

Sem vložte zadání Vaší práce.

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
KATEDRA SOFTWAREVÉHO INŽENÝRSTVÍ



Bakalářská práce

Program pro tvorbu poznámek sdílených přes sdílené síťové úložiště

Vedoucí práce: Ing. Pavel Kubalík, Ph.D.

16. května 2013

Poděkování

Tímto chci poděkovat vedoucímu bakalářské práce, panu Ing. Kubalíkovi, Ph.D., za jeho čas, velmi přínosné konzultace a zkušené vedení. Také bych rád poděkoval svým rodičům a partnerce za jejich obrovskou podporu.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů, zejména skutečnost, že České vysoké učení technické v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V Praze dne 16. května 2013

.....

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2013 Tomáš Homolka. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.

Odkaz na tuto práci

Homolka, Tomáš. *Program pro tvorbu poznámek sdílených přes sdílené síťové úložiště*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2013.

Abstract

The aim of this design-implemetation work is to create a cross-platform desktop application for managing short text notes, which will be shareable by a file hosting service.

Keywords Java, Swing, note manager, file hosting service

Abstrakt

Úkolem této návrhově-rešeršní bakalářské práce bylo vytvořit multiplatformní desktopovou aplikaci pro správu krátkých textových poznámek, které bude možno sdílet přes sdílené síťové úložiště.

Klíčová slova Java, Swing, správce poznámek, sdílené síťové úložiště

Obsah

Úvod	1
1 Analýza zadání, určení cílů a požadavků na aplikaci	3
1.1 Popis problému	3
1.2 Funkční požadavky:	4
1.3 Nefunkční požadavky	4
1.4 Vymezení cílů:	4
2 Rešerše již existujících aplikací	7
2.1 Google Keep	7
2.2 Evernote	8
2.3 Sticky notes	9
2.4 TomBoy	10
2.5 Shrnutí	10
3 Analýza a návrh implementace aplikace	13
3.1 Případy užití	13
3.2 Programovací jazyk	16
3.3 Formát zdrojových souborů	17
3.4 Uložení dat v aplikaci	18
3.5 Synchronizace dat	18
3.6 Vizualní podoba	19
4 Popis realizace implementace	23
4.1 Model	23
4.2 View	24
4.3 Controller	26

5	Testování aplikace a analýza výsledků testování	27
5.1	Dotazník	27
5.2	Betatestování	28
5.3	Test kompatibility	28
6	Srovnání s již existujícími aplikacemi	31
6.1	Nezávislost na operačním systému:	31
6.2	Rozložení poznámek v aplikacích	31
6.3	Možnosti ukládání a sdílení poznámek	32
7	Zhodnocení splnění cílů	35
7.1	Splnění funkčních požadavků:	35
7.2	Splnění nefunkčních požadavků:	36
7.3	Vymezené cíle aplikace	36
Závěr		39
	Návrhy na budoucí vývoj a zlepšení aplikace	39
Literatura		41
A	Seznam použitých zkratk	43
B	Obsah přiloženého CD	45

Seznam obrázků

2.1	Screenshot Evernote	8
2.2	Screenshot Sticky notes	9
3.1	První vize GUI aplikace	19
4.1	Grafické znázornění dat kontrolovaných třídou Monitor	24

Seznam tabulek

2.1	Shrnutí rešerše již existujících aplikací	11
5.1	Tabulka kritických chyb	29

Úvod

Pokrok nezastavíš, to v informačních technologiích platí dvojnásob. Civilizace se nezadržitelně posouvá vpřed, vyvíjejí se nové technologie, lidé jsou zahlceni informacemi a přidáme-li ke všemu všudypřítomný stres působící na moderního člověka, stane se, že důležité informace člověk lehce zapomene. Mnoho lidí se proto uchýlilo k zaznamenávání si důležitých informací pomocí krátkých textových poznámek.

V roce 1980 představila americká firma 3M svůj nový produkt: Post-It. Jedná se o čtvercové papírové štítky na poznámky, opatřené z jedné strany lepidlem umožňujícím opakované strhnutí a znovunalepení na různé povrchy[5]. Tento nový vynález měl ohromný úspěch a lidé ho používají dodnes k zaznamenávání svých myšlenek či úkolů. Ale i tento geniálně jednoduchý vynález má své nevýhody. Hlavní nevýhodu vidím v jeho fyzické existenci. Post-It můžeme mít jen na jednom místě. Z domova proto nemáme možnost si přečíst Post-It, který máme nalepený na monitoru v kanceláři. Další nevýhodou je jejich hledání. Je téměř nemožné najít jednu konkrétní poznámku, pokud máte desítky poznámek na více než jednom místě.

Výstupem této bakalářské práce je aplikace s vlastním grafickým prostředím pro tvorbu, správu a sdílení krátkých textových poznámek. Poznámky budou uloženy na všech uživateli dostupných zařízeních a díky externí službě pro sdílení souborů (ESSS) budou mezi zařízeními aktualizovány přes webové uložiště. Tuto aplikaci můžeme chápat jako evoluci Post-It. Pro potřeby této práce pojmenuji tuto aplikaci „Noter“.

Analýza zadání, určení cílů a požadavků na aplikaci

1.1 Popis problému

Oblast správy a sdílení poznámek není v informačních technologiích neznámá. Existuje mnoho možností správy poznámek, ať už se jedná o webovou službu nebo desktopovou aplikaci. Také u většiny moderních operačních systémů můžeme najít integrovanou utilitu pro správu poznámek (například Sticky notes v OS Windows 7). Očekávám tedy, že Noter nebude první zkušeností uživatele s aplikací tohoto typu. Proto v této práci zohledním dosavadní zvyklosti z nejpoužívanějších aplikací pro poznámky.

Noter bude pracovat s jednoduchými poznámkami. Za jednoduchou poznámku se v rámci této práce považuje text bez nutnosti pokročilejší editace, jako je změna barvy nebo velikosti písma. Délka tohoto textu je neomezená, ovšem nejčastěji se pracuje s textem s délkou pohybující se do 200 znaků.

ESSS je služba umožňující uživateli synchronizaci dat s webovým uložištěm a jejich následné rozšíření na všechna synchronizovaná zařízení v reálném čase. Tato služba také nabízí sdílení mezi více uživateli a stará se o aktuálnost a jednotnost těchto dat. Nejnámější zástupci takovýchto služeb jsou Dropbox, Disk od společnosti Google a služba SkyDrive vyvíjená firmou Microsoft.

Hlavní cíle a požadavky na Noter vycházejí ze zadání této bakalářské práce a jsou doplněny o požadavky vyplývající z uživatelských potřeb a jeho předešlých zkušeností:

1.2 Funkční požadavky:

1. Správa poznámek — vytvoření, editace a odstranění poznámek
2. Synchronizace poznámek — aktualizace poznámek v aplikaci v závislosti na změně zdrojových souborů poznámek
3. Třídění poznámek — přiřazení poznámky do skupiny a možnost grafického odlišení různých skupin
4. Drag and Drop — možnost přesouvání poznámek metodou „Táhni a pusť“
5. Určení kořenové složky poznámek (místo pro uložení zdrojových souborů poznámek)

1.3 Nefunkční požadavky

1. Desktopová aplikace
2. Kompatibilita s OS Windows
3. Rozšiřitelnost i na další OS např.: Linux
4. Synchronizace zajištěna pomocí ESSS
5. Offline mód aplikace (bez připojení k internetu)
6. Čitelná forma zdrojových souborů poznámek

1.4 Vymezení cílů:

1.4.1 Třídění poznámek do různých skupin

Poznámky budou tříděny do uživatelem definovaných skupin, v Noteru jsou tyto skupiny pojmenovány kategoriemi. Počet kategorií nebude omezen. Uživatel bude mít možnost těmito kategoriím nastavit viditelnost v rámci Noteru a také barvu pozadí poznámek. Barva pozadí poznámek bude sloužit jako hlavní prvek pro jejich odlišení. Název každé kategorie bude jedinečný. S kategoriemi bude souviset i funkce „řazení podle kategorií“, která seskupí poznámky podle jejich kategorií.

1.4.2 Správa jednoduchých poznámek

Aplikace bude pracovat s jednoduchými poznámkami. Bude podporovat editaci textu poznámky, změnu velikosti poznámkového lístku, přiřazení do kategorie a stanovení priority. Z pohledu všech poznámek Noter nabídne seřazení těchto poznámek podle různých kritérií (datum vytvoření, datum poslední změny, kategorie poznámky apod.)

Editace textu poznámky bude umožněna dvěma způsoby. Prvním způsobem bude změna zdrojového souboru poznámky, která se po uložení do souboru ihned projeví v samotné aplikaci, druhým způsobem bude editace pomocí editačního okna aplikace.

Priorita poznámek bude ve třech úrovních. Aby priorita měla smysl a nebylo to jen bezvýznamné číslo, je nutné na ni navázat další funkcionalitu. V Noteru půjde o řazení podle priority a grafické odlišení poznámek různých priorit, kterého bude dosaženo tak, že Noter v závislosti na prioritě zobrazí pozadí záhlaví poznámky červeně (nejvyšší priorita), oranžově (střední priorita) nebo v barvě poznámky (nejnižší priorita).

1.4.3 Sdílení poznámek

V rámci aplikace budou data rozdělena na lokální a sdílená. Toto rozdělení neumožní, aby uživatel mohl ovlivnit nastavení aplikace jiného uživatele.

K samotnému sdílení dat bude Noter využívat ESSS. Jedná se o rozšířený a uživateli vyzkoušený způsob sdílení dat mezi více zařízeními či uživateli. K synchronizaci dat může být využito libovolné ESSS.

1.4.4 Grafické rozhraní aplikace

Pravděpodobnost, že se nový uživatel Noteru v minulosti setkal s aplikací pro správu poznámek, je velmi vysoká. Proto nesmí být v návrhu grafického rozhraní opomenuty zřejmé uživatelské návyky. Mezi návyky lze zařadit dvojklik vyvolávající editaci nebo podpora Drag and Drop. Drag and Drop je způsob přesouvání objektů v GUI¹, kdy uživatel držením levého tlačítka myši může přemístit objekt do cílové oblasti, kde ho puštěním umístí. Podporou těchto způsobů ovládání se podaří zvýšit uživatelskou přívětivost aplikace a vyvolat v uživateli pocit jednoduchého a intuitivního ovládání.

¹Graphical user integrace (GUI) — rozhraní, které umožňuje interakci mezi aplikací a uživatelem

Rešerše již existujících aplikací

Jak jsem již zmínil, Noter není prvním programem pro správu poznámek. Tato oblast je v současnosti zastoupena velkým počtem aplikací, a proto se v rešerši budu zabývat jen těmi, které jsou mezi uživateli nejvíce rozšířeny a také jsou zdarma k použití. Z těchto kritérií vychází následující, abecedně seřazený, seznam:

2.1 Google Keep

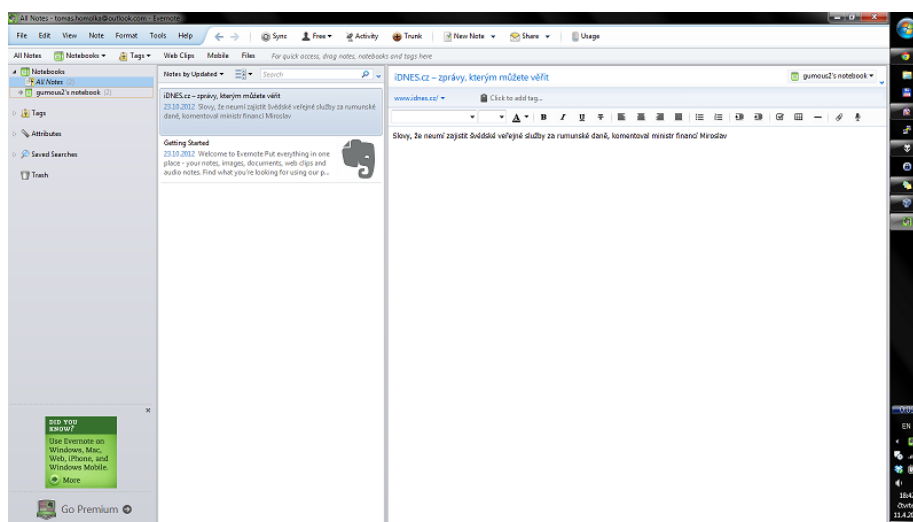
Google Keep je rozšíření známé služby Google Docs. Toto rozšíření Google představil na jaře roku 2013, tedy v době již probíhajícího vývoje Noteru. Google Keep může být chápán jako aplikace pro chytré telefony alternativně přístupná přes webovou stránku. Rešerše se zabývá pouze webovou verzí této služby z důvodu přímé konkurence vůči Noteru, který je navržen jako desktopová aplikace.

Vzhled webové verze je velmi jednoduchý a strohý, vyvolávající dojem pouhého doplňku k verzi pro mobilní telefony, nechybí zde však žádná z funkcionalit. Poznámky jsou zobrazovány v seznamech pod sebou nebo v mřížce, bez možnosti jejich přesouvání. Seznam zde plní funkci kategorie v názvosloví Noteru. Uživatel může pro každý seznam nastavit jednu z osmi přednastavených barev, která je pak zobrazena ve formě tenkého pruhu v záhlaví.

Poznámky mohou být tvořeny obrázky, textem, kombinací textu a obrázků, nebo zvukovým záznamem, který Google Keep dokáže převést do textové formy. Služba podporuje obrázky formátu .jpeg, .gif a .png do velikosti 10 MB.

Webová verze Google Keep nabízí nezávislost na platformě a přístup k aktuálním datům. Záparem této služby je nutnost připojení k internetu.

2. REŠERŠE JIŽ EXISTUJÍCÍCH APLIKACÍ



Obrázek 2.1: Screenshot Evernote

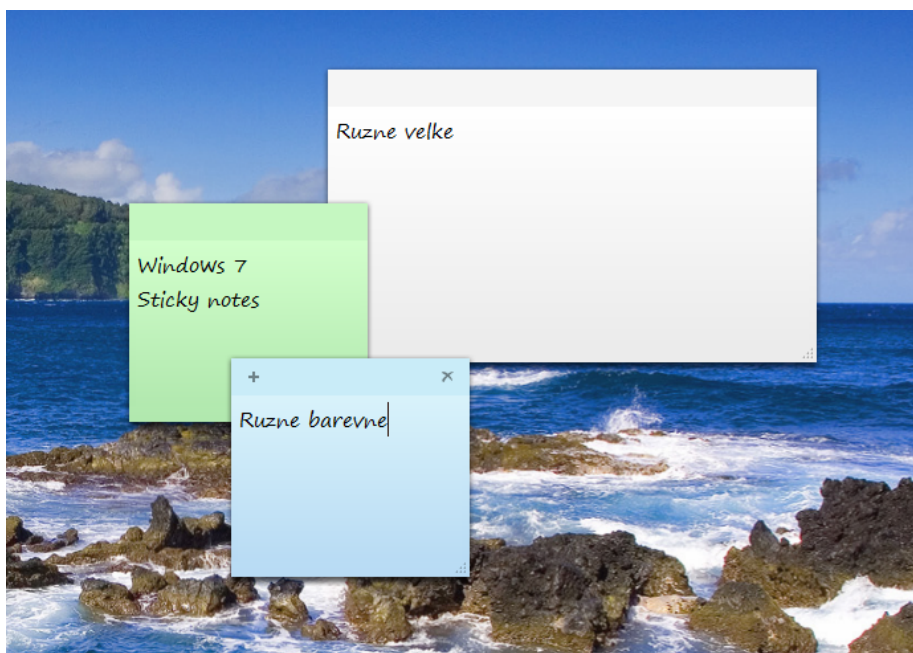
2.2 Evernote

Evernote je velmi propracovaná služba nabízející webovou aplikaci a klienty pro většinu současných operačních systémů. Evernote oficiálně podporuje operační systémy Microsoft Windows, MacOS a mobilní operační systémy Android, iOS, Windows Phone, Blackberry a WebOS[3].

V otázce vzhledu se u Evernotu zaměřím na klienta pro operační systém Windows. Rozložení aplikace je intuitivní, můžeme se s ním setkat například u emailových klientů. Toto rozložení rozděluje okno aplikace na tři svíslé informační oblasti a pracuje s uživatelskou interakcí zleva směrem doprava. Levá oblast je určena pro přehled Notebooků (jiný název pro skupinu poznámek), tagů, atributů, koše nebo notifikaci o upgradu na placenou verzi. V prostřední části je zobrazen seznam poznámek odpovídající vybrané položce z levé oblasti. Poznámka, vybraná v prostřední oblasti, se zobrazí v pravé oblasti, která pak slouží zejména k editaci vybrané poznámky.

Evernote nabízí rozsáhlé možnosti formátování textu poznámky, editační oblast je velmi podobná textovému editoru. V poznámce lze tedy nastavit zarovnání textu, velikost a barvu písma a použitý font s možností kurzívy, tučnosti nebo tučné kurzívy. Evernote podporuje číselné a nečíselné seznamy uvnitř poznámky, dále pak podporuje, podobně jako Google Keep, vkládání obrázků nebo zvukových záznamů do těla poznámky.

Evernote ukládá poznámky i lokálně do archivu, který je aplikací synchronizován s daty uloženými na serverech Evernotu. Tento archiv je editovatelný pouze touto aplikací.



Obrázek 2.2: Screenshot Sticky notes

Výhoda Evernotu spočívá ve velkém množství funkcí a podpoře většiny současných operačních systémů. Nevýhody Evernotu tkví v omezeních FREE verze jako například zobrazování reklamy, absence historie poznámek nebo omezená kapacita pro poznámky.

2.3 Sticky notes

Sticky notes je nativní aplikace operačního systému Windows 7 pro správu poznámek, která vyniká svou jednoduchostí.

Aplikace nemá vzhled klasické okenní aplikace, poznámky jsou volně rozmístěny přímo na ploše a budí tak dojem, že jsou na ni nalepeny.

Co se týká formátování textu poznámek, aplikace nabízí pouze možnost změny velikosti písma. Toho lze dosáhnout pomocí gesta pro přiblížení nebo oddálení, klávesy Ctrl v kombinaci s kolečkem myši. Také správa poznámek je značně omezená. Poznámky jdou libovolně přesouvat po pracovní ploše, nelze je však řadit podle jakýchkoliv kritérií a ani jim přiřazovat kategorie. Jediná možnost grafického odlišení poznámek je změna barvy pozadí jednotlivé poznámky. Barvu pozadí můžeme vybrat z šesti přednastavených barev.

Hlavní předností Sticky notes je jejich již zmíněná jednoduchost. Nenabízí nic víc, než prostor pro napsání krátké poznámky, která je hned po ruce. Největšími nevýhodami jsou závislost na operačním systému a absence sdílení poznámek mezi více zařízeními nebo uživateli.

2.4 TomBoy

TomBoy je aplikace známá mezi uživateli grafického prostředí GNOME, jehož je nedílnou součástí. GNOME je jedno z nejpoužívanějších grafických prostředí pro různé linuxové distribuce a UNIXové operační systémy. Pro tuto aplikaci byla vyvinuta i verze pro operační systém Windows, ve kterém však není tolik rozšířena.

Správa poznámek v této aplikaci je rozsáhlejší než například u Sticky notes, na druhou stranu ale nedosahuje možností aplikace Evernote. TomBoy nabízí třídění do skupin, které nazývá Notebooky. Formátování textu je realizováno pomocí několika přednastavených velikostí písma a základních stylů (tučné písmo, kurzíva apod.), zcela pak chybí volba nastavení barvy pozadí poznámek.

TomBoy je vyvíjen v programovacím jazyce C#, využívá .NET frameworku Mono a grafické knihovny Gtk#, která je součástí většiny linuxových distribucí. GUI TomBoye je proto velmi podobné GUI používanému těmito distribucemi. Rozvržení aplikace pracuje se 2 informačními oblastmi. V levé zobrazuje seznam uživatelových Notebooků (skupin), vpravo pak seznam poznámek v daném notebooku. TomBoy nabízí sdílení pomocí vlastního serveru nebo exportu do dané složky, která může být synchronizována ESSS.

Hlavní nevýhoda TomBoye je v editaci a prohlížení poznámek řešeném otevřením nového okna. Toto je v dnešní době zastaralé a překonané, pro příklad nemusíme chodit daleko. Internetové prohlížeče také zobrazovaly webové stránky v samostatných oknech, pak přišla revoluce zobrazování stránek v panelech (anglicky tab) v rámci jednoho otevřeného okna prohlížeče. Dnes je toto nepsaným standartem současných webových prohlížečů.

2.5 Shrnutí

Shrnutí rešerše je přehledně znázorněno v tabulce 2.1

Tabulka 2.1: Shrnutí řešerše již existujících aplikací

Funkce	Sticky notes	Evernote	TomBoy	Google Keep	Noter
Offline mód	ano	ano	ano	ne	ano
Multiplatformost	ne	ano	ano	ano	ano
Nutná registrace	ne	ano	ne	ano	ne
Vlastní rozložení poznámek	ano	ne	ne	ne	ano
Kategorie	ne	ano	ano	ano	ano
Grafické odlišení poznámek	ano	částečné	ne	částečné	ano

Analýza a návrh implementace aplikace

3.1 Případy užití

V aplikaci Noter se pracuje pouze s jednou rolí aktéra — uživatelem. Na začátek analýzy proto rozeberu uživatelské případy užití:

- Vytvoření nové poznámky
 1. Uživatel zvolí „Přidat poznámku“
 - a) přes menu „Nastavení poznámek“ v horní liště tlačítek
 - b) po kliknutí pravým tlačítkem na plochu aplikace mimo stávající poznámky
 2. Upraví požadované údaje v okně „Přidání nové poznámky“
 3. Dokončí operaci
 - a) stisknutím tlačítka „Uložit“. Aplikace zkontroluje jedinečnost jména, v případě úspěchu poznámku uloží a zobrazí na pracovní ploše aplikace, v opačném případě uživatele upozorní a vyzve jej ke změně jména poznámky
 - b) stisknutím tlačítka „Zrušit“. Aplikace poznámku nevytvoří a veškeré údaje zahodí
 4. Aplikace se přepne zpět na svou pracovní plochu
- Editace stávající poznámky
 1. Uživatel vyvolá zobrazení editačního okna poznámky:

3. ANALÝZA A NÁVRH IMPLEMENTACE APLIKACE

- a) dvojklikem nad danou poznámku
 - b) zvolením „změnit“ po kliknutí pravým tlačítkem myši nad danou poznámku
2. Změní požadované atributy poznámky
 3. Dokončí operaci
 - a) stisknutím tlačítka „Uložit“. Aplikace zkontroluje jedinečnost jména, při úspěchu uloží změny, v opačném případě vyzve uživatele k zadání nového jména
 - b) stisknutím tlačítka „Zrušit“. Aplikace ponechá poznámku v původním stavu a editační okno se zavře
 4. Aplikace se přepne zpět na svou pracovní plochu
- Smazání stávající poznámky
 1. Uživatel smaže poznámku:
 - a) kliknutím na křížek v pravém horním rohu poznámky
 - b) zvolením „Smazat“ po kliknutí pravým tlačítkem myši nad danou poznámku
 2. Aplikace zobrazí potvrzovací okno pro smazání poznámky
 - a) potvrdí-li uživatel svou volbu, aplikace smaže poznámku z pracovní plochy a zdrojovému souboru poznámku změní příponu na .del
 - b) nepotvrdí-li uživatel svou volbu, aplikace ponechá danou poznámku beze změn
 3. Aplikace se přepne zpět na svou pracovní plochu
 - Změna umístění složky obsahující zdrojové soubory poznámek
 1. Uživatel zvolí možnost „Změnit umístění poznámek“ v menu „Nastavení poznámek“
 2. V zobrazeném okně vybere složku pro umístění a
 - a) stisknutím tlačítka „Open“ potvrdí svou volbu. Aplikace přesune všechny zdrojové soubory poznámek do nově zvolené domovské složky
 - b) stisknutím tlačítka „Cancel“ je okno výběru zavřeno a domovská složka zůstává nezměněna
 3. Aplikace se přepne zpět na svou pracovní plochu

- Vytvoření nové kategorie
 1. Uživatel otevře okno „Editace kategorií“ stiskem tlačítka „Editace kategorií“ v horní liště
 2. Stiskem tlačítka „Nová...“ je okno „Editace kategorií“ nahrazeno oknem pro přidání nové kategorie
 3. Uživatel vyplní název nové kategorie, zvolí její viditelnost v rámci aplikace a nastaví barvu pozadí
 - a) Po stisknutí tlačítka „Ulož“ aplikace zkontroluje jedinečnost jména nové kategorie. V případě jedinečnosti jména aplikace vytvoří novou kategorii, jinak je uživatel vyzván ke změně jména nové kategorie
 - b) stisknutím tlačítka „Zrušit“ jsou data zahozena, kategorie není vytvořena
 4. Aplikace se přepne zpět na svou pracovní plochu
- Editace stávajících kategorií
 1. Uživatel stiskne tlačítko „Editace kategorií“ v horní liště tlačítek
 2. Uživatel změní požadované vlastnosti kategorií, může označit kategorii ke smazání
 3. Dokončí operaci stiskem tlačítka
 - a) „Ulož“. Aplikace zkontroluje jedinečnost všech jmen (při neúspěchu uživatele vyzve ke změně duplicitních jmen) a poté uloží změněné hodnoty vlastností kategorií. Uživatelovým potvrzením kategorií určených ke smazání jsou tyto kategorie smazány.
 - b) „Zrušit“. Aplikace neprovede žádné změny kategorií
 4. Aplikace se přepne zpět na svou pracovní plochu
- Seřazení poznámek
 1. Uživatel klikne na tlačítko „Seřazení poznámek“ v horní liště tlačítek a vybere požadované seřazení poznámek
 2. Aplikace seřadí poznámky podle uživatelské volby

3.2 Programovací jazyk

Po definování případů užití uživatele přichází na řadu volba programovacího jazyka, ve kterém bude aplikace implementována. Neexistuje žádný univerzální programovací jazyk, který by byl ideální volbou pro libovolnou implementaci, protože každý programovací jazyk má své přednosti a své nedostatky. Vývojář na počátku implementace musí zvážit všechny okolnosti a možnosti programovacího jazyka pro danou aplikaci a na základě toho rozhodnout o nejvhodnějším řešení.

Výběr programovacího jazyka pro implementaci aplikace Noter jsem provedl z jazyků C++, C#, Java a Python, které jsou aktuálně nejlepší volbou pro desktopové aplikace s vlastním GUI. Všechny tyto jazyky jsou objektově orientované, umožňují vytvářet aplikace pro nejrozšířenější operační systémy jako Windows, Linux nebo Mac OS a existují pro ně grafické knihovny pro tvorbu GUI.

3.2.1 JAVA

Po důkladném prostudování všech aspektů, výhod a nevýhod pro implementaci aplikace Noter jsem vybral jazyk Java. Javu jsem upřednostnil pro:

- snadnou distribuci výsledné aplikace uživatelům
- lepší správu paměti
- mé bohaté předchozí zkušenosti s tímto jazykem

Snadnou distribucí je zde myšlena aplikace v podobě .jar souboru, která pro své spuštění potřebuje pouze nainstalovaný Java Runtime Environment (JRE) v uživatelské počítači. Ten je spravován společností Oracle, která nabízí verze JRE pro většinu současných operačních systémů. V případě operačního systému Windows lze vytvořit spustitelný .exe soubor pomocí nástroje launch4j. Tento nástroj také umožňuje přibalení vlastního JRE k aplikaci, aplikace tak není závislá na přítomnosti JRE v uživatelské počítači. V případě velkého množství linuxových distribucí je JRE součástí základní instalace.

Pro správu paměti Java využívá nástroje Garbage collector (GC). Garbage v informatice jsou objekty, data nebo jiné úseky paměti, které již nebudou programem v budoucnosti použity. GC tyto garbage rozpozná a jejich uvolněním získá zpět paměťové prostředky pro další použití[4].

3.2.2 C++

Další možností byl jazyk C++, který jsem nevybral z důvodu absence výše zmíněného garbage collectoru. Programátor má sice větší volnost pro práci s pamětí, ale také více starostí s jejím následným uvolňováním.

Největší výhodou C++ je rychlost výsledné aplikace, ale v případě Noteru není rozdíl v rychlostech těchto dvou jazyků natolik velký, aby byl směrodatný ve výběru jazyka.

C++ je multiplatformní jazyk, ale je zde nutnost kompilace kódu kompilátorem pro danou platformu. Přidáním grafické knihovny pro vytvoření GUI se tato kompilace ještě zesložití.

3.2.3 C#

C# vznikl jako odpověď společnosti Microsoft na komerční úspěchy programovacího jazyka Java. Pro účely Noteru není mezi těmito dvěma jazyky výraznějších rozdílů, například oba dva používají GC pro správu paměti a jsou multiplatformní. O výběru Javy na úkor C# tedy nakonec rozhodly mé předchozí zkušenosti s jazykem Java a nutnost instalace C# knihoven, například v OS Windows nepříliš rozšířené grafické knihovny Gtk#.

3.2.4 Python

V programovacím jazyce Python jsem nikdy nepracoval a to ho v tomto výběru velmi znevýhodnilo oproti ostatním. Mezi další nevýhody tohoto jazyka patří nižší rychlost v porovnání s Javou (a tedy i ostatními) a vyšší paměťová náročnost, které jsou způsobené tím, že Python mimo jiné využívá vysokoúrovňové datové typy[2]. Na druhou stranu je díky tomu vývoj aplikací v Pythonu rychlejší a intuitivnější, tímto ušetřený čas na vývoji Noteru bych však musel použít na seznámení se s tímto jazykem.

3.3 Formát zdrojových souborů

Zdrojové soubory poznámek jsou základním prvkem aplikace. Zadání této práce požaduje, aby tyto soubory byly čitelné pomocí jednoduchého textového editoru. Tuto vlastnost splňují základní formáty s příponami .txt a .xml. Textový formát je pro Noter vhodnější díky své menší paměťové náročnosti. Zvolením tohoto formátu sice vzniká nutnost vlastních parametrů, jejichž nízký počet neznamená výhodu pro zbytečně rozsáhlý formát XML

Pro potřeby Noteru byly zvoleny parametry text, priorita, datum vytvoření a datum poslední změny. Neobsahuje-li zdrojový soubor tyto parametry, Noter přiřadí obsahu textového souboru parametr text a zbylé parametry do souboru sám doplní, parametrům datum vytvoření a datum poslední změny nastaví aktuální datum a nastaví nejnižší prioritu. Pro parametry byly zvoleny identifikátory:

- #TEXT následovaný textem poznámky až po další parametr,
- #DATE následovaný datem vytvoření poznámky ve tvaru „YYMMDDhhmmss“,
- #DATECHANGE následovaný datem poslední změny poznámky ve stejném tvaru jako #DATE,
- #PRIORITY následovaný číselnou hodnotou reprezentující prioritu poznámky (kde 0 znamená nejnižší, 1 střední a 2 nejvyšší prioritu).

3.4 Uložení dat v aplikaci

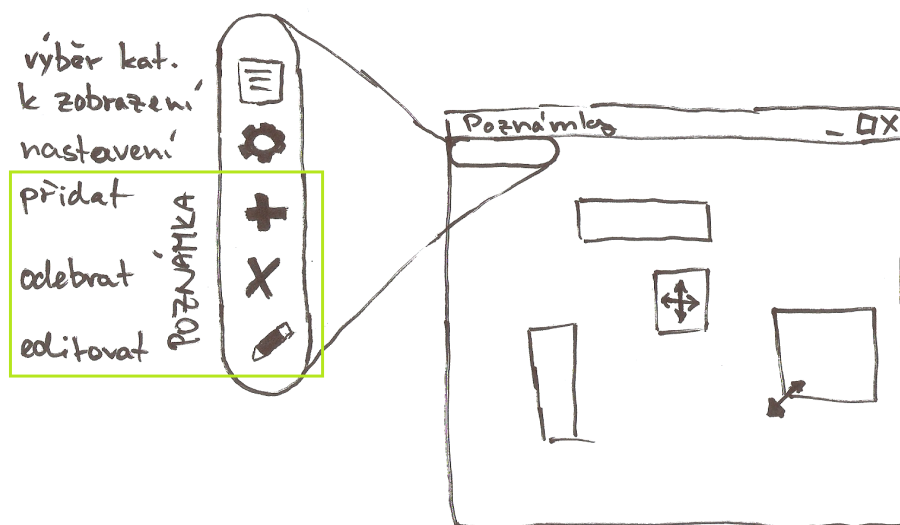
Poznámky budou v aplikaci implementovány pomocí objektu Note, který bude obsahovat všechny důležité informace o poznámce v podobě atributů tohoto objektu. Obdobně bude vytvořen objekt Category pro správu informací o kategoriích. Každý objekt Note bude obsahovat referenci na objekt Category jako jeden ze svých atributů. Objekt Category bude poté obsahovat celočíselný atribut informující o počtu objektů Note, které na něj odkazují.

Noter bude též obsahovat třídu Config obsluhující jeho aktuální nastavení. Změny nastavení budou ukládány do souboru, ze kterého budou při spouštění aplikace načteny.

3.5 Synchronizace dat

Noter bude pracovat se dvěma druhy dat, a to s daty lokálními a sdílenými.

Jedinými dvěma zástupci lokálních dat budou viditelnost kategorií, zvolené vizuální téma a rozmístění poznámek v aplikaci. Tato data není dobré sdílet, protože přímo ovlivňují chování poznámek v aplikaci. Uživatelé, kteří mezi sebou sdílí poznámky, se nemusí shodovat v ní jejich rozmístění nebo viditelnosti různých kategorií. Uživatelovi tak bude znemožněno, aby změnou svého nastavení měnil nastavení i jiných uživatelů.



Obrázek 3.1: První vize GUI aplikace

Mezi sdílená data budou patřit zdrojové soubory poznámek a barva pozadí kategorií. Díky tomu bude dosaženo aktuálnosti obsahu sdílených poznámek.

Spolupráce Noteru s ESSS je postavena na rozdělení kompetencí. Noter se bude starat o interpretaci zdrojových souborů a dalších parametrů do podoby poznámek a ESSS zase o to, aby sdílená data byla bezpečnou a nejrychlejší cestou přenesena na jiné sdílené zařízení, popř. k jinému uživateli, se kterým jsou tato data sdílena. Na Noter je tedy kladen požadavek na možnost volby umístění složky, kam se budou ukládat všechna sdílená data.

3.6 Vizuální podoba

Vytvoření přívětivého GUI je věc složitá, každý uživatel má jinou představu o přívětivém vzhledu a osobitý vkus. GUI aplikace se tedy může jednomu uživateli zamlouvat a jinými být zatracena. V aplikaci Noter byl zvolen minimalisticky funkční design, tedy bez zbytečných grafických elementů, které mohou na první pohled zaujmout, ale při delším používání uživatele rozptylují a ruší od splnění jeho cíle.

Návrh GUI začal jednoduchými náčrty na papíře pro lepší pochopení souvislostí mezi ovládacími prvky (viz. obrázek 3.1. Tento prvotní návrh byl konzultován s potenciálními uživateli. Tyto konzultace byly velmi přínosné, díky nim se podařilo získat cenné pohledy na způsob interakce

uživatelů s ovládacími prvky a též řadu inspirujících nápadů. Původní návrh byl poté podle zpětné vazby upraven. Například původní koncept úpravy poznámky, tzn. uživatel označí poznámku, kterou chce upravit a klikne na společné tlačítko editace v liště aplikace, jsem zamítnul, jelikož ani jeden z uživatelů na tento sled úkonů nepřišel.

3.6.1 Vizuální podoba aplikace

Horní lišta aplikace Noter nahrazuje klasické záhlaví okenní aplikace, známé z většiny současných operačních systémů, obsahující tlačítka pro minimalizaci, změnu velikosti okna a zavření aplikace. Díky této odlišnosti vypadá Noter ve všech OS shodně, nemusí řešit například rozdílné umístění tří ovládacích tlačítek v OS Windows oproti Ubuntu s grafickou nástavbou Unity, která zobrazuje tato tlačítka v levém horním rohu okna aplikace.

Noter nabízí dva základní vzhledy. První, defaultně nastavený, imituje korkovou tabuli s dřevěným rámem. Tento vzhled dává na první pohled najevo účel této aplikace. Druhý vzhled je více moderní, avšak stejně jako předchozí vzhled má i tento předobraz v reálném světě, a to v magnetické tabuli.

Prostředí samotné aplikace je rozděleno do dvou částí, horní lišty s tlačítky a pracovní plochy, kde se zobrazují poznámky. Horní lišta se skládá ze 6 tlačítek.

První je tlačítko pro volbu vzhledu aplikace.

Druhé tlačítko, nazvané Nastavení poznámek, umožňuje vytvoření nové poznámky a změnu umístění poznámek. Umístění poznámek je reprezentováno složkou v uživatelské počítači, kam Noter ukládá všechna data určená ke sdílení.

Třetí tlačítko otevře okno pro editaci kategorií a čtvrté tlačítko nabídne uživateli seřazení poznámek na pracovní ploše podle kritérií: datum vytvoření, datum poslední změny, název poznámky, priorita poznámky a také seskupení poznámek do větších shluků podle jejich kategorií.

Skupina prvních čtyř tlačítek je umístěna na levé straně, zbývající dvě jsou v pravém horním rohu. Jedná se o tlačítko pro přepnutí do režimu celé obrazovky a tlačítko pro vypnutí aplikace.

Pracovní plocha aplikace slouží k zobrazení poznámek. Aplikace si pamatuje pořadí poznámek i v ose Z a umožňuje jejich vzájemné překrývání s ohledem na toto pořadí. Při kliknutí na konkrétní poznámku je přenesena do popředí. Posouvání poznámek, jež mají jednoduchým stínem a postupným ztmavením směrem dolů připomínat přilepený lístek, je velmi přirozené díky podpoře Drag and Drop, stejně jako v reálném životě uživatel chytne lístek a přesune ho.

3.6.2 Vizualní podoba poznámek

Po vizualní stránce lze poznámku rozdělit na informační a textovou část. Obě části jsou graficky odděleny vodorovnou čarou.

V informační části najdeme název poznámky a v pravém horním rohu tlačítko pro smazání. Název poznámky slouží mimo jiné i jako grafické znázornění priority. S nejnižší prioritou je písmo názvu stejné jako písmo pro text, při střední prioritě je název napsán tučně a při nejvyšší prioritě je název tučný v červené barvě pro upoutání uživatelské pozornosti.

V textové části bude text zalamován s ohledem na šířku poznámky. Text bude zalamován pouze podle mezer, Noter nebude umožňovat zalamování například po slabikách. Pokud je název nebo nejdelší slovo poznámky delší než její šířka, pak je text nahrazen posloupností znaků „(...)“ značící uživateli, že není zobrazen celý.

V pravém dolním rohu je vyznačena oblast, pomocí které lze libovolně roztahovat poznámku. Velikost poznámky je omezená minimální šířkou, minimální výškou a maximální šířkou. Maximální výška poznámky je omezena pouze výškou monitoru, za předpokladu nekonečně vysokého monitoru můžeme teoreticky prohlásit, že Noter zobrazí nekonečně dlouhý text poznámky.

Popis realizace implementace

Aplikace Noter je postavena na architektuře Model-View-Controller (MVC) rozdělující aplikaci do tří komponent: datový model, grafické uživatelské rozhraní a řídicí logika. Tato architektura usnadňuje vývoj, implementaci a následné testování aplikace. Také nabízí vysokou modularitu aplikace, díky které můžeme upravovat danou komponentu s minimálním ovlivněním ostatních komponent, a odděluje data od jejich grafické reprezentace v aplikaci.

4.1 Model

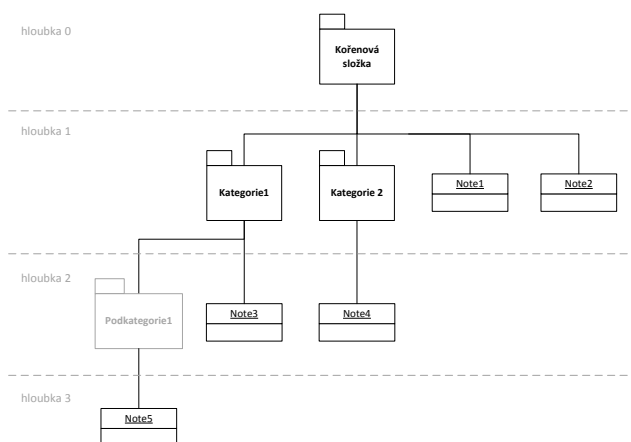
Model je vrstva reprezentovaná daty, se kterými aplikace pracuje, a metodami, které tato data obsluhují.

V rámci Noteru se jedná o objekty Note a Category a třídy Monitor, NoteColorUtility a Config, který obsluhuje uložení, nahrání a změnu nastavení aplikace.

4.1.1 Třída Monitor

Třída Monitor kontroluje kořenovou složku poznámek. Při vytvoření, editaci nebo smazání textového souboru nebo složky vyvolá příslušnou metodu pro danou akci. K tomuto účelu implementuje FileAlterationListenerAdaptor z knihovny commons-io-2.4 [1], zejména pak jeho metody onDirectoryCreator(File file), onDirectoryDelete(File file), onFileCreate(File file), onFileChange(File file) a onFileDelete(File file).

Určíme-li kořenovou složku poznámek jako kořen datové struktury stromu, poté uzly v hloubce rovno jedné jsou kategoriemi a kořenový uzel je základní kategorií. Složky s hloubkou vyšší než jedna jsou v Noteru považo-



Obrázek 4.1: Grafické znázornění dat kontrolovaných třídou Monitor

vány za podkategorie bez dalšího využití v aplikaci. Třída Monitor pracuje do hloubky stromu rovno třem (viz obrázek 4.1).

4.1.2 Třída NoteColorUtility

Pro práci s barvami v GUI aplikace Noter je využíváno balíku `java.awt.color`. Tento balík bohužel neposkytuje pokročilejší funkce pro vzájemné převody mezi `awt.color` a stringem, vytvořil jsem proto třídu `NoteColorUtility`. Jejími metodami jsou `getColorIcons`, `getColorIndex`, `getColorString`, `getColors` a `stringToColor`.

Metoda `getColor` vrací pole názvů jednotlivých barev, Noter podporuje 19 přednastavených barev. Metoda `getColorIcons` namísto názvů barev vrací pole ikon pro dané barvy.

Metody `getColorIndex` a `getColorString` prochází pole názvů barev a vrátí index nebo název barvy, kterou přijali jako vstupní parametr.

Poslední metoda, `stringToColor`, převede řetězec na barvu pokud je podporována. Pokud ne, vrátí bílou barvu.

4.2 View

View je komponenta převádějící data do uživateli přijatelné grafické formy. Tato komponenta spravuje veškeré úkony spojené s grafikou a grafickým prostředím, a to pomocí grafické knihovny Swing, jež je součástí JRE.

Komponenta View v aplikaci Noter obsahuje vše, co se týká vykreslování grafiky. Můžeme zde nalézt třídy vykreslující různé části aplikace, DrawBorder pro vykreslení okrajů a horní tlačítkové lišty, DrawNote pro vykreslení poznámky, PopUpMenus pro vytváření a zobrazování vyskakovacích menu nebo třídy zajišťující zobrazení popisků tlačítek v horní liště.

4.2.1 Třída DrawNote

Třída DrawNote vykresluje poznámku na pracovní plochu aplikace. Vykreslování je implementováno pomocí metod abstraktní třídy Graphics, jež je součástí grafické knihovny Abstract Window Toolkit (AWT), která je předkem grafické knihovny Swing.

Nejdříve je vykreslen stín poznámky, který je tvořen obrázky dolního stínu, pravého stínu a rohu mezi nimi. Tyto obrázky jsou tvořeny barevnými přechody z černé do bílé s transparentním pozadím a 50 % celkovou průhledností obrázku. Použití tří obrázků namísto jednoho eliminuje deformaci stínu způsobenou změnou velikosti poznámky.

Nad stíny je vykreslen obdélník pomocí metody fillRect. Tento obdélník je vykreslen v barvě nastavné v kategorii dané poznámky. Přes tento obdélník je následně vykreslen obrázek s přechodem z průhledné barvy do černé se 70 % celkovou průhledností. Přechod dodává poznámce reálnější a plastičtější dojem, jako kdyby byla k pracovní ploše aplikace přilepena ve své horní části.

Název a text poznámky, spolu s dělicí linkou mezi nimi, jsou vykreslovány pomocí metod drawString, která je použita pro text, a drawLine, která vykresluje oddělovací linku.

4.2.2 Třída DrawBorder

DrawBorder vykresluje ohraničení celé aplikace v případě zvoleného vzhledu „Korková tabule“ nebo horní lištu ve vzhledu „Whiteboard“. Dále pak vykresluje ovládací tlačítka aplikace, kterých je šest. Každé tlačítko je tvořeno dvěma obrázky se stejnou velikostí a pozicí. První z nich je vykreslován, pokud je tlačítko v normálním stavu. Druhý, výraznější obrázek je vykreslován, pokud se nad tlačítkem nachází kurzor. Platí, že vždy je vykreslován pouze jeden z těchto obrázků.

Pro tlačítka by bylo vhodnější použít komponentu JButton, ovšem možnosti správce rozložení byly pro tento účel nedostačující. Bylo proto nutné implementovat tlačítka pomocí vykreslování obrázků a definováním ovládacích oblastí v NoteMouseListeneru. Popisky, zobrazující se po najetí na

tlačítko, jsou tvořeny komponentou JFrame. Této komponentě se podle situace nastavuje viditelnost a zobrazovaný text.

4.3 Controller

Controller je komponenta zpracovávající uživatelské vstupy a v závislosti na těchto vstupech upravuje data. V této aplikaci je k tomu využito zejména tříd `NoteMouseMotionListener` a `NoteMouseListener`. Díky těmto dvěma listenerům je aplikace schopna pracovat s polohou kurzoru a dle potřeby na ni reagovat. Komponenta dále obsahuje třídu `Confirms`, požadující potvrzení uživatelských akcí, a třídu `Controller`, která obsahuje jednoduché podpůrné metody pro práci s poznámkami, např. přenesení poznámky do popředí.

4.3.1 NoteMouseMotionListener

Tento listener zpracovává pohyby uživatelského kurzoru pomocí dvou metod, `mouseMoved` a `mouseDragged`.

`MouseMoved` upravuje příznaky nesoucí informace o poloze kurzoru. Podle těchto příznaků se řídí například metoda `DrawBorder` při vykreslování ovládacích tlačítek.

`MouseDragged` spravuje proměnné potřebné pro správné fungování `Drag and Drop` způsobu přesouvání poznámek.

4.3.2 NoteMouseListener

`NoteMouseListener` implementuje rozhraní třídy `MouseListener`; v aplikaci `Noter` jsou překryty metody `MouseClicked` a `MousePressed`. Na první pohled není mezi nimi patrný rozdíl. Odlišují se způsobem volání. Metoda `MousePressed` je volána v okamžiku pouhého stlačení tlačítka, oproti tomu metoda `MouseClicked` je volána až v okamžiku puštění tlačítka po jeho stlačení. Z toho plyne i určení těchto dvou metod, v `Noteru` je metoda `MousePressed` překryta pro potřeby podpory `Drag and Drop`. `MouseClicked` obsluhuje funkční prvky aplikace, tedy určí, kde uživatel kliknul a to, jaká metoda má být zavolána.

Testování aplikace a analýza výsledků testování

Kladné přijetí Noteru jeho uživateli velkou měrou závisí na použitelnosti a intuitivně jeho GUI. Pokud bude uživatel zmaten grafickým prostředím, vytvoří si o aplikaci negativní mínění, které aplikace nezlepší sebelepší funkcionalitou. Z tohoto důvodu jsem věnoval velkou pozornost testování aplikace z pohledu uživatele pro minimalizaci jeho případné chybovosti způsobené nejasným rozhraním. Využil jsem tří testovacích metod: dotazník, betatestování a test kompatibility.

5.1 Dotazník

Dotazník se soustředí na hromadný sběr dat. Využívá k tomu jasně definovaných otázek s konkrétním výběrem odpovědí. Větším souborem dat tato metoda eliminuje subjektivitu dotazovaných, proto je vhodná pro zkoumání vnímání vzhledu uživateli. Výstupem jsou data, která lze generalizovat. Pro tvorbu dotazníku jsem využil služby Google Docs, která nabízí vytvoření jednoduchého dotazníku, lehkou distribuci k dotazovaným a snadné zpracování získaných dat.

V rámci dotazníku jsem se tázal hlavně na vzhled aplikace, její použitelnost a vybavenost uživatelů synchronizačními službami. Celkový počet respondentů byl 29 s věkovým rozptylem od 17 do 45 let a třetinovým zastoupením ženského pohlaví.

V otázce vzhledu lze z dat vypožorovat zajímavý trend, kdy se většině dotazovaných líbil buď jeden, nebo druhý vzhled aplikace. Pouze třem lidem se oba vzhledy líbily stejnou měrou. V celkovém počtu je lépe hodno-

cen vzhled „korková tabule“, který dosáhl průměrné známky 2.413, druhý vzhled „whiteboard“ dosáhl známky 2.750.

Z dotazníku také vyplývá, že 89 % procent dotázaných má předchozí zkušenost s aplikacemi pro správu poznámek, mezi kterými jasně dominují Sticky notes z OS Windows.

V současné době lze spatřovat zvyšující se oblibu synchronizačních služeb, což je tímto dotazníkem potvrzeno. Celých 82 % dotázaných aktivně používá síťovou synchronizační službu, 66 % uživatelů synchronizační služby by využilo možnosti sdílení poznámek.

5.2 Betatestování

Betatestování je testovací technika prováděná na samém konci vývojového cyklu. Betatestování se provádí s omezeným počtem uživatelů, kteří testují aplikaci jako v reálném nasazení. Díky různým konfiguracím počítačů, verzím operačních systémů a rozdílným uživatelským zkušenostem jsou nalezeny chyby, které se projevují jen v určitých kombinacích reálného prostředí.

Betatestování odhalilo čtyři kritické chyby, které způsobovaly pád aplikace nebo její zamrznutí. Přehled kritických chyb se nachází v tabulce 5.1. Také bylo objeveno množství malých chyb, které přímo neovlivňovaly samotný běh programu. Jednalo se například o nesprávné zobrazení tlačítek nebo nesrozumitelné popisky.

5.3 Test kompability

V rámci tohoto testu byl Noter spouštěn na několika operačních systémech, byli to Microsoft Windows ve verzích 7 a 8 a dále pak linuxové distribuce založené na jádře Ubuntu, konkrétně Kubuntu 13.04, Linux Mint 14.1 a Ubuntu 12.04.

Tabulka 5.1: Tabulka kritických chyb

Chyba	Operační systém	Příčina	Řešení
Zamrznutí aplikace při pokusu o uložení poznámky	Linux	Použití znaku „ / “ jako oddělovače souborů	Oddělovač nastaven oddělovač nastaven dle OS
Po vypnutí klávesovou zkratkou Alt +F4 nejde aplikacemi znovu spustit	Všechny	Tímto příkazem bylo zavřeno pouze okno aplikace, samotná aplikace běžela dál	Nastavení standardního vypnutí aplikace při zavření okna
Posloupnost úkonů: „Správa kategorií“, „Přidání nové kategorie“, zavření okna vede k zamrznutí aplikace	Všechny	Otevření okna „přidání nové kategorie“ udělá hlavní okno pasivním, při zavření editačního okna	Zaktivování hlavního okna po zavření jiného okna
Nefunkční spouštěcí skript	Linux	Špatný parametr příkazu uvnitř skriptu	Oprava parametru

Srovnání s již existujícími aplikacemi

V této kapitole se budu věnovat srovnání aplikace Noter s již existujícími aplikacemi pro správu poznámek, jmenovitě to jsou Evernote, Google Keep, Sticky notes a TomBoy. V tomto srovnání budu postupovat podle klíčových vlastností aplikací pro správu poznámek:

6.1 Nezávislost na operačním systému:

Podíváme-li se na aplikace z pohledu jejich závislostí na dané platformě, můžeme spatřit velmi podobné parametry s výjimkou Sticky notes. Sticky notes jsou pevně provázány s operačním systémem Windows, který je tak jediným operačním systémem, kterým lze tuto aplikaci spustit. V tomto případě tedy nelze mluvit o nezávislosti na operačním systému.

Zbývající aplikace jsou nezávislé na spuštěné platformě. Evernote nabízí klienty pro mnoho operačních systémů, TomBoy je ve verzích pro Windows a pro Linux a Google Keep je webová služba spuštěná v internetovém prohlížeči, tedy je naprosto nezávislá na operačním systému.

Noter ke svému spuštění potřebuje nainstalované JRE na uživatelské počítači. JRE je v dnešní době velmi rozšířeno napříč operačními systémy.

6.2 Rozložení poznámek v aplikacích

Zaměříme-li se na způsob rozložení poznámek v aplikaci, můžeme najít podobnosti (Noter a Sticky notes), ale také zcela odlišné přístupy k řešení této problematiky.

Noter je v tomto ohledu velmi podobný aplikaci Sticky notes, kde obě aplikace zobrazují poznámky v podobě jejich skutečného předobrazu, tedy Post-It. S těmito poznámkami lze pak libovolně hýbat a umístit je tak kdekoliv podle potřeby, jde o velmi intuitivní model se vzorem v reálném světě.

Dalšími aplikacemi s podobným způsobem rozložení poznámek, jsou Evernote a TomBoy. Můžeme si zde všimnout rozdělení hlavního okna na informační oblasti, ve kterých jsou zobrazována data v hierarchické struktuře.

Google Keep přišel s vlastním rozložením. Oproti ostatním aplikacím zde není poznámce přiřazen vlastní grafický objekt, nýbrž je poznámka jen částí většího seznamu. Tyto seznamy lze zobrazovat do řádků pod sebe nebo do mřížky vedle sebe.

6.3 Možnosti ukládání a sdílení poznámek

Aplikace se liší ve způsobu uložení poznámek do souborů a také v následném sdílení těchto souborů.

Sticky notes nabízí základní vlastnosti. Všechny poznámky ukládají do jediného zdrojového souboru a nativně nenabízí možnost sdílení poznámek. Na internetu lze nalézt různé postupy synchronizace zdrojového souboru poznámek ze Sticky notes, ovšem tyto postupy jsou pro běžné uživatele složité z důvodu vytváření symbolických odkazů vedoucích ze složky Dropboxu na zdrojový soubor. Sticky notes též nejsou připraveny na tuto možnost, objevují se proto problémy, kdy například Dropbox nemůže úspěšně dokončit synchronizaci kvůli častému zápisu do zdrojového souboru poznámek.

Aplikace TomBoy dovoluje uživateli zvolení si složky v počítači, do které bude poté ukládat zdrojové soubory poznámek. TomBoy, podobně jako Noter, ukládá každou poznámku do samostatného souboru, a to ve formátu XML. TomBoy nabízí synchronizaci poznámek přes vlastní server, zarážející je ale fakt, že nikde není vyžadováno žádné uživatelské jméno spojené s heslem. Sdílet poznámky v TomBoy lze podobným způsobem jako v Noteru, tedy nastavit ukládání poznámek do složky synchronizované pomocí ESSS. TomBoy ovšem neukládá změny při každé změně v poznámkách, proto toto sdílení poznámek také není okamžité.

Evernote synchronizuje poznámky pomocí vlastního serveru, změny jsou uloženy v rámci desktopového klientu a se serverem mohou být automaticky synchronizovány v uživatelem nastaveném intervalu. Tento interval může být nastaven na 15 minut, 30 minut, 60 minut a jeden den. Uživatel má možnost okamžité synchronizace se serverem. Veškeré používání Ever-

notu může provádět pouze přihlášený uživatel, riziko odcizení dat ze serveru Evernote je tedy nižší než u aplikace TomBoy.

Google Keep uchovává zdrojové soubory poznámek na svých serverech bez možnosti přístupu k nim jiným způsobem, než pomocí této služby. Synchronizace poznámek je okamžitá, protože veškeré změny jsou ukládány přímo na serveru.

Noter v sobě spojuje klíčové vlastnosti předešlých aplikací. Nabízí offline mód, tedy přístup k poznámkám bez připojení k internetu, synchronizaci pomocí libovolné ESSS a možnost prohlížení zdrojových souborů poznámek ve formě čitelné pro člověka. Pro přehlednost ve zdrojových souborech poznámek Noter tyto soubory pojmenovává názvem poznámky v aplikaci.

Zhodnocení splnění cílů

7.1 Splnění funkčních požadavků:

1. Správa poznámek — Noter umožňuje vytvoření, editaci a odstranění poznámky
2. Synchronizace poznámek — Noter kontroluje zdrojové soubory poznámek a při změně těchto souborů aktualizuje data zobrazovaná uživateli
3. Třídění poznámek — uživatel může seřadit své poznámky do matice nebo do skupin podle kategorií.
 - a) Řadit poznámky do matice lze podle data vytvoření, data poslední změny, priority a názvu poznámky. Pracovní plocha aplikace je rozdělena na potřebný počet oblastí tak, aby velikost těchto oblastí byla stejná a zároveň největší možná. Velikost poznámek je v případě potřeby upravena tak, aby nebyla větší než velikost oblast jí přiřazené. Výsledkem jsou v matici seřazené poznámky, kde se žádné dvě poznámky navzájem nepřekrývají
 - b) Řazení poznámek do skupin podle kategorií vytvoří skupiny poznámek. Skupiny jsou zobrazeny na pracovní ploše vedle sebe, šířka poznámek je případně zmenšena. Poznámky v rámci skupiny jsou skládány na sebe, každá další poznámka je posunuta směrem dolů o výšku názvu předchozí poznámky ze skupiny. Výsledkem je přehledné zobrazení všech poznámek podle jejich kategorií, názvy poznámek jsou viditelné, čímž je ulehčeno hledání konkrétní poznámky

7. ZHODNOCENÍ SPLNĚNÍ CÍLŮ

4. Drag and Drop — pro přesun poznámek po pracovní ploše je využíváno metody Drag and Drop
5. Určení kořenové složky poznámek — uživateli je umožněno nastavení kořenové složky poznámek. Po změně této složky Noter zkopíruje potřebná data do nově zvolené složky a původní data odstraní.

Co se týká funkčních požadavků, aplikace Noter všechny splňuje v plné míře.

7.2 Splnění nefunkčních požadavků:

1. Desktopová aplikace — Noter je od samého začátku vyvíjen jako desktopová aplikace
2. Kompatibilita s OS Windows — Noter nevyžaduje instalaci, v systému Windows je spouštěn spustitelným .exe souborem. Jediný problém může nastat v okamžiku, kdy na cílovém počítači není nainstalován JRE (v dnešní době je toto riziko nízké). Z tohoto důvodu byla vytvořena verze obsahující vlastní JRE. Tudíž i uživatel bez nainstalovaného JRE může Noter používat
3. Rozšířitelnost i na další OS, např. Linux — Noter je v spustitelný i v operačním systému Linux v podobě .jar souboru
4. Synchronizace zajištěna pomocí ESSS — vybráním kořenové složky poznámek (Funkční požadavek 5) je uživateli umožněno přesunutí sdílených souborů do složky synchronizované ESSS
5. Offline mód aplikace — Noter není nijak závislý na ESSS a ani nevyžaduje připojení k internetu pro svůj běh
6. Čitelná forma zdrojových souborů poznámek — poznámky jsou ukládány do textového formátu .txt, a to v čitelné formě

7.3 Vymezené cíle aplikace

7.3.1 Třídění poznámek do různých skupin:

Třídění do různých skupin je realizováno pomocí kategorií. V aplikaci lze kategoriím nastavit jejich viditelnost, barvu pozadí a název kategorie, který

musí být jedinečný (aplikace tuto jedinečnost kontroluje a nedovoluje změnu názvu kategorie na již použitý název).

Pokud budeme brát poznámku jako soubor v počítači, pak s kategorií budeme pracovat jako se složkou. Poznámky patřící do dané kategorie jsou ukládány do společné složky, jejíž název se shoduje s názvem kategorie. Tuto složku můžeme sdílet pomocí ESSS a tím dosáhnout sdílení jedné kategorie.

7.3.2 Správa jednoduchých poznámek

Noter umožňuje vytvoření, editaci a smazání poznámek. Všechny tyto akce respektují zažitá uživatelské návyky a jsou přístupné i přes položky menu, tudíž uživatel neznalý těchto návyků není v aplikaci ztracen a je schopen dokončit požadovanou akci.

Aplikace reaguje na změny ve zdrojových souborech poznámek s prodlevou činící maximálně 5 sekund, uživatel tedy pracuje s aktuálními daty.

7.3.3 Sdílení poznámek

Ze srovnání s již existujícími aplikacemi (Kapitola 6) vyplývá, že tento cíl byl splněn a Noter tedy umožňuje sdílení poznámek pomocí ESSS.

7.3.4 Grafické rozhraní aplikace

Pro aplikaci Noter jsem navrhl vlastní grafické uživatelské rozhraní se vzhledem připomínající korkovou tabuli. Nad rámec zadání jsem vytvořil i druhý vzhled, který má být alternativou, která je méně nápadná a neruší tak citlivější jedince při práci s aplikací.

Závěr

V rámci této bakalářské práce jsem vytvořil první verze multiplatformní desktopové aplikace pro správu krátkých textových poznámek. Samotné realizaci předcházela podrobná rešerše nejpoužívanějších aplikací v této kategorii. Výsledkem je konkurenceschopná alternativa k těmto zavedeným aplikacím, mezi které přichází jako aplikace s vlastním uživatelským rozhraním, možností okamžitého sdílení poznámek pomocí ESSS a zdrojovými soubory poznámek, které je možné upravovat pomocí jakéhokoliv textového editoru.

Hlavním přínosem pro mne samotného je prohloubení zkušeností s programovacím jazykem Java, získání znalostí o práci s grafickou knihovnou Swing a v neposlední řadě jsem si prakticky vyzkoušel vývoj aplikace od jejího návrhu až k její realizaci. Všechny tyto zkušenosti a znalosti určitě dále využiji ve svém studentském a profesním životě.

Návrhy na budoucí vývoj a zlepšení aplikace

Budoucí vývoj aplikace lze rozdělit do dvou rovin, a to vývoj mobilního klienta a integrace ESSS:

- Evernote a Google Keep mají konkurenční výhodu oproti Noteru v podobě klientů pro mobilní operační systémy jako Android nebo iOS. Noter v současné podobě nabízí možnost vytváření poznámek v mobilních telefonech díky mobilní verzi aplikace Dropbox, která umí vytvořit textový soubor. Tento stav je spíše provizorní a uživatelsky ně příliš komfortní. Vytvoření mobilního klienta by tak velmi rozšířilo možnosti využití Noteru.

- Integrace ESSS přímo v aplikaci Noter neexistuje z důvodu požadavku na zachování volby ESSS na uživateli. Zajímavou volbou pro budoucí vývoj je tedy integrace několika ESSS přímo do aplikace s možností nastavení sdílení přímo v aplikaci. Tím by se výrazně zlepšil komfort uživatelů požadujících sdílení.

Literatura

- [1] Commons IO - Commons IO Overview. [cit. 2013-05-14]. Dostupné z: <http://commons.apache.org/proper/commons-io/>
- [2] Comparing Python to Other Languages. [cit. 2013-05-14]. Dostupné z: <http://www.python.org/doc/essays/comparisons/>
- [3] The Evernote family of products | Evernote. [cit. 2013-05-14]. Dostupné z: <http://evernote.com/products/>
- [4] Garbage collection (computer science). [cit. 2013-05-14]. Dostupné z: [http://en.wikipedia.org/wiki/Garbage_collection_\(computer_science\)](http://en.wikipedia.org/wiki/Garbage_collection_(computer_science))
- [5] History: 3M UK & Ireland. [cit. 2013-05-14]. Dostupné z: http://solutions.3m.co.uk/wps/portal/3M/en_GB/Post-Its/Post-It/Solutions/History/

Seznam použitých zkratk

AWT Abstract Windows Toolkit

ESSS externí služba pro sdílení souborů

GUI Graphical user interface

JIT Just In Time Compilation

JRE Java Runtime Envirotment

MVC Model-View-Controller

OS operační systém

XML Extensible markup language

Obsah přiloženého CD

readme.txt	stručný popis obsahu CD
app	adresář se spustitelnou formou aplikace
├── Linux	verze pro OS Linux
├── Windows	verze pro OS Windows
│ ├── bundledJRE	s přibaleným JRE
│ └── nonBundledJRE	s nepřibaleným JRE
src	adresář se zdrojovými soubory
├── app	zdrojové kódy aplikace ve formě projectu pro NetBeans
├── thesis	zdrojová forma práce ve formátu L ^A T _E X
text	text práce
├── HomolkaTomasBP2013.pdf	text práce ve formátu PDF