkrétne tovary (televízory) „Televízor zn. X“ a podobne.

Tie isté atribúty tých istých objektov sa môžu opakovať vo viacerých podmienkach. Týmto spôsobom je možné špecifikovať intervaly hodnôt (použitím „väčšie ako“ a „menšie ako“). Reálne čísla je možné zadávať aj v exponenciálnom tvare, napr.:  $2e+3$  je 2000.

Pri logických hodnotách je porovnávaná hodnota súčasťou operátora („pravda“, „nepravda“).

The screenshot shows a search dialog box with the following fields and options:

- Search for:
- Attribute:
- Operator:
- Comparison:
- Value:
- Attribute:
- Operator:
- Comparison:
- Value:

Buttons at the bottom:

Obr. 6. Vyhľadanie elementov modelu s danými vlastnosťami

Všetky podmienky musia mať zadané platné hodnoty a nesmú si navzájom odporovať. Zadávanie vylučujúcich sa podmienok, napríklad „x je menšie ako 3“ a súčasne „x je väčšie ako 5“ nie je povolené.

Súčasťou podmienky je okrem hodnoty, operátora a názvu atribútu, aj typ príslušného objektu. Toto riešenie umožňuje okrem jednoduchých foriem, napríklad nájsť všetky pokladni s tržbou väčšou ako 1000 aj omnoho zložitejšie formy. Je možné vyhľadávať objekty na základe vlastností objektov, ktoré obsahujú (sú na nižšej úrovni hierarchie), napríklad nájsť oddelenie s pokladňou, ktorá má tržbu väčšiu ako 1000 alebo dokonca vyhľadávať objekty na základe vlastností iných objektov na tej istej úrovni hierarchie: nájsť regále takého oddelenia, ktorého pokladňa má tržbu väčšiu ako 1000.

Stlačením tlačidla „OK“ sa začne vyhľadávanie. Simulácia je počas vyhľadávania zastavená. Výsledky vyhľadávania sú zobrazené zozname v samostatnom okne, dvojitým kliknutím na položku zoznamu sa daný objekt zobrazí a označí v paneli s definíciou modelu obchodného domu.

### 5.3 Externé moduly

Externé moduly zabezpečujú zber, vyhodnocovanie a zobrazovanie informácií o objektoch obchodného domu priradených užívateľom. Externé moduly zabezpečujú špecifické funkcie, ktoré aplikácia neposkytuje. Moduly sa musia nachádzať v podadresári „Externé moduly“ v hlavnom adresári aplikácie. Sú to dynamicky linkované knižnice vytvárané vo

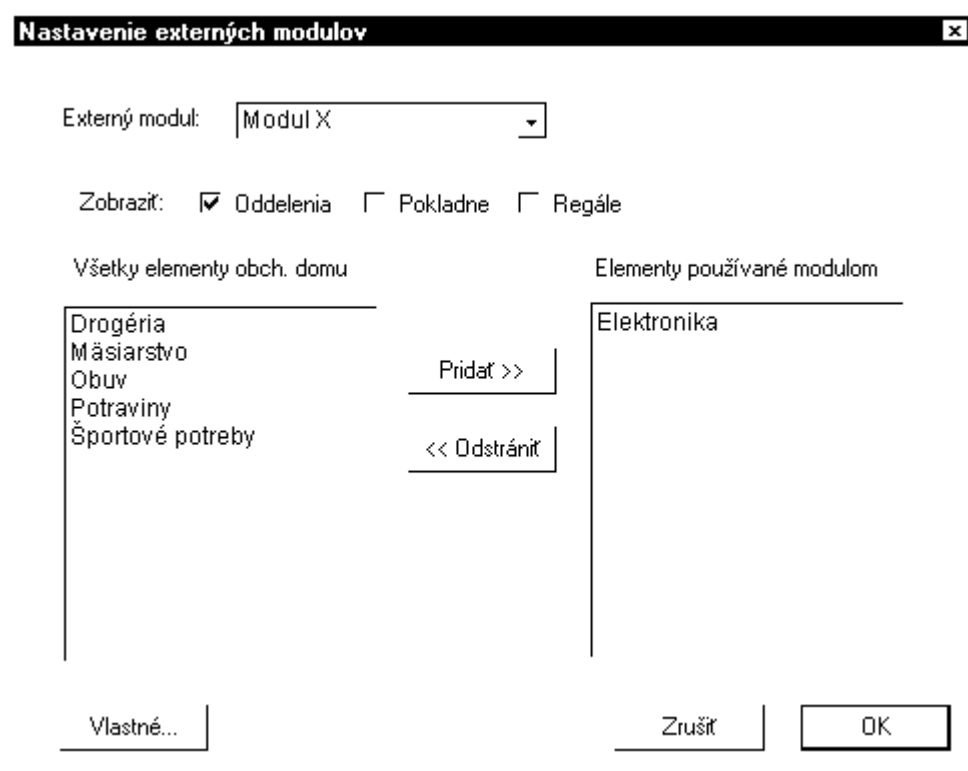
vyšších programovacích jazykoch. Spôsob vytvárania externých modulov nie je súčasťou užívateľskej príručky.

### 5.3.1 Nastavenie externých modulov

Vybráním položky „Externé moduly“ z ponuky „Simulácia“ sa otvorí dialóg, v ktorom sa dajú vybrať externé moduly a nastaviť ich parametre. Zvolené nastavenie je špecifické pre každý model.

Pri nastavovaní parametrov modulov je potrebné najskôr vybrať modul, ktorému sa má zmeniť jeho nastavenie zo zoznamu vo vrchnej časti dialógu. Tento zoznam obsahuje všetky nainštalované moduly. Ďalej treba nastaviť informácie, ktoré modul požaduje od simulačného programu. Môže sa pridať ľubovoľný počet sledovaných objektov. Pre sprehľadnenie používaných údajov je možné zapnúť filtrovanie kontextu údajov (oddelenia, pokladne, regále). Modul, ktorý nemá priradené žiadne sledované objekty, nie je aktívny. Špecifické nastavenia vlastné jednotlivým modulom sa dajú zmeniť pomocou tlačidla „Vlastné...“. Moduly nemusia podporovať všetky druhy objektov.

Nastavenie modulov potvrdíme stlačením tlačidla „OK“. V opačnom prípade dialóg ukončíme stlačením „Zrušiť“.



Obr. 7. Nastavenie parametrov externých modulov

## **6. NASTAVENIE PARAMETROV APLIKÁCIE**

Táto časť obsahuje popis nastavenia niektorých globálnych vlastností aplikácie a spôsoby prispôsobenia aplikácie potrebám užívateľa.

### **6.1 Nastavenie aplikácie**

Nastavenie špecifických parametrov simulácie umožňuje položka „Nastavenie“ v ponuke „Simulácia“. V otvorenom dialógu sa dá nastaviť typ zobrazovania grafu, vlastnosti vyhľadávania a automatické odstraňovanie použitých bodov zastavenia.

#### **Základný typ grafu**

Pri zobrazovaní časového priebehu stavu elementov obchodného domu je možné zvoliť typ grafu na zobrazenie tejto závislosti (body, čiary, histogram, plochy). Zvolený typ grafu sa uchováva pre každý graf spolu s definíciou modelu obchodného domu. Táto voľba špecifikuje počiatočné nastavenie pre každý graf.

#### **Parametre vyhľadávania**

Nastavenie týchto prepínačov (Ignorovať diakritiku, Rozlišovať veľké a malé písmená) ovplyvňuje vyhľadávanie elementov obchodného domu, konkrétne interpretáciu reťazcových atribútov vyhľadávaných elementov. Ďalšie informácie o vyhľadávaní sa nachádzajú v kapitole 5.2.

#### **Parametre automatického odstraňovania bodov zastavenia**

System bude automaticky odstraňovať body zastavenia po splnení podmienky zastavenia, aby systém nebol zastavovaný viackrát pri výskyte rovnakej udalosti. V niektorých prípadoch je užitočné čakať len na prvý výskyt konkrétnej udalosti. Užívateľ môže špecifikovať konkrétne typy bodov zastavení, ktoré sa budú odstraňovať (zastavenia podľa času, pri zmene objektu a pri kúpe tovaru). Odstraňovanie bodov zastavenia je samozrejme možné vykonať aj manuálne. Túto funkciu možno oceniť skúsení užívatelia. Pre začínajúcich užívateľov sa používanie tejto funkcie neodporúča, pretože môže dôjsť k neželaným odstráneniam bodov zastavenia.

### **6.2 Úpravy pracovnej plochy**

Simulačný systém umožňuje kvôli prehľadnosti zobrazovať a schovávať panely nástrojov. V systéme existuje viacero nástrojových panelov - štandardný panel („štandard“), panel riadenia simulácie („riadenie“), panel bodov zastavenia („body zastavenia“), panel definície simulačného modelu („model“) a panel záznamov o činnosti („záznam o činnosti“). Tieto panely sa dajú vypnúť a prípadne znovu zapnúť vybraním príslušnej položky v ponuke „Okno“, kde názov položky je zhodný z názvom panelu.

## 7. ČASTO KLADENÉ OTÁZKY

### **Ako zistím, či a kedy si zákazník z konkrétnej kategórie kúpil konkrétny tovar?**

Nastavíme bod zastavenia konkrétnemu tovaru na „Zastavenie pri kúpe“ a podmienku „Kategória zákazníka“ „je“ a napíšeme zvolené meno kategórie.

### **Ako zistím, že sa práve minul tovar v regále tovar?**

Nastavíme bod zastavenia danému regálu na „Zastavenie pri zmene“ a podmienku „Je naplánované dopĺňanie“ na „Pravda“.

### **Ako zistím, že oddelenie je preplnené?**

Nastavíme bod zastavenia danému oddeleniu na „Zastavenie pri zmene“, „Priemerná doba čakania“ „je väčšia ako“ a napíšeme zvolenú dobu čakania zákazníka, pri ktorej pokladáme oddelenie za preplnené.

### **Ako zistím kedy nastávajú výpadky pokladní?**

Nastavením bodu zastavenia pokladne na „Zastavenie pri zmene“, „Počet výpadkov“, „sa zmení o“ a napíšeme parameter 1. Pri každom výpadku sa simulácia zastaví.

### **Ako zistím stav obchodného domu v konkrétnom čase?**

V ponuke „Simulácia“ vyberieme položku „Zastavenie o...“ a nastavíme čas zastavenia požadovanú hodnotu. Po zastavení simulácie si môžeme prezrieť stav štandardným spôsobom.

### **Ako zistím, ktoré pokladne sú nevyťažené?**

Zvolíme položku „Hľadať“ z ponuky „Úpravy“. Nastavíme vyhľadávanie pokladní. Ako podmienku nastavíme „Pokladňa“, „Využitie“, „Je menej ako“ a napíšeme využitie, pre ktoré pokladáme pokladňu za nevyťaženú.

### **Mám otvorené veľké množstvo okien. Ako sprehladním obrazovku?**

Vyberieme položku „Rozložiť“ z ponuky „Okno“. Okná sa usporiadajú rovnomerne po celej obrazovke.

## **8. INŠTALÁCIA A POŽIADAVKY NA SYSTÉM**

### **8.1 Požiadavky programu na systém**

#### **Minimálna konfigurácia**

Operačný systém Windows 95/NT

512 KB diskového priestoru

Nie sú dané špecifické požiadavky na pamäť a rýchlosť procesora.

#### **Doporučená konfigurácia**

Procesor: Intel Pentium 133 MHz alebo kompatibilný

Pamäť: 32 MB RAM (48 MB Windows NT)

Rozlíšenie monitoru: 1024x768, HiColor

#### **Používanie externých modulov**

Používanie externých modulov môže výrazne zvýšiť nároky na procesor a pamäť. Uvedené hodnoty nezahŕňajú používanie náročnejších externých modulov. Konzultujte dokumentáciu k externým modulom, ktoré používate.

### **8.2 Inštalácia**

Program je distribuovaný na jednej 3.5“ diskete. V hlavnom adresári sa nachádza program „install.exe“, ktorý po spustení nainštaluje program do zvoleného adresára na pevnom disku. Na diskete sa nachádzajú aj posledné informácie týkajúce sa inštalovanej verzie programu - známe problémy, prípadne kompatibilita s inými produktami.

Poznámka 1: počas inštalácie sa nemodifikujú parametre systému (registry) ani ponuka „Štart“. Inštalácia pozostáva iba z dekompresie potrebných súborov.

Poznámka 2: autori si vyhradzujú právo zmeniť v prípade potreby obsah inštaláčnej diskety. V tomto prípade sa riadte ďalej pokynmi uvedenými na diskete.

### **8.3 Odstránenie inštalácie**

Vzhľadom na poznámku 1 uvedenú v bode 8.2 sa odstránenie programu realizuje vymazaním celého adresára, v ktorom sa nachádza program, štandardným postupom.

### **III. NÁVRH A IMPLEMENTÁCIA**

# 1. ÚVOD

Tento dokument obsahuje dokumentáciu vytvorenú v letnom semestri predmetu tímový projekt, ktorého náplňou je návrh a implementácia systému. Témou projektu je animácia simulačných experimentov s modelom diskrétného systému; cieľom je vytvoriť simulačný systém umožňujúci simuláciu bežných procesov prebiehajúcich v obchodnom dome, ich sledovanie a vyhodnocovanie.

Dokument obsahuje plán projektu, záznamy o jeho plnení, opis návrhu, implementácie a testovania a tiež dohodnuté konvencie používané členmi tímu. Dokument slúži ako návod pri implementácii systému pre členov tímu a dokumentuje prácu tímu v letnom semestri.

Na riešení úloh sa zúčastňujú všetci členovia tímu 11, tak ako boli predstavení v úvode špecifikácie v zimnom semestri. Úlohy členov tímu sú uvedené v pláne projektu. Pri návrhu a implementácii sa bude postupovať podľa špecifikácie vytvorenej v letnom semestri.

Užívateľská príručka bola vytvorená už v zimnom semestri ako súčasť dokumentácie k prototypu; prípadné zmeny v nej budú uvedené v dodatku.

## 2. PLÁN PROJEKTU

Plán projektu vychádza zo znalostí členov tímu a ich možností. Program a dokumentáciu je potrebné dokončiť do konca letného semestra.

Činnosti v letnom semestri možno rozdeliť do štyroch skupín: podrobný návrh, implementácia užívateľského rozhrania, implementácia simulátora a tvorba dokumentácie. Vzhľadom na obmedzený čas budú musieť tieto činnosti prebiehať viac-menej paralelne.

Užívateľské rozhranie zatiaľ nie je úplne implementované (nie sú vytvorené všetky dialógy, okná), je možné na ňom pracovať nezávisle od ostatných častí. Pokiaľ nebude úplne dokončený návrh, je možné implementovať elementárne, všeobecne použiteľné triedy.

Po dosiahnutí základnej funkčnosti simulátora, ktorý bude dovedty realizovaný a testovaný ako samostatná „faceless“ aplikácia sa simulátor spojí s užívateľským rozhraním (dodatočné spojenie musí byť zohľadnené v návrhu).

Základnou stratégiou je čo najskôr vytvoriť funkčný program a až potom sa plne venovať tvorbe dokumentácie. Časť dokumentácie však bude nutné vytvoriť už skôr, pretože bude slúžiť ako návod pre tých členov tímu, ktorí budú systém implementovať.

Rozdelenie úloh medzi členov tímu:

Juraj Mikluš - Implementácia simulátora a testovanie

Andrej Oriško - Návrh a dokumentácia, koordinácia práce

Miroslav Panák - Užívateľské rozhranie a dokumentácia

Boris Steiner - Implementácia užívateľského rozhrania

Martin Szolgay - Návrh a implementácia simulátora a testovanie

Rozdelenie úloh sa môže doplniť podľa aktuálnej situácie.

| Týž. | Činnosť                         |                                       |   |  |   |
|------|---------------------------------|---------------------------------------|---|--|---|
| 3    | Podrobný návrh tried simulátora | Implementácia užívateľského rozhrania | Implementácia základných tried                |  | Dopracovanie dokumentácie                                 |
| 4    | a simulovaného modelu           |                                       |   | Implementácia tried simulátora a simulovaného modelu | Vytvorenie testovacích dát a testovanie na rôz. úrovniach |
| 5    |                                 |                                       |   |  |   |
| 6    |                                 |                                       |   |  |   |
| 7    |                                 |                                       | Spojenie užívateľského rozhrania a simulátora |  |   |
| 8    | Tvorba dokumentácie             |                                       |   |  |   |
| 9    |                                 |                                       |   |  |   |
| 10   |                                 |                                       |   |  |   |
| 11   |                                 |                                       | Finálna verzia programu                       |  |   |
| 12   |                                 |                                       | Príprava prezentácie                          |  |   |
| 13   |                                 |                                       | Prezentácia                                   |  |   |

Obr. 1. Plán projektu na letný semester



### 3. ARCHITEKTÚRA SYSTÉMU

Navrhovaný systém sa skladá z dvoch subsystémov: zo simulátora obchodného domu a vizualizačného front-endu. Subsystém simulátora zabezpečuje funkcie týkajúce sa samotnej simulácie: simuláciu, nastavovanie a vyhodnocovanie bodov zastavenia, vyhľadávanie entít v obchodnom dome, čítanie a zápis definície obchodného domu zo súboru a štatistické vyhodnocovanie. Simulátor je založený na knižnici Tiny.

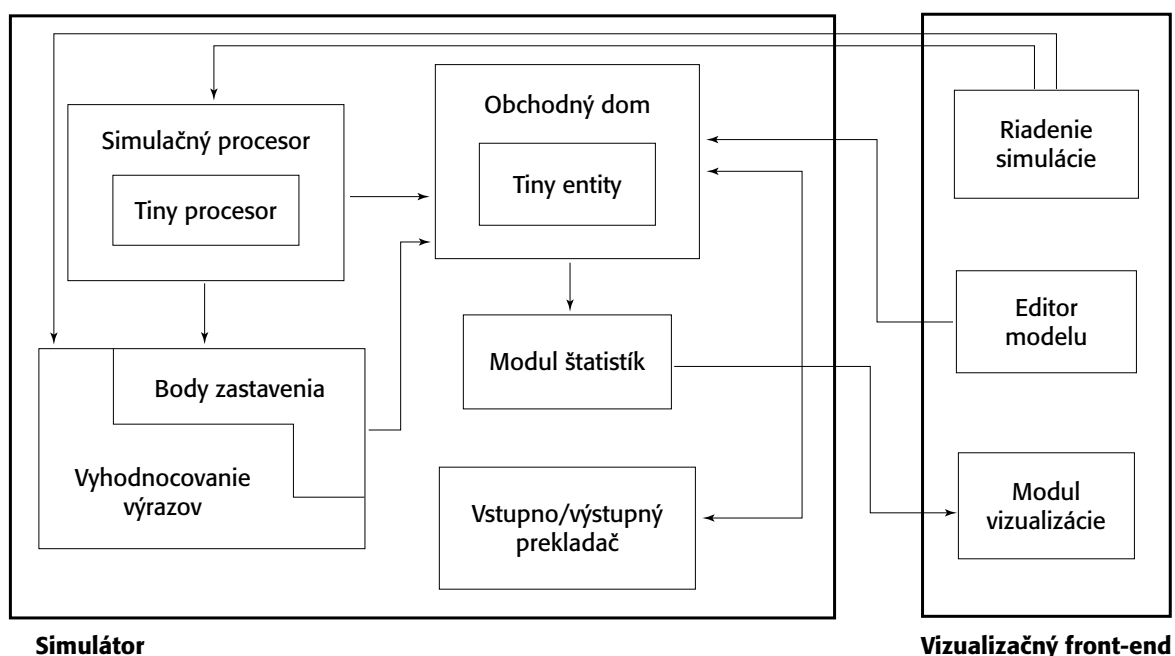
Vizualizačný front-end prispôsobuje rozhranie simulátora štandardnému užívateľskému rozhraniu cieľovej platformy. Úplne nahrádza užívateľské rozhranie simulátora, ktorý má iba jednoduché znakové rozhranie. Subsystém umožňuje riadiť simuláciu, určovať body zastavenia, modifikovať simulovaný model a zobrazovať stav entít.

Načrtnuté rozdelenie na subsystémy je logické; vo finálnej aplikácii sú obidva subsystémy spojené staticky a nie je možné ich od seba oddeliť. Možnosť rozdelenia má význam najmä pri implementácii a testovaní. Prípadné vytvorenie iného front-endu by bolo pomerne jednoduché.

Obidva subsystémy môžu byť aktívne súčasne; simulácia môže prebiehať paralelne s akciami vykonávanými užívateľom (otvorenie okna s informáciami o entite, výber z ponuky), pokiaľ je to možné.

Komunikácia subsystémov zodpovedá logicky modelu master-slave. Simulátor sa spustí a zastaví akciou užívateľa prostredníctvom užívateľského rozhrania vizualizačného front-endu. Modifikáciu bodov zastavenia a entít modelu taktiež vykonáva front-end. Jediným prípadom kedy simulátor je iniciátorom komunikácie je zaslanie správy o zmene stavu modelu alebo entity.

Zviazanosť subsystémov je slabá, subsystémy na sebe závisia iba v minimálnej miere. Subsystém simulátora je schopný pracovať aj samostatne. Spojenie je realizované odvodením nových tried a pomocou vzoru Observer (pozri ďalej).



Obr. 2. Dekompozícia systému

## 4. FYZICKÝ MODEL ÚDAJOV SYSTÉMU

### Použité typy atribútov

|        |   |
|--------|---|
| INT    | všeobecne použiteľné celé číslo (32 bitov, so znamienkom)                             |
| BOOL   | boolean hodnota (true/false)  |
| STRING | reťazec ľubovoľných znakov; predpokladá sa kódovanie CP-1250                          |
| LIST   | zoznam objektov; agregácia ak nie je uvedené inak                                     |
| OBJ    | objekt (agregácia)  |
| TIME   | čas v sekundách, celé číslo, 32 bitov, bez znamienka, počíta sa od 0:00:00 s          |
| RNP    | rozdelenie náhodnej premennej, obsahuje typ rozdelenia a podľa typu závislé parametre |
| CASH   | množstvo peňazí v nešpecifikovanej mene (double)                                      |
| U      | jedinečné   |
| K      | klúč  |
| V      | atribút nie je súčasťou dát uložených v súbore  |
| C      | atribút nie je modifikovateľný užívateľom   |

### Obchodný dom

|         |   |
|---------|---|
| TIME    | Otváracia doba (od)   |
| TIME    | Otváracia doba (do)   |
| LIST    | Oddelenia   |
| TIME CV | Simulačný čas   |
| INT CV  | Počet zákazníkov, ktorí navštívili obchodný dom                 |
| INT CV  | Počet zákazníkov, ktorí sú práve v obchodnom dome               |
| INT CV  | Celkové množstvo minútých peňazí                                |
| TIME CV | Celková doba, strávená v obchodnom dome                         |
| TIME CV | Celková doba čakania v rade                                     |
| INT CV  | Počet netrpezlivých zákazníkov, ktorí sa nedostali do oddelenia |
| INT CV  | Počet netrpezlivých zákazníkov, ktorí sa nedostali k regálu     |

### Oddelenie

|        |                          |
|--------|--------------------------|
| STRING | Názov                    |
| INT    | Kapacita (počet košíkov) |
| LIST   | Regále                   |
| OBJ    | Pokladňa                 |
| INT CV | Aktuálny počet ľudí      |
| INT CV | Maximálny počet ľudí     |
| INT CV | Počet netrpezlivých ľudí |

### Pokladňa

|     |                         |
|-----|-------------------------|
| INT | Počet obslužných miest  |
| RNP | Doba obsluhy            |
| RNP | Pravdepodobnosť výpadku |

|         |                                   |
|---------|-----------------------------------|
| RNP     | Dĺžka prestávky                   |
| INT CV  | Počet obsadených obslužných miest |
| CASH CV | Suma peňazí                       |
| INT CV  | Počet výpadkov                    |

## **Regál**

|         |   |
|---------|---|
| INT     | ID tovaru                                     |
| INT     | Maximálne množstvo tovaru                     |
| RNP     | Čas potrebný na doplnenie                     |
| INT     | Hranica dopĺňania tovaru (počet kusov)        |
| INT CV  | Aktuálne množstvo tovaru v regáli             |
| BOOL CV | Je/nie je naplánované dopĺňanie regálu        |
| INT CV  | Počet dopĺňaní                                |
| INT CV  | Celkový počet predaného tovaru                |
| INT CV  | Počet neuspokojených zákazníkov (spolu)       |
| INT CV  | - kvôli nedostatku tovaru v regáli            |
| INT CV  | - kvôli veľkému počtu zákazníkov pred regálom |
| INT CV  | Počet ľudí, ktorí chceli navštíviť regál      |
| INT CV  | Počet ľudí, ktorí navštívili regál            |

## **Tovar**

|         |                     |
|---------|---------------------|
| INT CKU | ID tovaru           |
| INT C   | ID kategórie tovaru |
| STRING  | Názov               |
| CASH    | Cena                |

## **Kategória tovarov**

|         |                            |
|---------|----------------------------|
| INT CKU | ID kategórie               |
| STRING  | Názov kategórie            |
| LIST    | Zoznam tovarov v kategórii |

## **Kategória zákazníkov**

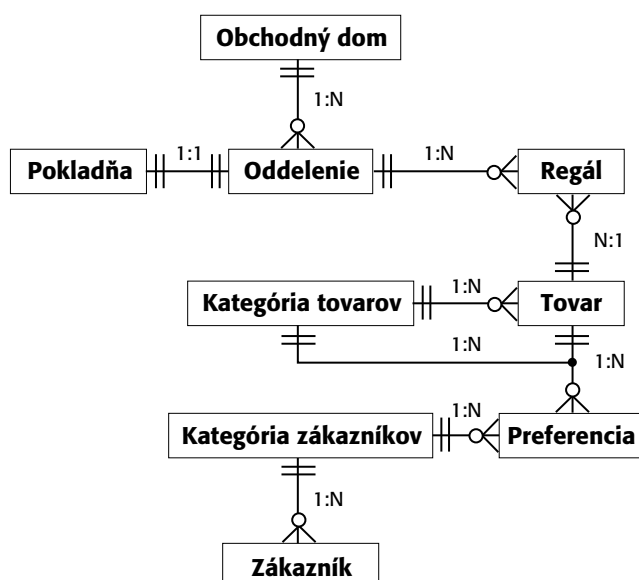
|         |  |
|---------|--|
| INT CKU | ID kategórie   |
| STRING  | Názov kategórie  |
| RNP     | Intervaly príchodu (v minútach)  |
| RNP     | Trpezlivosť - max. dĺžka radu, do ktorého sa je ochotný postaviť (rádovo 10) |
| RNP     | Videl reklamu (0...1)  |
| RNP     | Peniaze - má   |
| RNP     | Peniaze - chce minúť   |
| RNP     | Peniaze - môže minúť navyše  |
| LIST    | Zoznam preferencií   |

## Preferencia

|     |   |
|-----|---|
| INT | ID tovaru alebo kategórie tovaru                        |
| RNP | Časové intervaly medzi kúpou tovaru skupinou zákazníkov |

## Zákazník

|         |   |
|---------|---|
| INT CV  | ID kategórie  |
| INT CV  | Trpezlivosť   |
| CASH CV | Suma peňazí, ktoré chce minúť                               |
| CASH CV | Suma peňazí, ktoré môže minúť navyše                        |
| BOOL CV | Pozeráva reklamu  |
| CASH CV | Aktuálny obsah peňaženky                                    |
| CASH CV | Suma minulých peňazí  |
| TIME CV | Čas strávený čakaním v radoch                               |
| INT CV  | Počet neúspešných pokusov na vstup do oddelenia (dlhý rad)  |
| INT CV  | Počet neúspešných pokusov na vstup k regálu (dlhý rad)      |
| INT CV  | Počet neúspešných pokusov na vstup k regálu (prázdny regál) |
| LIST CV | Zoznam referencií na oddelenia, ktoré sa chystá navštíviť   |
| LIST CV | Zoznam referencií na tovary, ktoré kúpil                    |



Obr. 3. Entitno-relačný diagram pre obchodný dom

## **5. OHRANIČENIA, ZMENY ŠPECIFIKÁCIE, PRIORITY RIEŠENIA**

Základným ohraničením je čas na dokončenie projektu, z ktorého vyplývajú ďalšie, konkrétne ohraničenia, týkajúce sa (ne)implementovaných častí. S ohraničeniami súvisia zmeny špecifikácie, ktoré sú dôsledkom jednak už spomínaného časového obmedzenia a tiež dôkladnejšieho porozumenia problémom alebo nepresností v špecifikácii.

### **5.1 Zmeny atribútov entít a vyhodnocovaných údajov**

#### **5.1.1. Oddelenie**

##### **Nové atribúty**

- Obsadenie oddelenia
- Počet, ktorí nevošli do oddelenia lebo bolo preplnené
- Obsadenie oddelenia v čase
- Návštevy skupinami zákazníkov v dennej dobe

##### **Zrušené atribúty**

- Minimálna dĺžka radu
- Obsadenie oddelenia v závislosti od dennej doby
- Tržby v oddelení v čase

#### **5.1.2 Zákazník**

Atribúty entity zákazník (kapitola 5.1 špecifikácie) boli rozdelené medzi entity: kategória zákazníka, preferencia a zákazník. Taktiež bol presne určený význam termínov „potreba“ a „solventnosť“ zo špecifikácie. Potreba zodpovedá množine preferencií. Solventnosť je určená tromi premennými s náhodným rozdelením: suma peňazí v peňaženke, suma peňazí, ktorú sa zákazník rozhodol minúť a suma peňazí, ktoré môže prípadne minúť navyše.

Niektoré atribúty boli premenované (časové intervaly medzi kúpou tovaru; hotovosť, ktorou zákazník disponuje).

##### **Nové atribúty**

Eviduje sa počet radov, do ktorých sa zákazník nepostavil, lebo boli príliš dlhé. Rad môže byť rad na košíky alebo rad pri regáli; tieto rady sa rozlišujú.

##### **Zrušené atribúty**

- Suma minútých peňazí za reklamovaný tovar
- Počet tovarov, ktoré zákazník nemohol kúpiť

#### **5.1.3 Pokladňa**

##### **Nové atribúty**

- Dĺžka prestávky pokladne

- Priemerná tržba na zákazníka
- Počet prestávok pokladne
- Priemerný čas neaktivity pre obslužné miesto

#### **Zrušené atribúty**

- Minimálna dĺžka radu

#### **5.1.4 Regál**

Pri otvorení obchodného domu je každý regál plný.

#### **Nové atribúty**

- Počet dopĺňaní regálu
- Počet neuspokojených zákazníkov
- - kvôli nedostatku tovaru v regáli
- - kvôli veľkému počtu zákazníkov pred regálom

#### **Zrušené atribúty**

- Minimálna dĺžka radu

#### **5.1.5 Tovar**

- Pôvodne sa uvažovalo s tromi úrovňami hierarchie tovaru. Implementované sú len dve, návrh počíta s neobmedzenou hierarchiou.
- Atribút „reklamovanosť“, t.j. či existuje na tovar reklama bol presunutý do entity kategória zákazníkov ako atribút „Videl reklamu“

### **5.2 Zmeny algoritmu pohybu zákazníkov**

- Nielen pri vyberaní tovaru z regálu ale aj všeobecne pri rozhodovaní a nákupe tovaru sa uvažuje len jeden kus tovaru.
- Zákazník sa pri regáli (po tom ako sa dostane na rad) zdrží nulový čas.
- Zoznam oddelení, ktoré treba navštíviť sa vytvára na základe zoznamu tovaru, ktorý sa zákazník rozhodne kúpiť. Ak sa ten istý tovar nachádza vo viacerých oddeleniach, zákazník navštívi všetky takéto oddelenia. Tento problém sa dá odstrániť použitím oddelení obsahujúcich navzájom disjunktné množiny tovarov.
- Zákazník sa snaží kupovať všetky tovary zo skupiny, ak má priradenú preferenciu na kategóriu tovaru, nie na konkrétny tovar.

### **5.3 Priorita riešení**

#### **Vysoká priorita**

- Simulačný procesor
- Model obchodného domu (entity)
- Algoritmus rozhodovania a pohybu zákazníka po obchodnom dome
- Čítanie a zápis definície obchodného domu z/do súboru

- Vyhodnocovanie bodov zastavenia
- Vyhľadávanie objektov
- Štatistiky
- Užívateľské rozhranie sprístupňujúce horeuvedené vlastnosti

### **Stredná priorita**

- Ukladanie dynamického stavu objektov
- Modifikovanie vlastností obchodného domu počas spustenej simulácie
- Podpora externých modulov a ukázkový externý modul
- Zobrazovanie grafov - zmena mierky

### **Nízka priorita**

Nasledujúce vlastnosti systému sú uvedené v špecifikácii, ich priorita riešenia je nízka a pravdepodobne nebudú implementované:

- História udalostí, zoznam naplánovaných udalostí a presúvanie udalostí v čase (kapitola 4.2 špecifikácie)
- Pomoc (help)
- Akcie kopírovať/vložiť pre entity obchodného domu a body zastavenia
- Aktívne a neaktívne body zastavenia: všetky vytvorené body zastavenia budú aktívne.

## 6. URČENIE SPÔSOBU TESTOVANIA A ÚDAJOV NA TESTOVANIE

### 6.1 Testovanie transakcií v obchodnom dome

Transakcie prebiehajúce v obchodnom dome predstavujú základ celej simulácie. Preto bolo nutné dostatočné testovanie. Keďže v čase jej vytvárania ešte nebol k dispozícii editor obchodného domu, bolo nutné vytvoriť obchodný dom manuálne priamo v programe.

Zákazníci (transakcie) postupne prechádzajú všetkými časťami obchodného domu, čo bolo použité na kompletné otestovanie. Postup prechodu transakcie systémom je pomerne zložitý a pozostáva z viacerých významných udalostí. Najprv sa vygeneruje sa zákazník podľa údajov zo skupiny zákazníkov, ktoré sú dané užívateľom. Týmto vznikne transakcia, ktorá sa bude pohybovať po systéme. Vytvorí sa zoznam tovarov, ktoré zákazník chce kúpiť a zoznam oddelení, ktoré chce zákazník navštíviť (vzhľadom na vygenerované tovary). Zákazník vojde do CSMEntry, čo je prázdne oddelenie, ktorým sa vstupuje do obchodného domu. Naplánuje sa vstup zákazníka do prvého oddelenia. Odteraz zákazník vykonáva ďalej popísané činnosti, až kým neodíde z obchodného domu. Pri vstupe do každého oddelenia sa naplánuje vstup do prvého regálu. Zákazník navštívi regál a pri odchode naplánuje ďalší regál. Keď už nemá v danom oddelení čo kúpiť, naplánuje si príchod do pokladne. Po zaplatení opustí oddelenie a naplánuje vstup do nasledujúceho. Keď sa vyčerpá zoznam oddelení, zákazník vstúpi do CSMEExit, čo je prázdne oddelenie reprezentujúce východ. Zákazník opustí obchodný dom a transakcia sa zruší.

Na komplexné otestovanie bolo nutné vytvoriť viac skupín zákazníkov a ich preferencie. Takisto sme vytvorili viac skupín tovarov a v každej viac z nich niekoľko druhov tovarov. V obchodnom dome sme vytvorili oddelenia prislúchajúce skupinám tovarov a v oddeleniach potrebné regále.

### 6.2 Testovanie bodov zastavenia

Body zastavenia (breakpoints) boli testované samostatne pre všetky použiteľné entity, teda pre oddelenie, tovar a samotný obchod. Trieda CBreakPoint je odvodená od triedy CExpressionConjunction, ktorá predstavuje zložený výraz. Vyhodnotením výrazu je jedna z hodnôt pravda - nepravda. V prípade pravdivého vyhodnotenia je podmienka splnená a zastaví vykonávanie simulácie. Na splnenie podmienky musia byť pravdivo vyhodnotené všetky podvýrazy zloženého výrazu. Takisto ako v prípade testovania transakcií boli body zastavenia testované vkladaním do kódu programu, pretože nebolo k dispozícii užívateľské prostredie. Na pokrytie všetkých testovacích ciest sa použili údaje, ktoré vždy spĺňali podmienku (zastavenie sa vykonalo okamžite po spustení), údaje, ktoré podmienku nespĺnili nikdy (zastavenie sa nevykoná) a údaje, ktoré podmienku splnili počas behu simulácie, ale nie okamžite po spustení simulácie. Údaje obsahovali viacnásobné podmienky, aby sa odskúšala funkčnosť spájania jednotlivých podmienok.

### 6.3 Testovanie archivácie

Archivovanie spočíva v zapisovaní a čítaní objektov a ich atribútov do súboru a zo súboru. Na archivovaní sa podieľa objekt triedy CMyArchive, ktorý predstavuje súbor, do ktorého sa atribúty zapisujú, resp. z ktorého sa atribúty čítajú (CMyArchive nie je v žiadnom vzťahu s triedou MFC CArchive, kvôli ktorej musela byť aj premenovaná). Archivujú sa elementárne atribúty (bool, long, double...) a objekty tried odvodených od abstraktnej triedy CArchivable. Archivovateľné objekty môžu obsahovať zoznamy iných objektov a re-



ferencie na iné objekty, tieto sú taktiež archivovateľné. Archivácia elementárnych atribútov je implementovaná priamo v triede CMyArchive metódami readTag a writeTag. Archiváciu objektov zabezpečujú metódy read a write implementované v triedach odvodených od CArchivable.

Testovanie implementácie tried CMyArchive a CArchivable bude prebiehať nezávisle od testovania iných modulov a tried metódou white box.

## 6.4 Testovanie užívateľského rozhrania a integrácia

Grafické rozhranie, z pohľadu zdrojového kódu pozostáva z veľkej časti z knižnice MFC, funkcií obsluhujúcich užívateľom vyvolané udalosti (výber z ponuky, stlačenie tlačidla a funkcií zabezpečujúcich výmenu dát s dialógovými oknami.

Z hľadiska testovania možno preto testovanie grafického rozhrania rozdeliť na dve časti:

- testovanie funkcií zabezpečujúcich výmenu dát s dialógmi
- testovanie obslužných funkcií jednotlivých udalostí; tieto funkcie sú väčšinou pomerne jednoduché a spočívajú v zavolaní jednej alebo viacerých metód simulačného procesora.

Jednotlivé odladené moduly simulátora, a nakoniec aj celý simulátor, sa budú postupne spájať s grafickým rozhraním, ktoré sa potom otestuje ako celok. Možno teda povedať, že testovanie grafického rozhrania bude zároveň aj integračným testom.

## **7. VÝBER IMPLEMENTAČNÉHO JAZYKA A PROSTREDIA**

Ako implementačný jazyk bol zvolený jazyk C++, ako implementačné prostredie bolo zvolené Microsoft Visual C++ 5.0. Toto rozhodnutie bolo vykonané (aj keď nie úplne definitívne) už pri vypracúvaní ponuky a bolo aj jej súčasťou. Dôvody pre toto rozhodnutie sú uvedené v kapitole 1.2 špecifikácie. Možno ich zhrnúť do nasledovných bodov:

- členovia tímu ovládajú jazyk C++ a majú s ním dostatočné skúsenosti
- výber platformy (Windows 95/NT) bol ovplyvnený doterajšími skúsenosťami členov tímu a ich možnosťami.

Na základe predchádzajúcich dvoch bodov, doterajších skúseností a dostupnosti produktov bolo zvolené implementačné prostredie (Visual C++ 5.0).

## 8. ŠTANDARDY KÓDOVANIA

Spôsob kódovania a jeho možnosti vychádzajú zo zvoleného programovacieho jazyka (C++), sú však ovplyvnené aj zvoleným implementačným prostredím a operačným systémom. Nasledovné konvencie boli prijaté tímom na prvom stretnutí v letnom semestri, ich používanie sa pre všetkých členov tímu doporučuje.

Výsledkom implementácie má byť zdrojový text jazyku C++. Zdrojový text pozostáva z viacerých textových súborov. Spôsob rozdelenia zdrojového textu do súborov je daný vlastnosťami implementačného jazyka (jedná sa o objektový jazyk).

### 8.1 Zdrojové súbory

Vo všeobecnosti možno povedať, že implementácia každého objektu je uložená v samostatnom súbore, resp. v dvoch súboroch - deklarácia triedy v súbore pomenovanom názvom triedy s príponou .h a definíciou uloženou v súbore s príponou .cpp, čo je štandardné označenie v C++, s nasledovnými výnimkami:

- vzhľadom na fakt, že niektorí členovia tímu budú používať pri implementácii aj operačný systém MS-DOS, názov súboru obsahuje len maximálne prvých 8 znakov názvu triedy.
- v názve súboru sú dodržané veľké a malé písmená tak, ako sú uvedené v názve triedy; pretože pri prenose súborov zo systému MS-DOS na Windows NT môže dôjsť k strate tejto informácie, bude vhodné to opraviť, aj keď Windows NT nerozlišuje veľké a malé písmená. V príkazoch pre preprocesor (#include) sa uvádzajú vždy „správne“ názvy.
- abstraktné triedy/rozhrania môžu mať len súbor s deklaráciou
- deklarácia a definícia pomocných tried, ktoré nemajú byť viditeľné pre ostatné objekty môžu byť uvedené v rámci cpp súboru s triedou, ktorá ich používa
- v jednom hlavičkovom súbore (.h) sa môže nachádzať a aj viac ako jedna deklarácia triedy v prípade, že sa jedná napr. o pomocné triedy odvodené od tej istej triedy malého rozsahu

Súbory používajú na oddeľovanie riadkov dvojicu znakov CR LF, text sa formátuje pomocou tabulátorov (ASCII 9), veľkosť tabulátora je štyri znaky.

Úprava jazyka (napr. umiestňovanie zátvoriek) nie je daná. Je možné používať akékoľvek druhy komentárov; nesmú však obsahovať dvojicu znakov \*\*, ktorá označuje nedokončenú alebo zatiaľ neimplementovanú časť. Každý súbor so zdrojovým textom má záhlavie, ktoré obsahuje názov súboru, označenie tímu, školský rok a zoznam zmien.

### 8.2 Objekty a atribúty

Objekty a ich atribúty sa pomenúvajú v anglickom jazyku. Pokiaľ je názov zložený z viacerých slov, slová sú oddelené veľkým písmenom nasledujúceho slova. Ďalej platí:

- názvy tried začínajú na „C“
- názvy atribútov začínajú na „m“, nasledujúce písmeno je veľké
- názvy konštánt začínajú na „c“, nasledujúce písmeno je veľké
- globálne premenné začínajú na „g“; používanie globálnych premenných sa nedoporučuje. Pokiaľ je to nutné, je treba použiť static premennú v rámci triedy, ktorá ju používa; táto premenná začína na „s“
- názvy lokálnych premenných začínajú malým písmenom a nesmú byť zameniteľné s atribútmi objektu alebo konštantami. Iné požiadavky na názvy lokálnych premenných

nie sú dané, je možné používať konvencie používané vo Windows API (názov premennej obsahuje aj jej typ), ale nie je to však nutné ani preferované.

- názvy metód začínajú malým písmenom
- pokiaľ je to možné, nepoužívať #define. Namiesto #define je možné použiť konštantu alebo enumeráciu; ak je potrebné vytvoriť makro, potom vytvoriť inline funkciu. Názvy enumerácií začínajú na „e“
- atribúty objektov by nemali byť public, pokiaľ nie sú const
- všetky funkcie (okrem main) by mali patriť do nejakej triedy
- konštanty a enumerácie by mali byť deklarované v rámci triedy, ku ktorej patria
- metódy, ktoré nemodifikujú atribúty objektu by mali byť označené ako const.

Z objektívnych príčin platia nasledovné výnimky:

- súbory generované MS Visual C++ môžu používať iné konvencie a obsahovať generované komentáre, ktoré majú špecifický význam pre prostredie
- súbory, ktoré sú súčasťou knižnice Tiny používajú tieto konvencie len v obmedzenej miere; názvy tried, metód a atribútov spĺňajú zvolené konvencie.

## 9. OPIS IMPLEMENTÁCIE JEDNOTLIVÝCH MODULOV

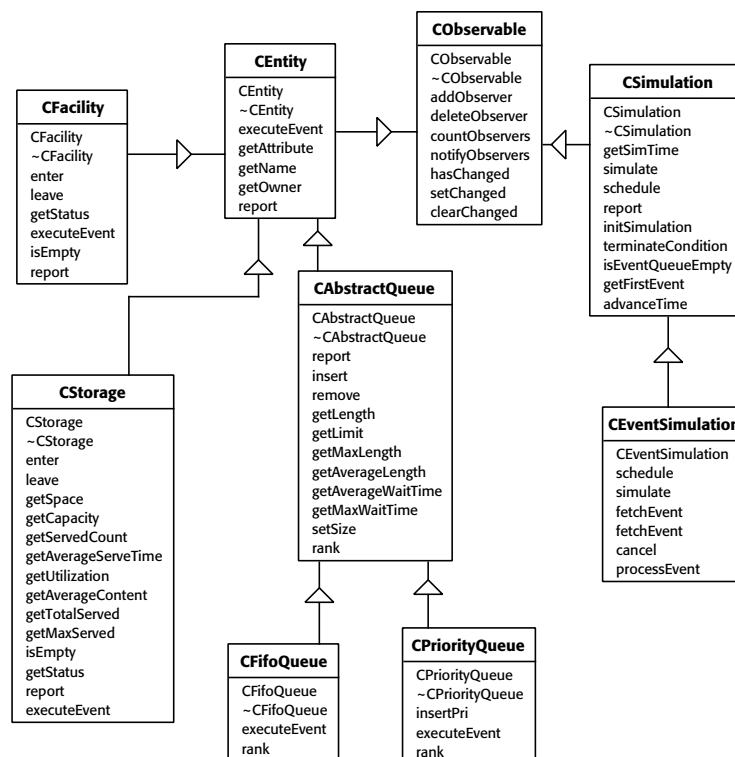
Ako už bolo uvedené v kapitole 3, systém je zložený z dvoch subsystémov: zo simulátora obsahujúceho simulačný procesor, simulovaný model a štatistiky; a vizualizačného front-endu slúžiaceho na zobrazovanie stavu simulácie a vytvárajúceho užívateľské rozhranie.

### 9.1 Modifikácie knižnice Tiny

Subsystém simulácie vychádza z knižnice Tiny. Knižnicu Tiny sme rozšírili v dvoch rovinách: odvodením nových tried, špecifických pre obchodný dom (napr. regál, pokladňa), odvodením nových tried rozširujúcich funkčnosť existujúcich tried (doplnenie bodov zastavenia do simulačného procesora) a tiež priamou modifikáciou knižnice. Aj keď sme pôvodne neplánovali použiť poslednú použitú metódu, ukázalo sa jej použitie vhodnejšie. Nevýhodou tejto metódy je, že nie je možné jednoducho vymeniť knižnicu Tiny za jej novšiu implementáciu.

Charakter zmien vykonaných v knižnici je rôzny: doplnenie funkcií prístupných atribúty objektov, premenovanie atribútov (podľa konvencií tímu), zmena deklarácií funkcií na const, doplnenie nových atribútov objektov a odvodenie existujúcich tried od pridaných tried (CObservable). Dosaženie cieľa bez modifikácie knižnice Tiny pomocou prekrytia by znamenalo duplikáciu existujúcich funkcií s doplnením minimálnych zmien; mnohokrát by však ani nebolo možné, pretože nie všetky funkcie sú deklarované ako virtuálne.

V knižnici Tiny nie je rozlíšený simulačný procesor a simulovaný model. Obidve činnosti implementuje trieda CSimulation. Hoci by sme pri návrhu vlastnej knižnice



Obr. 4. Hierarchia tried modifikovanej knižnice Tiny

postupovali inak, rozhodli sme sa to v knižnici Tiny nezmeniť. Nová hierarchia tried v knižnici Tiny je zobrazená na obr. 4.

## 9.2 Vzor Observer

Asi najvýznamnejšou zmenou v knižnici bolo jej rozšírenie o koncept Observable/Observer. Jedná sa o často používaný návrhový vzor, slúžiaci na propagáciu zmeny stavu objektu Observable priradeným objektom (Observer). Objekt Observable informuje po zmene svojho stavu priradené objekty Observable. Vzťah Observable:Observer je 1:M.

V novej verzii knižnice sú všetky entity a aj samotný model odvodené od Observable. Použitie vzoru nám umožňuje oddeliť simulátor od užívateľského rozhrania, čo je výhodné pri oddelenej implementácii a v neposlednom rade aj pri testovaní. Informácie o stave entít sa zobrazujú prostredníctvom objektov Observer (napríklad s okno „Vlastnosti“).

## 9.3 Body zastavenia

Ďalším rozšírením simulátora bolo doplnenie bodov zastavenia. Body zastavenia umožňujú automatické zastavenie simulácie ak je splnená zadaná podmienka (pozri 9.4). Implementácia využíva už spomínaný model Observable/Observer. Observerom každého objektu, na ktorý je možné nastaviť bod zastavenia (obchodný dom, oddelenie, pokladňa, regál, tovar a kategória tovarov) je simulačný procesor.

Objekt Observer dostáva správu pri akejkoľvek zmene stavu pozorovaného objektu, čím je zabezpečené, že mu žiadne eventuálne zastavenie „neunikne“. Podmienky zastavenia sa vyhodnocujú práve v tomto okamihu. Ak je splnená aspoň jedna podmienka, simulácia sa zastaví.

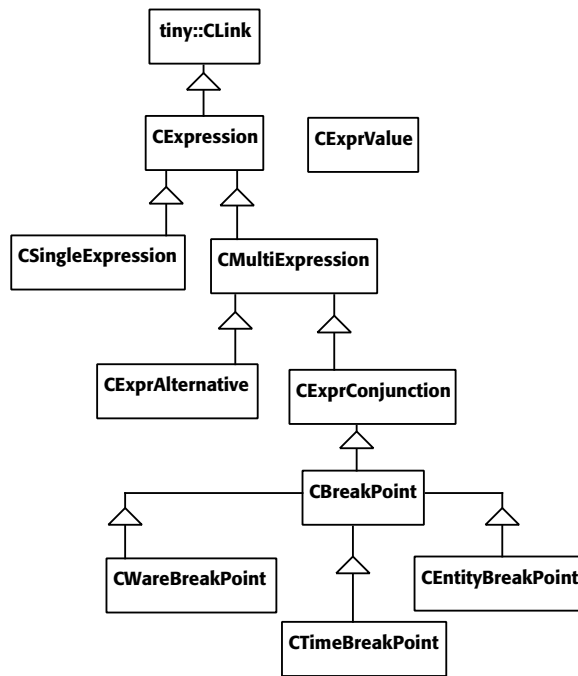
So zastavením simulácie je spojená interakcia s užívateľom, minimálne čakanie na stlačenie klávesu. Samotný simulátor by mal zostať nezávislý od použitého grafického rozhrania. Implementované riešenie využíva návrhový vzor Template Method, ktorého podstatou je vytvorenie prázdnych virtuálnych metód ponechaných na prekrytie. Takéto metódy sú v našom prípade go, step, stop, kill a allowContinue; sú prekryté v triede simulácie pre Windows (CSuperMarketWin). Prvé štyri funkcie zodpovedajú priamo tlačidlám na nástrojovej lište na riadenie simulácie, význam funkcie allowContinue súvisí s implementáciou simulácie vo Windows.

Trieda CSuperMarketWin je implementáciou simulátora pracujúcom v samostatnom threade. Tento prístup je v prostredí Windows takmer nevyhnutný - spustená simulácia by blokovala celý proces teda aj funkcie užívateľského rozhrania. Funkcie go, step, stop a kill v skutočnosti simuláciu nespúšťajú ani nezastavujú; modifikujú iba synchronizačné premenné, ktorých stav sa kontroluje vo funkcii allowContinue. Simulačný procesor volá túto funkciu vždy, keď je teoreticky možné zastaviť simuláciu (po vykonaní elementárneho kroku simulovaného modelu). Funkcia allowContinue rieši zastavovanie a opätovné spustenie threadu simulácie kontrolou atribútov nastavovaných v go, step, stop a kill.

## 9.4 Vyhodnocovanie výrazov

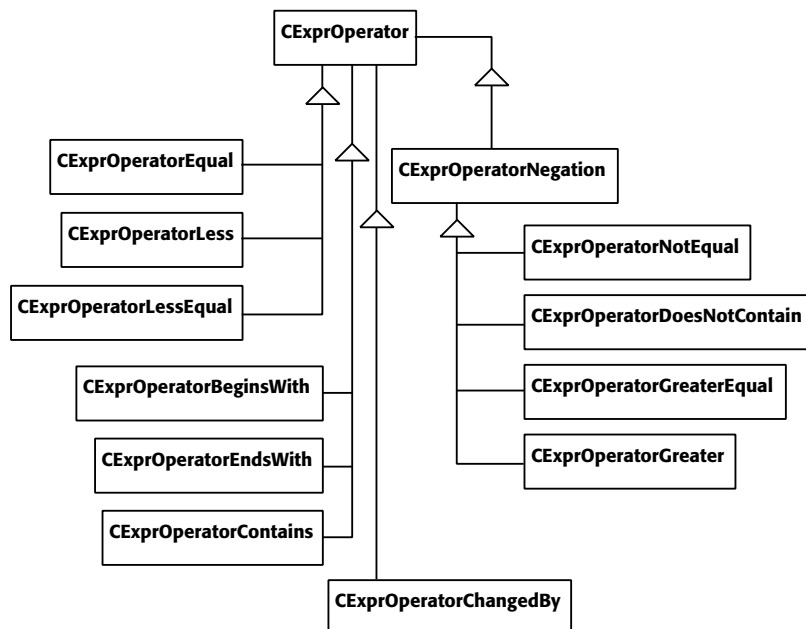
Pri implementácii bodov zastavenia bolo potrebné vyriešiť problém špecifikovania a vyhodnocovania zadaných výrazov. Navrhnuté riešenie muselo spĺňať viacero požiadaviek:

- podpora pre jednoduché a zložené výrazy (alternatíva a konjunkcia); podmienka pre bod zastavenia je konjunkcia, všetky body zastavenia tvoria alternatívu (pozri obr. 5)
- podpora rôznych typov (bool, int, double, string)



Obr. 5. Hierarchia tried výrazov

- operátory: >, <, ≥, ≤, =, !=, začína na, končí na, obsahuje, neobsahuje; nie všetky operátory sú použiteľné na všetky typy argumentov (pozri obr. 6)
- porovnávanie argumentov nerovnakých typov (napr. porovnávanie int a double)
- porovnávanie atribútov objektov, ktoré sa dynamicky menia (vyhodnotenie výrazu, napr. počet peňazí v pokladni je väčší ako 100, musí reflektovať aktuálny stav pokladne, nie stav vo chvíli vytvorenia výrazu)



Obr. 6. Hierarchia tried operátorov

- umožniť relatívnu komparáciu (napr. nie  $a > 3$  ale:  $a$  sa zmení o 3; takáto podmienka je mnohokrát výhodnejšia; napr. ak čakáme na udalosť, kedy nejaký zákazník vojde do oddelenia - počet zákazníkov v oddelení sa zmení o 1)

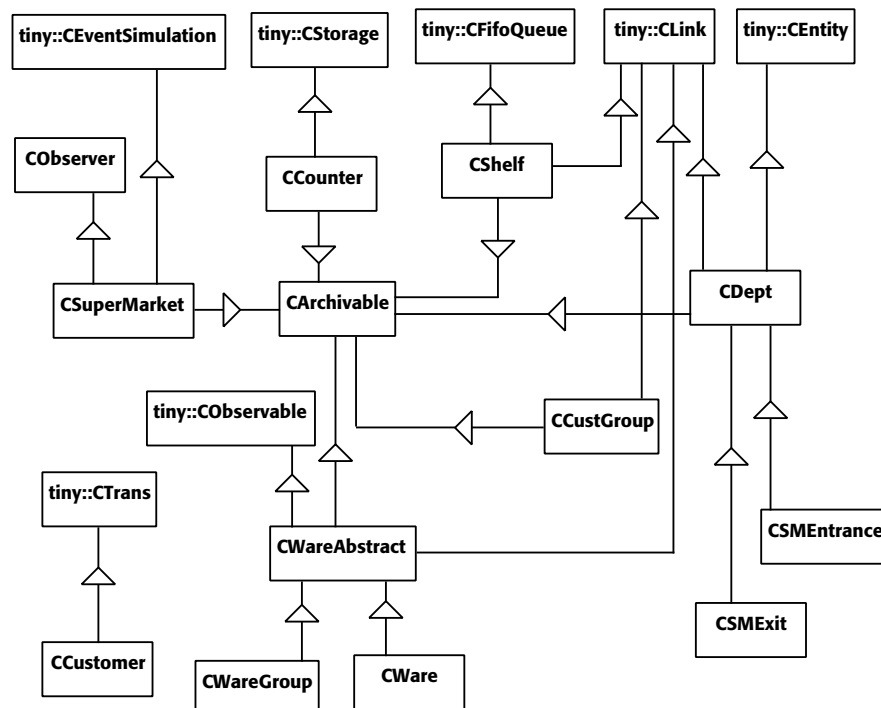
Hierarchia výrazov je znázornená na obr. 5. Základná abstraktná trieda CExpression je odvodená od CLink aby bolo možné vkladať ľubovoľný výraz do výrazu (CMultiExpression). CExpression obsahuje pure virtual metódu bool evaluate(), ktorú od nej odvodené triedy prekrývajú. CBreakPoint rozširuje triedu CConjunction o objekt, ktorému je bod zastavenia priradený.

## 9.5 Entity obchodného domu

Obchodný dom a jeho entity vznikli odvodením od tried Tiny (obr. 7). Okrem toho, že prekrývajú funkciu executeEvent entít Tiny pre udalosti špecifické pre obchodný dom implementujú rozhranie CArchivable slúžiace na čítanie a zapisovanie objektov a ich atribútov do súboru. CArchivable produkuje textový súbor. Triedy CDept (oddelenie) a CSelf (regál) sú odvodené od CLink aby mohli byť prvkom zoznamu oddelení, resp. regálov. Oddelenie obsahuje práve jednu pokladňu (CCounter) s viacerými obslužnými miestami.

Triedy CWare a CWareGroup predstavujú druh tovaru a kategóriu tovaru. Základná trieda umožňuje hierarchické členenie (vkladanie tovarov do skupín) a štatistické vyhodnocovanie počtu predaných kusov.

Triedy CSMEntrance a CSMExit predstavujú vchod a východ z obchodného domu, v skutočnosti sú to prázdne oddelenia (pozri odstavec 9.6).

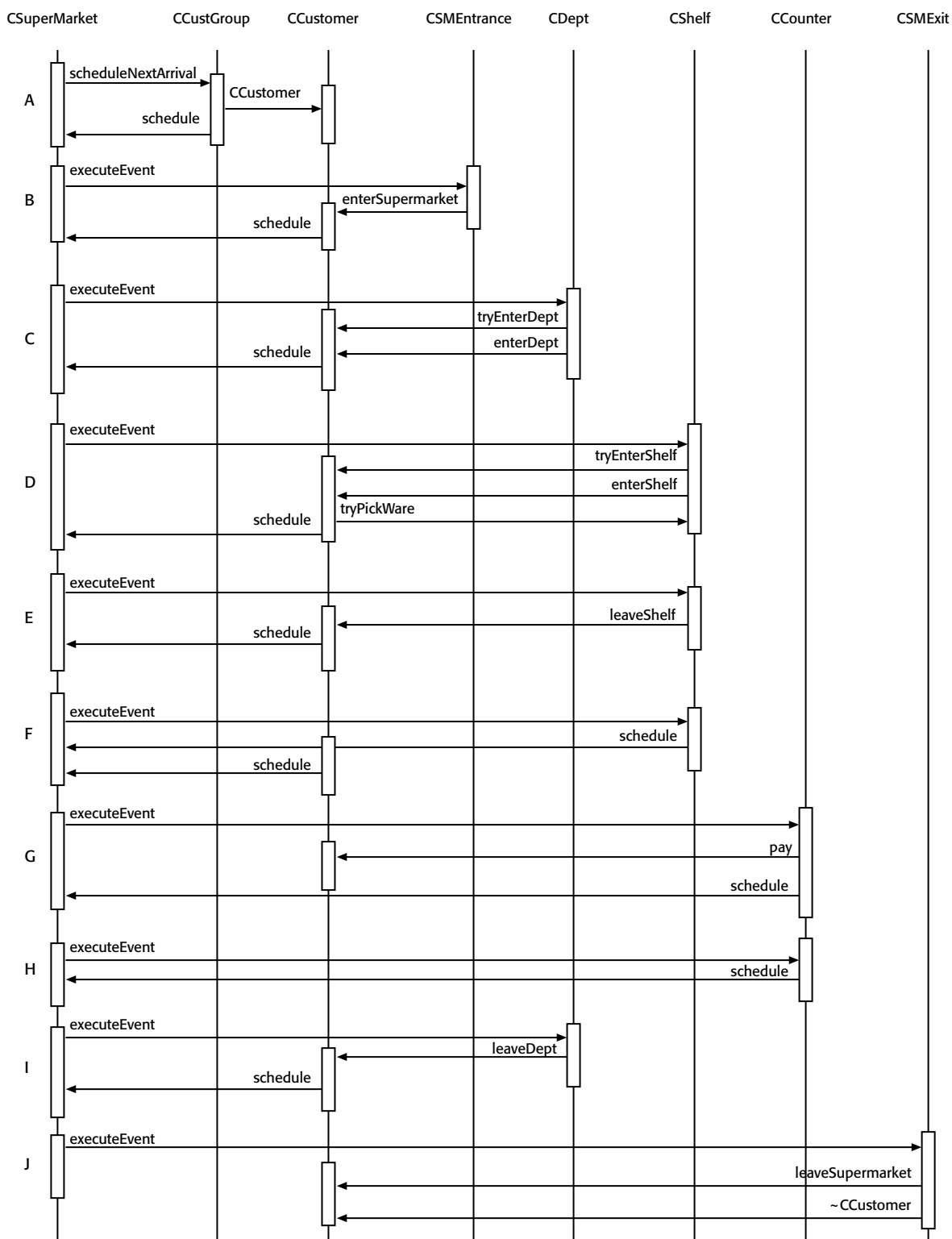


Obr. 7. Hierarchia tried obchodného domu



## 9.6 Algoritmus pohybu zákazníka

Simulačný procesor (Tiny::CEventSimulation) je implementáciou udalosťami riadenej simulácie. Entity spracúvajú prijaté udalosti a plánujú nové. Každá entita môže vytvoriť ľubovoľnú udalosť určenú pre ľubovoľnú entitu (predpokladá sa, že entity nebudú vytvárať



Obr. 8. Diagram sledu udalostí pri pohybe zákazníka v obchodnom dome

navzájom si odporujúce si udalosti). Dynamickou časťou systému je transakcia, v našom prípade zákazník (CCustomer). Všetky zmeny v obchodnom dome súvisia s pohybom a rozhodovaním zákazníka - príchod, čakanie v rade, výber z regálu, dopĺňanie regálov a zaplatenie.

Keďže sa jedná o udalosťami riadenú simuláciu neexistuje funkcia, ktorá by obsahovala úplne celý algoritmus rozhodovania zákazníka. Rozhodovanie je implementované vo viacerých funkciách obsahujúcich reakciu na špecifickú udalosť. Entity priamo volajú tieto metódy podľa konkrétnej situácie (opäť sa predpokladá konzistentné poradie volaní).

Diagram sledu udalostí pri pohybe zákazníka v obchodnom dome je znázornený na obr. 8. Návštevu jedného zákazníka možno rozdeliť do nasledovných bodov:

- A. CCustGroup (kategória zákazníkov) vytvára nových zákazníkov a naplánuje ich príchod (do CSMEntrance).
- B. CSMEntrance (vchod) informuje zákazníka, že vošiel do obchodného domu. Zákazník si naplánuje príchod do prvého oddelenia, ktoré sa rozhodol navštíviť.
- C. Zákazník skúša vojsť do oddelenia. Pokiaľ na tryEnterDept odpovie true (rad do oddelenia sa mu nezdá príliš dlhý), vojde do oddelenia (enterDept). Zákazník si naplánuje príchod k prvému regálu, ktorý chce navštíviť.
- D. Zákazník pri regáli. Pokiaľ na tryEnterShelf odpovie true (rad pri regáli sa mu nezdá príliš dlhý), vojde ku regálu, resp. sa postaví do radu. Keď sa dostane na rad (enterShelf) skúša si z regálu vybrať tovar (tryPickWare); regál môže byť aj prázdny. Potom si naplánuje odchod z regálu.
- E. Odchod z regálu: zákazník si naplánuje návštevu ďalšieho regálu alebo ide ku pokladni.
- F. Zaplatenie: pokladňa vyzve zákazníka aby zaplatil za tovary, ktoré nakúpil.
- G. Pokladňa naplánuje odchod zákazníka z pokladne (odchod z pokladne závisí od jej doby obsluhy)
- H. Odchod z pokladne: pokladňa naplánuje odchod zákazníka z oddelenia (okamžite po obslúžení)
- I. Odchod z oddelenia: zákazník si naplánuje návštevu nasledujúceho oddelenia. Posledným oddelením na zozname oddelení, ktoré treba navštíviť je vždy CSMExit - východ.
- J. Zákazník odchádza z obchodného domu a je zrušený.

## 9.7 Štatistiky

Knižnica Tiny obsahuje takmer všetky potrebné funkcie, ktoré sme potrebovali (počítanie priemeru, odchýliek a histogramy). Doplnili sme triedu CGroupCounter podobnú histogramu. Trieda počíta objekty konečného množstva skupín - to isté ako histogram s tým rozdielom, že oborom hodnôt nie je interval ale množina typov objektov.

Pomerne jednoduchou ale významnou zmenou bolo vytvorenie univerzálneho náhodného generátora s nastaviteľným typom rozdelenia náhodnej premennej implementujúceho rozhranie CArchivable.

## **10. TESTOVANIE**

### **10.1 Testovanie transakcií**

Testovanie prebiehalo po určitú dobu, čím sa súčasne otestovala zastavovacia podmienka (otvorený obchodný dom po určitý čas). Počas tejto doby sa viackrát vygenerovali zákazníci z oboch skupín s náhodnými požiadavkami na tovary, resp. skupiny tovarov. Všetci zákazníci korektno vošli do obchodného domu, prešli po oddeleniach nakoniec obchodný dom opustili. Záznam ich pohybu sa vypisoval na konzolu.

### **10.2 Testovanie bodov zastavenia**

Pri testovaní sa objavila nepríjemná chyba, ktorá bola dôsledkom nezhodnosti typov ukazovateľov. Jej príčinou bolo, že pretypovaním objektu sa zmení hodnota ukazovateľa a pri vyhodnocovaní výrazu, ktorého súčasťou je aj zisťovanie, či je sledovaný objekt vlastníkom daného výrazu, sa tieto rozdielne hodnoty prejavili. Riešením je použitie operátora na dynamické pretypovanie (`dynamic_cast`).

### **10.3 Testovanie archivovania údajov**

Pri testovaní sme postupovali tak, že sme najprv implementovali triedu `CMyArchive` a testovali sme metódy na archiváciu elementárnych atribútov `writeTag` a `readTag` pre všetky typy elementárnych atribútov. Po úspešnom testovaní metód sme otestovali archiváciu objektov na objekte pokusnej triedy odvodenej od `CArchivable`. Nakoniec sme implementovali a úspešne otestovali metódy `read` a `write` pre všetky triedy odvedené od triedy `CArchivable`. Použitie tried `CMyArchive` a `CArchivable` sú popísané v kapitole 9.5. Príklad testovacieho súboru, ktorý obsahuje všetky atribúty a objekty simulovaného modelu je uvedený v prílohe H.

### **10.4 Testovanie užívateľského rozhrania**

Užívateľské rozhranie sme testovali metódou bielej skrinky s využitím sledovacích prostriedkov prostredia Microsoft Developer Studio (body zastavenia). Testovaná bola jednak funkčnosť prostredia, jednak prepojenie prostredia a simulátora. Pri testovaní bola využitá aj konzola simulátora, ktorá slúžila ako záznam o činnosti systému.

## 11. ZÁVER

Tímová práca bola prínosom pre všetkých členov tímu. Naučili sme sa spolupracovať s ostatnými kolegami na jednom projekte. Súčinnosť bola dôležitá najmä pri rozdeľovaní jednotlivých úloh a ich konečnej agregácii do projektu ako celku. Pri rozdeľovaní povinností bolo potrebné rovnomerne rozdeliť úlohy jednotlivým členom tímu.

Z časových dôvodov sme boli nútení upustiť od niektorých požiadaviek, ktoré sú popísané v špecifikácii produktu. V kapitole 5 v časti Návrh a implementácia sa nachádza rozdelenie častí produktu podľa priority. Vypustili sme tie časti, ktoré mali nižšiu prioritu a neboli nutné pre beh simulátora. Medzi vynechané časti patrí pripájanie externých modulov a ukladanie dynamického stavu systému. Bez týchto dvoch častí je produkt plne funkčioschopný. Ďalej sme sa rozhodli neimplementovať kopírovanie a drag-and-drop objektov. Tieto časti majú nízku prioritu a týkajú sa skôr komfortu pri používaní systému a neobmedzujú jeho funkčnosť.

Pri oddelenej práci členov tímu na zdrojovom kóde sme používali program WinDiff pri porovnávaní a spájaní verzií, ale vhodnejšie by bolo použiť revízný kontrolný systém (revision control system); zlučovanie zdrojového kódu nám zaberalo množstvo času a bolo neefektívne.

Práca prebiehala najmä vo výpočtovom laboratóriu D132 v priestoroch školy. Jeho vybavenie nám poskytlo možnosť pracovať aj v prípade, že niektorí členovia tímu nemali k dispozícii potrebný softvér a hardvér doma. Prácu sťažovala dlhá doba kompilácie, spôsobená nie veľmi vhodnou inštaláciou produktu Microsoft Developer Studio, ktoré nebolo na jednotlivých strojoch, ale na sieťovom disku. Takýto prístup spôsoboval zahltenie siete a výrazné zníženie rýchlosti.

## **Príloha A: Ponuka**

Juraj Mikluš • Andrej Oriško • Miroslav Panák  
Boris Steiner • Martin Szolgay

# **ANIMÁCIA SIMULAČNÝCH EXPERIMENTOV S MODELOM DISKRÉTNEHO SYSTÉMU**

## **Motivácia**

Získať skúsenosti s tímovou prácou  
v oblasti simulačných systémov

## **Plán projektu**

Navrhnuť a implementovať systém, ktorý umožňuje:

- špecifikovať model diskretného systému pomocou základných objektov (entita, rad, obslužné miesto)
- vytvoriť užívateľské rozhranie umožňujúce zobrazovať a riadiť priebeh simulácie
- vyhodnotiť výsledky simulácie
- vyriešiť spôsob komunikácie modulov a umožniť nahradiť moduly systému (COM)
- využiť existujúce knižnice (TINY, C++, príp. MFC)

## **Predpokladané zdroje**

- implementačný jazyk: C++, prostredie: Visual C++, Metrowerks CodeWarrior, OS: Windows 95/NT
- alternatíva: Java
- čas: 5 mesiacov

## Príloha B: Príklad vstupného súboru

Kľúčové slová sú označené VELKÝMI písmenami. Znaky // znamenajú komentár.

|                              |   |
|------------------------------|---|
| KATEGORIA „Potraviny“        | // kategória potraviny                  |
| SKUPINA „Pekárenské výrobky“ | // skupina pekárenské výrobky           |
| TOVAR „Chlieb“               | // tovar chlieb                         |
| CENA 16                      | // cena tovaru chlieb                   |
| RELAMOVANOST FALSE           | // tovar nebol reklamovaný              |
| TOVAR „Pečivo“               | // tovar pečivo                         |
| CENA 2                       | // cena tovaru pečivo                   |
| REKLAMOVANOST FALSE          | // tovar nebol reklamovaný              |
| SKUPINA „Pochutiny“          | // skupina pochutiny                    |
| TOVAR „Mars“                 | // tovar mars                           |
| CENA 9, 90                   | // cena tovaru                          |
| REKLAMOVANOST TRUE           | // tovar bol reklamovaný                |
| SKUPINA „Mliečne výrobky“    | // skupina mliečne výrobky              |
| TOVAR „Mlieko“               | // tovar mlieko                         |
| CENA 13, 20                  | // cena tovaru                          |
| REKLAMOVANOST FALSE          | // tovar nebol reklamovaný              |
| <br>                         |   |
| KATEGORIA „Elektronika“      | // kategória tovaru elektronika         |
| SKUPINA „Televízory“         | // skupina televízory                   |
| TOVAR „Tesla Orava OTF 457“  | // tovar...                             |
| CENA 12 000                  | // cena tovar                           |
| REKLAMOVANOST TRUE           | // bol reklamovaný                      |
| <br>                         |   |
| ODDELENIE „Potraviny“        | // oddelenie obchodného domu            |
| KOSIKY 100                   | // počet košíkov v oddelení             |
| POKLADNA                     | // blok pokladní                        |
| POCET 10                     | // počet pokladní                       |
| VYPADOK 5%                   | // pravdepodobnosť výpadku pokladne     |
| REGAL                        | // regál                                |
| TOVAR „Chlieb“               | // obsahuje tovar chlieb                |
| MAXIMUM 100                  | // maximálne množstvo tovaru v regáli   |
| MINIMUM 10                   | // množstvo, pri ktorom sa tovar doplní |
| ONESKORENIE 600              | // oneskorenie doplnenia tovaru         |
| REGAL                        | // iný regál                            |
| TOVAR „Mars“                 | // obsahuje tovar pečivo                |
| MAXIMUM 1000                 | // max. množstvo tovaru                 |
| MINIMUM 10                   | // dopĺňacie množstvo tovaru            |
| ONESKORENIE 600              | // oneskorenie doplnenia                |

|                                       |   |
|---------------------------------------|---|
| ODDELENIE „Elektro“                   | // oddelenie elektroniky  |
| KOSIKY 10                             | // počet košíkov  |
| POKLADNA                              | // pokladne   |
| POCET 2                               | // ich počet  |
| VYPADOK 5%                            | // a pravdepodobnosť výpadku  |
| REGAL                                 | // regál  |
| TOVAR „Tesla Orava OTF 457“           | // obsahuje tento tovar   |
| MAXIMUM 10                            | // max. počet tovaru  |
| MINIMUM 8                             | // doplniace množstvo tovaru  |
| ONESKORENIE 3600                      | // oneskorenie doplnenia  |
| ZAKAZNIK „Študent“                    | // zákazník typu študent  |
| TRPEZLIVOST 10                        | // počet ľudí v rade, do ktorého sa<br>// je ochotný postaviť             |
| HOTOVOST 200                          | // suma, ktorú má pri sebe  |
| SOLVENTNOST 100                       | // suma, ktorú je ochotný minúť za<br>// reklamovaný tovar                |
| POTREBA „Pekárske výrobky“ 20%        | // pravdepodobnosť kúpenia jedného<br>// potrebného typu tovaru           |
| POTREBA „Mlieko“ 40%                  | // pravdepodobnosť kúpenia iného tovaru                                   |
| PREFERENCIA „Mars“ 80%                | // pravdepodobnosť kúpenia nepotrebného<br>// tovaru, ak zhlíadol reklamu |
| ZAKAZNIK „Domáci kutil“               | // iný zákazník   |
| TRPEZLIVOST 50                        | // trpezlivosť čakania v rade   |
| HOTOVOST 6000                         | // suma, ktorú má pri sebe  |
| SOLVENTNOST 2000                      | // suma, ktorú je ochotný minúť   |
| POTREBA „Elektronika“ 100%            | // pravdepodobnosť kúpenia nutného tovaru                                 |
| PREFERENCIA „Pekárske<br>výrobky“ 30% | // pravdepodobnosť kúpenia tovaru, ak bol<br>// reklamovaný               |
| PREFERENCIA „Mliečne<br>výrobky“ 20%  | // iný tovar, ktorý sa kúpi, len ak bol<br>// reklamovaný                 |

## **Príloha C: Záznamy zo stretnutí**

### **Zápisnica zo stretnutia tímu 19. 10. 1998**

#### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Boris Steiner  
Martin Szolgay

#### **Zapisovateľ:**

Juraj Mikluš

#### **Úlohy na tomto stretnutí:**

1. Organizačné dohody: Za vedúceho tímu bol zvolený Andrej Oriško. Dohodli sme sa na rozdelení 11 bodov za ponuku nasledovne: Andrej Oriško 3 body, ostatní po 2 body.
2. Formulácia základných požiadaviek na systém zadávateľom. Upresnenie textu témy, určenie základných služieb a ohraničení systému na základe konzultácie so zadávateľom.
3. Dohoda na základných parametroch obchodného domu z hľadiska užívateľa, špecifikácie obchodného domu užívateľom, správania sa zákazníkov.
4. Vizualizácia sledovaných veličín obchodného domu ako celku, ale aj jednotlivých elementárnych objektov (napr. pokladní).
5. Možnosti modifikovania modelu počas simulácie. Možnosti zmeny parametrov objektov v obchodnom dome.
6. Odkladanie stavu simulácie do súboru, prerušenie simulácie.
7. Príprava na formuláciu požiadaviek na systém, identifikáciu základných objektov a ich vzťahov.

#### **Úlohy na budúce stretnutie:**

1. Štruktúra modelu (vlastnosti zákazníka obchodného domu a jeho rozhodovacie funkcie).
2. Premyslieť funkčnosť programu.
3. Zvoliť implementačné prostriedky. Preskúmať možnosti poslednej verzie knižnice Tiny. V učebni D132 preskúmať dostupné vývojárske nástroje.



## **Zápisnica zo stretnutia tímu 26. 10. 1998**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Boris Steiner  
Martin Szolgay

### **Zapisovateľ:**

Boris Steiner

### **Úlohy z predchádzajúceho stretnutia:**

Rozdelenie bodov za prezentáciu, prezentácia zoznamu objektov a ich atribútov

Andrej Oriško prezentoval zoznam objektov a hierarchiu objektov

Diskusia o rozhodovaní v modeli (rozhodovanie zákazníka podľa preferencií)

#### 1. Statická časť:

reprezentácia obchodného domu (štruktúra hierarchie)

#### 2. Dynamická časť:

hierarchia tovarov a ich atribúty

reprezentácia tovarov a väzba na oddelenia

reprezentácia zákazníka, model reprezentácie atribútov

- problém „trpezlivosť“ zákazníka, hľadanie riešenia situácie, keď zákazník čaká na rad, ktorý je ochotný akceptovať (jeho dĺžku a zaradiť sa doň)
- výber tovaru v oddelení - zoradenie tovarov podľa reklamy a ceny, spôsob prezerania regálu zákazníkom - len raz zhora dole
- preferencia jednotlivých tovarov bude uvedená vo vstupnom súbore - rozdelenie a pravdepodobnosť nákupu
- usporiadanie jednotlivých oddelení v obchodnom dome - zretáženie obchodov, vstupná hala
- výstupy
- riadenie simulácie
- režimy práce simulátora - vytváranie, beh, modifikácia, vyhodnocovanie

### **Úlohy pre budúce stretnutie:**

Vypracovať predbežnú verziu špecifikácie - len v digitálnej forme

### **Rozdelenie úloh pre špecifikáciu:**

|                |                   |
|----------------|-------------------|
| Juraj Mikluš   | Úvod              |
| Andrej Oriško  | Plán projektu     |
| Boris Steiner  | Funkcie v systéme |
| Martin Szolgay | Údaje v systéme   |
| Miroslav Panák | Správanie systému |

## Zápisnica zo stretnutia tímu 2. 11. 1998

### Prítomní:

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Boris Steiner  
Martin Szolgay

### Zapisovateľ:

Andrej Oriško

Kontrola úloh pridelených na minulom stretnutí (predbežná verzia špecifikácie požiadaviek). Členovia tímu spoločne prehodnotili vypracované úlohy pridelené na minulom stretnutí s dôrazom na obsah a konzistenciu vypracovanej dokumentácie.

1. Do úvodu (Juraj Mikluš) bude potrebné okrem všeobecnejších cieľov projektu, doplniť údaje o členoch riešiteľského kolektívu. Tieto údaje budú prístupné aj na web stránke. Do budúceho stretnutia členovia tímu poskytnú potrebné údaje.
2. Plán systému (Andrej Oriško) - zohľadniť nové dostupné informácie na web stránke k predmetu; podrobnejšie určiť úlohy a priority.
3. Špecifikácia funkcií systému (Boris Steiner). Táto časť dokumentu by mala poskytovať užívateľský pohľad na systém; detailná špecifikácia bude uvedená v ďalších častiach dokumentácie. Boli vybrané ukazovatele, ktoré sa budú vyhodnocovať; okrem lokálnych (týkajúcich sa jedného objektu) je potrebné podávať aj globálny pohľad na systém (napr. rozdelenie príchodov do obchodného domu v závislosti od času). V dokumente bude potrebné popísať presnejšie možnosti vizualizácie. Taktiež už bola presne konkretizovaná požiadavka na rozhranie systému. Na základe dohovoru so zadávateľom, so zreteľom na obmedzený čas na vývoj a užitočnosť bolo stanovené, aký typ externých modulov bude systém podporovať. Prostredníctvom externých modulov bude možné rozširovať možnosti vyhodnocovania údajov o systéme. Externý modul bude pasívne zaznamenávať namerané údaje alebo stavy objektov, ktoré sa budú zobrazovať v samostatnom okne. Externé moduly budú používať technológiu COM.
4. Špecifikácia údajov v systéme (Martin Szolgay) - identifikácia údajových entít a ich atribútov. Úpravy týkajúce sa konzistencie pomenovania entít, rozlišovanie základných objektov používaných pri simulácii a k nim zodpovedajúcich (odvodených) objektov potrebných pri simulácii obchodného domu. Zdôvodnenie atribútov a vlastností objektov. Kvôli prehľadnosti bude potrebné zobrazovať graficky.
5. Opis správania systému (Miroslav Panák). Vytvorený dokument sa čiastočne prekrýva so špecifikáciou údajov a špecifikáciou funkcií systému. Podrobnejšiu časť zo špecifikácie funkcií systému bude potrebné premiestniť do tejto časti a doplniť presnú softvérovú špecifikáciu. Identifikovať základné moduly a ich funkcie. Popísať slovnou činnosť systému v štandardných a aj neočakávaných situáciách. Popísať algoritmus pohybu transakcie (zákazníka) v systéme, ktorý bol špecifikovaný na predošlom stretnutí.
6. Bol vyskúšaný prototyp web stránky (Andrej Oriško) a základná funkčnosť - nastavenie prístupových práv. Stránky bude potrebné upraviť, kvôli optimálnejšiemu

vzhľadu na monitoroch s menším rozlíšením. Vzhľadom na to, že obsah stránky nebol kompletný na viac než 50%, stránka nebola zverejnená. Stránka bude obsahovať až finálnu verziu dokumentácie.

7. Na budúcom stretnutí sa bude musieť prvá časť dokumentácie úplne uzavrieť, vy publikovať na web stránke a vytlačiť.

## **Zápisnica zo stretnutia tímu 9. 11. 1998**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Boris Steiner  
Martin Szolgay

### **Zapisovateľ:**

Martin Szolgay

### **Kontrola úloh z minulého stretnutia**

Každý člen tímu doplnil a prepracoval zadané časti špecifikácie:

### **Úlohy splnené v rámci stretnutia**

1. Do úvodu doplnené údaje o členoch riešiteľského kolektívu, podrobnejší popis cieľov projektu (Juraj Mikluš).
2. Plán systému, doplnenie nových informácií na domovskej stránke, určenie úloh a priorít (Andrej Oriško).
3. Špecifikácia funkcií systému, užívateľský pohľad na systém. Presnejší popis možnosti vizualizácie (Boris Steiner).
4. Špecifikácia údajov v systéme, úpravy týkajúce sa konzistencie pomenovania entít, rozlišovanie základných objektov používaných pri simulácii a k nim zodpovedajúcich (odvođených) objektov potrebných pri simulácii obchodného domu. Podrobnejší popis atribútov a vlastností objektov. Vytvorenie prílohy: príklad vstupného súboru (Martin Szolgay).
5. Opis správania systému. Identifikácia základných modulov a ich funkcií. Slovný popis činnosti systému v štandardných a aj neočakávaných situáciách nie je úplný. Popísať algoritmus pohybu transakcie (zákazníka) v systéme, ktorý bol špecifikovaný na predošlom stretnutí (Miroslav Panák).
6. Doplnenie obsahu bodu Kontext systému (Andrej Oriško, Miroslav Panák) a Ďalšie požiadavky a ohraničenia (Andrej Oriško)
7. Kompletizácia zápisníc. Prepisovanie do digitálnej formy a zjednotenie štýlu zápisu (Martin Szolgay). Dopracovávanie a zjednocovanie špecifikácie. Vytvorenie preberacieho protokolu (Juraj Mikluš) a návrh titulnej strany špecifikácie (Martin Szolgay).
8. Určenie člena tímu na tlač dokumentu (Andrej Oriško). Zabezpečenie väzby dokumentácie (Boris Steiner).
9. Práca na domovskej stránke. Domovská stránka bude obsahovať kompletnú dokumentáciu priamo na prezeranie a možnosť nahrať si kompletnú dokumentáciu vo formáte PDF (Andrej Oriško).

## **Zápisnica zo stretnutia tímu 16. 11. 1998**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Boris Steiner  
Martin Szolgay

### **Zapisovateľ:**

Miroslav Panák

### **Obsah stretnutia**

Vytvorenie posudku špecifikácie a hrubého návrhu tímu č. 7 (Animácia simulačných experimentov s modelom diskrétného systému). Dokument prebral Andrej Oriško dňa 12.11.1998. Diskusia o prečítanej práci. Hodnotenie formálnej (úpravy, dodržanie požadovaných náležitostí, pravopisné chyby) a obsahovej stránky (zrozumiteľnosť, jasnosť, čitateľnosť, presvedčivosť).

## Zápisnica zo stretnutia tímu 23. 11. 1998

### Prítomní:

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Boris Steiner  
Martin Szolgay

### Zapisovateľ:

Juraj Mikluš

### Obsah stretnutia

1. Oboznámenie sa s novým obsahom web stránky k tímovému projektu: Prezentácie prototypov musia byť najneskôr do štvrtka 17.12.1998. Dokumentáciu k prototypu tvorí používateľská príručka vytváraného systému. V príručke bude spracovaný systém ako celok, t.j. aj tie časti, ktoré sa z rôznych dôvodov nenachádzajú v prototypu, ale plánujú sa do výsledného systému a disketa, ktorá obsahuje prototyp a návod na jeho inštaláciu.
2. Nedostatky špecifikácie identifikované zadávateľom. Treba doplniť aká notácia bola použitá v kontextovom diagrame. Treba doplniť k obrázku, ktorý zobrazuje hierarchickú štruktúru tovarov väzbu na funkčnosť. V algoritme v kapitole 6 treba doplniť krok medzi inicializáciu simulátora a výber prvej udalosti. Menu web stránky obsahuje na jednej úrovni odkazy, ktoré by nemali byť logicky na jednej úrovni.
3. Preštudovali sme si posudok od tímu 10, ale len neoficiálne, keďže nám kópiu posudku nedodali v stanovenom termíne. Z posudku sme vybrali nasledujúce body, v ktorých bude realizovaná zmena:
  - do úvodu dokumentu treba dopísať jeho význam
  - v odstavci 4.5. „počet ľudí v závislosti od času“ znamená počet ľudí, ktorí navštívili oddelenie
  - do kapitoly 4.1 bude pravdepodobne potrebné doplniť popis funkcií entity oddelenie
  - v odstavci 6.1.1 by sa dalo uvažovať o graf. vyjadrení algoritmu v prirodzenom jazyku
  - upozornili nás na potrebu členenia dokumentu na vyššej úrovni (špecifikáciu, užívateľská príručka, hrubý návrh), toto bude zohľadnené pri rozširovaní dokumentu.
  - slovo „toolbar“ bude nahradené slovenským ekvivalentom
4. Zmeny v dokumente špecifikácie budú realizované formou dodatkov.
5. Pridelenie bodov za špecifikáciu:

|                |    |
|----------------|----|
| Juraj Mikluš   | 16 |
| Andrej Oriško  | 17 |
| Miroslav Panák | 15 |
| Boris Steiner  | 17 |
| Martin Szolgay | 16 |

## **Zápisnica zo stretnutia tímu 30. 11. 1998**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Boris Steiner  
Martin Szolgay

### **Zapisovateľ:**

Martin Szolgay

### **Obsah stretnutia**

1. Oboznámenie sa s výsledkami práce jednotlivých členov pripravenej na toto stretnutie.  
Boris Steiner predviedol pripravený systém ponúk vo zvolenom implementačnom prostredí. Taktiež predviedol voľne širiteľné triedy na implementáciu tzv. docking windows. Andrej Oriško pripravil podklady pre vzhľad obrazoviek.
2. Rozdelenie úloh spracovávaných na tomto stretnutí:  
Martin Szolgay - doplnenie dodatkov do špecifikácie  
Juraj Mikluš, Miroslav Panák - tvorba dialógov rozhrania  
Boris Steiner - tvorba hlavnej ponuky prototypu  
Andrej Oriško - dopĺňanie dialógov, koordinácia práce
3. Spracovávanie úloh, konzultácia vzhľadu ponuky a dialógových obrazoviek.
4. Návrh tried prototypu a ich rozhraní.
5. Návrh častí manuálu:  
Úvod (zhrnuté možnosti a význam softvéru)  
Postup pri inštalácii (vrátane požiadaviek na HW a SW)  
Kapitoly popisujúce štandardný postup pri používaní programu:  
Práca s modelom (vytvorenie, modifikácia, uloženie)  
Riadenie simulácie  
Prezeranie výsledkov simulácie  
Popis všetkých položiek ponuky  
Podrobný popis dialógov  
Riešenie často kladených otázok.
6. Rozdelenie úloh:  
Boris Steiner - implementácia prostredia  
Andrej Oriško - príprava objektov na zobrazovanie grafov, zadávanie času a pravdepodobností  
Martin Szolgay - popis ponúk  
Juraj Mikluš - popis dialógov  
Miroslav Panák - úvod

## **Zápisnica zo stretnutia tímu 7. 12. 1998**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Boris Steiner  
Martin Szolgay

### **Zapisovateľ:**

Miroslav Panák

### **Obsah stretnutia**

1. Práca na prototyp aplikácie. (Oriško, Steiner, Mikluš)  
Integrácia pripravených tried na zobrazovanie grafov a zadávanie parametrov rozdelenia náhodných premenných do hotovej časti prototypu. Vloženie dialógových okien navrhnutých na predchádzajúcom stretnutí, pokračovanie vo vytváraní dialógových okien. Do budúceho stretnutia, alebo aj skôr, bude potrebné vytvoriť prototyp, na základe ktorého bude možné napísať užívateľskú príručku.
2. Príprava užívateľskej príručky. (Szolgay, Panák)  
Vytváranie častí užívateľskej príručky na základe návrhu a hotovej časti prototypu. Príprava a zlučovanie častí príručky vytvorených jednotlivými členmi. Popis základných činností pri vytváraní, modifikácii a ukladaní modelu obchodného domu. Oboznámenie sa s obsahom užívateľských príručiek známych produktov (Acrobat Reader, Internet Explorer).



## **Zápisnica zo stretnutia tímu 14. 12. 1998**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Boris Steiner  
Martin Szolgay

### **Zapisovateľ:**

Andrej Oriško

### **Obsah stretnutia**

1. Práca na prototype aplikácie. (Oriško, Steiner, Mikluš)  
Tvorcovia prototypu (Steiner, Oriško) dodali v piatok (11. 12.) verziu prototypu s implementovanou väčšinou plánovaných funkcií. Počas stretnutia sa vykonávali posledné úpravy na prototype (nové ikony, ukážkové dáta). Úprava vzhľadu dialógov (Mikluš).
2. Príprava užívateľskej príručky. (Szolgay, Panák, Mikluš)  
Opisy dialógov (Panák - vyhľadávanie, nastavenie bodov zastavenia, Mikluš - nastavenie aplikácie, Szolgay - nastavenie externých modulov).  
Často kladené otázky a odpovede (Szolgay).  
Popis vlastností objektov (Mikluš).
3. Príručku spojí, doplní chýbajúce časti a vytlačí Andrej Oriško.

## **Zápisnica zo stretnutia tímu 23. 2. 1999**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Boris Steiner  
Martin Szolgay

### **Zapisovateľ:**

Andrej Oriško

### **Obsah stretnutia**

1. Prvé stretnutie v letnom semestri. Stretnutia tímu v letnom semestri budú v utorok od 10.00 do 13.00. Vzhľadom na plánované zmeny v laboratóriu D132 bude nutné dokončiť program skôr ako bolo pôvodne plánované, konkrétne do konca apríla.
2. Členovia tímu odovzdali 19. 2. 1999 posudok prototypu tímu 7, obdržali posudok od tímu 10 a oboznámili sa s jeho obsahom. Prezentácia bola hodnotená pozitívne, až na príliš vysoké rozlíšenie monitoru, ktoré znemožňovalo rozoznať detaily z väčšej vzdialenosti. Vytvorený prototyp splnil očakávania, v prototypy chýbali príklady výsledkov (grafy). Užívateľská príručka bola celkovo hodnotená po obsahovej a formálnej stránke pozitívne s nasledovnými nedostatkami:
  - nepresne definovaný termín „dynamický stav modelu“
  - členenie kapitoly 4.1.2, prípadne zobrazenie obsahu ponúk
  - význam a spôsob použitia regálov
  - popis vlastností objektov a ich atribútov je všeobecný
  - neúplné vysvetlenie použitia externých modulov
  - v požiadavkách na systém chýba požiadavka na verziu knižnice MFC.
3. Diskusia o obsahu práce v letnom stretnutí, ktorá bude spočívať v podrobnom návrhu, implementácii a vytvorení dokumentácie. Tím bude pokračovať vo vytváraní užívateľského rozhrania (z vytvoreného prototypu) a paralelne implementovať simulovaný model a simulátor. Vytváranie dokumentácie sa presunie pravdepodobne do druhej polovice semestra.
4. Andrej Oriško predstavil prvú (neúplnú) verziu návrhu tried, ich hierarchiu a v skratke oboznámil členov tímu s funkčnosťou knižnice Tiny, ktorá sa využije pri implementácii. Do budúceho stretnutia sa členovia tímu oboznámia dôkladne s knižnicou Tiny a súčasným stavom návrhu, implementujú niekoľko elementárnych tried (Juraj Mikluš, Martin Szolgay). Boris Steiner pokračuje v práci na prototypy; zoznam nedostatkov a neimplementovaných funkcií bol súčasťou inštalácie prototypu. Andrej Oriško pokračuje na návrhu tried a pripraví plán projektu na letný semester.
5. Bude tiež potrebné dopracovať nedostatky v odovzdanej dokumentácii formou dodatku.

## **Zápisnica zo stretnutia tímu 2. 3. 1999**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Boris Steiner  
Miroslav Panák  
Martin Szolgay

### **Zapisovateľ:**

Martin Szolgay

### **Obsah stretnutia**

1. Kontrola zadaných úloh z minulého stretnutia. Vypracované úlohy zadané na minulom stretnutí (Juraj Mikluš, Martin Szolgay) boli počas stretnutia do knižnice pridané. Návrh bol rozšírený o triedy zabezpečujúce v/v operácie nad modelom a jeho objektami a vyhodnocovanie výrazov, napr. podmienky bodov zastavenia (Andrej Oriško).
2. Pridelovanie úloh pokračuje podľa plánu. Juraj Mikluš, Miroslav Panák, Martin Szolgay pracujú na implementácii samostatných modulov simulátora, Boris Steiner na užívateľskom rozhraní a Andrej Oriško na podrobnom návrhu. Rozdelenie nových úloh (do budúceho stretnutia):
  - Implementácia objektov na čítanie a zapisovanie simulovaného modelu a jeho entít (a testovanie). Jedná sa o všeobecne použiteľné triedy, využitie týchto tried bude realizované neskôr (Juraj Mikluš)
  - Implementácia vyhľadávacích výrazov a atribútov objektov (Miroslav Panák)
  - Notifikácia objektov CObserver v objektoch tiny, algoritmus rozhodovania zákazníka (Martin Szolgay)
  - Pokračovanie v implementácií užívateľského rozhrania, doplnenie špecifikácie kategórií zákazníkov užívateľom, návrh spojenia užívateľského rozhrania so simulátorom (Boris Steiner)
  - Pokračovanie v podrobnom návrhu. Vytváranie hierarchie tried, tvorba dokumentácie - diagram hierarchie tried a diagram sledu udalostí pri pohybe zákazníka po obchodnom dome (Andrej Oriško).
3. Oboznámenie sa s plánom stretnutí na letný semester. Plán projektu je zatiaľ tu. Doplnenie dokumentácie vzhľadom na pripomienky hodnotiaceho tímu (Miroslav Panák, Juraj Mikluš).

## **Zápisnica zo stretnutia tímu 9. 3. 1999**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Boris Steiner  
Miroslav Panák  
Martin Szolgay

### **Zapisovateľ:**

Andrej Oriško

### **Obsah stretnutia**

1. Kontrola zadaných úloh z minulého stretnutia a pridelenie nových úloh  
Doplnená dokumentácia vzhľadom na pripomienky hodnotiaceho tímu (Miroslav Panák). Dokumentácia bude dostupná aj na web stránke (Andrej Oriško).
2. Doplnená funkčnosť elementov obchodného domu o operácie rozhrania Observable (Martin Szolgay).
3. Vytvorený diagram sledu udalostí pri pohybe zákazníka po obchodnom dome (Andrej Oriško), ktorý je potrebný pri implementácii pohybu a rozhodovania zákazníka po obchodnom dome (Martin Szolgay, pokračovanie v implementácii rozhodovania zákazníka). Počas stretnutia bol upresnený význam atribútu „preferencia zákazníka“.
4. Dopĺňanie užívateľského rozhrania - interné zmeny (Boris Steiner).
5. Implementované všeobecné triedy na vyhodnocovanie výrazov - bodov zastavenia (Miroslav Panák). Miroslav Panák bude implementovať funkcie rozhrania na vyhodnocovanie výrazov pre konkrétne objekty v obchodnom dome. Oboznámi sa tiež s možnosťami vyhodnocovania údajov v knižnici tiny.
6. Triedy na ukládanie a čítanie atribútov obchodného domu zo/do súboru. Prehodnotenie návrhu týkajúceho sa čítania objektov. Juraj Mikluš bude pokračovať v implementácii. V druhej fáze bude implementovať čítanie a zapisovanie atribútov konkrétnych elementov obchodného domu.
7. Andrej Oriško pokračuje v návrhu a vytváraní dokumentácie.

## **Zápisnica zo stretnutia tímu 16. 3. 1999**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Boris Steiner  
Miroslav Panák  
Martin Szolgay

### **Zapisovateľ:**

Andrej Oriško

### **Obsah stretnutia**

1. Kontrola zadaných úloh z minulého stretnutia a pridelenie nových úloh  
Dokončená implementácia všeobecných tried na vyhodnocovanie výrazov, pokračovanie v implementácii pre konkrétne triedy obchodného domu (Miroslav Panák), ktoré sú potrebné pri vyhľadávaní objektov a vyhodnocovaní bodov zastavenia. Doplnený návrh; navrhnutý algoritmus vyhľadávania objektov v obchodnom dome a rozšírené triedy na štatistiku v knižnici Tiny, doplnená trieda CSimConsole, ktorá slúžia ako štandardný vstup/výstup, upravená knižnica Tiny (Andrej Oriško).  
Kompletný zoznam zmien je k dispozícii členom tímu.  
Práca na ostatných úlohách sa opozďuje kvôli termínom odovzdávania zadaní z iných predmetov. Termín sa posúva na budúce stretnutie.
2. Príprava dokumentácie  
Návrh štruktúry dokumentácie, ktorú treba vytvoriť v letnom semestri. Niektoré kapitoly (Úvod, Plán, Štandardy kódovania) sú už hotové. Dokumentáciu bude z väčšej časti vytvárať Andrej Oriško, ostatní členovia tímu prispievajú do kapitoly Opis implementácie, kde popíšu nimi implementované časti systému.  
Diskusia o obsahu niektorých častí. Obsah týchto častí ešte bude potrebné presne objasniť: Fyzický model údajov systému, Určenie spôsobu testovania a údajov na testovanie, Testovanie.

## **Zápisnica zo stretnutia tímu 23. 3. 1999**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Boris Steiner  
Miroslav Panák  
Martin Szolgay

### **Zapisovateľ:**

Andrej Oriško

### **Obsah stretnutia**

1. Kontrola zadaných úloh z minulého stretnutia a pridelenie nových úloh  
Dokončená implementácia triedy CArchive slúžiacej na textový vstup/výstup vlastností objektov. Prv než sa začne s implementáciou rozhrania CArchivable objektov obchodného domu, bude ju potrebné [CArchive] otestovať (Juraj Mikluš).
2. Implementované rozhranie umožňujúce vyhľadávanie objektov v obchodnom dome. Pri porovnávaní reťazcov je potrebné zohľadniť užívateľské nastavenie (ignorovať diakritiku). Dopracovať porovnávanie atribútov pre triedy CSuperMarket a CWareAbstract. Preveriť (všeobecne) algoritmus vyhľadávania. Doplňiť zaznamenávanie údajov pre grafy (Miroslav Panák).
3. Dokončená implementácia pohybu zákazníka po obchodnom dome. Algoritmus bude potrebné odladiť, obchodný dom bude potrebné vytvoriť dynamicky, keďže čítanie zo súboru ešte nie je implementované. Po odladení samotného algoritmu sa bude testovať funkčnosť bodov zastavenia (Martin Szolgay).
4. Rôzne interné zmeny v GUI, príprava spojenia so simulátorom (Boris Steiner).  
Príprava spojenia so simulátorom - sprístupnenie hodnôt v dialógoch vyhľadávania a bodov zastavenia, implementované a odladený objekt GUI na špecifikovanie rozdelenia náhodnej premennej. Navrhnuť lgoritmus vyhľadávania objektov (pseudokód). (Andrej Oriško).

## **Zápisnica zo stretnutia tímu 30. 3. 1999**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Boris Steiner  
Miroslav Panák  
Martin Szolgay

### **Zapisovateľ:**

Andrej Oriško

### **Obsah stretnutia**

1. Kontrola zadaných úloh z minulého stretnutia a pridelenie nových úloh  
Simulačný procesor algoritmus pohybu zákazníka sú implementované a čiastočne odladené (Martin Szolgay). Je potrebné otestovať regále a oddelenia, potom príde na rad testovanie štatistík a bodov zastavenia.
2. Dokončené vyhodnocovanie výrazov atribútov obchodného domu a tovaru (Miroslav Panák). Doplnené časti označené ako nedokončené týkajúce sa práce so zoznamami. Pokračuje sa implementáciou algoritmu vyhľadávania v obchodnom dome a výmeny dát s dialógovými oknami.
3. Úprava objektu dockable tree na view, spojenie s triedou CDocTemplate, CSimulView. Prenos funkcionality s CTreeCtrl na CTreeView (Boris Steiner).
4. Implementácia čítania a zapisovania špecifikácie obchodného domu ešte nie je hotová (Juraj Mikluš), mala byť začatá už pred dvoma týždňami.
5. Návrh algoritmu vyhľadávania, implementácia dialógov pre zadanie podmienok pre body zastavenia, vyhľadávanie a objektu na zadávanie rozdelenia náhodnej premennej (Andrej Oriško). Bude potrebné modifikovať triedu rad v knižnici Tiny (kvôli štatistikám), vyriešiť vyhodnocovanie atribútov niektorých objektov, týkajúcich sa bodov zastavenia a pracovať na dokumentácii.

## **Zápisnica zo stretnutia tímu 13. 4. 1999**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Martin Szolgay

### **Zapisovateľ:**

Andrej Oriško

### **Obsah stretnutia**

1. Kontrola zadaných úloh z minulého stretnutia a pridelenie nových úloh  
Odstránená nekonzistencia všetkých atribútov entít obchodného domu.  
Implementované porovnávanie reťazcov ak sa ignoruje diakritika (iba kódovanie 1250).  
Dorobené zaznamenávanie štatistík vo všetkých entitách obchodného domu.  
Implementované vyhľadávanie. (Miroslav Panák)
2. Zmenená trieda CAbstractQueue v tiny- už obsahuje histogram. Rozsahy sa nastavujú pomocou setHistogramParams. Dokončené časti dokumentácie - Architektúra systému, Fyzický model údajov systému, Ohraničenia, zmeny špecifikácie, priority riešenia, Výber implementačného jazyka a prostredia. Pokračovanie v práci na dokumentácii. (Andrej Oriško)
3. Odladený simulačný procesor, model a algoritmus pohybu zákazníka. Pokračuje sa ladením štatistík a bodov zastavenia. (Martin Szolgay)
4. Odladené implementácie CArchive a CArchivable, čiastočne implementované rozhranie CArchivable pre konkrétne triedy (opozdenie 3 týždne). (Juraj Mikluš)



## **Zápisnica zo stretnutia tímu 20. 4. 1999**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Boris Steiner  
Martin Szolgay

### **Zapisovateľ:**

Andrej Oriško

### **Obsah stretnutia**

1. Odladený algoritmus vyhľadávania, implementovaná výmena dát s dialógom vyhľadávania. Pokračuje sa implementáciou výmeny dát s dialógmi s bodmi zastavenia (Miroslav Panák).
2. Odladené štatistiky, hodnoty CTrans->TotalActivity sú neplatné. Do budúceho stretnutia vyriešiť spojenie tried CHistogram a CGroupCounter s HWND triedami SimPieChart resp. SimXYChart (napísať funkciu, ktorá nastaví hodnoty grafu podľa hodnôt CHistogram alebo CGroupCounter). Odladiť body zastavenia na základe pripraveného príkladu na vytvorenie bodu zastavenia (Martin Szolgay).
3. Doplnenie funkcií na sprístupnenie atribútov objektov CSimSettings a CWareAbstract. Pripravený príklad na vytváranie bodu zastavenia potrebný pre testovanie. Premenovanie triedy CArchive na CMyArchive kvôli MFC. Zmena dátového modelu - relácie medzi preferenciou, regálom a tovarom/skupinou. Pokračovanie v tvorbe dokumentácie, spájanie GUI so simulátorom, odstránenie nájdenej chyby v CTrans (Andrej Oriško).
4. Dokončená implementácia rozhrania CArchivable u odvodených tried. Rôzne drobné zmeny - doplnenie metód na sprístupnenie atribútov niektorých objektov. Implementované funkcie je potrebné otestovať. Preveriť obsah dialógových okien programu obsahujúcich atribúty objektov na základe pripraveného zoznamu (Juraj Mikluš).
5. Spájanie GUI so simulátorom. Presmerovanie CSimConsole do okna. Zmena lišty riadenia simulácie - obsahuje už aj simulačný čas a priebeh simulácie. Vytváranie zobrazovaného stromu (obch. dom, tovar) na základe štruktúry modelu. Pokračovanie v spájaní GUI so simulátorom: vytváranie modelu, čítanie, zapisovanie, spustenie a zastavenie simulácie, neskôr body zastavenia a vyhľadávanie (Boris Steiner).

## **Zápisnica zo stretnutia tímu 27. 4. 1999**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Boris Steiner  
Martin Szolgay

### **Zapisovateľ:**

Martin Szolgay

### **Kontrola zadaných úloh z minulého stretnutia**

- Miroslav Panák: Testovanie vyhľadávania - odstránené chyby vo funkčnosti operátorov a vo vyhľadávaní.
- Martin Szolgay: Naprogramované spojenie tried CHistogram a CGroupCounter s HWND triedami SimPieChart resp. SimXYChart (funkcie nastavujú hodnoty grafu podľa hodnôt CHistogram alebo CGroupCounter). Testovanie bodov zastavenia: simulácia sa vôbec nezastaví. Chybu odstráni Andrej Oriško.
- Andrej Oriško: Pokračovanie v tvorbe dokumentácie, spájanie GUI so simulátorom, odstránenie nájdenej chyby v CTrans.
- Juraj Mikluš: Otestovanie implementácie archivácie. Odsadzovanie textu v súbore podľa hĺbky v hierarchii. Úvodzovky sa nepoužívajú pokiaľ to nie je nutné.
- Boris Steiner: Spájanie GUI so simulátorom: vytváranie modelu, čítanie, zapisovanie, spustenie a zastavenie simulácie, neskôr body zastavenia a vyhľadávanie.

### **Práca na stretnutí**

Spájanie posledných zmien. Opravy chýb vzniknutých pri spájaní.  
Pokračovanie v odladovaní a spájanie simulácie s GUI.

### **Pridelenie nových úloh**

- Miroslav Panák: Implementácia výmeny dát s dialógmi s bodmi zastavenia.
- Martin Szolgay: Kapitoly "Určenie spôsobu testovania" a "Testovanie" týkajúce sa testovania algoritmu rozhodovania zákazníka.
- Juraj Mikluš: Kapitoly "Určenie spôsobu testovania" a "Testovanie" týkajúce sa testovania čítania a zapisovania definície modelu do súboru, kontrola obsahu okien GUI a vytvorenie chýbajúcich okien zobrazujúcich vlastnosti objektov.
- Andrej Oriško: Práca na dokumentácii, rozhranie na nastavovanie bodov zastavenia z GUI, odstránenie chyby v bodoch zastavenia.
- Boris Steiner: GUI - body zastavenia a vyhľadávanie, odstránenie chyby v CLogBar pri volaní z iného threadu.

## **Zápisnica zo stretnutia tímu 4. 5. 1999**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Boris Steiner  
Martin Szolgay

### **Zapisovateľ:**

Juraj Mikluš

### **Obsah stretnutia**

#### **Kontrola zadaných úloh z minulého stretnutia a pridelenie nových úloh**

- Miroslav Panák: Dokončená implementácia výmeny dát s dialógmi s bodmi zastavenia. Doplň funkčnosť operátora „zmení sa o“ do dialógov s bodmi zastavenia. Vložiť grafy do okien zobrazujúcich vlastnosti objektov.
- Martin Szolgay: Kapitoly „Určenie spôsobu testovania“ a „Testovanie“ týkajúce sa testovania algoritmu rozhodovania zákazníka. Bude pokračovať ďalej a otestuje všetky typy bodov zastavenia.
- Juraj Mikluš: Kapitoly „Určenie spôsobu testovania“ a „Testovanie“ týkajúce sa testovania čítania a zapisovania definície modelu do súboru, kontrola obsahu okien GUI a vytvorenie chýbajúcich okien zobrazujúcich vlastnosti objektov (dokončené).
- Andrej Oriško: Práca na dokumentácii, rozhranie na nastavovanie bodov zastavenia z GUI, odstránená chyba v bodoch zastavenia. Príprava web stránky a práca na implementácii GUI.
- Boris Steiner: GUI - body zastavenia a vyhľadávanie, odstránenie chyby v CLogBar pri volaní z iného threadu. Práca na implementácii GUI - okná zobrazujúce vlastnosti objektov.

#### **Práca na stretnutí**

Spájanie posledných zmien. Opravy chýb vzniknutých pri spájaní. Pokračovanie v odladovaní a spájanie simulácie s GUI.

## **Zápisnica zo stretnutia tímu 11. 5. 1999**

### **Prítomní:**

Juraj Mikluš  
Andrej Oriško  
Miroslav Panák  
Boris Steiner  
Martin Szolgay

### **Zapisovateľ:**

Juraj Mikluš

### **Obsah stretnutia**

#### **Kontrola zadaných úloh z minulého stretnutia**

- Miroslav Panák: Doplnil funkčnosť operátora "zmení sa o" do dialógov s bodmi zastavenia. Vložil grafy do okien zobrazujúcich vlastnosti objektov. Nastavovanie prvkov dialógových okien (štatistiky).
- Martin Szolgay: Práca na dokumentácii (testovanie, záver). Nastavovanie prvkov dialógových okien (štatistiky).
- Juraj Mikluš: Testovanie web stránky. Tlačenie dokumentácie.
- Andrej Oriško: Pripravil web stránku. Pripravil dokumentáciu.
- Boris Steiner: GUI - Dorobil prácu s bodmi zastavenia v GUI. Práca na implementácií GUI. Spájanie simulátora s GUI - vytváranie a vymazávanie objektov. Integračné testy.

## Príloha D: Posudok špecifikácie a hrubého návrhu

Tento dokument obsahuje posudok špecifikácie a hrubého návrhu tímu č. 7 z predmetu Tímový projekt. Hodnotenie vykonali všetci členovia tímu č. 11. Hodnotená bola formálna a obsahová stránka odovzdaného dokumentu.

### Zhodnotenie jednotlivých častí

#### 1. Úvod

Kapitola má obsahovať ciele projektu, ohraničenia a riešiteľský kolektív.

Pozitívne hodnotíme predstavenie riešiteľského kolektívu, kde sú uvedené okrem schopností členov tímu aj ich funkcie, ktoré v tíme zastávajú.

Nedostatkom úvodu je absencia ohraničení projektu a všeobecného cieľa projektu okrem cieľa vytvorenia dokumentu.

#### 2. Plán projektu

Kapitola má obsahovať rozdelenie úloh členov riešiteľského kolektívu a plán stretnutí na jednotlivé týždne.

Pozitívne hodnotíme, že pridelenie úloh členom obsahuje okrem plánovania budúcich rozhodnutí a úloh aj plán štúdia metodiky objektovo - orientovaného návrhu, vyhľadania a štúdia existujúcich podporných prostriedkov v oblasti simulácie, štatistiky a vizualizácie.

#### 3. Kontext systému

Obsahuje rozhranie systému, jeho základnú funkciu a iné systémy, ktoré budú so systémom komunikovať.

Kapitola má požadovaný obsah, s nasledujúcimi nedostatkami:

V časti 3.1 je uvedený okrem zákazníkov a pokladní aj personál, ktorý sa ďalej v texte neuvádza.

V obrázku na strane 5 sú identicky pomenované dva rôzne moduly, čo môže byť zavádzajúce. Z obrázku nie je zrejma interakcia užívateľa so simulátorom. Tok dát *parameter simulácie* by mohol priamo smerovať od užívateľa do modulu *1.1 Simulácia*. Úmyslom autorov bolo pravdepodobne znázorniť, že riadenie simulácie užívateľom sa uskutočňuje pomocou užívateľského rozhrania (modul *1.3 Vizualizácia*).

Nie je explicitne uvedené, či sa rozlišujú triedy užívateľov pracujúcich so systémom (modifikovanie modelu, prezeranie simulácie).

#### 4. Špecifikácia funkcií systému

Kapitola obsahuje popis funkcií poskytovaných systémom. Určuje sa aj priorita pre jednotlivé funkcie.

Bolo by možno vhodnejšie rozdeliť funkcie nie podľa modulov, ale podľa kategórií poskytovaných funkcií z hľadiska systému (funkcie na údržbu modelu, riadenie simulácie a prezentácie výsledkov).

V simulačnom module nie je špecifikované, akým spôsobom možno vytvárať a modifikovať simulačný model a či je možné meniť parametre simulácie.

Štatistický modul neobsahuje presne určené typy štatistík.

Priorita jednotlivých funkcií nie je určená.

## 5. Špecifikácia údajov v systéme

Obsahuje okrem podrobného popisu atribútov aj správanie jednotlivých entít, ich základné funkcie, vzťahy k ostatným entitám a vysvetlenia zvolených zjednodušení systému.

V odstavcoch 5.1 a 5.2 však možno nájsť protichodné tvrdenia. V časti 5.1 sa uvádza, že generovanie zákazníkov postačuje pre každé oddelenie zvlášť, čo je v protiklade s tvrdením v časti 5.2, že simulácia si kladie za cieľ simulovať životný cyklus zákazníka.

Časť 5.3 obsahuje formálnu chybu: na začiatku je uvedené, že modifikovateľné atribúty sú vytlačené kurzívou. Napriek tomu žiaden atribút kurzívou napísaný nie je, takže nie je zrejmé, ktoré atribúty sú vlastne modifikovateľné.

Z názvu niektorých neopísaných atribútov nie je jasný ich význam a typ (napríklad rýchlosť obsluhy, stredná hodnota maximálnej akceptovateľnej dĺžky radu).

## 6. Špecifikácia správania systému

Kapitola obsahuje popis architektúry systému, návrh komunikácie modulov systému klient-server. Navrhovaná architektúra systému umožňuje vymeniť vizualizačný front-end, ktorý je spomínaný v zadaní projektu, hoci autori túto alternatívu neuvažovali.

Hlavný nedostatok jednak tejto kapitoly, ale aj celej špecifikácie vzhľadom na zrozumiteľnosť je, že nie je možné určiť, či:

- a) všetci užívatelia sledujú tú istú simuláciu toho istého modelu,
- b) každý užívateľ nezávisle simuluje ten istý model,
- c) užívatelia nezávisle sledujú simulácie rôznych modelov,

čo považujeme za kľúčový bod na pochopenie funkčnosti celého systému. V súvislosti s týmto nedostatkom by bolo vhodné uviesť riešenie konfliktov v prípade prístupu k simulovanému modelu viacerými užívateľmi súčasne.

## 7. Ďalšie požiadavky a ohraničenia

Kapitola zhodnocuje najmä technické požiadavky a obmedzenia, z ktorých je možné si vytvoriť reálnu predstavu nárokov systému (podporovaná platforma, požiadavky na hardvér a softvér).

## 8. Dodatky

Dodatky obsahujú ponuku a záznamy zo stretnutí tímu. Záznamy zo stretnutí poskytujú v prehľadnej forme informácie o obsahu stretnutí, plnení a pridelovaní úloh. Obsahujú aj možné alternatívy diskutovaných problémov, ich výhody a nevýhody.

### Formálna stránka

Vytvorený dokument obsahuje 12 strán textu a 4 strany príloh zviazaný hrebeňovou väzbou v tvrdom obale. Po formálnej stránke dokumentu nemožno takmer nič vytknúť - dokument má prehľadnú a jednotnú grafickú úpravu, obsahuje 3 obrázky na doplnenie textu. Obrázkom chýba popis a číslovanie, dokument nemá obsah, ktorý vzhľadom na jeho rozsah možno nie je ani potrebný.

## **Príloha E: Preberacie protokoly**

Slovenská technická univerzita v Bratislave  
Fakulta elektrotechniky a informatiky  
KATEDRA INFORMATIKY A VÝPOČTOVEJ TECHNIKY  
Študijný odbor: SOFTVÉROVÉ INŽINIERSTVO

# **PREBERACÍ PROTOKOL**

ŠPECIFIKÁCIE TÍMOVÉHO PROJEKTU  
ANIMÁCIA SIMULAČNÝCH EXPERIMENTOV  
S MODELOM DISKRÉTNEHO SYSTÉMU

## **TÍM 11**

Bc. Juraj MIKLUŠ  
Bc. Andrej ORIŠKO  
Bc. Miroslav PANÁK  
Bc. Boris STEINER  
Bc. Martin SZOLGAY

## **ZADÁVATEĽ**

Ing. Ivan Lališ

Potvrdzujem prebratie špecifikácie projektu.

V Bratislave, 12.11.1998

Slovenská technická univerzita v Bratislave  
Fakulta elektrotechniky a informatiky  
KATEDRA INFORMATIKY A VÝPOČTOVEJ TECHNIKY  
Študijný odbor: SOFTVÉROVÉ INŽINIERSTVO

## PREBERACÍ PROTOKOL

ŠPECIFIKÁCIE TÍMOVÉHO PROJEKTU  
ANIMÁCIA SIMULAČNÝCH EXPERIMENTOV  
S MODELOM DISKRÉTNEHO SYSTÉMU

### TÍM 11

Bc. Juraj MIKLUŠ  
Bc. Andrej ORIŠKO  
Bc. Miroslav PANÁK  
Bc. Boris STEINER  
Bc. Martin SZOLGAY

### ZADÁVATEĽ

Ing. Ivan Lališ

Potvrdzujem prebratie špecifikácie projektu.

V Bratislave, 12.11.1998



Slovenská technická univerzita v Bratislave  
Fakulta elektrotechniky a informatiky  
KATEDRA INFORMATIKY A VÝPOČTOVEJ TECHNIKY  
Študijný odbor: SOFTVÉROVÉ INŽINIERSTVO

## PREBERACÍ PROTOKOL

POSUDKU ŠPECIFIKÁCIE TÍMOVÉHO PROJEKTU  
ANIMÁCIA SIMULAČNÝCH EXPERIMENTOV  
S MODELOM DISKRÉTNEHO SYSTÉMU

### TÍM 11

Bc. Juraj MIKLUŠ  
Bc. Andrej ORIŠKO  
Bc. Miroslav PANÁK  
Bc. Boris STEINER  
Bc. Martin SZOLGAY

### ZADÁVATEĽ

Ing. Ivan Lališ

Potvrdzujem prebratie posudku špecifikácie projektu.

V Bratislave, 19.11.1998

Slovenská technická univerzita v Bratislave  
Fakulta elektrotechniky a informatiky  
KATEDRA INFORMATIKY A VÝPOČTOVEJ TECHNIKY  
Študijný odbor: SOFTVÉROVÉ INŽINIERSTVO

## PREBERACÍ PROTOKOL

POSUDKU ŠPECIFIKÁCIE TÍMOVÉHO PROJEKTU  
ANIMÁCIA SIMULAČNÝCH EXPERIMENTOV  
S MODELOM DISKRÉTNEHO SYSTÉMU

### TÍM 11

Bc. Juraj MIKLUŠ  
Bc. Andrej ORIŠKO  
Bc. Miroslav PANÁK  
Bc. Boris STEINER  
Bc. Martin SZOLGAY

### ZADÁVATEĽ

Ing. Ivan Lališ

Potvrdzujem prebratie posudku špecifikácie projektu.

V Bratislave, 19.11.1998

Slovenská technická univerzita v Bratislave  
Fakulta elektrotechniky a informatiky  
KATEDRA INFORMATIKY A VÝPOČTOVEJ TECHNIKY  
Študijný odbor: SOFTVÉROVÉ INŽINIERSTVO

## PREBERACÍ PROTOKOL

DOKUMENTÁCIE K PROTOTYPU TÍMOVÉHO PROJEKTU  
ANIMÁCIA SIMULAČNÝCH EXPERIMENTOV  
S MODELOM DISKRÉTNEHO SYSTÉMU

### TÍM 11

Bc. Juraj MIKLUŠ  
Bc. Andrej ORIŠKO  
Bc. Miroslav PANÁK  
Bc. Boris STEINER  
Bc. Martin SZOLGAY

### ZADÁVATEĽ

Ing. Ivan Lališ

Potvrdzujem prebratie dokumentácie k prototypu projektu.

V Bratislave, 17.12.1998

Slovenská technická univerzita v Bratislave  
Fakulta elektrotechniky a informatiky  
KATEDRA INFORMATIKY A VÝPOČTOVEJ TECHNIKY  
Študijný odbor: SOFTVÉROVÉ INŽINIERSTVO

# PREBERACÍ PROTOKOL

DOKUMENTÁCIE K PROTOTYPU TÍMOVÉHO PROJEKTU  
ANIMÁCIA SIMULAČNÝCH EXPERIMENTOV  
S MODELOM DISKRÉTNEHO SYSTÉMU

## TÍM 11

Bc. Juraj MIKLUŠ  
Bc. Andrej ORIŠKO  
Bc. Miroslav PANÁK  
Bc. Boris STEINER  
Bc. Martin SZOLGAY

## ZADÁVATEĽ

Ing. Ivan Lališ

Potvrdzujem prebratie dokumentácie k prototypu projektu.

V Bratislave, 17.12.1998

Slovenská technická univerzita v Bratislave  
Fakulta elektrotechniky a informatiky  
KATEDRA INFORMATIKY A VÝPOČTOVEJ TECHNIKY  
Študijný odbor: SOFTVÉROVÉ INŽINIERSTVO

# PREBERACÍ PROTOKOL

POSUDKU PROTOTYPU TÍMOVÉHO PROJEKTU  
ANIMÁCIA SIMULAČNÝCH EXPERIMENTOV  
S MODELOM DISKRÉTNEHO SYSTÉMU

## TÍM 11

Bc. Juraj MIKLUŠ  
Bc. Andrej ORIŠKO  
Bc. Miroslav PANÁK  
Bc. Boris STEINER  
Bc. Martin SZOLGAY

## ZADÁVATEĽ

Ing. Ivan Lališ

Potvrdzujem prebratie posudku prototypu projektu.

V Bratislave, 19.2.1999

Slovenská technická univerzita v Bratislave  
Fakulta elektrotechniky a informatiky  
KATEDRA INFORMATIKY A VÝPOČTOVEJ TECHNIKY  
Študijný odbor: SOFTVÉROVÉ INŽINIERSTVO

## PREBERACÍ PROTOKOL

POSUDKU PROTOTYPU TÍMOVÉHO PROJEKTU  
ANIMÁCIA SIMULAČNÝCH EXPERIMENTOV  
S MODELOM DISKRÉTNEHO SYSTÉMU

### TÍM 11

Bc. Juraj MIKLUŠ  
Bc. Andrej ORIŠKO  
Bc. Miroslav PANÁK  
Bc. Boris STEINER  
Bc. Martin SZOLGAY

### ZADÁVATEĽ

Ing. Ivan Lališ

Potvrdzujem prebratie posudku prototypu projektu.

V Bratislave, 19.2.1999

Slovenská technická univerzita v Bratislave  
Fakulta elektrotechniky a informatiky  
KATEDRA INFORMATIKY A VÝPOČTOVEJ TECHNIKY  
Študijný odbor: SOFTVÉROVÉ INŽINIERSTVO

# PREBERACÍ PROTOKOL

PRODUKTU A DOKUMENTÁCIE K TÍMOVÉHO PROJEKTU  
ANIMÁCIA SIMULAČNÝCH EXPERIMENTOV  
S MODELOM DISKRÉTNEHO SYSTÉMU

## TÍM 11

Bc. Juraj MIKLUŠ  
Bc. Andrej ORIŠKO  
Bc. Miroslav PANÁK  
Bc. Boris STEINER  
Bc. Martin SZOLGAY

Potvrdzujem prebratie produktu a dokumentácie.

V Bratislave, 13. 6. 1999

Slovenská technická univerzita v Bratislave  
Fakulta elektrotechniky a informatiky  
KATEDRA INFORMATIKY A VÝPOČTOVEJ TECHNIKY  
Študijný odbor: SOFTVÉROVÉ INŽINIERSTVO

## PREBERACÍ PROTOKOL

PRODUKTU A DOKUMENTÁCIE K TÍMOVÉHO PROJEKTU  
ANIMÁCIA SIMULAČNÝCH EXPERIMENTOV  
S MODELOM DISKRÉTNEHO SYSTÉMU

### TÍM 11

Bc. Juraj MIKLUŠ  
Bc. Andrej ORIŠKO  
Bc. Miroslav PANÁK  
Bc. Boris STEINER  
Bc. Martin SZOLGAY

Potvrdzujem prebratie produktu a dokumentácie.

V Bratislave, 13. 6. 1999



## Príloha F: Posudok prototypu

Tento dokument obsahuje posudok prototypu tímu č. 7 z predmetu Tímový projekt. Hodnotenie vykonali členovia tímu č. 11. Hodnotená bola formálna a obsahová stránka odovzdaného dokumentu, prototyp a jeho prezentácia.

### Dokumentácia

Odovzdaný dokument obsahuje 17 strán textu a 3 strany príloh zviazaných hrebeňovou väzbou. Dokument má prehľadnú grafickú úpravu a obsah. Súčasťou dokumentu je 8 číslovaných a pomenovaných obrázkov. Text je členený na 3 kapitoly - Úvod, Charakteristika systému a Použitie systému.

Dokument je pravdepodobne novou verziou dokumentácie odovzdanej 19.11.1998. Chýbajúce a nepresné časti v dokumentácii boli doplnené priamo v texte (teda nie formou prílohy), zmenila sa aj štruktúra celej dokumentácie a niektoré časti dokonca chýbajú.

Predstavenie členov tímu a plán projektu sú uvedené ako príloha, záznamy zo stretnutí chýbajú úplne. Uvedené fakty sú v rozpore s informáciami k predmetu uvedenými na web stránke, kde sa okrem iného uvádza „Vždy treba odovzdať úplnú doteraz vytvorenú dokumentáciu“ a tiež „Z každého stretnutia tím vypracuje zápis. Tieto zápisy tvoria súčasť dokumentácie“. Súčasťou dokumentácie by mali byť aj preberacie protokoly.

V súvislosti s vytvoreným prototypom bola doplnená kapitola 3 (Použitie systému) obsahujúca užívateľskú príručku. Kapitola obsahuje popis spôsobu práce s programom, popisy príkazov, poradie ich vykonania a popis užívateľom zadávaných informácií (vlastnosti objektov). Príručka obsahuje obrázky niekoľkých okien programu a príklady grafických výstupov. Bolo by vhodné, keby všetky objekty grafického rozhrania v oknách obsahovali nejaké charakteristické hodnoty (napr. obr. 5). Autori zaradili do príručky aj tabuľku so stručným prehľadom ponúk programu.

V kapitole 3.1 (Inštalácia a spustenie systému) by bolo vhodné uviesť:

- čo je to Java JRE
- či inštalácia programu obsahuje aj Java JRE alebo nie
- ak nie, kde sa dá získať
- informácie o konfigurácii Java JRE alebo odkaz na ne
- inštrukcie na spustenie programu, keďže nie je úplne triviálne

V kapitole 3.2 v odstavci Nastavenie by bolo vhodné uviesť doporučené rozsahy čísiel používaných portov.

Keďže vytváraný systém má architektúru klient-server je tiež potrebné uviesť, ktorá časť príručky sa týka časti klient, ktorá časti server a či sa jedná o dve samostatné aplikácie alebo nie.

### Prototyp

Členovia tímu 7 nám umožnili zhotoviť si kópiu vytvoreného prototypu, ktorý bol vo forme dvoch skomprimovaných súborov vo formáte ZIP. Prototyp sme spustili pomocou súboru „simulacia.bat“ zo súboru ALLEVELS.ZIP, keďže žiaden iný súbor neprichádzal do úvahy.

Prototyp je v anglickom jazyku, význam jednotlivých funkcií je uvedený v dokumentácii. Pri implementácii prototypu bola použitá knižnica JSIM.

Program pracuje v jednom okne s ponukou, ako celok je pomerne neprehľadný a používanie nie je v takej miere intuitívne ako by sa očakávalo. Príčinou môže byť nie úplne štandardné užívateľské rozhranie.

Ponuka „Edit“ nie je prítomná, jej funkčnosť nahrádza ponuka „Object“. Jednotlivé položky v ponukách nie sú vhodne pomenované, napr. „Create Department or Cashdesk“. Položky v ponukách nemajú klávesové skratky, program nemá nástrojové lišty.

Práca so simulovaným modelom na jednotlivých úrovniach je dosť neprehľadná; celkovú hierarchiu modelu nie je možné zobrazit'. Modifikuje sa práve zobrazená časť modelu, vykonanie rýchlych zmien na rôznych úrovniach rozsiahlejšieho modelu je zdĺhavé. Niektoré prvky dialógových okien (otváracie ponuky) sú nefunkčné alebo nereagujú správne (napr. tlačidlá „open“ a „closed“, ktoré musia vylučovať). Spôsob špecifikovania rozdelenia náhodnej premennej a jeho parametrov nie je zrejmy a zdá sa nedoriešený.

Pozitívne hodnotíme vytváranie objektov modelu a ich umiestňovanie na pracovnej ploche.

Otázka simulácie jedného modelu viacerými užívateľmi bola rozobraná v diskusii (pozri časť Prezentácia).

Vzhľadom na fakt, že hodnotený program je iba prototypom, je pochopiteľné, že nie všetky funkcie sú (a správne) implementované. Nemožno však prehliadnuť navrhnuté užívateľské rozhranie, ktoré by sa určite dalo vylepšiť.

## **Prezentácia**

Program prezentovali dvaja členovia tímu. Druhý člen tímu odpovedal na položené otázky už počas prezentácie, čím bola prezentácia plynulejšia. Úroveň prezentácie hodnotíme pozitívne; výklad bol vecný a zrozumiteľný.

Prezentované bolo užívateľské rozhranie, funkcie vytvorenia modelu, nastavenia simulácie, vlastností objektov. Neboli prezentované všetky funkcie uvedené v menu, časť klient nebola prezentovaná v HTML prehliadači.

Otázky v diskusii sa týkali komunikácie pripojených užívateľov, používania jedného simulovaného modelu viacerými užívateľmi a rozlišovania ich priorít.

Pri zodpovedaní niektorých otázok sa ukázali možné nedostatky v návrhu, ktoré sa môžu prejaviť na použiteľnosti výsledného systému; systém nerozlišuje oprávnenia užívateľov a neumožňuje ich komunikáciu.

## **Záver**

Spomínané nepresnosti v dokumentácii boli objasnené v rámci prezentácie a v diskusii. Bolo by však vhodnejšie, keby sme sa niektoré dôležité fakty dozvedeli už z dokumentácie. Doporučujeme preto doplniť dokumentáciu o diskutované otázky a prípadne sa zamyslieť nad odhalenými nedostatkami.

## Príloha G: Dopracovanie nedostatkov špecifikácie

Táto časť obsahuje doplňujúce informácie, ktoré riešia problémy a nedostatky identifikované zadávateľom a hodnotiacim tímom v odovzdanej špecifikácii.

- Na prvej strane ani v úvode dokumentu nie je explicitne uvedený význam dokumentu, ktorý nemusí byť zrejmý (keďže tento dokument bude súčasťou celkovej dokumentácie projektu). Dokument obsahuje špecifikáciu a hrubý návrh systému.
- Do časti 4.1 patria špecifické údaje o oddelení, ktoré sa čítajú zo vstupného súboru:  
názov oddelenia  
kapacita oddelenia  
Význam týchto parametrov je zrejmý z kapitoly 5.2.5.
- Slovenským ekvivalentom anglického slova „toolbar“ v časti 4.4 je „nástrojová lišta“, prípadne „panel nástrojov“.
- Význam položky v odstavci 4.5 „počet ľudí v závislosti od času“ znamená počet ľudí, ktorí navštívili oddelenie počas simulácie.
- Obrázok v časti 5.2.4 znázorňuje hierarchiu tovarov. Hierarchia zvyšuje prehľadnosť systému a umožňuje zadať nielen jednotlivé tovary, ktoré zákazník preferuje, ale aj celé skupiny. Pravdepodobnosť kúpy potom platí pre všetky tovary patriace do danej skupiny.
- V časti 6.1.1 sa vyskytol nekorektný krok, keď by sa mala okamžite po inicializácii vybrať udalosť z prázdneho kalendára. Tento nedostatok sa odstráni vygenerovaním prvotnej udalosti na začiatku simulácie a jej naplánovanie pred prvým výberom udalosti z kalendára. Aj keď by sa dalo uvažovať o grafickom vyjadrení algoritmu, považujeme jeho popis v prirodzenom jazyku za dostatočne presný a zrozumiteľný.

## Príloha H: Testovacie dáta - súbor s definíciou simulovaného modelu

```
root .begin
  mOpenTime 0
  mCloseTime 100
  mSettings .begin
    mCaseSensitive false
    mASCIIOOnly true
    mTimeBreakpoints false
    mWareBreakpoints false
    mEntityBreakpoints false
  .end
  mWareGroups .begin
    mName "Mliečne výrobky"
    mWareId 4608720
    mWare .begin
      mName Mlieko
      mPrice 15.000000
      mAdvertized false
      mWareId 4609200
    .end
    mWare .begin
      mName Maslo
      mPrice 30.000000
      mAdvertized false
      mWareId 4612384
    .end
    mWare .begin
      mName Jogurt
      mPrice 10.000000
      mAdvertized false
      mWareId 4615296
    .end
  .end
  mWareGroups .begin
    mName "Pekarenske výrobky"
    mWareId 4618528
    mWare .begin
      mName Chlieb
      mPrice 18.000000
      mAdvertized false
      mWareId 4619008
    .end
    mWare .begin
      mName Pecivo
      mPrice 1.500000
      mAdvertized false
      mWareId 4622064
    .end
    mWare .begin
      mName Bageta
      mPrice 7.000000
      mAdvertized false
      mWareId 4625232
    .end
  .end
  mWareGroups .begin
    mName Elektronika
    mWareId 4628272
    mWare .begin
      mName Televizor
      mPrice 18.000000
      mAdvertized false
      mWareId 4634160
    .end
    mWare .begin
      mName Veza
      mPrice 18.000000
      mAdvertized false
      mWareId 4637392
    .end
    mWare .begin
      mName Video
      mPrice 18.000000
      mAdvertized false
      mWareId 4640304
    .end
  .end
  .end
  mDepts .begin
    mName DEPT
    mCapacity 100
    mCounter .begin
      mServeGen .begin
        mMeth 3
        mPar1 0.000000
        mPar2 10.000000
      .end
      mBreakGen .begin
        mMeth 3
        mPar1 0.000000
        mPar2 10.000000
      .end
      mBreakLenGen .begin
        mMeth 3
        mPar1 0.000000
        mPar2 10.000000
      .end
      mCapacity 1
    .end
    mShelves .begin
      mFullCount 1
      mReloadLimit 1
      mWareId 4609200
      mReloadGen .begin
        mMeth 3
        mPar1 0.000000
        mPar2 10.000000
      .end
    .end
    mShelves .begin
      mFullCount 1
      mReloadLimit 1
      mWareId 4612384
      mReloadGen .begin
        mMeth 3
        mPar1 0.000000
        mPar2 10.000000
      .end
    .end
    mShelves .begin
      mFullCount 1
      mReloadLimit 1
      mWareId 4615296
      mReloadGen .begin
        mMeth 3
        mPar1 0.000000
        mPar2 10.000000
      .end
    .end
    mShelves .begin
      mFullCount 1
      mReloadLimit 1
      mWareId 4619008
      mReloadGen .begin
        mMeth 3
        mPar1 0.000000
        mPar2 10.000000
      .end
    .end
    mShelves .begin
      mFullCount 1
      mReloadLimit 1
      mWareId 4622064
      mReloadGen .begin
        mMeth 3
        mPar1 0.000000
        mPar2 10.000000
      .end
    .end
    mShelves .begin
      mFullCount 1
      mReloadLimit 1
      mWareId 4625232
      mReloadGen .begin
        mMeth 3
        mPar1 0.000000
        mPar2 10.000000
      .end
    .end
    mShelves .begin
      mFullCount 1
      mReloadLimit 1
      mWareId 4634160
      mReloadGen .begin
        mMeth 3
        mPar1 0.000000
        mPar2 10.000000
      .end
    .end
    mShelves .begin
      mFullCount 1
      mReloadLimit 1
      mWareId 4637392
      mReloadGen .begin
        mMeth 3
        mPar1 0.000000
        mPar2 10.000000
      .end
    .end
    mShelves .begin
      mFullCount 1
      mReloadLimit 1
      mWareId 4640304
      mReloadGen .begin
        mMeth 3
        mPar1 0.000000
        mPar2 10.000000
      .end
    .end
  .end
  .end
```



## **Príloha I: Údaje na testovanie bodov zastavenia**

Na testovanie oddelenia sme použili tieto podmienky:

Počet zákazníkov  $> 0$

- splnená od okamžiku príchodu prvého zákazníka

Percento odmietnutia  $> 100$

- nikdy nesplnená

Počet zákazníkov  $> 10$

- niekedy splnená

Na testovanie tovaru sme použili tieto podmienky:

Predaných kusov  $> 0$

- splnená od okamžiku predaja prvého kusu

Počet predaných kusov  $< 0$

- nikdy nesplnená

Počet predaných kusov  $> 0$  a zákazník je "Študent"

- splnená občas

Na testovanie obchodného domu sme použili tieto podmienky:

Simulačný čas  $> 0$

- splnená od začiatku simulácie

Simulačný čas  $< 0$

- nikdy nesplnená

Zmena času je  $\geq 10$

- splnená občas

## Príloha J: Dopracovanie nedostatkov užívateľskej príručky

Táto časť obsahuje doplňujúce informácie k odovzdanej užívateľskej príručke. Na nedostatky upozornili v posudku členovia hodnotiaceho tímu.

- Definícia termínu „dynamický stav modelu“: dynamický stav modelu predstavujú atribúty objektov, ktoré nie sú súčasťou atribútov špecifikovaných vo vstupnom súbore, konkrétne:
- Obchodný dom - počet zákazníkov, ktorí ho navštívili, suma minútých peňazí a celkový strávený čas a štatistiky týkajúce sa kúpených tovarov
- Oddelenie - počet zákazníkov, ktorí ho navštívili a počet neuspokojených zákazníkov
- Regál - počet zákazníkov, ktorí ho navštívili, počet neuspokojených zákazníkov, počet kusov tovaru v regále, štatistiky týkajúce skupín zákazníkov, ktorí z neho vyberali
- Pokladňa - suma peňazí v pokladni, počet obslužených zákazníkov, priemerná doba obsluhy a využitie.
- Význam a spôsob použitia regálov: Každý regál obsahuje práve jeden tovar. Regále sú umiestnené v jednotlivých oddeleniach. Pri poklese tovaru pod kritickú hodnotu sa naplánuje doplnenie na plný stav o určitý čas.
- Neúplné vysvetlenie použitia externých modulov: Externé moduly sú COM komponenty, ktoré slúžia na štatistický zber a zobrazovanie údajov získaných pri simulácii.
- V požiadavkách na systém chýba požiadavka na verziu knižnice MFC: Pri implementácii tímového projektu bude použitá knižnica MFC 4.21, ktorá je dodávaná v inštalácii Microsoft Visual C++ 5.0.
- Obsah ponúk aplikácie:

### **Súbor**

Nový  
Otvoriť  
Zatvoriť  
Uložiť  
Uložiť ako  
Uložiť dynamický stav  
Koniec

### **Okno**

Nové okno  
Kaskáda  
Rozložiť  
Usporiadať ikony  
Štandard  
Status bar  
Riadenie  
Model  
Body zastavenia  
Záznam o činnosti

### **Úpravy**

Späť  
Vyrezať  
Kopírovať  
Vložiť  
Vymazať  
Označiť všetko  
Vložiť nový objekt  
Hľadať  
Vlastnosti  
Nastavenie...

### **Pomoc**

O programe  
O externých moduloch  
Pomoc

### **Simulácia**

Spustiť  
Krokovať  
Zastaviť  
Ukončiť  
Zastaviť o...  
Zastaviť pri zmene...  
Zastaviť pri kúpe  
Externé moduly...