

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

## PROGRAM PRO NÁCVIK TECHNIKY RYCHLÉHO ČTENÍ

BAKALÁŘSKÁ PRÁCE

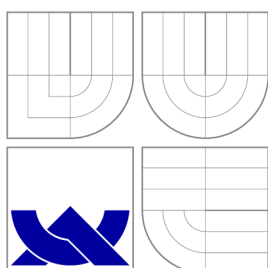
BACHELOR'S THESIS

AUTOR PRÁCE

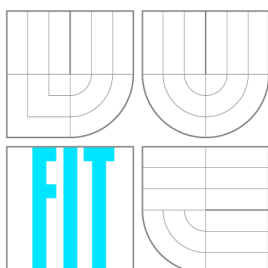
AUTHOR

MARTIN REPKA

BRNO 2010



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

# PROGRAM PRO NÁCVIK TECHNIKY RYCHLÉHO ČTENÍ

APPLICATION FOR PRACTICING RAPID READING TECHNIQUES

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

MARTIN REPKA

VEDOUCÍ PRÁCE  
SUPERVISOR

Ing. ADAM HEROUT, Ph.D.

BRNO 2010

## Abstrakt

Tato práce se zabývá návrhem a implementací programu na výuku rychlého čtení. Popisuje základní teorii, zejména samotný proces čtení a jednotlivé metody používané při výuce rychlého čtení. Dále se zabývá návrhem aplikace využívající popsané metody – rozšiřování zrakového rozpětí, snižování počtu fixací a měření čtenářského výkonu. Poté popisuje implementaci tohoto programu a výsledky testování na uživateli. Výsledkem je ucelená aplikace na výuku rychlého čtení, založená na metodách používaných zejména v příručkách a kursech, transformovaných do elektronické podoby.

## Abstract

This work deals with design and implementation of a program for rapid reading teaching. It describes fundamental theory, especially the very process of reading, and basic methods used for training of rapid reading. Then it deals with design of application which will be using described methods – eye span widening, reduction of eye fixations number and reader's performance measuring. Afterwards it describes program implementation and interprets user testing results. It results in a integrated application for rapid reading teaching, based on methods, used especially in handbooks and courses, transformed into electronic form.

## Klíčová slova

rychlé čtení, rychločtení, technika, výuka, GUI, oční fixace, zrakové rozpětí, čitatelský výkon, rychlost čtení

## Keywords

rapid reading, techniques, practicing, GUI, eye fixations, eye span widening, reader's performance, reading speed

## Citace

Martin Repka: Program pro nácvič techniky rychlého čtení, bakalářská práce, Brno, FIT VUT v Brně, 2010

# Program pro nácvik techniky rychlého čtení

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Adama Herouta, Ph.D. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....  
Martin Repka  
19. května 2010

## Poděkování

Rád by som poďakoval vedúcemu mojej práce, pánovi Ing. Adamovi Heroutovi, Ph.D. za to, že som mohol robiť na práci, ktorá ma bavila a najmä za jeho cenné rady a pripomienky, ktoré mi pri práci veľmi pomohli. Tiež by som chcel poďakovať svojej rodine a Janke, za prejavenu pomoc a podporu.

© Martin Repka, 2010.

*Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.*

# Obsah

<b>1 Úvod</b>	<b>3</b>
<b>2 Výučba rýchleho čítania</b>	<b>4</b>
2.1 Technika čítania	4
2.2 Proces zrýchľovania čítania	5
2.3 Didaktika rýchleho čítania	6
2.3.1 Rozširovanie zrakového rozpätia	6
2.3.2 Znižovanie počtu fixácií	8
2.4 Meranie čitateľského výkonu	9
2.5 Nástroje na nácvik techniky rýchleho čítania	11
2.5.1 Elektronické nástroje	11
2.5.2 Iné nástroje	12
<b>3 Návrh aplikácie</b>	<b>13</b>
3.1 Formulácia cieľa	13
3.2 Cvičenie na rozširovanie zrakového rozpätia ( <i>RZR</i> )	14
3.3 Cvičenie na znižovanie počtu fixácií ( <i>Fixácie</i> )	16
3.4 Meranie rýchlosti čítania a miery chápavosti ( <i>Testy</i> )	18
3.5 Ostatné súčasti programu	20
3.6 Intergrácia do výsledného programu	21
<b>4 Implementácia</b>	<b>23</b>
4.1 Vývojové nástroje	23
4.2 RZR	23
4.3 Fixácie	24
4.4 Testy	24
4.5 Ostatné súčasti programu	25
4.6 Integrácia do výsledného programu	26
<b>5 Testovanie</b>	<b>27</b>
5.1 Metodika testovania	27
5.2 Výsledky testovania	28
5.3 Záver testovania	30
<b>6 Záver</b>	<b>32</b>
<b>Literatúra</b>	<b>33</b>
<b>Zoznam príloh</b>	<b>34</b>

<b>A Obsah CD</b>	<b>35</b>
<b>B Dotazník</b>	<b>36</b>

# Kapitola 1

## Úvod

Súčasná doba je charakteristická veľkým množstvom prijímaných informácií, ktoré sme každodenne nútení spracovať. Či už sú to novinové články, odborné práce, e-maily, knihy alebo časopisy, objem textu je často nemalý a my potrebujeme efektívny spôsob, aby sme ho zvládli. Z tohto dôvodu vznikli a stále vznikajú metódy, ktoré nám majú pomôcť sa v tejto informačnej záplave zorientovať a informácie účinne spracovávať.

Jedným z takýchto spôsobov je metóda nazývaná *rýchle čítanie*. Jej prednosť spočíva v tom, že sa jedná o *plnohodnotné* čítanie, ktoré je rovnako dôsledné ako bežné čítanie, avšak realizované niekoľkonásobnou rýchlosťou. Tej je možné docieľiť zdokonalením očného aparátu, aby bol schopný jedným pohľadom zachytiť väčšiu časť textu. Práve týmto zdokonalením sa zaoberajú techniky na výučbu rýchleho čítania, ktorých elektronickú podobu sa pokúša spracovať táto práca. Jej cieľom je teda tieto techniky a metódy preskúmať a na základe nich vytvoriť program, pomocou ktorého by bolo možné svoje čitateľské schopnosti zdokonaľiť.

V úvode prvej kapitoly je vysvetlená technika čítania, pochopenie ktorej je dôležité na porozumenie samotného procesu zrýchľovania čítania. Jednotlivé časti tohto procesu sú popísané ďalej. Hlavná časť tejto kapitoly je však zameraná na opísanie hlavných metód, prostredníctvom ktorých je výučba rýchleho čítania realizovaná. Záver je venovaný stručnému prehľadu nástrojov na nácvik tejto techniky.

Druhá kapitola popisuje návrh aplikácie. Na začiatku vysvetľuje hlavný cieľ práce, prevažná časť kapitoly je však venovaná návrhu jednotlivých dielčích rozhraní, ktoré sú vo výslednom programe obsiahnuté. Koniec kapitoly sa zaoberá návrhom niektorých ďalších súčastí a integráciou do výslednej aplikácie.

Tretia kapitola objasňuje implementáciu programu. Je tu diskutovaný najmä spôsob riešenia niektorých netriviálnych problémov, ktoré pri procese tvorby vznikali. Ďalej sú tu stručne popísané niektoré triedy, ktoré boli pre účely tohto programu vytvorené.

Posledná, štvrtá kapitola je zameraná na testovanie aplikácie na užívateľoch. Táto časť najprv vysvetľuje metodiku testovania a následne analyzuje výsledky získané z dotazníkov, ktoré užívatelia v závere testovania vyplňali.

## Kapitola 2

# Výučba rýchleho čítania

### 2.1 Technika čítania

V tejto časti bude objasnený samotný proces čítania, aby bolo zrejmé, v čom je rozdiel medzi čitateľom využívajúcim tzv. „tradičné“ metódy čítania, od čitateľa využívajúceho techniky *rýchleho* čítania a aby bolo možné vytvoriť si obraz o tom, ktorým oblastiam je vhodné sa venovať, pre dosiahnutie efektívneho zvýšenia čitateľského výkonu (viac o čitateľskom výkone v kapitole 2.4).

Pre pochopenie procesu čítania nie je nutné byť detailne oboznámený s biologickou stavbou očného ústrojenstva, je však dobré si uvedomiť, že na očné svaly sa po svalstve srdcovom a dýchacom kladú najväčšie nároky [13]. Preto je veľmi dôležité, aby očný aparát nebol zbytočne zaťažovaný a aby sme sa snažili jeho činnosť rozumne usmerňovať. Použitie správnych čitateľských návykov smeruje práve k tomuto cieľu.

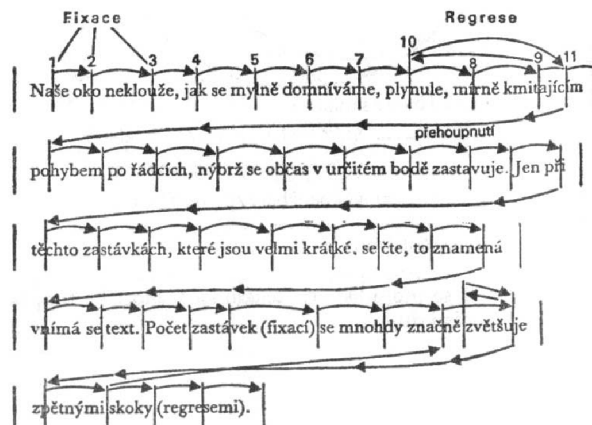
Vráťme sa ale späť k činnosti očí pri procese čítania. Rozšíreným názorom je, že oči pri čítaní po riadku prechádzajú plynule. Oko však, práve naopak, musí byť pri čítaní v pokoji, a po riadku skáče – po väčších či menších úsekoch, podľa toho, aký veľký úsek je schopné zachytiť na jeden pohľad. To znamená, že oko na riadku vykoná určitý počet tzv. *fixácií* a na každom fixačnom bode istý čas zotrúva, aby zachytilo, aké dáta sú v okolí tohto bodu uvedené. Tento proces je znázornený na obr. 2.1. Čím viac fixácií na riadku oko vykoná, tým viac je očný aparát namáhaný a tým nižšia je samotná rýchlosť čítania. Spomaľujúca a únavná je aj prítomnosť tzv. *regresii*, teda spätných pohybov na text, ktorý už sme prečítali.

To, že rýchlosť čítania je u každého človeka iná, má na svedomí viacero faktorov ako napr. úroveň vzdelania, s ňou čiastočne súvisiaci vek a celková sčítanosť čitateľa [13, 9]. Najmä spomenutá sčítanosť umožňuje aj neškolenému čitateľovi dosiahnuť relatívne vysokú rýchlosť čítania a to vďaka tomu, že si pri čítaní vytvárame *slovné obrazy*, podobne, ako sme si v prvej triede základnej školy vytvorili obrazy jednotlivých písmen. Čím väčšiu zásobu slovných obrazov máme vytvorenú, tým je čítanie plynulejšie. Bolo takisto zistené, že pre správne zachytenie slova je rozhodujúca jeho horná polovica [13] a že najdôležitejšie je jeho prvé a posledné písmeno, zvyšok môže byť často poprehadzovaný, príp. niektoré písmená vynechané [9]. Na ilustráciu tohto javu uveďme jeden populárny príklad [12]:

Na uiverznitě v Cbramidgi uladěli vzýukm a vlšyo njeavo, že náezleží na přaodí psmíen ve svloě. Džiúletě je, aby pnrví a peoldsní píemsna blya spvánřě.

Treba však poznamenať, že čítanie nemôže byť len mechanická záležitosť, ale že to musí byť proces aktívny [9]. To inými slovami znamená to, že pri čítaní je okrem iného nutné





Obrázok 2.1: Pohyby očí pri čítaní [13]

uvedomovať si, čo čítame a snažiť sa „jazyk autora“ transformovať do „vlastného jazyka“ – teda „nemôžno myslieť tými slovami, prostredníctvom ktorých sa text prijíma“ [9]. S vyššie uvedeným však súvisí aj nutnosť uvedomiť si, že každý druh literatúry vyžaduje osobitý prístup a tomu treba prispôbovať aj rýchlosť a štýl čítania<sup>1</sup>.

Vyvodíme na záver nedostatky tradičných metód čítania, teda časté zlé návyky neškoleného čitateľa. Okrem už uvedených, ako *veľký počet fixácií* (malé zrkové rozpätie) a *regresné pohyby*, je to často *zlá koncentrácia*, čítanie bez cieľa (tzv. *pasívne čítanie*), *artikulácia pri čítaní* (teda „vnútorná reč“, keď všetko, čo čítame, si v duchu alebo potichu hovoríme), príp. používanie napr. prsta či pera, ako *vodidla* v texte.

## 2.2 Proces zrýchľovania čítania

Na úvod je nutné zdôrazniť, že samotný proces zrýchľovania čítania by mal byť len dôležitou súčasťou zvyšovania čitateľského výkonu. Tou ďalšou je „zvyšovanie schopnosti chápania, čo je najmä výchova k pozornejšiemu mysleniu“ [13] a rovnako „odstránenie zlých návykov pri čítaní“ [10].

Čo je však zárukou toho, že proces čítania možno podstatne zrýchliť? W.Zielke v [13] uvádza, že „schopnosť myslenia niekoľkonásobne predstihuje rýchlosť čítania zaťaženu telesne, motorickými zábrami“, a zároveň, že školením možno tieto zábrany posúvať, a tak dosiahnuť minimálne dvoj- až trojnásobné zvýšenie čitateľského výkonu. Podľa [10] je tento proces možné urýchliť najmä vďaka tomu, že v textoch dochádza k redundancii, teda že mnoho slov v texte je doplnkových a možno ich preskočiť. Ďalej sú to existujúce značné rezervy mozgu a nakoniec existujúce nedostatky tradičných metód čítania (popísané v kap. 2.1).

Celý proces zrýchľovania čítania delí W.Zielke v [13] do siedmich bodov:

1. Vytvorenie optimálnych podmienok na čítanie a vylúčenie všetkého, čo by mohlo rozptyľovať našu pozornosť.
2. Preskúšanie pamäti a prípadné odstránenie zistených nedostatkov.

<sup>1</sup>Touto problematikou sa zaoberá tzv. *racionálne čítanie*, diskutované napr. v [7] kde je *rýchle čítanie* chápané len ako jeden z druhov *racionálneho čítania*.

3. Preskúšanie slovnej zásoby, nakoľko (ako už bolo uvedené) rýchlosť čítania závisí aj od rozsahu slovnej zásoby.
4. Samotné zvyšovanie rýchlosti čítania, ako funkcie dokonalejších čitateľských návykov.
5. Identifikácia a následné odstránenie závad a zlých čitateľských návykov pri čítaní.
6. Precvičovanie, a tým pádom zvyšovanie chápavosti (miery porozumenia textu), čím rastie celkový čitateľský výkon.
7. Rozširovanie zrakového rozpätia, teda systematické školenie zraku, ktoré má umožniť jedným pohľadom zachytiť širšiu časť textu.

Ešte raz zdôraznime dôležitosť predposledného bodu – precvičovanie a zvyšovanie chápavosti, pretože bez toho by sa rýchle čítanie mohlo veľmi rýchlo stať čítaním pasívnym, čo je presný opak cieľa o ktorý sa snažíme.

Táto práca, a teda najmä program, ktorý je jej výsledkom, sa zameriava až na posledný bod tohto procesu – rozširovanie zrakového rozpätia, pomocou techník, ktoré budú bližšie priblížené v kapitole 2.3. Tieto techniky však napriek tomu nie sú samoúčelné a majú rovnako za cieľ znížiť mieru niektorých zlých čitateľských návykov (napr. zvyšovaním koncentrácie [9]). Podľa môjho názoru však už samotné tréningy týchto zručností, abstrahovaných od ostatných častí procesu, môže rýchlosť čítania v nezanedbateľnej miere zvýšiť.

## 2.3 Didaktika rýchleho čítania

Ako už bolo spomenuté, táto práca sa zaoberá až samotnými technikami na zvyšovanie rýchlosti čítania, ktoré majú za cieľ rozšíriť zrakové rozpätie a znížiť s tým súvisiaci počet fixácií na riadok. Na dokázanie toho, aká účinná môže byť kombinácia týchto dvoch faktorov, uvádza W.Zielke v [13] zaujímavý príklad:

Veďmíme rýchlosť čítania 250 slov/min., ktorú má dosiahnuť zbehlý normálny čitateľ a položíme si otázku, či je možné toto číslo skutočne zdvojnásobiť. Veta ... obsahuje približne 10 slov na jeden riadok. 500 slov teda dosahuje približne 50 riadkov strojopisu. K prečítaniu týchto 50 riadkov je teda k dispozícii 1,2 sekundy na riadok. Ak sa jeden riadok zvládne šiestimi fixáciami, ... potom na každú fixáciu pripadá 0,2 sekundy času, tzn., že jedným pohľadom je nutné zachytiť 1,7 slova. Ak si vymedzíme na každý pohľad dvojnásobok času, t.j. 0,4 sekundy, museli by sme zakaždým zachytiť 3,3 slova, aby sme dosiahli rýchlosť 500 slov/min. Oboje je možné a dosiahnuteľné cvikom. Kombináciou 5 slov a tretiny sekundy pre vnem možno dokonca dosiahnuť ... 900 slov/min. (dokonalý čitateľ).

Podme už ale k priblíženiu samotných techník. Predtým je však ešte dôležité uviesť, že vysvetľované a potom aj aplikované metódy patria k odporúčaným vo všetkých knihách o rýchlom čítaní, ktoré sú v tejto práci citované, s miernymi odlišnosťami v samotnom prevedení a osobných preferenciách autorov.

### 2.3.1 Rozširovanie zrakového rozpätia

Rozširovanie zrakového rozpätia (ďalej *RZR*) je systematickým školením v periférnom videní. Cvičený čitateľ má totiž dva– až trikrát širšie zrakové rozpätie, než neškolený čitateľ

[13]. Školený čitateľ takto dokáže jedným pohľadom zachytiť 20–30 znakov v texte dávajúcom zmysel [7]. Toto cvičenie zároveň zlepšuje postreh, prispieva k lepšej koncentrácii, precvičuje pamäť a cibrí asociačné schopnosti [13]. Tréning je možné realizovať buď na skutočných slovách, alebo na tzv. *abstraktnom materiáli*. Väčšina autorov odporúča skôr druhú možnosť, nakoľko zhotovenie materiálu so skutočnými slovami je náročnejšie a takéto cvičenia sú menej účinné [13]. Abstraktným materiálom rozumieme (vzostupne, podľa rastúcej náročnosti):

1. skupiny číslíc,
2. skupiny malých písmen,
3. skupiny veľkých písmen,
4. skupiny číslíc a písmen.

Medzi samotnými znakmi v skupine môžu byť rôzne typy medzier (takisto vzostupne, podľa rastúcej náročnosti):

1. skupiny s rovnomernými medzerami, t.j. jednou medzerou za každým znakom (napr. 3 8 4, b q t a, r g 6 L 2),
2. skupiny bez medzier (napr. 2954, AQJPZ, Fk56L),
3. skupiny s nepravidelnými medzerami variabilnej šírky (napr. 89 0, X OP S, 3k 5 T1).

Náročnosť je samozrejme daná najmä počtom neprázdnych znakov<sup>2</sup>, ktoré sa v tejto skupine nachádzajú a tento počet sa postupne zvyšuje. V prípade varianty na papieri, je potom určitý počet skupín, tzv. *sada skupín* (podľa [13] je to 25 skupín) rovnakého typu (teda rovnakej náročnosti) usporiadaných pod seba, pričom pri každej skupine je miesto na zaznačenie odpovede (príp. podľa [7] na zaznačenie chyby). Časť cvičného materiálu, náročnosti *štyroch číslíc a písmen variabilnej šírky*, potom môže vyzeráť napríklad tak, ako je to znázornené na obr. 2.2. Na čo najväčšiu časť tejto sady potom priložíme napr. kartičku

41X 1	
n 4j t	
k eW I	
6 zIR	
z 3 Zd	
W ugL	
6 T aP	

Obrázok 2.2: Ukážka časti sady skupín na RZR

z kartónu tak, aby boli vrchné skupiny zakryté. Na kartičku zľahka položíme prsty ľavej ruky<sup>3</sup> a najvrchnejšiu skupinu potom prstami na malý moment ([13] udáva 100–200ms, [7] menej ako 0,3s.) odkryjeme a hneď zakryjeme tak, aby sme odkryli len jednu skupinu

<sup>2</sup>Preklad z angl. *non-whitespace character*, teda znak, ktorý je „tlačiteľný“ a nie je len horizontálnym alebo vertikálnym prázdny miestom.

<sup>3</sup>Resp. prsty nedominantnej ruky, u ľavákov to teda bude pravá ruka.

a aby sme ju zachytili len letným pohľadom (t.j. aby sme nemali čas urobiť viac než jednu očnú fixáciu) a túto skupinu sa potom snažíme zaznačiť na príslušnú čiaru. Potom skupinu znovu odkryjeme, porovnáme s tým, čo sme zaznamenali a poznačíme, či sme skupinu identifikovali správne, alebo nie. Takto prejdeme celú sadu cvičení, a v prípade, že chybu spravíme najviac trikrát (t.j. nesprávne identifikujeme najviac 3 skupiny z 25), môžeme prejsť na vyššiu náročnosť.

Podľa [13] stačí, keď dokážeme spoľahlivo prečítať skupiny šiestich znakov, aby sme dosiahli výrazného zlepšenia čitateľských schopností. Bolo totiž zistené, že pri čítaní materiálu, ktorý dáva zmysel, možno počítať s 3,5-násobnou šírkou zrkového rozpätia [13]. Keď teda budeme schopní spoľahlivo za zlomok sekundy zachytiť 6 znakov, budeme môcť naraz prečítať časť textu dĺžky až 20 znakov. Tohto zlepšenia je možné doceliť systematickým cvičením v priebehu 2 týždňov (1,5 hodiny päťkrát za týždeň), pričom začať by sme mali so skupinou troch číslíc s rovnomernými medzerami [13].

Cvičenie RZR je možné uskutočňovať aj na materiáli dávajúcom zmysel, avšak táto technika sa viacerým autorov javí ako menej účinná a navyše je zostavenie takýchto cvičení náročnejšie. R.Papík [10] ďalej odporúča pri cvičení využívať napríklad aj tzv. *fixačné pyramídy* (viď obr. 2.3), no iní autori toto cvičenie odporúčajú najmä na zistenie počiatočného, príp. aktuálneho stavu šírky zrkového rozpätia, nie na aktívne cvičenie [13, 7].

2	čo
4	bude
6	príliv
8	množstvo
10	informačný
12	racionálneho
14	najdôležitejší
16	charakteristické
18	vedeckotechnických
20	zaujímavého problému

Obrázok 2.3: Fixačná pyramída

### 2.3.2 Znižovanie počtu fixácií

Na začiatok je nutné si uvedomiť, že rozšírením zrkového rozpätia neznížime automaticky počet fixácií napriek tomu, že by sa to tak mohlo javiť. Tréningom RZR však len získame určitú schopnosť a výhodu, ktorú musíme ďalej preniesť aj do praktického čítania. K zníženiu počtu fixácií preto treba pristupovať rovnako systematicky ako k RZR. D.Gruber v [7] uvádza, že keď si osvojíme pomocou RZR schopnosť na jednu fixáciu zachytiť 20 znakov dávajúcich zmysel, sme schopní za 4 dni, tréningom počas niekoľkých desiatok minút denne, naučiť sa čítať riadok približne tromi fixáciami. Treba ešte pripomenúť, že aj pri schopnosti zachytiť jednou fixáciou 20 znakov je nemožné, a hlavne je chybou, čítať všetko na tri fixácie. Fixácie by mali stále rešpektovať hranice slov, resp. ešte lepšie logických celkov, teda častí viet [7].

Čo sa týka samotného postupu pri znížovaní počtu fixácií, je k nemu možné prisupovať dvomi základnými spôsobmi.

Prvým spôsobom je postupné znižovanie fixácií, keď počet fixácií znižujeme od stavu, v ktorom sa na začiatku nachádzame, k stavu, kde sa chceme dostať (tri fixácie na riadok o šírke približne 60 znakov).

Druhým spôsobom je nastavenie východzej pozície na žiadaný stav hneď od začiatku, teda tréning rovno troch fixácií na riadok. V tomto prípade sa v priebehu cvičenia rozširuje každý stĺpec, t.j. každé fixačné miesto o jeden znak, až kým sa riadok nezaplní. Tento spôsob ilustruje obr. 2.4. Táto varianta je po prekonaní niektorých počiatočných problémov veľmi dobre použiteľná [13]. Treba však pripomenúť, že napriek tomu, že sa toto cvičenie realizuje na skutočných slovách, stále precvičuje najmä mechanický úkon zachytenia časti textu jedným pohľadom bez nároku na myslenie. Až po zvládnutí samotnej techniky je možné – vhodné, prejsť na normálne texty dávajúce zmysel a zachytávať aktívne ich obsah.

2.	čo	na	od
3.	sám	pre	kto
7.	postava	problém	rytmika
12.	možno neskôr	neprijateľný	rozširovanie
20.	Najdôležitejšie býva pochopenie poslednej časti tohto cvičenia.		

Obrázok 2.4: Postup pri znížení počtu fixácií z počiatočného stavu na tri

Bolo vymyslených taktiež niekoľko opatrení, hlavne čo sa týka vyznačovania fixácií na riadku, ktoré mali toto cvičenie uľahčiť. Tieto opatrenia podľa [13] spočívali napr. v tom, že sa medzi fixáciami striedal typ písma (napr. normálne – kurzíva), jednotlivé fixácie sa vyznačovali bodkami nad alebo pod písmenami, príp. sa fixácie označovali čiarami, rámkami a pod. Pri týchto opatreniach však podľa autora prevažujú nevýhody, preto sa pri cvičeniach bežne nepoužívajú. Taktiež autor tejto práce zastáva názor, že tieto prvky môžu byť dosť rušivé už len tým, že svojou prítomnosťou ešte viac cvičenia odďaľujú bežnému čítaniu, keďže čitateľ nezanedbateľnú časť svojej pozornosti, či už vedome alebo nevedome, venuje práve týmto vedľajším javom.

## 2.4 Meranie čitateľského výkonu

V predchádzajúcom texte boli už niekoľkokrát uvádzané termíny *rýchlosť čítania*, *miera chápanosti*, či *čitateľský výkon*. Poďme si preto tieto pojmy jasnejšie definovať, aby sme následne mohli lepšie porozumieť samotným testom, ktoré tieto veličiny merajú.

*Rýchlosť čítania* je objem prečítaného textu za časovú jednotku [10]. *Objem* môže byť uvedený napr. v počte slov, písmen, či strán, zatiaľ čo *časová jednotka* býva najčastejšie minúta, sekunda alebo hodina. Zvyčajne sa však stretávame s vyjadrením tejto veličiny v *slovách za minútu*. V tomto prípade sa teda rýchlosť vypočíta podľa vzťahu (2.1).

$$\text{rýchlosť [slová/min]} = \frac{\text{počet slov} \cdot 60}{\text{doba čítania [s]}} \quad (2.1)$$

*Miera chápanosti* vyjadruje, s akou percentuálnou úspešnosťou bol text pochopený. Dôležité však je, že sa oveľa viac hodnotí pochopenie hlavnej myšlienky textu, hlavných bodov a detailov, ktoré sú pre zachytenie významu kľúčové, než znalosť vedľajších myšlienok a menej dôležitých častí textu [7]. Vypočíta sa z podielu správnych odpovedí na celkovom počte

otázok, ktoré sú k testovaciemu textu pripojené. Je však potrebné zdôrazniť, že pri miere chápanosti úplne neplatí priama úmera a za vhodnú sa v praxi považuje hodnota okolo 70 % [10].

*Čitateľský výkon* je veličina, ktorá dáva do pomeru *rýchlosť čítania* a *mieru chápanosti*. Má teda vyjadrovať rýchlosť, znalosť prečítaného textu a v tejto znalosti uprednostňovať prvoradá pred podružným [7]. Čo sa týka posledného bodu, musíme priznať že je subjektívny. Na testovanie čitateľského výkonu existuje niekoľko metód, no žiadna nie je úplne objektívna [10]. Avšak nejakým spôsobom testovať čitateľský výkon je potrebné. Pokiaľ už poznáme hodnotu rýchlosti čítania a miery chápanosti, odvodíme hodnotu čitateľského výkonu zo vzťahu (2.2) [13].

$$výkon \text{ (skalár)} = \frac{\text{rýchlosť [slová/min]} \cdot \text{výkon [\%]}}{100} \quad (2.2)$$

V priebehu cvičení na rozvoj rýchleho čítania je veľmi vhodné priebežne merať pokrok práve na *testoch čitateľského výkonu*. Takýto test pozostáva z textovej časti, po prečítaní ktorej má čitateľ zodpovedať otázky k tomuto textu. Počas čítania je čitateľovi meraný čas, z ktorého sa následne, podľa počtu slov v teste, vypočíta rýchlosť. Potom sú vyhodnotené odpovede na otázky, na základe čoho sa vypočíta miera chápanosti a následne celkový čitateľský výkon. Aby mal takýto test zmysel, musí byť, pochopiteľne, zostavený treťou osobou.

Pri zostavovaní testu je vhodné použiť text z bežných informačných prameňov, teda maximálne stredne náročné texty, ktoré sú na rýchle čítanie všeobecne najlepšie [10]. Čo sa týka otázok, naskytujú sa tu dve hlavné možnosti [7], ktoré však možno vzájomne kombinovať.

Prvou variantou je použitie otázok, kde čitateľ svoju odpoveď *tvorí* (angl. *creation*). Môžu to byť napr. otázky na uvedenie určitej informácie, hlavnej myšlienky, či kľúčových slov, ktoré by s textom mohli byť spojené. Hodnotenie odpovedí na tento typ otázok môže byť o niečo miernejšie, keďže tzv. *správnych* odpovedí môže byť niekoľko.

Druhou variantou sú otázky s *výberom z možností* (angl. *multiple choice*). Pri tejto variante dostane čitateľ k otázke niekoľko možností odpovede, z ktorých je jedna *najsprávnejšia* a podľa toho sa hodnotí správnosť čitateľovho výberu.

Obidve tieto varianty majú svoje výhody aj nevýhody, preto sa nedá jednoznačne povedať, ktorá je objektívne lepšia. V kurzoch a tlačných testoch sa však obvykle používa skôr prvá alternatíva [7].

Absolvovanie takýchto testov má dvojitý účinok. Jednak merajú aktuálny pokrok čitateľa a rovnako umožňujú prakticky si vyskúšať nadobudnuté zručnosti. Čo sa týka merania pokroku, je potrebné zdôrazniť, že na to, aby mohli odrážať realitu, je nutné k nim pristupovať ako k *meraniu* a nesnažiť sa kľúčovo dokázať si, že už sme urobili nejaký pokrok. To isté sa týka aj odpovedania na otázky k textu – nie je vhodné hádať a vymýšľať si odpovede, pokiaľ si skutočne nemyslíme, že odpoveď poznáme. Podľa [13] je takisto dôležité, aby čitateľ k testu vždy pristupoval s nejakým cieľom (napr. čítať riadok na tri fixácie).

Otázkou ďalej je, ako často takéto testy absolvovať. Je odporúčané na začiatku výcviku rýchleho čítania urobiť si takýto vstupný test na získanie východzieho stavu a potom absolvovať testy (pochopiteľne zakaždým iný) vždy po intenzívnejšom nácviku (zhruba 3–4 hodiny [10]). Výsledky testov môžu v priebehu tréningu kolísať, čo je normálny stav, rov-

nako ako mierne vnútorné napätie, či pociťované nedostatočné privyknutie na nové návyky [10].

## 2.5 Nástroje na nácvik techniky rýchleho čítania

### 2.5.1 Elektronické nástroje

Existujúcich kvalitných elektronických nástrojov na výučbu rýchleho čítania nie je veľa. Navyše obvykle býva ich cena pomerne vysoká a často býva problém ich vôbec otestovať bez toho, aby bolo nutné najprv za ne zaplatiť (aj keď s prípadnou garanciou vrátenia peňazí). Na americkom trhu je v ostatných rokoch populárny a oceňovaný napr. program *The Reader's Edge* [11]. Tento program ponúka prepracované prostredie na výučbu rýchleho čítania, zahŕňajúce nástroje na rozširovanie horizontálneho aj vertikálneho zrkového rozpätia, cvičenia na skupinách slov, testy porozumenia a mnohé ďalšie nástroje. Bohužiaľ nie je možné plnohodnotne ho vyskúšať, k dispozícii je len interaktívna ukážka na internete, ktorá je znázornená na obrázku 2.5.

Zaujímavým programom je tiež napr. program *AceReader* [4], ktorý ponúka viaceré nástroje na rozširovanie zrkového rozpätia, znižovanie počtu fixácií a rovnako poskytuje nástroj na meranie čitateľského výkonu. Ponúkané cvičenia sú však niekedy pomerne zdĺhavé a podľa môjho názoru často odpútavajú pozornosť nesprávnym smerom.

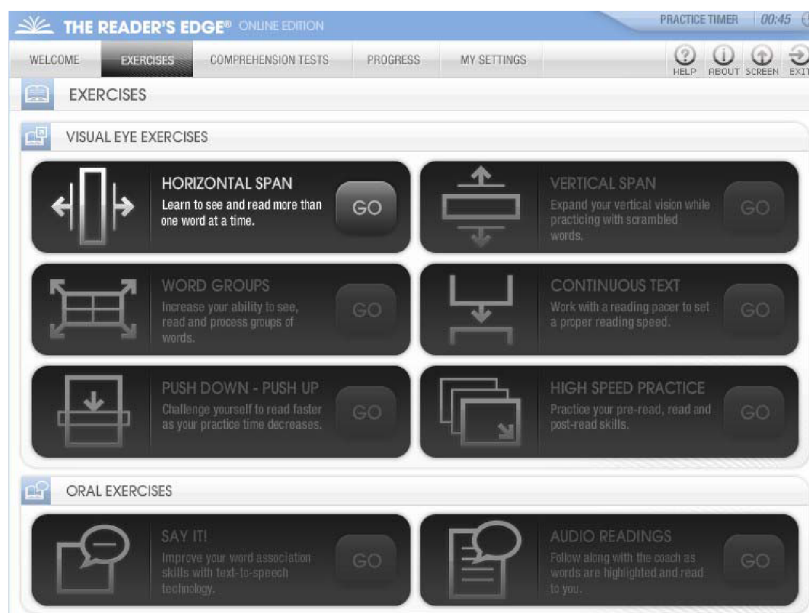
Z ďalších programov na výučbu rýchleho čítania možno spomenúť napr. *Speed Reader-X* [3], či *RapidReader* [2]. Týmto programom ale nie je potrebné venovať viac pozornosti, keďže v zásade obsahujú podobné cvičenia, ako vyššie uvedené aplikácie.

Zahraničné nástroje však majú pre nás značnú nevýhodu v tom, že sú postavené na iných jazykoch (zväčša angličtine), t.j. aj na inej slovnej zásobe, čo môže byť vo výučbe rýchleho čítania značne obmedzujúce, pokiaľ nemáme tento cudzí jazyk zvládnutý na dostatočne vysokej úrovni.

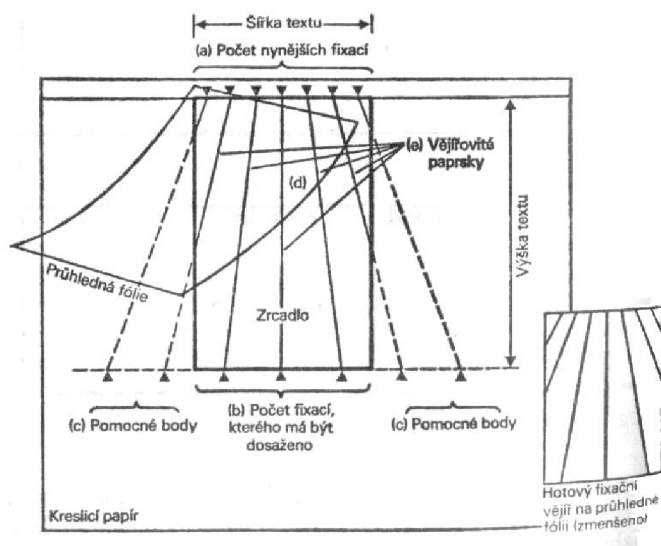
Na česko-slovenskom trhu je ešte ťažšie nájsť program, ktorý by sa tejto oblasti venoval. Zmienku o programoch na rozširovanie zrkového rozpätia *RZR Win* a *Pyramida* je možné nájsť na stránkach D.Grubera [5], kde je možné si ich aj zakúpiť, no je možné získať ani demoverziu týchto programov a k dispozícii nie sú ani ukážky napr. vo forme tzv. *screenshotov*, preto nie je možné posúdiť, na akej úrovni sa tieto programy nachádzajú.

### 2.5.2 Iné nástroje

Naučiť sa rýchlo čítať je však možné aj inak ako pomocou počítačových programov. Medzi metódy odporúčané napr. [13] patrí cvičenie na rozširovanie zrkového rozpätia (vid. kap. 2.3.1 na str. 6) metódou tzv. *bleskovej kartičky*, keď sú skupiny písmen zakrývané a odkrývané rukou pomocou kartónovej kartičky. Na znižovanie počtu fixácií odporúča [13] napr. používanie *fixačného vejára*, čo je priehľadná fólia s vyznačenými fixačnými čiarami, ktorá sa položí priamo na text a číta sa cez ňu. Zhotovenie takejto pomôcky ilustruje obrázok 2.6.



Obrázok 2.5: Ukážka prostredia programu The Reader's Edge [11]



Obrázok 2.6: Zhotovenie fixačného vejára [13]



## Kapitola 3

# Návrh aplikácie

### 3.1 Formulácia cieľa

Cieľom tejto práce je vytvoriť aplikáciu podľa zadania, ktorá by bola užívateľsky príjemná, jednoduchá, no zároveň dostatočne komplexná na to, aby poskytovala ucelené prostredie, ktoré by bolo plnohodnotným nástrojom na výučbu rýchleho čítania. Dôraz by mal byť, odhliadnúc od zamerania aplikácie, rovnako kladený na to, aby bol program vytvorený podľa súčasných trendov v grafických užívateľských rozhraniach, prípadne prinášal nové prvky, ktoré však budú pre užívateľa znamenať zjednodušenie práce a nebudú naopak znamenať skomplikovanie, prípadne zneprehľadnenie ovládania. Výsledný produkt by sa mal však okrem súčasných trendov pridržovať aj niektorých základných, všeobecne známych princípov tvorby grafických užívateľských rozhraní ([8, 1]), a to najmä:

1. *Jednoduchosť* – poskytnúť čo najjednoduchšie užívateľské rozhranie, s predvolenými možnosťami (bez nutnosti všetko nastavovať). Spraviť bežné úkony čo najjednoduchšími na úkor toho, že menej bežné budú zložitejšie. Poskytnúť jednotnosť a konzistentnosť.
2. *Jasnosť* – rozhranie by malo byť vizuálne, koncepčne a lingvisticky jasné a čisté, zahŕňajúc vizuálne prvky, funkcie, metafory a text.
3. *Prehľadnosť, zrozumiteľnosť* – systém by mal byť jednoduchý na pochopenie a naučenie. Užívateľ by mal vedieť: *čo robiť, kam sa pozeráť, kedy to urobiť, kde to urobiť, prečo to urobiť a ako to urobiť.*
4. *Intuitívnosť* – snažiť sa, aby mal užívateľ pred sebou všetko, čo potrebuje a zároveň nebola jeho pozornosť odpútavaná nežiadúcim smerom. Najbežnejšie veci by mali byť pre užívateľa čo najintuitívnejšie.
5. *Efektívnosť* – minimalizovať pohyby očí, rúk a ďalšie kontrolné úkony. Prechody medzi rôznymi prvkami systému by mali byť pozvoľné. Pohyby očí po obrazovke by mali byť prirodzené a postupné. Čo najviac predvídať užívateľove potreby všade, kde je to možné.
6. *Konzistentnosť* – systém by mal vyzeráť, správať sa a ovládať sa jednotným spôsobom. Podobné komponenty by mali mať podobný vzhľad, podobné použitie, mali by podobne fungovať, rovnakou akciou by mal byť vždy docielený rovnaký výsledok a ich funkcia by sa nemala meniť. Umiestnenie štandardných prvkov by sa takisto nemalo meniť.

7. *Konfigurovateľnosť* – poskytnúť jednoduché možnosti personalizácie, konfigurácie a rekonfigurácie nastavení. Podporovať užívateľa v aktívnom spoznávaní programu.

Keďže pre väčšinu užívateľov bude kontakt s týmto programom prvým krokom do sveta rýchleho čítania, je takisto veľmi dôležité, aby im bola poskytnutá dostatočná podpora a nápoveda pri tejto činnosti, aby jednoducho pochopili zmysel a formu cvičení, jednotlivých nastavení a podobne. Je takisto rozumné vymedziť cieľovú skupinu užívateľov, ktorí budú s programom prichádzať do styku najčastejšie. Napriek tomu, že na zefektívňovanie čítania nie je nikdy neskoro, najjednoduchšie je to približne od 18. do 35. roku života [10], čo je vek, kedy sú ľudia na nové veci a metódy adaptibilnejší.

Čo sa týka metód výuky, tie hlavné a najčastejšie používané už boli popísané v kapitole 2. Vychádzajme preto z toho, že sú overené a bolo by teda vhodné ich do programu zahrnúť. Program tým pádom bude obsahovať:

- cvičenie na rozširovanie zrkového rozpätia,
- cvičenie na znižovanie počtu fixácií,
- nástroj, ktorým bude možné zmerať užívateľov pokrok, vo forme testov s otázkami.

Tieto cvičenia by však nemali byť samoúčelné. Ich spoločným cieľom by mala byť aj čo najväčšia flexibilita voči užívateľovým potrebám. Užívateľ, zaoberajúci sa povedzme ekonomickou oblasťou, by teda mal dostať možnosť trénovať sa aj v slovnej zásobe, s ktorou prichádza do styku najčastejšie, či trénovať svoju rýchlosť čítania a chápanosť na ekonomických testoch.

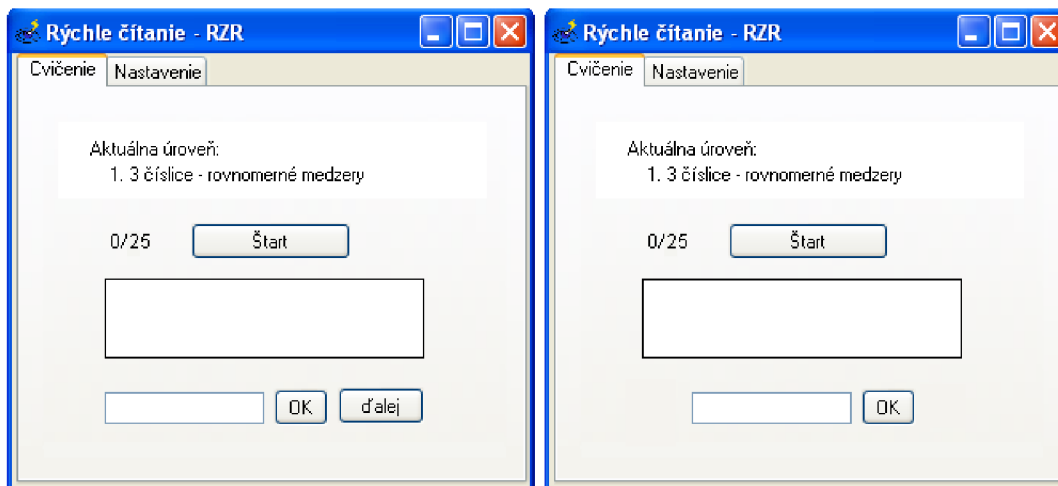
### 3.2 Cvičenie na rozširovanie zrkového rozpätia (*RZR*)

Cvičenie na RZR budeme podľa odporúčaní viacerých autoroch realizovať na abstraktnom materiáli. Keďže cvičenie v elektronickej forme sa bude líšiť od cvičenia na papieri, určíme si najprv formu, akou bude realizované. V elektronickej podobe nie je nutné, aby bola generovaná celá sada skupín naraz, skupiny sa môžu generovať a zobrazovať postupne. Postupnosť cvičenia však zostáva zachovaná. Najprv sa teda na obrazovke na okamih zobrazí skupina slov, ktorá v zápätí zmizne a užívateľ má túto skupinu správne zapísať a potvrdiť. Vyhodnotenie správnosti odpovede realizuje program, ktorý by mal vhodným spôsobom užívateľovi signalizovať, či jeho odpoveď bola správna a po absolvovaní všetkých skupín odporučiť prechod do vyššej úrovne, resp. zotrvanie v aktuálnej úrovni.

Z pohľadu grafického užívateľského rozhrania (ďalej *GUI*) tieto požiadavky znamenajú pole, kde sa skupina zobrazí, ďalej pole, kde má užívateľ zaznačiť svoju odpoveď a tlačidlo/tlačidlá na potvrdenie odpovede a vyvolanie ďalšej skupiny. Okrem toho by bolo vhodné zahrnúť do GUI aj tlačidlo na zahájenie celého cvičenia, informáciu o aktuálnej náročnosti cvičenia a prípadne informáciu v priebehu cvičenia o tom, v akom štádiu rámci celého cvičenia sa užívateľ práve nachádza, teda koľko skupín z celkového počtu už má za sebou.

Súčasťou GUI by mala byť aj možnosť základných nastavení cvičenia, ako nastavenie úrovne, rýchlosti zobrazovania, či ďalšie nastavenia. V rámci zachovania pravidla jednoduchosti by však mali byť nastavenia a samotné cvičenie oddelené.

Všetky tieto požiadavky boli spojené, a na základe nich boli vytvorené dva mierne odlišné koncepty základného GUI (obrázok 3.1). Obidve varianty tvorí základný formulár



Obrázok 3.1: Návrhy GUI cvičenia RZR

so záložkami, čo je spôsob, ktorý je v súčasnosti značne rozšírený a užívateľsky dostatočne intuitívny. Uvedené návrhy sa odlišujú len (ne)prítomnosťou tlačidla **ďalej**, no pre užívateľa to predstavuje dosť podstatný rozdiel v ovládaní cvičenia. V prípade absencie tohto tlačidla sa ďalšia skupina zobrazí automaticky po určitej dobe od stlačenia tlačidla **OK** a indikácie (ne)správnosti odpovede. Naopak, v prípade jeho prítomnosti užívateľ zobrazenie ďalšej skupiny vyvolá sám, práve jeho stlačením. Prvá varianta pre užívateľa znamená menšiu kontrolu nad priebehom cvičenia, výmenou za komfort v tom zmysle, že nemusí všetko potvrdzovať sám. Tieto dve možnosti boli testované na vybranej vzorke užívateľov a väčšina sa priklonila k variante s tlačidlom naviac, čo bolo nakoniec zvolené za výsledné riešenie.

Aby bolo ovládanie cvičenia čo najjednoduchšie, bude okrem možnosti stláčania tlačidiel pomocou myši implementovaná aj možnosť ovládania pomocou klávesnice, a to stláčaním klávesy **Enter** pomocou automatického prenášania zaostrenia tlačidiel (tzv. *focus-u*), takže ho nebude nutné prenášať manuálne stlačením klávesy **Tab**. Tento spôsob ovládania bude užívateľovi doporučený aj v nápovede k cvičeniu, ktorú popíšeme neskôr.

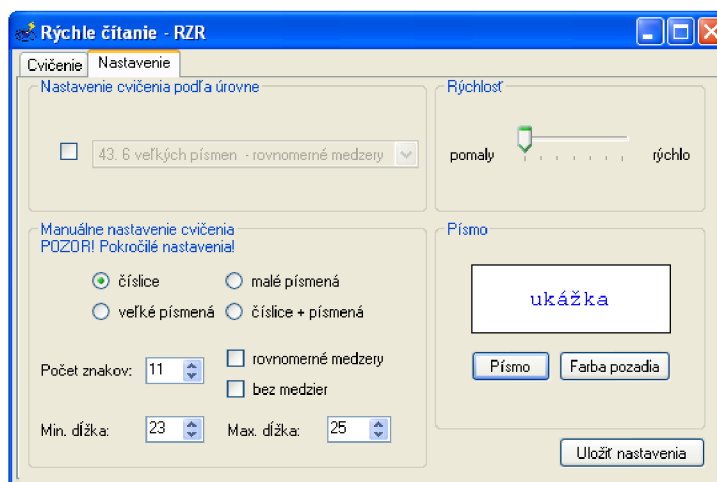
V priebehu cvičenia by mal byť užívateľ informovaný o tom, kedy skupinu rozpoznal správne a kedy nesprávne. Táto informácia je užitočnou spätnou väzbou a zároveň môže byť motiváciou na napredovanie ďalej. Ako najlepšie riešenie sa ukázalo farebné zvýraznenie (podfarbenie pola na odpoveď) zelenou, resp. červenou farbou ak je zadaná odpoveď správna, resp. nesprávna. Možnosťou takisto bolo, po zadaní nesprávnej odpovede ukázať správnej varianty, no ukázalo sa, že to užívateľovu pozornosť odpútava a môže to byť rušivým faktorom.

Ďalšou časťou GUI cvičenia RZR je záložka s nastaveniami, kde má užívateľ možnosť si cvičenie prispôbiť. Tu je potrebné nájsť strednú cestu medzi tým, aby užívateľ mal dostatok voľnosti na nastavenie všetkého potrebného, no nebol na druhej strane obťažovaný zbytočne zložitými a podrobnými nastaveniami, ktoré s najväčšou pravdepodobnosťou nikdy nevyužije. Užitočné nastavenia by teda mohli byť:

- nastavenie úrovne základného tréningu (úrovne sú popísané v [13]),
- nastavenie rýchlosti zobrazovania, teda doby, na ktorú sa skupina na obrazovke zobrazí,

- pre pokročilých užívateľov by bola vhodná možnosť nastaviť si cvičenie manuálne (najmä pre tých, ktorí sa aj po základnom tréningu chcú ďalej zdokonaľovať),
- nastavenie veľkosti, typu a farby písma, pre prispôsobenie užívateľským potrebám.

Grafický návrh záložky s nastaveniami je znázornený na obr. 3.2. Pri návrhu bola pozornosť kladená aj na to, aby nastavenia, s ktorými sa bude manipulovať najčastejšie boli na začiatku (vyššie), kým tie pokročilejšie sú uvedené až neskôr.



Obrázok 3.2: Návrh GUI nastavení cvičenia RZR

Zhrňme teda na záver, aký bude priebeh cvičenia RZR:

1. Po spustení programu užívateľ stlačí klávesu **Enter** (tým vyvolá stlačenie tlačidla **Štart**), čím zahájí cvičenie. Zobrazí sa prvá skupina znakov.
2. Túto skupinu zaznamená do pola na odpoveď a znovu stlačí **Enter** (stlačenie tlačidla **OK**). Pole s odpoveďou sa zafarbí podľa správnosti odpovede na zeleno, príp. červeno.
3. Keď je užívateľ pripravený pokračovať, stlačí opäť klávesu **Enter** (ekvivalent stlačenia tlačidla **ďalej**). Zobrazí sa ďalšia skupina znakov.
4. Body 2. a 3. sa opakujú celkom 25krát. Poradie aktuálne absolvovanej skupiny z celkového počtu sa zobrazuje naľavo od tlačidla **Štart**.
5. Po absolvovaní celého cvičenia sa vyhodnotí počet nesprávne zaznamenaných skupín a v prípade, že je nižší ako 4 (podľa [13]), je pomocou oznamovacieho okna (angl. *message box*) odporučený prechod do vyššej úrovne.

### 3.3 Cvičenie na znižovanie počtu fixácií (*Fixácie*)

Cvičenie na znižovanie počtu fixácií (ďalej *Fixácie*) má za cieľ trénovať zručnosti získané v RZR na skutočných slovách a priblížiť tak tréning rýchleho čítania normálnemu čítaniu. Treba však pripomenúť, že napriek tomu, že je cvičenie realizované na skutočných slovách, netvoria tieto slová zmysluplný celok a stále slúžia len na mechanické tréningovanie zraku.

Počet fixácií na riadku možno znižovať buď postupne, alebo naraz (viac v kap. 2.3.2). Znížiť počet fixácií naraz je problematické najmä pri cvičení „na papieri“, keďže príprava materiálu môžu byť pomerne náročná. Tento problém však v elektronickej verzii tohto cvičenia zaniká, a preto sa budeme držať tejto varianty, ktorú okrem iných odporúča aj [13].

Fixácie by sa mali ideálne trénovať súbežne s RZR, aby bol pokrok postupný a tieto metódy boli priebežne prevádzané aj do bežného čítania. Pomocou RZR rozšírime zrkové rozpätia na približne 20 znakov zmysluplného textu, čo je ekvivalentom jednej tretiny riadku normostrany. Preto budeme ďalej vychádzať z toho, že cieľom sú tri fixácie na riadok textu. Pri zmenenej šírke strany sa však pochopiteľne mení aj toto číslo, no jedným pohľadom by sme mali ku koncu tréningu zachytiť vždy zhruba 20 znakov zmysluplného textu. Toto číslo však samozrejme bude na začiatku podstatne nižšie a slová sa budú predlžovať postupne.

Aká by však mala byť forma tohto cvičenia? Jedná sa v podstate už o čítanie textu, preto by sme mohli vychádzať z textového pola, ktoré bude rozdelené do stĺpcov podľa počtu fixácií. Užívateľ bude týmto cvičením postupne prechádzať, pričom každým pohľadom zachytí práve jednu skupinu písmen, resp. slov (ďalej len *skupina*). Jednotlivé fixácie by mali byť užívateľovi nejako naznačované, keďže sa pravdepodobne bude jednať o začiatok, ktorý takýmto cvičením prichádza do styku po prvýkrát. Mojm pôvodným zámerom bolo toto naznačovanie spraviť len formou zvukového upozornenia – metronómu, aby si užívateľ novú rýchlosť doslova „zafixoval“, no toto riešenie sa ukázalo ako nedostačujúce, užívatelia nechápali zmysel cvičenia, preto bol do cvičenia pridaný ďalší vyznačovací prostriedok – zvyrazňovanie skupiny zmenou farby.

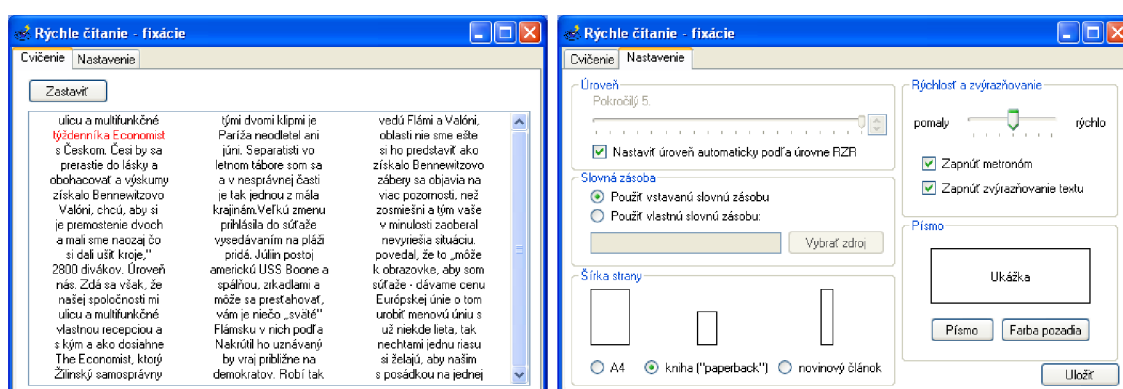
Cvičenie by sa v rámci jednotnosti potom malo zahájiť, ako aj ukončiť, stlačením tlačidla. V tomto cvičení však nie je celkom jednoznačný spôsob, akým sa má vyhodnocovať úspešnosť. Spôsob akým to robia niektoré podobne zamerané programy je, že testujú správnosť odpovede po každej zobrazenej skupine a to väčšinou výberom z viacerých možností. Tento spôsob však osobne nepovažujem za príliš efektívny, keďže z vlastnej skúsenosti viem, že som bol často schopný vybrať správnu odpoveď, aj keď som nezachytil pohľadom celú skupinu. Okrem toho, tento spôsob je dosť zdĺhavý a cvičenie vôbec nepribližuje skutočnému čítaniu. Za lepšiu variantu považujem nechať vyhodnotenie na užívateľa, ktorý sám najlepšie vie, či je schopný danú úroveň bez problémov zvládať, alebo ju ešte potrebuje precvičovať. Vďaka tomu sa cvičenie viac približuje normálnemu čítaniu, je realizované plynule, pričom pokiaľ užívateľ už na začiatku cíti, že cvičenie je príliš ľahké/ťažké, môže ho zastaviť a úroveň zmeniť.

Otázkou v tejto časti je aj samotná príprava cvičení, teda ich obsah. Narozdiel od RZR sa nejedná o náhodne generované zhluky znakov, ale o skutočné slová, preto je nutné použiť skutočnú slovnú zásobu, resp. skutočné texty. Na tento účel dobre vyhovujú napr. noviny, príp. vedecko-populárne články. Je však vhodné zamyslieť sa aj nad individuálnymi potrebami užívateľov, preto by bolo dobré zohľadniť aj fakt, že niektorí môžu mať záujem trénovať sa na slovnej zásobe, s ktorou prichádzajú do styku najčastejšie a poskytnúť im možnosť, aby cvičenia mohli byť vytvárané z ich vlastných textov akéhokoľvek zamerania. Samotní užívatelia v priebehu testovania ďalej prejavili záujem o nastavenie úrovne vo Fixáciach automaticky podľa úrovne v RZR. To je prakticky realizovateľné, keďže, ako už bolo uvedené, na zmysluplnom texte sme schopní jedným pohľadom zachytiť približne 3,5-násobok dĺžky náhodného zhluku písmen. Táto možnosť teda bude do nastavení pridaná.

Z vyššie uvedeného možno odvodiť nastavenia, ktoré by bolo dobré do tohto cvičenia zahrnúť:

- nastavenie úrovne s možnosťou automatického nastavenia podľa RZR,
- nastavenie doby fixácie (podobne ako pri RZR),
- vypnutie/zapnutie zvýrazňovania – zvukového aj vizuálneho,
- možnosť načítania užívateľského textu ako zdroja slovnej zásoby,
- nastavenie šírky strany (dĺžky riadka),
- a nakoniec možnosť nastavenia veľkosti, typu a farby písma.

Na základe všetkých týchto požiadaviek bol vytvorený návrh GUI cvičenia Fixácie. Tento koncept je znázornený na obr. 3.3. Pre ilustráciu je záložka *Cvičenie* zobrazená za behu cvičenia.



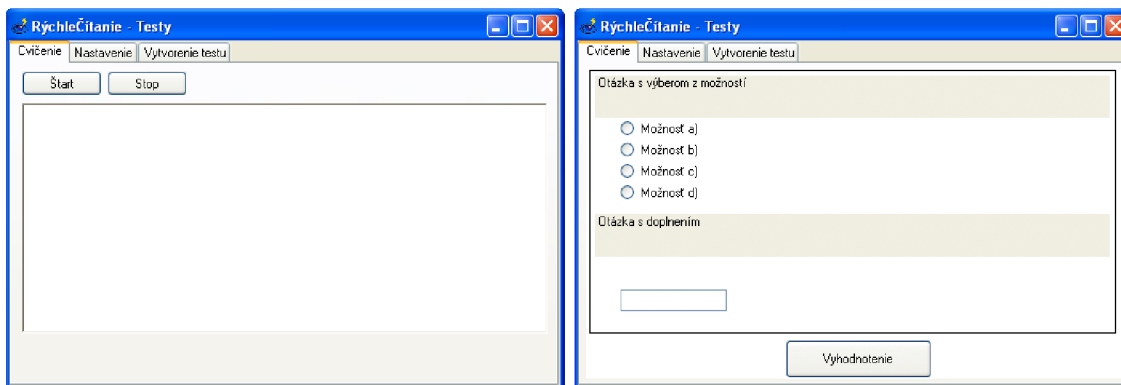
Obrázok 3.3: Návrh GUI cvičenia Fixácie

### 3.4 Meranie rýchlosti čítania a miery chápanosti (*Testy*)

Meranie rýchlosti čítania a miery chápanosti (ďalej *Testy*) má za cieľ priebežne merať užívateľov pokrok. Takéto meranie by mal užívateľ absolvovať približne po každých troch hodinách intenzívneho tréningu v RZR a Fixáciach. Zopakujme na začiatok, z čoho zostáva takéto testovanie. Najprv je to text, ktorý užívateľ prečíta s využitím získaných zručností. Počas tohto čítania je mu meraný čas, na základe ktorého sa potom vypočíta rýchlosť čítania. Po prečítaní nasleduje odpovedanie na otázky k tomuto textu, tie užívateľ zodpovie, po čom nasleduje vyhodnotenie testu. Toto vyhodnotenie obsahuje užívateľovu rýchlosť, mieru chápanosti a z toho vypočítaný celkový výkon.

Rovnako ako pri Fixáciach však aj tu treba vyriešiť otázku, čo bude tvoriť samotné testy, keďže ich nie je možné generovať automaticky. Je treba zaručiť, aby sa jednalo o kvalitné testy s vecnými otázkami, preto som pristúpil ku kroku použiť existujúce testy, publikované v [6], ktoré sú veľmi dobre spracované a navyše, narozdiel od mnohých iných testov obsahujú aj otázky s výberom z možností, takže sa na elektronické testovanie dobre hodia.

V rámci zachovania flexibility programu by však bolo vhodné umožniť aj v tejto časti, aby užívateľ mohol byť skúšaný aj na testoch, zapadajúcich svojím obsahom do jeho oblasti záujmu. Preto by bolo dobré pridať možnosť tvorby vlastných testov. Je pochopiteľné, že užívateľ nemôže vytvárať testy sám sebe, no môže ich vytvárať iným užívateľom, testy sa

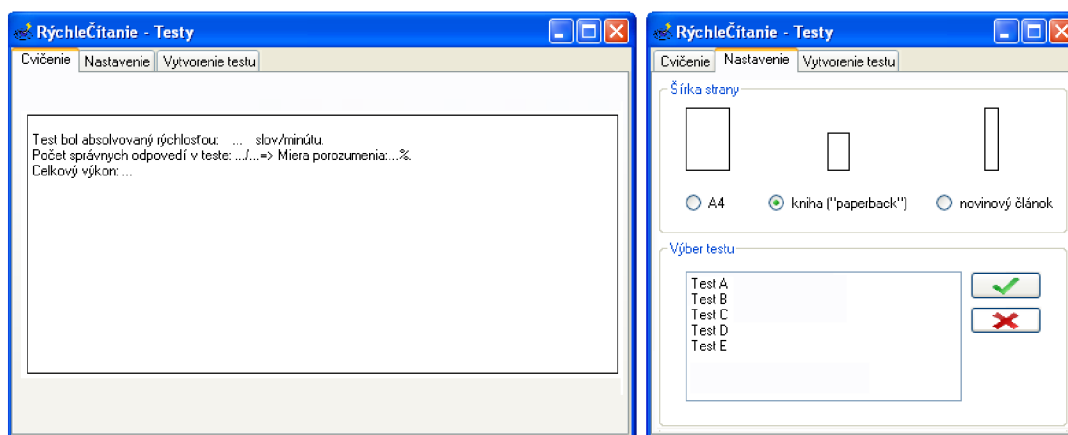


Obrázok 3.4: Návrh GUI cvičenia Testy – text testu a otázky k textu

môžu zdieľať prostredníctvom internetu a podobne. Táto možnosť sa rovnako môže hodiť napríklad v prípade, keď sa bude učiť rýchlo čítať skupina ľudí, ktorá si tak môže testy vytvárať navzájom.

Z vyššie uvedeného teda vyvodíme, čo všetko bude tvoriť GUI tohto cvičenia. Najdôležitejšou časťou bude záložka so samotným testom, ktorá bude obsahovať textové pole a tlačidlo na zahájanie/ukončenie čítania. Po ukončení čítania sa namiesto textu objavia otázky, ktoré treba zodpovedať, s tlačidlom, pomocou ktorého užívateľ bude môcť ukončiť ich zodpovedanie a zobrazí vyhodnotenie. Vyhodnotenie sa zobrazí na mieste, kde boli predtým zobrazené otázky, keďže tieto pri vyhodnotení už nemusia byť viditeľné.

Čo sa týka nastavení, bude treba, aby mal užívateľ možnosť vybrať si test, ktorý chce absolvovať a zvoliť šírku strany, akú chce používať. Čo sa týka výberu testu, bolo by možno tiež vhodné, keby mal užívateľ možnosť zvolený test aj vymazať – napr. v prípade skúšobných testov a pod. Možnosť nastavenia písma v tomto prípade nie je nutná, keďže testy sa nebudú absolvovať často a naviac je takto možno simulovať situáciu, keď čítame text predvolenou veľkosťou písma, napr. knihu či noviny.

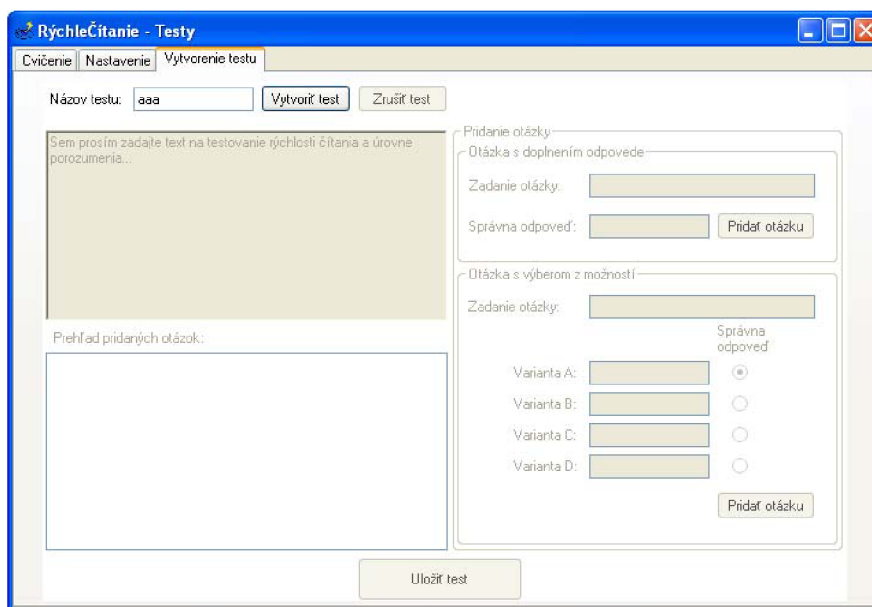


Obrázok 3.5: Návrh GUI cvičenia Testy – vyhodnotenie testu a nastavenie

Ďalej sa treba zamerať na možnosť tvorby vlastného testu. Tu bude potrebné mať pole na zadanie názvu testu, na vytvorenie, uloženie, či zrušenie testu, ďalej polia na pridanie

textu, otázok s výberom z možností a s doplnením odpovede a bolo by tiež vhodné umožniť užívateľovi, aby si pridané otázky mohol prezerat a mal tak približný prehľad o tom, čo bude ním vytváraný test obsahovať.

Zo všetkých požiadaviek bol vytvorený návrh jednotlivých záložiek, resp. obrazoviek, s ktorými bude užívateľ prichádzať do styku. Tieto sú zobrazené na obr. 3.4, 3.5 a 3.6



Obrázok 3.6: Návrh GUI cvičenia Testy – vytvorenie testu

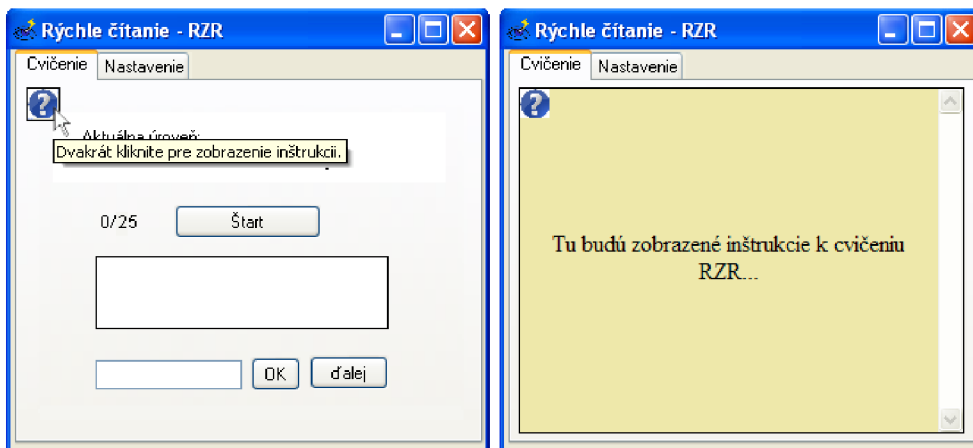
### 3.5 Ostatné súčasti programu

Okrem vyššie uvedených cvičení bude treba do programu implementovať niekoľko ďalších súčastí.

Dôležitým prvkom v programe bude určite nápoveda, keďže ako už bolo napísané, väčšina užívateľov sa pravdepodobne do styku s rýchlym čítaním dostane po prvýkrát. Okrem štandardných a v súčasnosti často používaných tzv. „bublinových nápoved“ (angl. *bubble tooltip*) by bolo dobré do programu zahrnúť aj o niečo obsiahlejšiu nápovedu. Je všeobecne známe, že nápoveda nie je medzi užívateľmi veľmi obľúbená, zvlášť keď je príliš objemná, užívateľ sa v nej stráca a často nie je schopný v rozumnom čase nájsť, čo hľadá. Pravdepodobne práve preto veľa užívateľov nápovedu nepoužíva skoro vôbec, resp. po nej siahne, až keď si s niečím skutočne nevie poradiť. Zrovna to však môže byť v prípade tohto programu veľmi nežiadúce, pretože pri tréningu je dôležité, aby boli cvičenia realizované podľa pokynov, keďže to môže zvýšiť ich efektívnosť. Preto som dospel k záveru, že najvhodnejšie bude nápovedu umiestniť priamo do jednotlivých cvičení, príp. nastavení, pričom ju užívateľ bude môcť rozbaľiť a zbaľiť, kedy bude chcieť. Táto nápoveda bude stručná a presne cieleňá na konkrétnu časť programu, vďaka čomu nebude nutné ňou zdĺhavo listovať, pretože všetky potrebné pokyny budú na jednom mieste. Podľa týchto požiadaviek bol vytvorený koncept systému nápovedy, ktorý je na príklade cvičenia RZR zobrazený na obr. 3.7.

Ďalším prvkom, ktorý býva v programoch často využívaný, je úvodná obrazovka. Pri





Obrázok 3.7: Návrh systému nápovedy na ukážke RZR

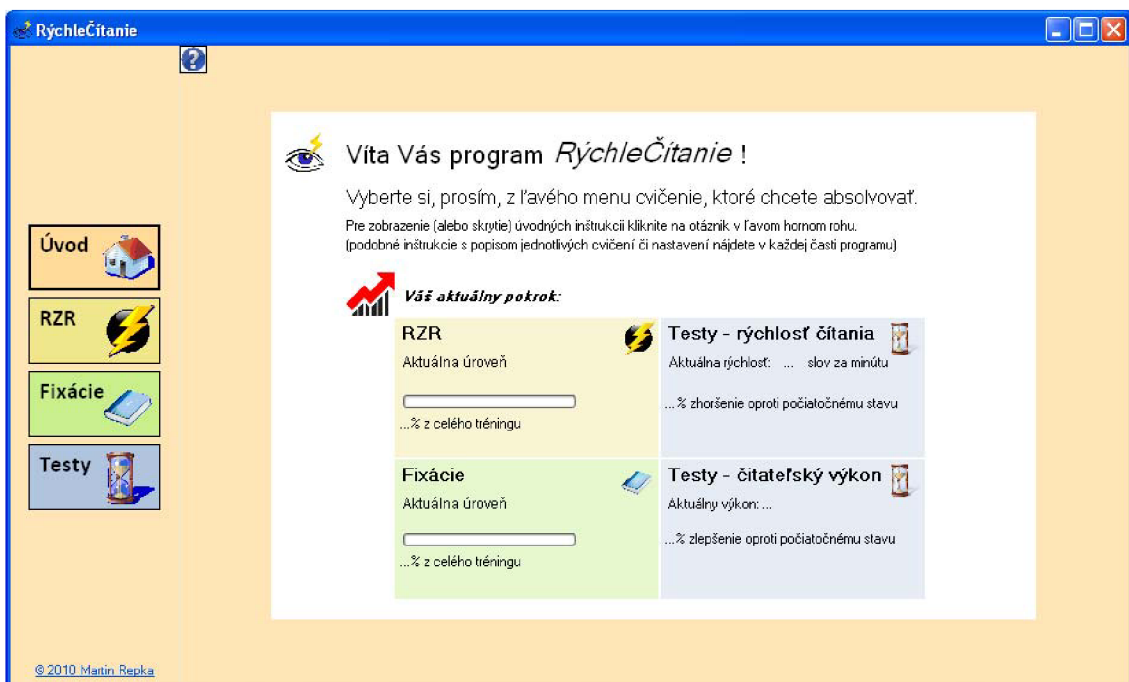
zoznamovaní s neznámym programom môže takáto súčasť výrazne pomôcť v orientácii a okrem toho môže mať svoj význam aj počas celého používania programu, keďže môže byť nositeľom rôznych aktualít, noviniek a pod. Z toho dôvodu sa domnievam, že použitie tohto prvku by mohlo byť v navrhovanom programe veľmi prínosné. Takáto úvodná obrazovka by teda mohla obsahovať úvodné inštrukcie, resp. radu, kde sa inštrukcie nachádzajú a rovnako by mohla obsahovať aktuálny pokrok užívateľa – teda aktuálnu úroveň v jednotlivých cvičeniach a prípadne zlepšenie oproti počiatočnému stavu. Na základe týchto požiadaviek bol vytvorený návrh GUI, ktorý je v kontexte GUI výsledného programu zobrazený na obr. 3.8.

### 3.6 Intergrácia do výsledného programu

Podľa zadania mali byť dielčie cvičenia implementované do výsledného, uceleného programu. Treba teda určiť, ako bude konečný produkt vlastne vyzeráť. Trendom pri podobných aplikáciach býva GUI rozdelené na dve časti – ovládaciú (menu) a tzv. pracovnú, kde sa zobrazuje užívateľom zvolená súčasť programu. Domnievam sa, že tento spôsob je veľmi príjemný a intuitívny, preto by mohol byť použitý aj v tomto programe. Výsledné rozhranie teda bude rozdelené na ľavú statickú časť, kde si užívateľ vyberie, ktorému cvičeniu sa chce venovať a na pravú časť, ktorá sa bude meniť podľa aktuálne vybraného cvičenia. Pri spustení potom bude zobrazená predvolene úvodná obrazovka a bude záležať na užívateľovi, ktorému cvičeniu sa chce venovať.

Počas priebežného testovania užívateľa prejavili záujem na tom, aby boli cvičenia farebne odlišené, čo by jednak prispelo k väčšej prehľadnosti a rovnako by to spravilo program vizuálne príjemnejším, preto som sa rozhodol tento nápad zapracovať do výslednej podoby. Základné farby boli vybrané podľa užívateľských preferencií a farebné odtiene boli doladené tak, aby prechody boli plynulé a farebná schéma pôsobila prirodzene.

Návrh výsledného GUI aplikácie je znázornený na obr. 3.8.



Obrázok 3.8: Návrh GUI výsledného programu

## Kapitola 4

# Implementácia

V tejto kapitole budú popísané využité implementačné nástroje a stručne vysvetlená implementácia jednotlivých súčastí, ako aj spoločných prvkov programu. Pri odkazovaní sa na metódy, premenné, či triedy bude použité **strojopisné písmo**.

### 4.1 Vývojové nástroje

Po konzultácií s vedúcim práce bolo na implementáciu zvolené vývojové prostredie Visual Studio 2008 s .NET framework v3.5 a programovací jazyk C#. Jedná sa o vyspelý vývojový nástroj, poskytujúci širokú škálu knižníc, ktoré boli pri implementácii použité. Návrh GUI bol realizovaný pomocou dizajnéra, ktorý Visual Studio ponúka, vrátane možnosti použitia pomerne veľkého množstva ovládacích prvkov.

### 4.2 RZR

Cvičenie RZR, rovnako ako ostatné časti, bolo pôvodne navrhnuté ako formulár (**Form**), ktorý bol pri integrácii do výsledného programu transformovaný na **Panel**. Pri návrhu boli použité základné štandardné prvky. Základ je tvorený záložkovým panelom **TabControl**, do ktorého sú vložené ďalšie **Panel**-y, aby bolo možné spoľahlivo zaistiť konzistenciu prvkov napr. pri zmene veľkosti okna a zároveň tak doceliť vizuálne oddelenie samotného cvičenia od ovládacích prvkov programu.

Záložka *Cvičenie* je tvorená prvkami **Label**, **Button** a **TextBox**. Záložka *Nastavenie* je pomocou **GroupBox**-ov rozdelená do logických celkov vrámci ktorých sú navyše použité prvky **ComboBox**, **TrackBar**, **NumericUpDown**, **RadioButton** a **CheckBox**. Dialógové okná na výber písma a farby sú inštancie tried **FontDialog** a **ColorDialog**. Na zobrazenie skupiny znakov počas cvičenia je použitý **Timer**, ktorému je podľa rýchlosti nastavený časový interval, počas ktorého má skupinu na obrazovke zobraziť.

Na generovanie skupín slov bola vytvorená trieda **RzrHandler**. Táto trieda obsahuje metódy na generovanie náhodných skupín znakov podľa špecifických požiadaviek a zároveň poskytuje prostriedky na overovanie správnosti odpovede.

Počas cvičenia je programom riešený automatický prenos zamerania tlačidiel (tzv. *focus*), aby bolo možné cvičenie pohodlne ovládať klávesnicou, len stláčaním klávesy **Enter** a písaním odpovedí.

## 4.3 Fixácie

Základ tohto cvičenia tvorí tabuľkový zoznam triedy `ListView`, v ktorom je cvičenie zobrazované. Pri návrhu GUI tohto cvičenia boli použité rovnaké prvky ako pri návrhu RZR, navyše s dialógovým oknom `FileDialog` na výber zdroja užívateľskej slovnej zásoby.

Zvukový a vizuálny zvýrazňovací prvok je riešený použitím `Timer`-a. Zvuk metronómu je prehrávaný použitím triedy `SoundPlayer`, vizuálne zvýrazňovanie skupín je realizované časovanou zmenou farby písma jednotlivých položiek. Vizuálne zvýrazňovanie, ktoré bolo pridané dodatočne, však spôsobovalo problém zapríčinený čiastočne neefektívnou implementáciou triedy `ListView`. Tá je implementovaná tak, že pri zmene vizuálnych vlastností jedinej bunky je prekreslený úplne celý zoznam, čo pri častej zmene farby jednotlivých buniek spôsobuje „blikanie“ celého `ListView`. Pri hľadaní riešenia na vývojárskych internetových fórach bolo nájdené východisko – vytvorenie vlastnej triedy odvodenej od pôvodného `ListView`, pričom sa v konštruktore tejto triedy aktivuje *dvojité bufferovanie*<sup>1</sup> a použije sa vlastná implementácia metódy `OnNotifyMessage()` tak, aby filtrovala správu na prekreslenie celého pozadia prvku `ListView`.

Netriviálnym problémom z pohľadu implementácie bolo ďalej spracovanie slovnej zásoby. Na spracovanie boli vytvorené metódy, ktoré prehľadávajú zadaný text a snažia sa nájsť vhodné skupiny slov, podľa požadovanej dĺžky. Spracovanie užívateľskej slovnej zásoby bolo vyriešené spôsobom, keď je zdroj slovnej zásoby (súvislý text) uložený v textovom súbore formátu `.txt` alebo `.rtf` (formát `.doc` a ďalšie formáty kvôli problémom s kompatibilitou, resp. komplikovanému spracovaniu nie sú podporované) a tento súbor je programom automaticky spracovaný – rozdelený na skupiny slov požadovanej dĺžky, ktoré sa potom pri vytváraní cvičenia používajú.

## 4.4 Testy

Testy boli z pohľadu implementácie najnáročnejším cvičením. Najskôr bolo potrebné vymyslieť rozumný spôsob ukladania testov do súboru a načítavanie zo súboru do programu. Ďalej bolo nutné dynamicky generovať GUI s otázkami, na ktoré užívateľ odpovedá po prečítaní textu a zabezpečiť správne prepínanie jednotlivých obrazoviek pri absolvovaní testu – zobrazenie textu, otázok a vyhodnotenia. Nakoniec bolo treba navrhnúť vhodné GUI pre tvorbu nových testov. Poďme si teda riešenie niektorých implementačných záležitostí stručne popísať.

Čo sa týka ukladania testu do súboru, bol zvolený formát XML, ktorý je v súčasnosti značne rozšírený a knižnice `.NET` poskytujú kvalitné nástroje pre prácu s týmto formátom. Z požiadaviek, ktoré boli na test kladené bol vytvorený jednoduchý model *DTD*<sup>2</sup>, ktorého jadro je zobrazené na obr. 4.1. Na tomto obrázku možno vidieť štruktúru výsledného dokumentu, ale popíšme ju stručne aj slovne. XML súbor s testom teda tvorí hlavný element `<Test>` s atribútmi meno testu, dĺžka (počet slov) a počet otázok. Vránci tohto elementu sa nachádza samotný text testu a element `<Questions>`, ktorý obsahuje jednotlivé otázky – elementy `<question>`, ktoré môžu byť typu `multChoice`, teda otázky s výberom, alebo typu `fill`, teda otázky s doplnením odpovede.

<sup>1</sup>Angl. *double buffering*, presnejšie *OptimizedDoubleBuffer* je možnosť, ktorú ponúka metóda `SetStyle()` triedy `ListView`.

<sup>2</sup>Z angl. *Document Type Definition* – definícia typu dokumentu je šablónou, ktorá určuje štruktúru daného dokumentu a vymedzuje povolené prvky, čím určuje štandardy, ktorými sa dokument riadi.

```

<!ELEMENT Questions ( question+ ) >
<!ELEMENT Test ( text, Questions ) >

<!ATTLIST Test length NMTOKEN #REQUIRED >
<!ATTLIST Test name NMTOKEN #REQUIRED >
<!ATTLIST Test questionsCount NMTOKEN #REQUIRED >
<!ELEMENT answer ( #PCDATA ) >

<!ELEMENT option ( #PCDATA ) >

<!ATTLIST option isCorrect ( False | True ) #REQUIRED >
<!ELEMENT question ( answer | option )* >

<!ATTLIST question text CDATA #REQUIRED >
<!ATTLIST question type ( multChoice | fill ) #REQUIRED >
<!ELEMENT text ( #PCDATA ) >

```

Obrázok 4.1: DTD XML súboru s uloženým testom

Pre prácu s testami bolo vytvorených niekoľko tried, kvôli abstrakcií a lepšej manipulácii. Samotný test je zapúzdrený triedou `Test`, ktorá obsahuje parametre spracovávaného testu a zoznam otázok – objektov triedy `Question`, ktorá zapúzdruje štruktúru otázky. `Test` je potom spracovávaný ďalšou triedou `TestHandler`, ktorá má na starosti vytvorenie, resp. načítanie a spracovanie testu na úrovni samotných XML súborov. Na spracovanie na tejto úrovni boli použité vstavané triedy `XmlTextWriter` a `XmlTextReader`. Trieda `TestHandler` obsahuje metódy na vytvorenie testu, pridanie otázok, uloženie testu a rovnako metódy na načítanie testu – otvorenie, načítanie parametrov a načítanie otázok. Pomocou tejto triedy ďalej možno určiť, do akého priečinka majú byť testy ukladané, určiť prípadný prefix/suffix názvov súborov s testami, ďalej získať zoznam uložených testov, či určitý test vymazať.

GUI záložky so samotným testom bolo riešené pomocou prekrývajúcich sa panelov, ktoré sú skrývané a zobrazované podľa potreby. Jedná sa o panel so samotným testom, panel s otázkami a nakoniec panel s vyhodnotením. Ich prepínanie má v réžii užívateľ tým, kedy ukončí jednotlivé fázy testu, t.j. čítanie textu a odpovedanie na otázky.

Užívateľské rozhranie na vytvorenie nového testu bolo riešené pridaním ďalšej záložky s potrebnými ovládacími prvkami. Kvôli tomu, aby mal užívateľ priebežnú kontrolu nad pridávaným testom, resp. pridanými otázkami, bol pridaný prvok `TreeView`, ktorý priebežne zobrazuje pridané otázky. Bola taktiež implementovaná funkčnosť, aby užívateľ mohol z tohto zoznamu otázky jednotlivou odstrániť. Pri pridávaní otázok s možnosťami bola zahrnutá aj podmienka, že takáto otázka musí mať minimálne dve varianty odpovede, keďže jedna by nedávala zmysel a takýmto spôsobom je potom možné pridávať aj otázky typu „áno/nie“ a iné.

## 4.5 Ostatné súčasti programu

Okrem vyššie popísaných cvičení bolo potrebné implementovať niektoré ďalšie, spoločné prvky programu, ako systém nápovedy, ukládanie nastavení, či úvodnú obrazovku. Posledným uvedeným sa zaoberať nebudeme, keďže sa v podstate jedná o triviálnu implementáciu, no pozrime sa stručne aspoň na prvé dve spomenuté súčasti.

Grafický návrh systému nápovedy bol už popísaný v kapitole 3.5 na str. 20. Z pohľadu implementácie bola vytvorená špeciálna trieda `PanelHelp`, odvodená od štandardnej triedy `Panel`. Tejto novej triede boli pridané vlastnosti a metódy, aby vyhovovala požiadavkám, ktoré sú na ňu kladené.

Obrázok s otáznikom, z ktorého sa po kliknutí rozbalí nápoveda, bol riešený pomocou prvku `PictureBox`, ku ktorému bola vytvorená vlastnosť, vďaka ktorému tento obrázok možno jednoducho meniť z vytvorenej inštancie triedy. Ďalej bola priamo tomuto obrázku pridaná riadková nápoveda `ToolTip`, ktorá sa zobrazí pri prejdení myšou nad týmto prvkom. Do samotného panelu bol pridaný prehliadač webových stránok `WebBrowser`, s vlastnosťou na umiestnenie zobrazovanej stránky. Vďaka tomu mohla byť vytvorená prehľadná HTML nápoveda, ktorá bola zabudovaná priamo do programu. Aby rozbalenie/zbalenie nápovedy bolo vizuálne príjemnejšie, boli do tejto triedy implementované metódy `Expand()` a `Collapse()`, ktoré využívajú `Timer-y`, na postupné zväčšovanie panela až do veľkosti podľa nastavenej vlastnosti `MaxExpandSize`. Vďaka tomu vzniká vizuálne zaujímavý efekt, ktorý môže spraviť používanie nápovedy príjemnejším, čo potenciálne môže zvýšiť frekvenciu jej využívania.

Ďalšou súčasťou celého programu, napriek tomu, že čiastočne skrytou, je ukladanie nastavení. Na správu nastavení bola vytvorená trieda `SettingsHandler`, ktorá sa používa v celom programe. Jej vlastnosťami sú všetky nastavenia programu. Nastavenia sú fyzicky ukladané do súboru `app.config`, čo je svojou štruktúrou XML súbor, vytváraný vstavanou triedou na správu konfigurácie programu – `Configuration`, resp. `ConfigurationManager`.

## 4.6 Integrácia do výsledného programu

Popíšme na záver, akým spôsobom boli jednotlivé súčasti integrované do programu.

Výsledné GUI je tvorené prvkom `SplitContainer`, ktorý formulár rozdeluje na dva panely – ovládaciu časť (menu) a pracovnú časť s aktuálnym cvičením. Pracovná časť z pohľadu implementácie obsahuje `Panel-y` všetkých cvičení, ktoré sa skrývajú a zobrazujú podľa požiadaviek užívateľa. Menu v ľavej časti je tvorené prvkami `Button`, so zmeneným štýlom, farbou pozadia a podkladovou ikonou. Použité ikony v menu aj v cvičeniach sú prebrané zo sád, voľne dostupných na internete. Pri spustení programu sa teda načítajú nastavenia programu, inicializujú jednotlivé cvičenia a ich prvky a od tohto momentu je program pripravený na použitie.

## Kapitola 5

# Testovanie

Program bol testovaný v dvoch fázach: prvá prebiehala počas vývoja, keď niektoré implementované prvky boli testované na vybraných užívateľoch, prípadne bol zisťovaný ich názor na ovládanie, skúmané ich reakcie a pod. V druhej fáze bol testovaný už výsledný produkt, ktorý bol poskytnutý väčšej skupine ľudí, aby ho nejaký čas používali a poskytli spätnú väzbu. V tejto kapitole sa budeme venovať druhej fáze tohto testovania, najmä analýzou výsledkov.

### 5.1 Metodika testovania

Program bol počas druhej testovacej fázy poskytnutý väčšej skupine ľudí, pričom im boli dané nasledujúce pokyny:

- ▶ používať program aspoň niekoľko dní, aspoň polhodinu denne,
- ▶ počas používania sa riadiť inštrukciami a náповедou, ktorú program obsahuje,
- ▶ zoznámiť sa so všetkými cvičeniami a používať ich,
- ▶ po otestovaní vyplniť dotazník k programu.

Dotazník bol zameraný jednak na prvý dojem z programu – najmä orientáciu v ňom, ale hlavne na skúsenosti z dlhodobejšieho používania. V dotazníku bolo skúmané najmä:

- ako sa užívatelia orientovali v programe pri prvom spustení,
- ktoré cvičenie sa im páčilo najviac, ktoré najmenej a prečo,
- ktoré cvičenie je z ich pohľadu najužitečnejšie a ktoré naopak najmenej užitočné,
- ako často je využívaná náповeda a či je zrozumiteľná,
- ako často sú využívané nastavenia a ktoré sú využívané najviac,
- čo konkrétne sa užívateľom na programe páčilo a čo naopak nepáčilo,
- čo by pridali a čo naopak z programu odstránili, keby mohli,
- ako na užívateľov pôsobia zvolené farby,
- ako sa im program celkovo ovládal,
- či sa im získané techniky darí používať pri bežnom čítaní,
- a či majú pocit, že sa ich čítanie po intenzívnejšom tréningu bude zlepšovať.

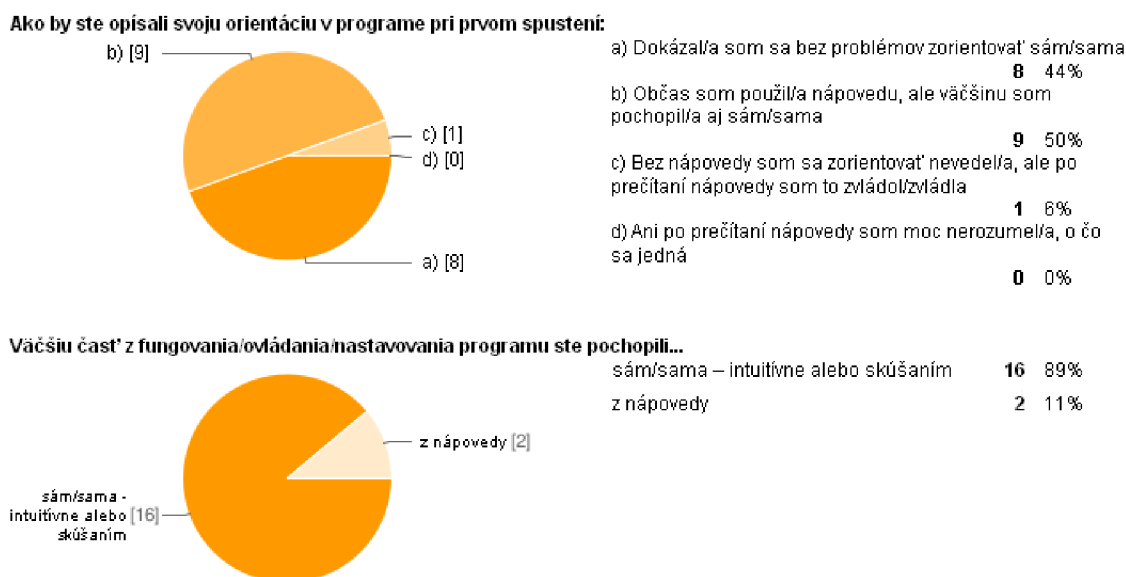
Kompletný dotazník vo forme, v akej bol poskytnutý užívateľom sa nachádza v prílohe **B**. Poďme sa ďalej pozrieť na výsledky testovania.

## 5.2 Výsledky testovania

Vyplnené dotazníky odoslalo 18 ľudí, z toho presne polovica žien a polovica mužov, prevažne vekovej kategórie 18–25 rokov. Väčšina z nich uvádzala aj detaily a svoje vlastné postrehy a pripomienky k programu, ktoré budú spomínané priebežne.

Orientácia v programe pri prvom spustení bola hodnotená prevažne kladne. Užívatelia väčšinou nemali problém zorientovať sa, či pochopiť fungovanie programu z nápovedy, ako je znázornené na obr. 5.1. Samotná nápoveda bola hodnotená takmer výlučne ako „dostatočná a dobre zrozumiteľná“. Z týchto výsledkov však možno pozorovať ešte jeden jav – užívatelia sa v mnohých prípadoch orientovali sami, čo môže byť pre efektívne absolvovanie cvičení nežiadúce. Preto by možno bolo dobré znovu sa zamyslieť nad prístupom k nápovede a viac užívateľov nabádať k jej používaniu.

Farby použité v programe boli hodnotené prevažne neutrálne, prípadne príjemne. V niektorých prípadoch sa užívatelia vyjadrili, že sú až moc nevýrazné, takže by bolo možno vhodné použiť o niečo tmavšie odtiene.



Obrázok 5.1: Orientácia v programe pri prvom spustení

Zamerajme sa ďalej na užívateľské preferencie jednotlivých cvičení.

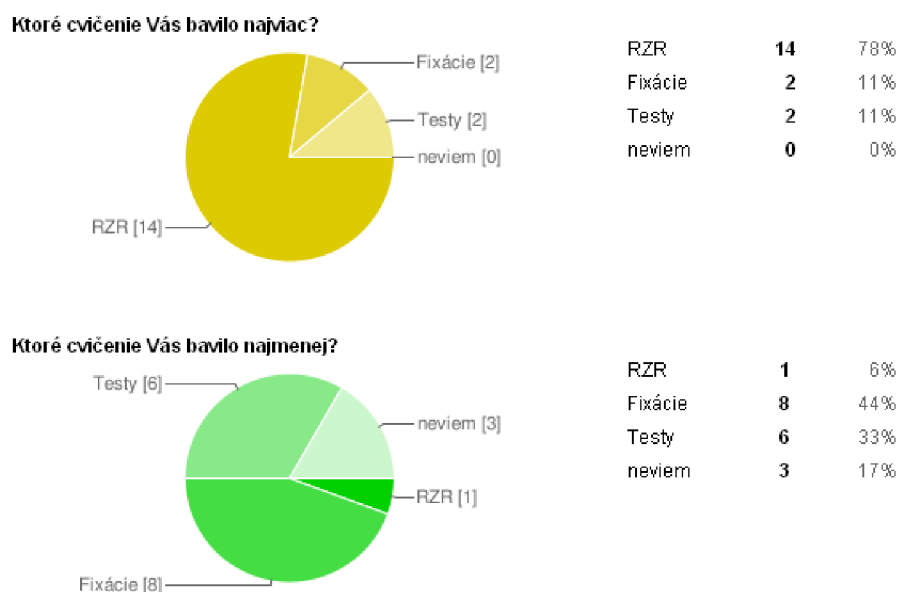
Najobľúbenejšie cvičenie medzi užívateľmi, ako ukazuje obr. 5.2, bolo RZR. V dotazníkoch boli spomínané najmä tieto dôvody:

- cvičenie bolo dynamické, „akčné“, vyžadovalo značnú koncentráciu,
- bol priebežne viditeľný výsledok, čo motivovalo pokračovať,
- dobre ovládateľné pomocou klávesnice, je možné ho kedykoľvek zastaviť.

Ako nedostatky tohto cvičenia boli v niektorých prípadoch uvádzané napr. malé okno s cvičením v strede, trochu neprehľadný začiatok, či voľba predvoleného písma.



Ako najmenej obľúbené cvičenia boli zvolené Fixácie a Testy (viď obr. 5.2). V prípade Fixácii užívatelia uvádzali, že cvičenie nie je dostatočne zaujímavé, že chýba výraznejšie ocenenie užívateľa, či absencia súvislého zmysluplného textu. V prípade Testov to bolo použitie „nekontrastného písma“, ťažkých otázok, či „zľhavosť“ cvičenia. Je však nutné pripomenúť, že Testy nie sú „štandardným“ cvičením, ale najmä prostriedkom na meranie pokroku, čo užívateľom pravdepodobne nebolo celkom jasné, takže by to bolo vhodné v nápovede zdôrazniť.

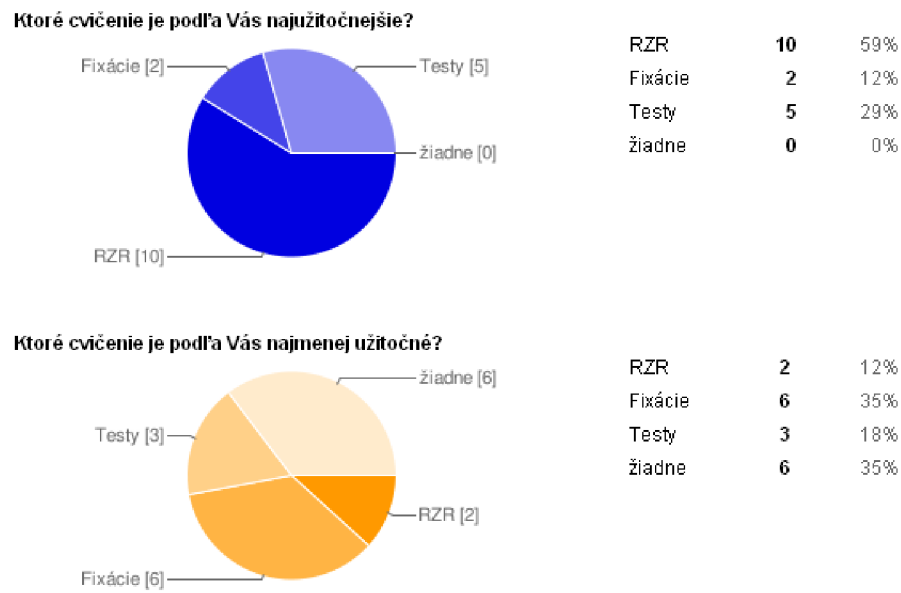


Obrázok 5.2: Užívateľské preferencie cvičení v programe

Podobne dopadlo aj užívateľské hodnotenie cvičenia podľa subjektívnej užitočnosti, čo znázorňuje obr. 5.3. Ako najužitečnejšie sa užívateľom väčšinou javilo cvičenie RZR, zatiaľ čo za najmenej užitočné považovali Fixácie, príp. žiadne cvičenie. Z toho rovnako vyplýva, že by bolo vhodné sa na Fixácie zamerať viac a spraviť ich užívateľsky príjemnejšie.

V dotazníkoch sa tiež ukázalo, ako často užívatelia využívajú nastavenia programu. Podľa výsledkov len približne pätina užívateľov používa nastavenia často, čo môže znamenať, že je program od začiatku nastavený tak, že vyhovuje väčšine užívateľov, alebo to potvrdzuje nepísané pravidlo, že väčšina užívateľov sa jednoducho snaží nastaveniam čo najviac vyhýbať. Obidve možnosti však potvrdzujú, že je potrebné počiatocnej konfigurácii venovať náležitú pozornosť, aby pre väčšinu užívateľov nebolo nutné od začiatku nastavenia rekonfigurovať im vyhovujúcim spôsobom.

Účinnosť techník na výučbu rýchleho čítania použitých v programe ilustrujú grafy na obr. 5.4. Väčšine respondentov sa do doby vyplňovania dotazníka nedarilo získané techniky efektívne používať pri bežnom čítaní. To však môže znamenať aj to, že program zatiaľ nebol používaný dostatočne intenzívne na to, aby sa výsledky prejavili, keďže testovaciu dobu tvorilo len niekoľko dní. Túto možnosť potvrdzuje aj druhý graf na obr. 5.4, kde, ako je zobrazené, užívatelia väčšinou vyjadrili názor, že sa po intenzívnejšom tréningu ich čítanie bude zlepšovať.

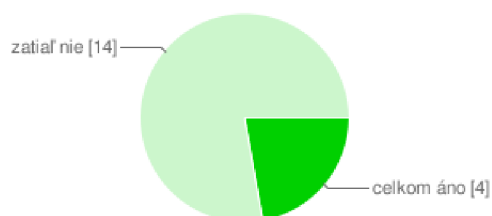


Obrázok 5.3: Úžitočnosť metód, hodnotená užívateľmi

### 5.3 Záver testovania

Testovanie prinieslo množstvo zaujímavých a podnetných užívateľských pripomienok k programu ako celku ako aj k jeho jednotlivým súčastiam. Z týchto pripomienok sa niektoré podarilo do výslednej implementácie zahrnúť, tie ostatné môžu byť cennými poznatkami pri ďalšom vývoji programu.

**Darí sa Vám získané techniky používať pri bežnom čítaní?**



celkom áno	<b>4</b>	22%
zatiaľ nie	<b>14</b>	78%

**Máte pocit, že sa Vaše čítanie po intenzívnejšom tréningu týmto programom bude zlepšovať?**



áno	<b>9</b>	50%
asi áno	<b>8</b>	44%
asi nie	<b>1</b>	6%
nie	<b>0</b>	0%

Obrázok 5.4: Účinnosť programu, hodnotená užívateľmi

## Kapitola 6

# Záver

Cieľom tejto práce bolo vytvoriť niekoľko užívateľských rozhraní na nácvik rýchleho čítania a následne ich integrovať do ucelenej aplikácie. Na začiatku bola preštudovaná metodika rýchleho čítania, na základe čoho boli vytvorené nástroje na rozširovanie zrkového rozpätia, znižovanie počtu fixácií a nakoniec nástroj na meranie čitateľského výkonu. Navyše bolo vytvorené rozhranie na vytváranie užívateľských testov v XML formáte. Všetky implementované nástroje sú založené na metódach rýchleho čítania používaných v praxi či už v kurzoch rýchleho čítania, alebo odporúčané autormi publikácii o rýchlom čítaní. Táto práca bola však do určitej miery aj experiment – snažila sa nástroje používané najmä v príručkách a kurzoch rýchleho čítania transformovať do elektronickej podoby.

Tento cieľ bol dosiahnutý, čo dokazujú výsledky testovania a pozitívne ohlasy užívateľov, ktorí program považovali za užitočný a sami mali pocit, že im môže pri výučbe rýchleho čítania pomôcť.

Vývoj programu však môže pokračovať aj naďalej. V budúcnosti by bolo dobré rozšíriť cvičenie na znižovanie počtu fixácií, aby bolo pre užívateľov zaujímavejšie – napr. pridať možnosť tréningu na nadväzujúcom texte. Takisto by mohlo byť prínosné pridať ďalších cvičení – napr. cvičenie na zameriavanie pozornosti na vrchnú polovicu slov, či pozorovacie cvičenie na skupinách podobných slov na zvyšovanie postrehu. Niektorí užívatelia ďalej navrhovali pridať pamäťových cvičení, čo by takisto mohlo byť zaujímavé rozšírenie.

Programovací jazyk C# a prostredie Visual Studio boli pre mňa novou skúsenosťou, počas vývoja som sa však s týmito nástrojmi dobre zoznámil a tvorba tohto programu ma veľmi bavila. Práve preto by som rád s jeho vývojom pokračoval aj naďalej a postupne z neho vytvoril produkt, ktorý bude konkurencieschopný aj na poli komerčných aplikácií. Túto prácu preto hodnotím veľmi prínosne a som rád, že som vďaka nej mohol príjemným spôsobom získať cenné skúsenosti s vývojom aplikácií.

# Literatúra

- [1] GUI Design Principles [online]. 2009 [cit. 2010-05-01].  
URL [http://en.wikibooks.org/wiki/GUI\\_Design\\_Principles](http://en.wikibooks.org/wiki/GUI_Design_Principles)
- [2] RapidReader: Speed Reading On-Demand [online]. 2010 [cit. 2010-05-01].  
URL <http://www.rapidreader.com>
- [3] Speed Reader-X – Speed Reading Made Easy – Memory and Comprehension Training Course [online]. 2010 [cit. 2010-05-01].  
URL <http://www.speedreaderx.com>
- [4] Speed Reading, Software for Reading Improvement and Reading Assessment [online]. 2010 [cit. 2010-05-01].  
URL <http://www.acereader.com>
- [5] David Gruber – Techniky duševní práce: David Gruber – Software [online]. 2010 [cit. 2010-05-01].  
URL [http://www.gruber.cz/index.php?option=com\\_content&task=blogcategory&id=25&Itemid=90](http://www.gruber.cz/index.php?option=com_content&task=blogcategory&id=25&Itemid=90)
- [6] GRUBER, D.: *Kdo to má všechno číst?!* Ostrava: Gruber – TDP, 1991.
- [7] GRUBER, D.: *Šetřme časem! Rychločtení – Rychlostudium*. Praha: MANAGEMENT PRESS, třetí vydání, 1995, ISBN 80-85943-04-2.
- [8] HOBART, J.: Principles of Good GUI Design [online]. 1995 [cit. 2010-05-01].  
URL <http://www.classicsys.com/css06/cfm/article.cfm?articleid=20>
- [9] MISTRÍK, J.: *Rýchle čítanie*. Bratislava: Slovenské pedagogické nakladateľstvo, 1979.
- [10] PAPIK, R.: *Naučte se číst!* Praha: Grada, 1992, ISBN 80-85424-93-2.
- [11] The Literacy Company: Top-Rated Speed Reading Software – The Reader’s Edge [online]. 2010 [cit. 2010-05-01].  
URL <http://www.readfaster.com/tre.asp>
- [12] TOMAN, I.: *Debordelizace hlavy: zprimitivněte k úspěchu*. Praha: TAXUS International, první vydání, 2009, ISBN 858-6-11-22023-8.
- [13] ZIELKE, W.: *Jak číst rychleji a lépe*. Praha: Nakladatelství Svoboda, druhé vydání, 1988.

# Zoznam príloh

**Príloha A: Obsah CD**

**Príloha B: Dotazník**

**Príloha C: CD**

**Príloha D: Plagát**

# Príloha A

## Obsah CD

Priložené CD je rozdelené nasledovne:

<code>README</code>	inštrukcie k inštalácii programu a prekladu zdrojových textov
<code>/bin</code>	priečinok so spustiteľným programom
<code>/src</code>	priečinok so zdrojovými textami programu
<code>/doc</code>	priečinok s užívateľským manuálom k programu vo formáte HTML (odporúča sa však používať vstavaná nápoveda programu)
<code>/misc</code>	priečinok s ostatnými súbormi – obsahuje text tejto správy vo formáte PDF a jej zdrojové texty, inštalátor .NET Framework 3.5 a prípadné ďalšie súbory

# Príloha B

## Dotazník

\* Povinné

Vaše pohlavie: \*

- žena  
 muž

Ako by ste opísali svoju orientáciu v programe pri prvom spustení: \*

- dokázal/a som sa bez problémov zorientovať sám/sama  
 občas som použil/a nápovedu, ale väčšinu som pochopil/a aj sám/sama  
 bez nápovedy som sa zorientovať nevedel/a, ale po prečítaní nápovedy som to zvládol/zvládla  
 ani po prečítaní nápovedy som moc nerozumel/a, o čo sa jedná

Ako na Vás pôsobili použité farby? \*

- veľmi príjemne  
 príjemne  
 neutrálne  
 skôr rušivo  
 veľmi rušivo

Ktoré cvičenie Vás bavilo najviac? \*

- RZR  
 Fixácie  
 Testy  
 neviem

Skúste napísať prečo Vás bavilo najviac.

Ktoré cvičenie Vás bavilo najmenej? \*

- RZR  
 Fixácie  
 Testy  
 neviem

Skúste napísať prečo Vás bavilo najmenej.

Ako často ste používali nastavenia jednotlivých cvičení? \*

- vôbec  
 len na začiatku  
 občas  
 často

V prípade, že ste používali nastavenia jednotlivých cvičení, čo ste napríklad nastavovali?

Nápoveda podľa Vás je... \*

- v niektorých prípadoch zbytočná  
 dostatočná a dobre zrozumiteľná  
 dostatočná, občas nezrozumiteľná  
 dostatočná, ale často nezrozumiteľná  
 nedostatočná

Pokiaľ sa Vám nápoveda niekde zdala zbytočná, nezrozumiteľná, príp. nedostatočná, napíšte to prosím sem.

Väčšiu časť z fungovania/ovládania/nastavovania programu ste pochopili... \*

- sám/sama – intuitívne alebo skúšaním  
 z nápovedy

Bola niektorá časť podľa Vás zložitá alebo nezrozumiteľná? Ak áno, ktorá?

Ako by ste celkovo ohodnotili ovládanie programu? \*

1 2 3 4 5

Najlepšie      Najhoršie



Skúste vymenovať aspoň 2-3 veci, ktoré sa Vám na programe páčili (pr. použité farby, systém náповedy, ovládanie konkrétneho cvičenia...)\*

\_\_\_\_\_

Skúste vymenovať aspoň 2-3 veci, ktoré sa Vám na programe nepáčili (pr. použité farby, systém náповedy, ovládanie konkrétneho cvičenia...:))\*

Zvládli (by) ste sami, pomocou inštrukcii, vytvoriť nový test (v sekcii Testy)? \*

- určite áno, rozumiem o čo ide
- asi áno, rozumiem o čo ide
- asi áno, ale nerozumiem o čo ide (na čo je to dobré)
- asi nie, nerozumiem o čo ide

Ktoré cvičenie je podľa Vás najužitočnejšie? \*

Ktoré cvičenie je podľa Vás najmenej užitočné? \*

\_\_\_\_\_

Darí sa Vám získané techniky používať pri bežnom čítaní? \*

- celkom áno
- zatiaľ nie

Napadá Vám niečo, čo by ste z programu odstránili, keby ste mohli?

\_\_\_\_\_

Napadá Vám niečo, čo by ste v programe určite ponechali, príp. rozšírili alebo vylepšili, keby ste mohli?

\_\_\_\_\_

Máte pocit, že sa Vaše čítanie po intenzívnejšom tréningu týmto programom bude zlepšovať? \*

- áno
- asi áno
- asi nie
- nie

Pokiaľ je ešte hocičo iné, čo by ste k programu chceli napísať, spravte tak prosím tu :)

