



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

WEBOVÝ NÁSTROJ PRO TESTOVÁNÍ POMOCÍ Q-ŘAZENÍ

WEB TOOL FOR TESTING BASED ON Q-SORTING

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

VOJTĚCH MIČKA

VEDOUCÍ PRÁCE

SUPERVISOR

prof. Ing. ADAM HEROUT, Ph.D.

BRNO 2016

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačové grafiky a multimédií

Akademický rok 2015/2016

Zadání bakalářské práce

Řešitel: **Mička Vojtěch**

Obor: Informační technologie

Téma: **Webový nástroj pro testování pomocí Q-řazení**

Web Tool for Testing Based on Q-Sorting

Kategorie: Uživatelská rozhraní

Pokyny:

1. Seznamte se s technikou Q-sorting pro zjišťování subjektivního hodnocení.
2. Vyhledejte a prostudujte existující nástroje pro hodnocení pomocí Q-řazení.
3. Prostudujte vhodné webové technologie pro realizaci webového nástroje pro Q-sorting.
4. Navrhněte prvky uživatelského rozhraní pro Q-sorting.
5. Implementujte dílčí prototypy navrženého uživatelského rozhraní. Testujte prototypy na uživateli a navrhněte vhodná vylepšení.
6. Navrhněte a implementujte webový nástroj pro Q-řazení.
7. Realizujte několik experimentů s použitím vytvořeného nástroje včetně jejich vyhodnocení a demonstруйте tak jeho použitelnost.
8. Zhodnoťte dosažené výsledky a navrhněte možnosti pokračování projektu; vytvořte plakátek a krátké video pro demonstrování výsledků projektu.

Literatura:

- dle pokynů vedoucího

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 až 3, značné rozpracování bodů 4 až 6.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Herout Adam, doc. Ing., Ph.D.**, UPGM FIT VUT

Datum zadání: 1. listopadu 2015

Datum odevzdání: 18. května 2016

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
602 00 Brno, Božetěchova 2



doc. Dr. Ing. Jan Černocký
vedoucí ústavu

Abstrakt

Práce pojednává o návrhu uživatelského rozhraní pro online provedení metody q-řazení. Stěžejním cílem je navrhnout takové rozhraní, které bude co nejvěrněji kopírovat řazení s fyzicky uchopitelnými objekty a vyřešit problém zobrazení velkého množství samostatně stojících prvků na obecně omezené ploše zobrazovacího zařízení. V textu práce je popsán proces návrhu a vývoje uživatelského rozhraní aplikace q-řazení a provedení experimentu potvrzujícího kvalitu navrženého řešení. Výsledkem této bakalářské práce je aplikace umožňující online provedení q-řazení spolu s textem práce, pojednávajícím o postupech použitých při její implementaci.

Abstract

This thesis describes process of designing user interface and implementing online q-sort application. The primary goal is to design the user interface as similar as possible to the classic q-sort with graspable elements, and to solve problem of easy understandable layout with many functional stand alone elements on the limited size of monitor. The thesis describes the process of designing, implementing and testing of the application and it includes experiment for validation of application useability. The outcome of this thesis is an online application for q-sorting and the text of thesis documanting whole process of implementation.

Klíčová slova

Grafické uživatelské rozhraní, návrh uživatelského rozhraní, q-řazení, q-řazení online, q-metodologie, studium subjektivního hodnocení

Keywords

GUI, UX, UX design, user interface, q-sort, q-sort online, q-methodology, study of people's subjectivity, drag&drop

Citace

MIČKA, Vojtěch. *Webový nástroj pro testování pomocí Q-řazení*. Brno, 2016. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Herout Adam.

Webový nástroj pro testování pomocí Q-řazení

Prohlášení

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

.....
Vojtěch Mička
16. května 2016

Poděkování

Děkuji především vedoucímu práce prof. Ing. Adamovi Heroutovi Ph.D. za odborné vedení, cenné rady a inspiraci v průběhu vypracovávání této bakalářské práce. Také chci poděkovat všem těm, kteří se podíleli na testování nebo provádění experimentů k ověření práce.

© Vojtěch Mička, 2016.

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

1 Úvod	2
2 Základy problematiky q-řazení	3
2.1 Teorie q-řazení a jeho přínos	3
2.2 Praktické využití online aplikace pro q-řazení	4
3 Zhodnocení existujících aplikací a nástrojů k provádění q-řazení	5
3.1 Přehled aplikací pro q-řazení	5
3.2 Přehled dostupných nástrojů k realizaci q-řazení	8
3.3 Shrnutí poznatků ze zkoumaných řešení	9
4 Technologie použité k realizaci aplikace	10
4.1 JavaScript a knihovna jQuery	10
4.2 Knihovna Dragula	10
4.3 HTML, CSS3, PHP	11
5 Návrh a implementace aplikace pro q-řazení	12
5.1 Teorie návrhu specifického uživatelského rozhraní	12
5.2 Stěžejní prvky q-řazení a jeho uživatelského rozhraní	13
5.3 Implementace serverové části a interaktivního rozhraní aplikace	19
5.3.1 Práce s daty aplikace pro q-řazení	20
5.3.2 Zajištění funkčnosti uživatelského rozhraní	22
6 Testování a ověření použitelnosti vytvořené aplikace pro q-řazení	24
6.1 Iterativní testování prototypů aplikace v průběhu vývoje	24
6.1.1 Testování intuitivního pochopení práce s uživatelským rozhraním	25
6.1.2 Testování grafické podoby uživatelského rozhraní	28
6.1.3 Testování použitelnosti nápovědy	29
6.2 Ověření relevantnosti výsledků q-řazení s použitím vytvořeného uživatelského rozhraní	30
7 Závěr	35
Literatura	36

Kapitola 1

Úvod

Cílem této práce je navrhnout a implementovat webový nástroj pro provádění průzkumu metodou q-řazení. Dalším cílem je navrhnout, realizovat a vyhodnotit experiment ověřující jeho použitelnost.

Nástroj použitelný k provádění průzkumu metodou q-řazení musí být velmi uživatelsky přívětivý a jednoduchý, zejména proto, že každé vyrušení uživatele během jeho práce má za následek nežádoucí ovlivnění výsledku jím provedeného q-řazení.

V této práci se zabývám metodou q-řazení a způsoby jejího provedení, teorií uživatelsky přívětivého rozhraní a testováním aplikace zakončené provedením experimentu s vytvořeným nástrojem.

V kapitole 2 shrnuji základní údaje o metodě q-řazení a jejím významu. Nastiňuji také možnosti použití této metody. V kapitole 3 shrnuji nejdůležitější poznatky získané zkoumáním existujících řešení. Kapitola 4 krátce shrnuje technologie použité pro realizaci webového nástroje. Kapitola 5 pojednává o teorii návrhu uživatelsky přívětivého rozhraní a její aplikaci při vytváření vlastního nástroje. V kapitole 6 je popsán způsob a průběh testování aplikace. Také pojednává o provádění experimentů ověřujících použitelnost výsledné aplikace.

Kapitola 2

Základy problematiky q-řazení

Stěžejním bodem mé bakalářské práce je vytvořit uživatelské rozhraní pro pohodlné provádění průzkumu metodou q-řazení. K tomu, abych dokázal navrhnout intuitivní aplikaci s přívětivým rozhraním, bylo potřeba nastudovat principy a význam metody q-řazení. O jejích základech pojednává tato kapitola.

2.1 Teorie q-řazení a jeho přínos

Q-řazení je pojem z oblasti psychologie a sociologie představený v roce 1935 britským psychologem Williamem Stephensonem [10]. Jedná se o metodu výzkumu lidské subjektivity, tedy osobního postoje k danému tématu. Kompletní výzkum metodou q-řazení se skládá ze sběru tvrzení k nějakému tématu, samotného provedení q-řazení a analýzy získaných výsledků s použitím jistého druhu faktorové analýzy.

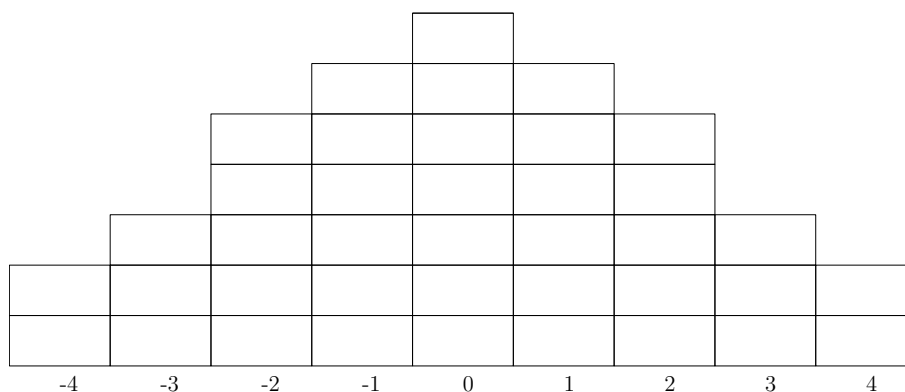
Q-řazení

Z pohledu této práce je na q-řazení nahlíženo jako na proces, ve kterém probíhá třídění papírových karet s tvrzeními, označovaných výrazem *q-set*. Q-set představuje sadu pečlivě vybraných vět k danému tématu získaných tázáním se velkého množství osob, jejichž postoje se co nejvíce liší. Z takto nasbíraných tvrzení je pečlivě vybrán stanovený počet těch, které jsou relevantní k prováděnému výzkumu. Při výběru musí být dbáno na to, aby bylo vyvážené zastoupení všech postojů k tématu. Tato vybraná tvrzení jsou promíchána, čímž vznikne výsledný q-set, který respondenti třídí.

Samotné q-řazení probíhá ve dvou krocích a třídí se podle názoru a pocitu respondenta vůči jednotlivým tvrzením. V prvním kroku dostane každý účastník průzkumu promíchaný q-set a je požádán o jeho rozdělení do tří kategorií. Souhlasný postoj, nesouhlasný postoj a neutrální postoj k tvrzením. Takto získané hromádky mají účastníci ve druhém kroku rozdělit na plochu vytištěné výsledkové mřížky q-řazení.

Podoba výsledkové mřížky tvarem připomíná rozložení podle Gaussovy křivky a slouží k umístování tvrzení dle míry nesouhlasu nebo souhlasu s nimi. Na krajích výsledkové mřížky jsou umístovány ty, ke kterým se respondent vyjadřuje nejdůrazněji. V prostřední části jsou tvrzení, na něž nemá tak vyhraněný názor [12].

Výsledný setříděný q-set představuje souhrn postojů respondenta k danému tématu. Množina seřazených q-setů slouží jako vstupní data pro faktorovou analýzu.



Obrázek 2.1: Výsledková mřížka q-řazení. Slouží k umístění karet s tvrzeními ohodnocených podle názoru respondenta na každou z nich

Faktorová analýza výsledků q-řazení

Faktorová analýza je v případě q-řazení považována za statistickou metodu, která redukuje velký počet tvrzení q-setu na několik těch, u kterých se shodli účastníci průzkumu na pozici umístění. Jejím výsledkem je pestrá škála různých informací, z nichž nejzajímavější jsou zjištěné *faktorové skóre* průzkumu [27]. Faktorové skóre představuje pozice umístění jednotlivých tvrzení ve výsledkové mřížce a určuje se pro každý z nalezených faktorů. Interpretace konkrétních výsledků průzkumu se provádí pomocí rekonstrukce výsledkové mřížky q-setu ze získaných faktorových skóre a následným vzájemným porovnáváním. Z výsledků můžeme například vyčíst data o tom, která tvrzení jsou pro většinu účastníků nejzásadnější. Zjistit převažující postoj k tématu nebo sledovat společné rysy skupin těch účastníků průzkumu, kteří jsou v názorové shodě. Uvedené příklady jsou však jen zlomkem z široké škály použitelnosti výsledných dat metody q-řazení.

2.2 Praktické využití online aplikace pro q-řazení

Nejrozšířenější uplatnění zaznamenává v současné době metoda q-řazení ve výzkumech týkajících se politiky, zdravotnictví, školství a sociologických disciplín. Uplatnitelná je však všude tam, kde je potřeba zjistit převažující názor. Využitelná je také k ověření předpokladů nebo etických otázek. Dokonce by našla uplatnění například k provádění průzkumu názoru na nějaký produkt nebo službu. Q-řazení by tak mohlo fungovat například k hodnocení produktu v průběhu vývoje nebo jako přesnější náhrada hvězdičkového hodnocení.

Ve školství je q-řazení používáno mimo jiné jako nástroj pro rozpoznání jednoty pedagogického sboru [20]. V takovém průzkumu je pedagogy provedeno q-řazení s tvrzeními týkajícími se například vzájemné spolupráce mezi vyučujícími, spolupráce s vedením školy nebo způsobu přístupu školy ke svým žákům. Z výsledků takového průzkumu může ředitel školy zjistit skutečnosti, ze kterých pramení problémy ve vzájemné komunikaci učitelského sboru.

Kapitola 3

Zhodnocení existujících aplikací a nástrojů k provádění q-řazení

Před samotným návrhem vlastní aplikace bylo nutné vyhledat a prozkoumat dostupná řešení elektronického provedení metody q-řazení. Jsou totiž zdrojem mnoha poučení, které mohu využít k návrhu lépe provedeného uživatelského rozhraní. Zkoumáním nalezených řešení jsem našel jejich chyby a problémy vznikající při práci s nimi, kterých se teď mohu vyvarovat. Jedno z dostupných řešení mě naopak inspirovalo. Obsahovalo totiž některé prvky příjemného uživatelského rozhraní. Bohužel jejich provedení bylo špatně odladěné.

Podkapitoly věnované existujícím řešením, dělí text na dvě části. První je věnována skupině aplikací představujících hotové řešení k provádění q-řazení, jež jsou dostupné online nebo jako offline programy. Ty nabízí možnost zorganizování průzkumu pomocí metody q-řazení. Druhá část kapitoly je věnována skupině připravených řešení, které umožňují snadnou realizaci vlastní aplikace k průzkumu metodou q-řazení.

3.1 Přehled aplikací pro q-řazení

Existují jen tři fungující aplikace pro q-řazení, přičemž všechna zkoumaná řešení jsou zastaralá. Kvalita jejich uživatelského rozhraní zdaleka nevyužívá dnešních technologických možností ani poznatků o práci uživatele ve webovém prostředí. Dvě řešení jsem prozkoumal důkladněji a jejich rozhraní jsou popsána níže. Třetí aplikací je online provedení metody s názvem Q-Assessor, disponující funkčním, avšak velmi starým a triviálním uživatelským rozhraním, které pro mou práci nepřináší žádné další užitečné poznatky.

POETQ, online aplikace q-řazení

POETQ je vyvíjen skupinou HSMC na Birminghamské univerzitě ve spolupráci s Stephenem Jeffaresem [15]. Jedná se o veřejně nepřístupnou aplikaci k online q-řazení s netypickým pojetím jeho průběhu. Před začátkem samotného řazení je uživateli položeno několik otázek relevantních ke zkoumanému tématu. Ty zahrnují například základní údaje o testujícím, to, jak se cítí být obeznámen s tématem, ale i další otázky vhodné pro konkrétní cíl výzkumu.

Samotné q-řazení pak probíhá ve třech etapách. Nejprve rozdělí uživatel pomocí klikání myši karty z q-setu do tří základních kategorií: souhlasná, nesouhlasná a neutrální. Následně je mu v několika krocích předložena nabídka předtím seřazených tvrzení a je vyzván k výběru dvou nebo tří těch, které se mu jeví jako nejvýznamnější. Tvrzení jsou vybírána kliknutím myši a potvrzena tlačítkem. Vybraná tvrzení se postupně ukládají a je z nich

4 STATEMENTS ABOUT JOINT COMMISSIONING YOUR AREA - DO YOU AGREE?

Help

Based on your experience of joint commissioning in your area, place the statement below into one of the three boxes depending on whether you agree or disagree. You can drag or click on the box itself. If you are not sure place it in neutral. There are 40 in total. In this stage follow your instincts, there is time later to refine your views.

<i>Disagree</i>	<i>Neutral</i>	<i>Agree</i>
Integration of care is about delivering more for less. Integrating leads to better working conditions for colleagues and helps boost morale. Integration of care is about improving choice for users.	Integration of care is opening up opportunities for the private and third sectors. Integrating care signals to others that we are in partnership.	Integrating can lead to individual teams becoming more insular. Integration of care is mostly about fulfilling government requirements to collaborate. Integration of care seems to be speed up referral Integration of care helps build the necessary trust between us.
	Integration of care requires larger management structures.	

Obrázek 3.1: Základně rozdělování tvrzení v prvním kroku q-řazení aplikace POETQ

5 REFINE YOUR PREFERENCES

Help

Below are the statements you agreed with in the previous step. To begin ordering them select 2 you most agree with. They will disappear from the screen when you click Next.

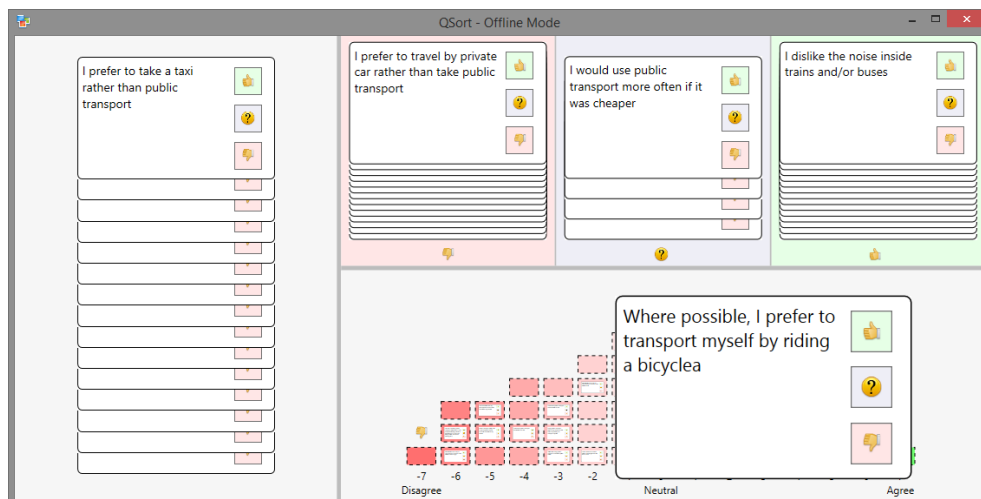
Integration of care enables better risk management.	Where we are co-located, it benefits professional discussion through the development of informal relationships.	Integrating can lead to individual teams becoming more insular.
Integration of care helps build the necessary trust between us.	Integrating is about delivering a seamless service for service users.	Integration of care enables greater information sharing.
Integration of care is about reducing duplication.	Integration of care seems to be speed up referral	Integration of care changes the way service users can influence the services they receive.
Integration of care requires larger management structures.	Integrating means fewer inappropriate referrals.	Integrating should be about reducing pressure on acute services.

Obrázek 3.2: Výběr nejsilněji vnímaných tvrzení v druhém kroku q-řazení aplikace POETQ

generována finální výsledková mřížka q-řazení. Ta se ve třetí fázi předloží testujícímu ke kontrole a případným opravám s použitím drag&drop funkcionalita.

V posledním kroku odpovídá uživatel fulltextovou formou na otázky, které shrnují jeho postoj v oblasti zkoumaného tématu.

Za dobrou myšlenku tohoto nástroje považují dotazníkovou formu zjišťování základních informací o testujícím uživateli a finální dotazník shrnující jeho postoje. Celkové pojetí q-řazení je však dle mého názoru špatně zpracováno. Postupným výběrem několika málo nejvýznamnějších tvrzení totiž není uživatel nucen k důslednému přemýšlení nad přesným zařazením každého z nich. Nástroj totiž nemá jak analyzovat to, jakou mírou uživatel souhlasil nebo nesouhlasil s tvrzeními rozdělenými v první fázi q-řazení. Proto je problematické generovat výsledkovou mřížku přesně seřazenou podle postojů uživatele. Ten sice má na konci řazení možnost výsledek přetřít, nic ho však již k tomu nenutí, protože vše je vlastně hotovo.



Obrázek 3.3: Obrazovka programu Qsort v průběhu q-řazení

QSort, offline program pro q-řazení

Qsort je offline program pro q-řazení vyvíjený společností Alintech v Perthu v Austrálii [1] společnost Alintech emailem.

Po spuštění programu se na úvodní obrazovce zobrazí nabídka výběru z několika připravených témat q-setu. Zcela však chybí možnost přidání vlastního. Samotné q-řazení pak probíhá celé na jediné obrazovce, do které jsou vmáčknuty oba standardně používané kroky. Uživatel si tak může zvolit, jestli si karty předem roztřídí do tří kategorií (souhlas, nesouhlas a neutrální kategorie) nebo rovnou vyplní výsledkovou mřížku q-řazení. Pro přesuny karet je využíváno drag&drop a pro rozčlenění tvrzení do tří základních kategorií je navíc možné využít jejich ikonky umístěné na každé kartě. Kliknutím na ikonku se karta sama přesune do požadované kategorie. Po rozřazení všech tvrzení z nabídky se okno aplikace samo přerorganizuje tak, aby výsledková mřížka a hromádky s kartami tří základních kategorií nad ní zabraly co nejvíce prostoru.

Karty v hromádkách jsou umístěny tak, že čitelná je pouze nejsvrchnější z nich, tedy ta nejvzdálenější výsledkové mřížce. Ostatním kartám vyčuhuje pouze jejich okraj, takže je možné odhadnout, kolik jich ještě zbývá roztřídít. Takto zvolená organizace je nevhodná z toho důvodu, že prodlužuje vzdálenost, kterou musí uživatel táhnout každou kartou do výsledkové mřížky.

Karty ve výsledkové mřížce q-setu jsou zmenšeny do fixní velikosti a nabízejí funkci zvětšení najetím kurzoru myši. Po zaplnění se výsledková mřížka sama roztáhne do plné velikosti okna programu. Takto si uživatel může pohodlně zkontrolovat seřazené výsledky a poté potvrdit dokončení q-řazení.

Uživatelské rozhraní tohoto programu je ze zhlédnutých řešení nejpropracovanější. Používá barev a obrázkových ikon k odlišení kategorií, automatickou organizaci prostoru na obrazovce a nástroj pro zvětšování malých karet s tvrzeními. Ačkoli jsou všechny tyto prvky dobré svou myšlenkou, jejich provedení není pro použití dobře odladěno. Ikonky na kartách zbytečně zabírají cenný prostor, takže se při delších tvrzeních objeví skrolovací lišta. Navíc na kartách zbytečně zůstávají i po jejich umístění do dané kategorie nebo do výsledkové mřížky q-řazení. Animace zvětšení karty po najetí kurzoru myši je nepříjemně rychlá a působí velmi rušivým dojmem. Zvětšování funguje i u karet, které jsou teprve v nabídce

k řazení.

To může být výhodné pro lidi se špatným zrakem, avšak způsobuje to problémy. Pokud chce uživatel využít klikací ikonky, najede kurzorem na kartu a chvíli mu trvá, než najde ikonku odpovídající kategorie. Přibližně ve stejné době, kdy na ni chce kliknout, dojde ke zvětšení karty a ikonka se o kousek posune. Toto není nijak významný problém, ale po desátém takovém posunutí se jeví jako velmi obtěžující. Dobře provedená je naopak možnost zvětšování malé karty i v případě, kdy uživatel kurzor právě používá k přemístování jiné karty. To usnadňuje a zrychluje celý proces q-řazení.

3.2 Přehled dostupných nástrojů k realizaci q-řazení

Během průzkumu existujících online řešení jsem našel tři nástroje, pomocí kterých je možné aplikaci pro q-řazení realizovat. Jedná se o připravená řešení, která lze využít i s minimální znalostí programování. Nevýhodou takto připravených řešení je nepropracované uživatelské rozhraní. Nástroje totiž nabízejí jen základní funkcionalitu k provedení q-řazení a nedbají na jeho grafickou úpravu.

Jedním z nich je i nástroj FlashQ, který je v následujícím textu opominut. Vývoj i podpora pro něj byly zastaveny v roce 2007 a díky použití technologie Flash nejde spustit v dnešních prohlížečích.

Nástroj HTMLQ

HTMLQ je vyvíjený německou společností Aproxima [3], která se zabývá výzkumem trhu a sociálními výzkumy. Tento nástroj umístila na server GitHub pod MIT licenci, jakožto adaptaci na nástroj FlashQ, která opravuje nedostatky původního provedení a jeho neaktualnost.

Proces třídění probíhá standardně ve dvou krocích, po kterých následuje krátký dotazník k tématu. Celý nástroj je velmi jednoduchý, jelikož implementuje jen základní provedení prvků používaných při procesu q-řazení. Neklade tedy žádný důraz na přívětivost práce s uživatelským rozhraním ani na jeho vizuální podobu. Bylo by možné využít jej jako základ pro implementaci vlastní online aplikace k provádění q-řazení. Nevýhodou však je, že pro vytvoření uživatelsky přívětivého nástroje bych z jednoduchého HTMLQ využil pouze drag&drop funkcionalitu, kterou by také bylo potřeba do značné míry upravit. Proto jsem se rozhodl pro kompletně vlastní řešení.

WebQ, historický nástroj pro q-řazení

Řešení v podobě nástroje WebQ uvádím spíše pro zajímavost. Jedná se totiž o velmi staré provedení metody q-řazení, které vyvinul Peter Schmolck v roce 2002 [19]. V nástroji jsou jednotlivá tvrzení q-setu rozložena v řádcích tabulky, která v záhlaví obsahuje číselné ohodnocení míry souhlasu (nesouhlasu) s tvrzením. Ohodnocení jde od záporných čísel pro nesouhlas ke kladným pro souhlas. Jednotlivým tvrzením přiřazuje uživatel hodnotu pomocí přepínacích tlačítek. Výsledek takto provedeného řazení se po dokončení zobrazí v tabulce s tvrzeními seřazenými podle ohodnocení, které jim bylo přiděleno.

Disagree					Agree	
-2	-1	0	+1	+2		

Disagree	Neutral	Agree
(3) What really matters is reaching my destination and getting back, the mode of travel does not matter much.	(2) As a result of all those different timetables and lines, travelling by public transport is too complicated.	(12) For my work I need a representative mode of transport.
(11) I had rather not drive in big cities... lots of traffic, lots of	(5) I had rather look out of the compartment window to the	(1) For private use I do not need a car.
		(10) This is the most comfortable

Obrázek 3.4: Průběh druhého kroku q-řazení realizovaného pomocí nástroje HTMLQ

3.3 Shrnutí poznatků ze zkoumaných řešení

Během průzkumu existujících řešení elektronického provedení metody q-řazení jsem došel k několika poznatkům užitečným pro implementaci vlastní aplikace.

Organizace q-řazení v provedení POETQ je příliš časově náročná. Účastníka průzkumu tak odradí několik delších otázek ještě před začátkem třídění. Samotný proces q-řazení je rozdělen do více než sedmi kroků, což také působí nepříjemně zdlouhavě. Za dobrý nápad však považuji závěrečnou možnost vyjádřit svůj postoj k danému tématu jako celku.

Za nejlépe provedené řešení považuji program Qsort. Ten používá barevné oddělení kategorií, dobrou organizaci rozložení prvků q-řazení na ploše programu a také nabízí možnost zvětšování karet ve výsledkové mřížce q-setu. Hlavní nevýhodou programu je jeho offline provedení a také špatně doladěné detaily jeho uživatelského rozhraní. Například zbytečné ikonky přesunu na kartách ve výsledkové mřížce q-setu nebo nepříjemně rychlá animace při zvětšování karty po najetí kurzoru.

Za nejvýznamnější bych označil zjištění, že žádný z existujících nástrojů není jednoduše použitelný pro zorganizování vlastního průzkumu metodou q-řazení, aniž by někdo účastníky průzkumu při vypracovávání provázel.

Kapitola 4

Technologie použité k realizaci aplikace

Dnes je možné vytvořit webovou aplikaci s použitím nesčetně mnoha technologií a programovacích jazyků. Já jsem k její implementaci zvolil ty, které jsou na webu používány nejčastěji. Výhodou této volby byla široká uživatelská základna a existence množství již vytvořených řešení a vyřešených problémů. Tento fakt mi dal jistotu, že nenastane situace, kdy budu nucen věnovat více času vyřešení jednoho technického problému než návrhu a testování použitelnosti uživatelského rozhraní jakožto náplni mé bakalářské práce.

4.1 JavaScript a knihovna jQuery

Pro ovládání interaktivních prvků webu, změny obsahu bez opětovného načítání a zachytávání událostí uživatelského rozhraní se ve webovém prostředí nejčastěji využívá JavaScript a jeho knihovna *jQuery*. Ta je nadstavbou JavaScriptu, jejímž cílem je nabídnout řešení často používaných funkcí při tvorbě webových stránek a sjednotit chování JavaScriptu napříč prohlížeči.

Při tvorbě webového rozhraní pro provádění metody q-řazení jsem JavaScript a jQuery využil pro implementaci interaktivních prvků webu a generování velké části HTML kódu. Ten bylo potřeba vytvářet v závislosti na velikosti q-setu, kdy se pro každý jeho rozměr generuje výsledková mřížka různé podoby a velikosti.

4.2 Knihovna Dragula

Dragula [7] je JavaScriptová knihovna pro snadné použití drag&drop, vyvíjená Nicolásem Bevacquavou. Knihovna disponuje velmi dobře navrženým API a všemi základními prvky drag&drop funkčnosti.

Dragula vytváří centrální objekt API s názvem „drake“. Ten je použit pro kompletní správu práce s knihovnou, pro definování drag&drop kontejnerů, odchyťování událostí a nastavování parametrů poskytované funkčnosti. Při inicializaci objektu „drake“ knihovna sama vybere všechny elementy, které jsou potomky definovaných drag&drop kontejnerů. Ty je pak možné mezi jednotlivými kontejnery přesouvat. API knihovny Dragula nabízí odchyťování všech myslitelných událostí, které mohou při drag&drop nastat a do nich posílá následující parametry:

- *el*: přemísťovaný element
- *target*: cílový kontejner pro umístění taženého elementu
- *source*: zdrojový kontejner ze kterého přemísťovaný element pochází
- *sibling*: element, který bude následovat za taženým elementem po jeho přemísťení

Hlavním důvodem pro použití právě knihovny Dragula bylo její API. Celé knihovna je totiž navržena tak, aby obsáhla pouze základní podstatu drag&drop funkčnosti a co nejlépe umožnila její rozšíření o vlastní funkcionalitu.

4.3 HTML, CSS3, PHP

HTML (Hypertext Markup Language) představuje značkovací jazyk pro tvorbu webových stránek. Zvolil jsem jej proto, že jeho použití je nezbytné pro využití frameworku Bootstrap [2], usnadňujícího tvorbu webů a sjednocujícího různé vykreslování elementů napříč prohlížeči a platformami. Jeho použití se však pro potřebu návrhu atypického layoutu mé aplikace neosvědčilo. Důvodem bylo především to, že Bootstrap tvoří layout v mřížce o maximálním počtu dvanácti sloupců. Ačkoli je možné tyto sloupce dále dělit, pro potřebu generování výsledkové mřížky q-řazení o proměnlivé velikosti jde o zbytečně komplikované řešení. Jednotlivé sloupce mřížky při využití Bootstrapu navíc nemohly plně využívat dostupný prostor a tím zmenšovaly již tak problematicky malé karty s tvrzeními. (To se projevovalo již při středně velkém q-setu).

CSS (Cascading Style Sheets) je jazyk definující způsob zobrazování webových stránek napsaných v HTML a příbuzných jazycích. Pro potřeby mé aplikace jsem využil jeho nejnovější verzi CSS3 [24]. Ta přináší možnosti jako průhlednost elementů, animace, 2D transformace nebo zaoblení rohů, které jsem mohl dobře uplatnit v navrhovaném uživatelském rozhraní.

PHP je skriptovací jazyk pracující na serverové straně aplikace. Funkční aplikace pro q-řazení potřebuje komunikovat s databází, ze které načítá data pro organizaci výzkumu (tvrzení q-setu, jeho rozměry, unikátní odkaz konkrétního výzkumu, atd.) a ukládá do ní výsledky jednotlivých q-řazení. Pro implementaci této rutinní funkčnosti se PHP hodí nejlépe. Jsou totiž dobře zdokumentované způsoby jeho komunikace s JavaScriptem a také nabízí rozšíření z názvem PDO, vhodné pro pohodlnou a bezpečnou práci s databází [5].

Kapitola 5

Návrh a implementace aplikace pro q-řazení

Při realizaci aplikace k provádění q-řazení jsem musel nastudovat problematiku návrhu uživatelských rozhraní. Tato kapitola proto pojednává zejména o základních pravidlech návrhu dobře použitelného, intuitivního uživatelského rozhraní a jejich uplatnění při vytváření aplikace. Také rozebírá způsob implementace serverové části aplikace a komunikace s ní. V neposlední řadě obsahuje i popis dynamického generování některých částí uživatelského rozhraní a implementace interaktivních prvků v něm.

5.1 Teorie návrhu specifického uživatelského rozhraní

Při tvorbě návrhu dobrého uživatelského rozhraní je potřeba přemýšlet tak, aby bylo dbáno na potřeby uživatele při provádění daného úkonu i během procesu získávání informací z uživatelského rozhraní [9]. Například v aplikaci sloužící k měření rychlosti internetového připojení je třeba dbát na to, aby obsáhla všechny údaje, které jsou pro měření nejdůležitější. Zároveň je však nutné zobrazit je natolik přehledně a srozumitelně, aby uživatel bez dlouhého přemýšlení pochopil jejich význam.

Abych dokázal implementovat aplikaci podle této myšlenky, bylo potřeba nastudovat především základy teorie návrhu dobře použitelného uživatelského rozhraní. Důležité předpoklady pro podobu takové aplikace jsou shrnuty v druhé části této podkapitoly.

Aplikace pro provádění q-řazení je unikátním typem webových stránek. Jedná se totiž o nástroj určený k provádění průzkumu. Nejrozšířenější podobou takových nástrojů na internetu je dotazníková forma. Ostatní způsoby provedení jsou již specifické pro použitou metodu průzkumu. Z toho důvodu se mi nepodařilo dohledat žádné obecné principy pro návrh takových uživatelských rozhraní. Proto jsem na základě nastudovaných znalostí o použití a účelu metody q-řazení stanovil několik předpokladů, které jsou popsány v první části této podkapitoly. Shrnují, co by měla aplikace splňovat, aby byla použitelným ekvivalentem k provádění q-řazení v papírové podobě.

Stanovení předpokladů pro použitelnost aplikace

Nejčastěji prováděnou formou průzkumu metodou q-řazení je její obdoba s papírovými kartami. To proto, že je pro respondenty nejlépe proveditelná. Lidem je totiž nejpřirozenější práce s fyzicky uchopitelnými předměty. Dalším důvodem je skutečnost, že v průběhu výzkumu metodou q-řazení bývá respondent navigován osobou průvodce. Ten pomáhá osvětlit

princip rozdělení karet do výsledkové mřížky q-řazení a provádí interview zaměřené na souhrn postojů respondenta k tématu průzkumu.

Uživatelské rozhraní této aplikace tak musí splnit dva hlavní cíle. Navodit pocit co nejpodobnější s řazením papírových karet a nahradit osobu provázející q-řazením.

Abych navodil pocit třídění skutečných karet s tvrzeními, bylo důležité přizpůsobit způsob práce s rozhraním aplikace potřebám uživatele. Stejně důležité je však i perfektní odladění plynulosti práce s rozhraním. V průběhu provádění q-řazení musí mít uživatel naprostý klid na práci. Z principu metody je zjevné, že se musí soustředit na to, co pro něj znamenají jednotlivá tvrzení. Pokud ho při tom cokoli ruší, ztrácí pozornost, což ovlivní výsledek jím provedeného q-řazení. Neodladěné uživatelské rozhraní by také mělo za následek znepríjemnění nálady uživatele. To může v menší míře vést k ovlivnění jeho výsledků a v extrémních případech k odrazení od dokončení q-řazení.

Pro nahrazení osoby provádějící q-řazením existují dvě možnosti. Textové provázení průběhem q-řazení nebo intuitivní uživatelské rozhraní. Protože textové průvodce lidé na webových stránkách často ignorují, zvolil jsem cestu návrhu intuitivně použitelné aplikace.

Teze intuitivního uživatelského rozhraní

Důležitým poznatkem pro návrh intuitivního webu je skutečnost, že je uživatelé nechtou [17]. Pouze je prohlížejí. Výjimkou jsou weby zaměřené na obsah. I u těch je však pozornost uživatelů mířena na hlavní text webu, zbytek textů na stránce bývá většinou přehlížen. Důvodů je několik.

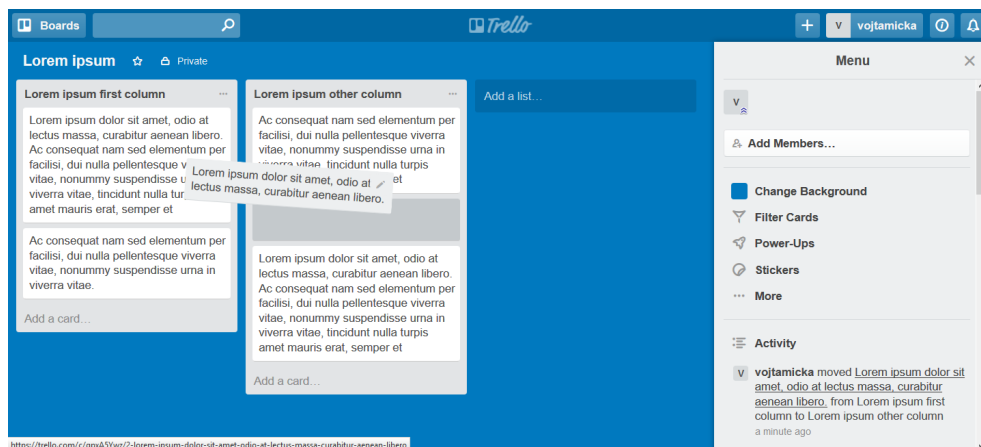
Prvním z nich je skutečnost, že uživatelé nechtějí na webu strávit ani o minutu déle, než je nutné. Počítač jim má přece usnadnit práce a ne ji přidávat. Druhým je povědomí uživatelů o tom, že nepotřebují vědět všechno. Každé uživatelské rozhraní nabízí spoustu funkcí, které uživatel nikdy nepoužije, proto nemá snahu dozvědět se o nich víc.

Právě na těchto poznacích je založen můj návrh intuitivního uživatelského rozhraní. Uživatelům musí být zdůrazněny ty jeho prvky, které nutně potřebuje k základní práci s rozhraním. Zbytek funkčnosti může objevit náhodou nebo v nápovědě. Důležité je zaměřit se na pochopení základního principu práce s aplikací.

Dalším důležitým poznatkem o práci s webem je to, že uživatelé nedbají na to, jak by se s aplikací mělo pracovat. Prostě vyzkouší všechny možnosti, které je napadnou, dokud nedojdou ke svému cíli. Nemyslí přitom na to, že návrhář předpokládal, že budou pracovat systematicky. Z toho důvodu je důležité poskytnout uživateli při provádění q-řazení co nejvíce volnosti. Jedinou podmínkou je, aby byl výsledný q-set seřazený v takové podobě jakou požadují pravidla q-řazení. Proto nabízí uživatelské rozhraní co nejvíce možností jak jednoduše s aplikací pracovat. Například různé způsoby přemísťování karet, žádná omezení v prvním kroku q-řazení nebo přemísťování karet ve druhém kroku q-řazení i napříč předtím zvolenými kategoriemi.

5.2 Stěžejní prvky q-řazení a jeho uživatelského rozhraní

Pro příjemnou a intuitivní práci uživatele je potřeba identifikovat základní prvky uživatelského rozhraní. Těmi mohou být elementy, jejichž význam je pro uživatele důležitý nebo prvky webu, se kterými pracuje více než s ostatními. U metody q-řazení jsou to karty s tvrzeními z q-setu, které figurují prakticky v každém úkonu, jež uživatel provádí. Dalším významným elementem je *výsledková mřížka q-řazení*, což je prostor, do kterého uživatel



Obrázek 5.1: Uživatelské rozhraní služby Trello, která hojně využívá přesunu karet mezi organizačními sloupci. K tomu disponuje sofistikovaným provedením elementů karet, usnadňujícím činnost uživatele

umísťuje karty z q-setu. A v neposlední řadě také nápověda. Ta sice nepatří k často využívaným prvkům aplikace, ale obsahuje všechny informace důležité pro člověka, který by si během provádění q-řazení nevěděl rady.

Karty s tvrzeními q-setu

Nejčastěji se vyskytujícím prvkem uživatelského rozhraní jsou jednotlivé karty obsahující tvrzení q-setu. Zároveň se jedná o nejvíce používaný *funkční prvek*. Tím je myšlena skutečnost, že není pouze statickým elementem webového rozhraní, ale interaktivním. Proto je nejen důležité perfektní odladění plynulosti práce s ním, ale i to, jak bude na uživatele působit.

Z významu karty vyplývá, že je pro ni nejdůležitější dobrá čitelnost. Z toho důvodu jsem se rozhodl se vzhledem karty neexperimentovat a zvolil jsem kombinaci černého písma na bílém pozadí. Jedná se o nejkontrastnější barvy a nejlépe se tak hodí pro čitelnost textu. Stejně konzervativně proběhla i volba sans-serif fontu Verdana [14]. Pro lepší vizuální oddělení textů v jednotlivých kartách dostala každá karta velký vnější okraj. Mezera mezi kartami je tak necelých 20% z průměrné velikosti karty.

Nejčastějším úkonem v průběhu q-řazení je pohyb karet mezi umístovacími plochami. Pro tento účel se nejlépe hodí drag&drop funkcionalita. Při jejím návrhu jsem se inspiroval u služby *Trello* [6].

Trello je webová služba pro organizaci čehokoliv. Pro tento účel nabízí uživatelské rozhraní v podobě volitelného počtu sloupců mezi kterými je možné přesouvat kartičky s úkoly. Drag&drop je v ní tedy hojně využíváno. Provedením analýzy jejich uživatelského rozhraní jsem odhalil několik efektů zpříjemňujících práci uživatele s pohyblivými kartami.

Prvním z nich je pootočení karty v úhlu přibližně 5 stupňů po směru hodinových ručiček. To zapříčiní, že je přesouvaná karta jednoznačně odlišena od ostatních. Snadná identifikace karty je významným vylepšením zejména při jejich velkém počtu. Uživatel ji díky tomu může rychleji vyhledat a znovu si přečíst tvrzení před tím, než se rozhodne pro finální umístění. Nevýhodou je snížení čitelnosti textu v kartě. Proto jsem se na tuto skutečnost zaměřil při testování, kde se ukázalo, že jde o zanedbatelný problém v porovnání s jeho přínosem.



Obrázek 5.2: Ilustrace navržených efektů karet s tvrzeními

Dalším efektem, který na Trello doplňuje drag&drop, je naznačení pozice karty po jejím umístění. To je provedeno zobrazením stínu karty na místě, kam při uvolnění dopadne. Díky tomuto efektu není uživatel nikdy nechán na pochybách, zda pouští kartu do správné pozice, což zpřijemňuje a urychluje jeho práci.

Ačkoli je přesouvání karet pomocí drag&drop nejintuitivnější volbou, během testování se ukázalo, že pro velký počet přemísťovaných prvků není ideální. Ti testující, kteří nejsou zvyklí na každodenní práci s myší, měli v průběhu řazení většího q-setu problémy s bolestmi zápěstí. Jednou z možných alternativ pro přemísťování karet bylo použití kláves. Jejich použití by však bylo zbytečně komplikované, oproti možnosti přemísťovat karty na kliknutí myši. Tato volba se při testování velice osvědčila. Problémem však bylo, že na ni nepřišli uživatelé intuitivně nýbrž náhodným kliknutím. Proto byla potřeba na tuto možnost nějak upozornit. K tomuto účelu jsem využil nápovědu, která je popsána v další z podkapitol stěžejních prvků uživatelského rozhraní.

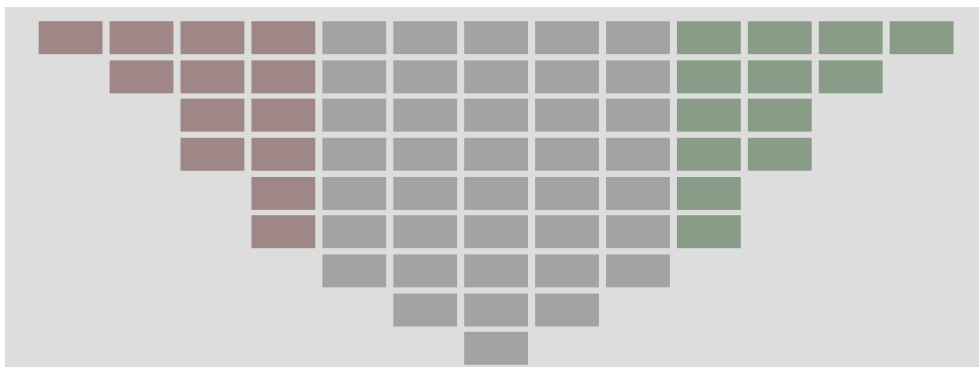
Nejvýznamnějším problémem při práci s kartami byla jejich velikost ve druhém kroku q-řazení. Na obrazovce při něm může být rozprostřeno až 65 karet s tvrzeními a další 3 větší prvky obsahující hromádky karet k roztrídění. Karty ve výsledkové mřížce q-řazení bylo potřeba zmenšit do té míry, že se text v nich stal nečitelným. Při řešení tohoto problému jsem se částečně inspiroval programem Qsort. Ten nabízí zvětšení karty s textem po najetí myši. Při delším textu pak přidává skrolovací lištu pro jeho posouvání. Použití posuvníku textu bylo sice funkčním, ale rušivým elementem. Proto jsem situaci vyřešil použitím animace — zvětšování karty do velikosti, jakou měla před umístěním do výsledkové mřížky.

Pro příjemnější formu zalamovaného textu ve zmenšených kartách jsem přidal zkracování textu tak, že poslední slovo nahradí zařítá klauzule tří teček. Ta v uživatelích evokuje pokračování zobrazeného textu.

Výsledková mřížka q-řazení

V kontextu této práce je výsledkovou mřížkou q-setu myšlen prostor, do kterého jsou tříděny tvrzení z nabídky. Ta jsou rozmísťována na základě toho, do jaké míry mírou s nimi uživatel nesouhlasí nebo souhlasí.

První krok q-řazení obnáší rozčlenění tvrzení do tří kategorií, kterými jsou souhlas s tvrzením, nesouhlas s tvrzením a neutrální postoj. Protože uživatelům se s myší lépe



Obrázek 5.3: První návrh podoby výsledkové mřížky q-setu

pracuje při pohybu do stran, zvolil jsem při návrhu layout rozložení se čtyřmi vedle sebe umístěnými sloupci. Sloupec s nabídkou tvrzení je od ostatních zdatelně odsazen a je také vyšší. Díky tomu působí výraznějším dojmem a je snadno odlišitelný. Jeho nezaměnitelnost s ostatními sloupci je výhodná v tom momentu řazení, kdy už všechny sloupce obsahují nějaká tvrzení a mohou se proto plést.

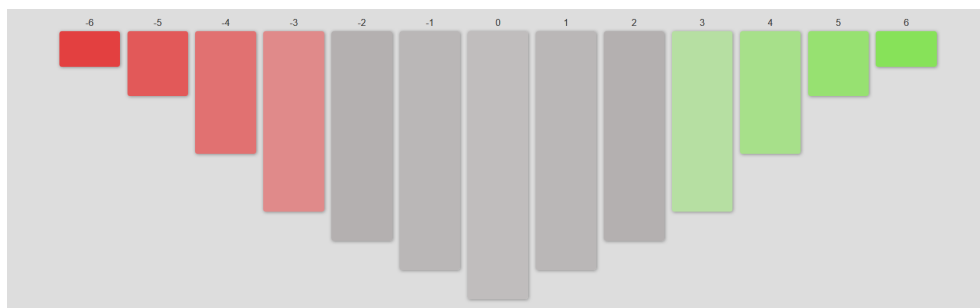
Pro další odlišení sloupců s výsledky se v papírové podobě metody q-řazení používá jejich textový popis. Tento způsob je sice dostačující, uživatelé však často zapomínali na které straně je sloupec pro kladná a na které sloupec pro záporná tvrzení. Protože uživatelé rychleji reagují na barvy než na text, přidal jsem barevné odlišení sloupců. Barvy jsem zvolil podle zažitých významů jejich slovního ekvivalentu, tedy červená pro nesouhlas, zelená pro souhlas a šedá pro neutrální postoj k tvrzení. Barevná pozadí sloupců nabývají pouze nádechu jednotlivých barev. To proto, že sloupce v plné barvě příliš přitahovaly oči a odváděly tak pozornost uživatelů od významnějšího sloupce s tvrzeními k seřazení.

Návrh layoutu ve druhém kroku řazení vyžadoval přenesení standardně používané výsledkové mřížky q-setu na plochu zobrazovacího zařízení. Mřížka se skládá z množství buněk sloužících k umístění karet s tvrzeními, které jsou sdruženy do tvaru Gaussovy křivky. Buňky z levé strany slouží k umístění tvrzení, se kterými uživatel nejvíce nesouhlasí. Napravo jsou buňky pro ta, se kterými nejvíce souhlasí. Prostřední část je pro tvrzení, ke kterým nemá tak vyhraněný názor. Buňky umístěné nad sebou mají stejná ohodnocení, kde ohodnocením je myšlena váha, kterou uživatel přikládá daným tvrzením z q-setu. Váha buněk roste od nuly v prostředním sloupci ke krajům výsledkové mřížky, kde jsou umístěny karty, k nimž je postoj uživatele provádějího třídění nejvyhraněnější.

Mřížka je i pro středně velké q-sety natolik široká, že zabere celou šířku monitoru. Proto bylo potřeba umístit hromádky s nabídkou karet k setřídění nad výsledkovou mřížku. A to i navzdory tomu, že se uživatelům lépe pracuje, když myši pohybují do stran.

Z obrázku 5.3 je patrné, že druhý krok q-řazení působil velmi nepřehledně. Tuto domněnku jsem potvrdil během testování, kdy většina testujících reagovala na tento krok kritikou. Proto jsem standardně používanou mřížku upravil tak, že jsem vizuálně sdružil spolu souvisejících elementy [17]. Provedl jsem spojení nad sebou umístěných buněk do sloupců a otestováním ověřil, že výsledná podoba mřížky působí uspořádanějším dojmem. To má za následek příjemnější podobu a hlavně přehlednější proces umístování karet.

Stejně jako v prvním kroku je i zde provedeno barevné oddělení sloupců tří základních kategorií (souhlas, nesouhlas a neutrální postoj). Pro zdůraznění toho, že záleží na tom, do kterého ze sloupců stejné barvy uživatel kartu umístí, jsem využil různých odstínů barev.



Obrázek 5.4: Finální návrh podobny výsledkové mřížky q-setu

Tmavší odstíny dostaly krajní sloupce pro nejvýznamněji ohodnocené karty a směrem ke středu se odstíny zesvětlují.

Při testování se však osvědčily jen z části. Účastníci testování sice ihned správně přiřazovali karty na základě barvy, neřadili je však podle míry souhlasu a nesouhlasu. Tento problém je způsoben pravděpodobně nedostatečným rozdílem odstínu jednotlivých sloupců, který se projevuje zejména u menších výsledkových mřížek q-setu. Řešením by bylo přidat barvám kategorií na důrazu. Silně červeně nebo zeleně zabarvené sloupce však působí rušivým dojmem. Situaci jsem proto vyřešil přidáním takové číselné osy ohodnocení sloupců, jaká je uváděna ve standardně používané výsledkové mřížce pro q-řazení.

Nápověda

Stěžejním bodem mnou navrhovaného uživatelského rozhraní byla snaha o to, aby uživatel rychle pochopil princip q-řazení, aniž by u sebe měl osobu, která jej procesem provází. Ačkoli se tento cíl do značné míry podařilo splnit, malé procento uživatelů mělo stále problém s okamžitým pochopením druhého kroku q-řazení. Proto jsem implementoval nápovědu, která rychle objasní principy q-řazení i možnosti práce s uživatelským rozhraním.

V otázce nápovědy však vyvstává (nejen u webových aplikací) významný problém. Lidé je nechtou [21] Nápověda je zpravidla otevírána jako poslední možnost v případě, kdy uživatel opravdu neví jak dál. V průběhu testování aplikace jsem se proto rozhodl udělat malý průzkum. Pokud si testující uživatel nevěděl ihned rady, většinou se tázal mě, co má dělat. Jako nezúčastněný pozorovatel jsem mu neporadil, ale zeptal jsem se, proč se obrátili na mně místo otevření nápovědy. Nejčastější odpovědi na tuto otázku lze shrnout do následujícího tvrzení: „Nápovědy obsahují dlouhé texty a spoustu informací které musím přečíst, ale nepoužiji“.

Proto jsem se při návrhu nápovědy musel vyhnout dlouhým a pro uživatele nezáživým textovým popisům. Inspiraci pro její podobu jsem našel v nápovědách chytrých telefonů a úsloví: „Jeden obrázek vydá za tisíc slov“.

Takto implementovaná nápověda se skládá z obrazovky aplikace na pozadí a krátkých textových popisů, samovysvětlujících obrázků, šipek a zvýrazňujících prvků na popředí. Není zde žádný dlouhý, odrazující text, což také urychlilo použití nápovědy. Zejména pokud uživatel hledá význam konkrétního prvku uživatelského rozhraní. V takovém případě se stačí podívat na místo, kde je hledaný prvek, a hned vedle něj si přečíst krátký popis. Není potřeba pracně hledat význam prvku v obsáhlém textu nápovědy.

První krok metody je přímočarý a při testování se jeho pochopení uživateli ukázalo snadné i bez použití nápovědy. Bylo ji však možné využít k upozornění na možnosti pohybu



Obrázek 5.5: Netextová část nápovědy ve druhém kroku q-řazení. Modré popisky upozorňují na funkční prostředky a pravidla v průběhu druhého kroku třídění

karet, zejména na jejich přesouvání kliknutím myši.

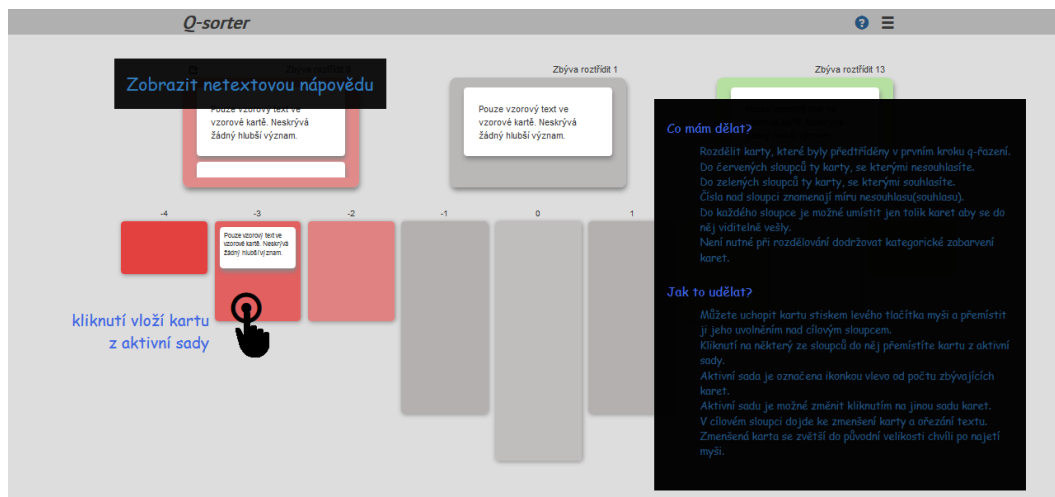
Širší uplatnění nápovědy je teprve ve druhém kroku metody. Q-řazení zde totiž definuje množství pravidel, kterými se musí třídící uživatel řídit. Ačkoli jsem při návrhu dbal na jejich zakomponování do uživatelského rozhraní, tak někteří z testujících uživatelů pochopili princip druhého kroku až v průběhu třídění. Většinou v momentě, kdy jim došly kartičky spadající do stejné barevné kategorie nebo když si všimli číselné osy ohodnocení jednotlivých sloupců.

V nápovědě pro druhý krok bylo třeba upozornit na následující:

- Princip podle něhož se třídí tvrzení do sloupců s různým ohodnocením
- Omezení maxima karet ve sloupcích výsledkové mřížky
- Zvětšení karty při najetí myši
- Označení aktivní sady pro funkci přesouvání karet kliknutím
- Možnost přesunu karty kliknutím na sloupec výsledkové mřížky q-setu

Provedení nápovědy způsobem vyznačení prvků uživatelského rozhraní tak, jak popisuje obrázek 5.5 se při testování ukázal jako nedostačující. Uživatelé dokázali nápovědu dobře použít pro hledání popisu konkrétního prvku nebo funkcionality. Pro objasnění principu q-řazení se však toto provedení ukázalo nepoužitelné. Důvodem bylo to, že na takto vypadající nápovědu nahlíželi uživatelé jako na několik samostatně stojících informací o prvcích uživatelského rozhraní. Neviděli v ní jeden logický celek objasňující práci s aplikací. Z toho důvodu byl do nápovědy v obou krocích q-řazení přidán i krátký textový popis s nadpisy: „Co mám dělat?“ a „Jak to udělat“. Ty v několika bodech shrnují nejzákladnější informace o principech q-řazení i o prvcích uživatelského rozhraní.

Takto obsáhlá nápověda ve druhém kroku q-řazení se však během testů neosvědčila z důvodu nepřehlednosti. Velké množství informací na jedné obrazovce odradilo většinu testujících ihned po jejím otevření. Proto jsem ji zde rozdělil na dvě samostatné části,



Obrázek 5.6: Textová část nápovědy ve druhém kroku q-řazení, umístěná ve ztmaveném prostoru obrazovky (Její špatná čitelnost se v prostředí aplikace neprojevuje)

mezi kterými se přechází dobře viditelným textovým odkazem. První z nich obsahuje výše zmíněný textový popis. Druhý pak vyznačení stěžejních prvků s krátkými popisky.

I skvěle propracovaná nápověda je však často zbytečná. Během testování jsem totiž zjistil, že uživatelé si zvykli ji ani neotevírat. Tento problém jsem vyřešil tak, že jsem na navigační panel aplikace umístil ikonku nápovědy v podobě otazníku. Tu nechám po načtení obsahu okna několikrát probliknout výrazně modrou barvou. Tím přiláká pozornost zraku uživatele, který může při prvních pochybách o správnosti prováděného řazení zkusí spíš otevřít nápovědu, protože na ni byl předem upozorněn.

5.3 Implementace serverové části a interaktivního rozhraní aplikace

Výsledná aplikace pro q-řazení má za cíl nabídnout možnost snadného uspořádání vlastního q-řazení. K tomu bylo potřeba nejen vytvořit uživatelsky přívětivé provedení metody q-řazení, ale také umožnit snadné nahrání dat a správu průběhu q-řazení. Vytvořil jsme tedy základní *backend* aplikace. Za backend je terminologií informačních technologií považováno programové pozadí aplikace, tedy to co uživatel v aplikaci využívá, ale nevidí. Pro usnadnění případné budoucí práce na aplikaci jsem vytvořil základní backend tak, aby byla snadná jeho výměna nebo rozšiřitelnost.

Aplikace funguje tak, že je možné v nabídce jejího menu založit nový průzkum pomocí q-řazení nebo prohlížet seznam existujících průzkumů. Pro založení průzkumu je potřeba pouze nahrání souboru ve formátu TXT, který obsahuje tvrzení q-setu oddělená prázdným řádkem. Tímto je založen průzkum, na který je vygenerován unikátní odkaz. Ten je vždy možné dohledat pod položkou „probíhající průzkumy“ z menu vrchního panelu aplikace. Vstupem na odkaz je uživatel nasměrován na první krok q-řazení s příslušným q-setem. Po dokončení prvního i druhého kroku q-řazení a potvrzení výsledků je do databáze zaznamenána podoba seřazené výsledkové mřížky.

Ta je uložena ve formátu JSON, protože se jedná o statická data, u kterých nebude docházet k žádným změnám a při práci s nimi nebude potřeba vyhledávat jednotlivé zá-

znamy podle obsahu uložených dat. Tento způsob uložení jsem zvolil především pro možnost budoucího ukládání a nahrávání stavu aplikace. Ten je totiž zachycen jako pole obsahující identifikátory HTML elementů jednotlivých obsahových prvků uživatelského rozhraní, které jsou oddělovačem pro textová tvrzení, jež obsahují. Z takového způsobu uložení je snadné v cyklu vytvořit původní podobu rozpracovaného q-setu.

V databázi aplikace bylo potřeba vytvořit tabulku pro záznamy jednotlivých průzkumů, která obsahuje identifikátor průzkumu, unikátní odkaz na průzkum, a q-set s tvrzeními. Identifikátor průzkumu je cizím klíčem do tabulky výsledků q-řazení, která obsahuje jedinou datovou položku, řetězec v JSON se zakódovaným výsledkem q-řazení.

Tato podkapitola se dělí na dvě části. První se zaměřuje na implementaci té části backendu starající se o organizaci navržené funkčnosti aplikace. Druhá popisuje zajištění funkcí uživatelského rozhraní, jako je logika funkcí karty s tvrzeními nebo generování výsledkové mřížky q-řazení.

5.3.1 Práce s daty aplikace pro q-řazení

Pro organizaci aplikace bylo potřeba zajistit její propojení s databází k uchování dat probíhajících průzkumů q-řazení. Pro práci s MySQL databází jsem využil PHP, které pro tento účel disponuje velmi dobře provedenou nástavbou PDO. Před jejím použitím však bylo potřeba vyřešit předání seřazeného q-setu z HTML do PHP. K tomuto účelu se nejlépe hodí JavaScript, který je pro práci s *HTML DOM* navržen. HTML DOM lze definovat jako standard pro manipulaci s HTML elementy [4]. Další, nezbytnou součástí aplikace je export výsledků seřazených q-setů z podoby, v jaké jsou ve formátu JSON uloženy v databázi, do CSV souboru. CSV je zkratkou slov Comma-separated values a jedná se o jednoduchý formát pro uložení tabulkových dat [23]. CSV je standardně využívaným formátem pro přenos elektronické podoby výsledků q-řazení, který je požadován jako vstupní formát pro analyzátoři dat.

Způsob uložení výsledků s použitím JavaScriptu a PHP

Jako první je potřeba vyextrahovat data z HTML DOM do jejich textové podoby a tu uložit do proměnné typu pole. To je v JavaScriptu tvořeno posloupností hodnot, oddělených čárkami a indexováno vždy číselně od nuly. Sběr tvrzení z HTML elementů probíhá postupně, od nejnižšího ohodnocení sloupce výsledkové mřížky q-řazení až po ten s nejvyšším ohodnocením. Protože je mřížka generována s proměnlivým počtem sloupců, byly při jejím vytváření zaznamenány identifikátory jednotlivých sloupců do pomocného pole. To je při sběru výsledků procházeno a pomocí JavaScriptové metody *getElementById()* jsou s využitím identifikátorů získány konkrétní HTML elementy.

Pro uložení výsledného seřazeného q-setu jsem použil další proměnnou typu pole. Do ní je nejprve zapsán identifikátor sloupce a poté postupně všechna tvrzení získaná z HTML elementů karet, které sloupec obsahuje. Ty lze z obalového prvku (sloupce) snadno získat použitím JavaScript atributu *childNodes*, který vrací pole potomků vybraného HTML objektu.

Nakonec je na získané pole s výsledky aplikována metoda *JSON.stringify()*, která převede hodnotu předanou JavaScriptem do ekvivalentní podoby ve formátu JSON řetězce.

Protože JavaScript nedisponuje prostředky pro práci s databází, je potřeba zajistit předání řetězce s výsledky do PHP skriptu, který komunikaci obstará. Pro tento účel se nejlépe hodí využití asynchronního HTTP požadavku. Jednou z metod pro jeho provedení je metoda *ajax()* z knihovny jQuery. Ta poskytuje lepší kontrolu provádění požadavku, než ostatní

metody za tímto účelem implementované. Při zavolání metoda *ajax()* přijímá jako parametr URL, specifikující cíl volání, a objekt pro nastavení dalších parametrů požadavku. V něm je možné například specifikovat jeho typ nebo určit data předávaná s využitím HTTP metody POST. Zavoláním metody *ajax()* s potřebnými parametry jsem tedy předal do PHP skriptu výsledky seřazeného q-setu, identifikátor řazení ke kterému výsledky přísluší a rozměry použité výsledkové mřížky q-řazení.

V PHP jsou data snadno k vyzvednutí ze superglobální proměnné `$_POST` kterou HTTP požadavek naplnil. Skript se poté postará o spojení s databází a pomocí *Prepare Statement* vloží data do příslušné tabulky databáze. Prepare Statement je technika opětovného použití připraveného SQL dotazu, do kterého je možné dynamicky vkládat různá data k uložení do databáze. Vkládané proměnné jsou v textu SQL dotazu zastoupeny pomocí otazníků nebo jiných řetězcových literálů, podle použité nastavby pro komunikaci s MySQL. Já jsem použil nastavbu PDO, která na objektu udržujícím komunikaci s databází nabízí metodu pro přípravu SQL dotazu a další metodu pro vložení proměnných do něj. Použití této techniky je výhodné jak z hlediska opětovného použití kódu. Nastavba PDO také sama zajišťuje ochranu před útokem SQL injection.

Skript pro export dat do CSV

Pro přenos výsledků metody q-řazení je potřeba zajistit export z formy, v jaké jsou uloženy v databázi, do CSV souboru. Za tímto účelem jsem napsal malý skript formátující data do podoby, kterou na vstupu očekávají existující analyzátoři výsledků metody q-řazení [26]. Ty přijímají výsledky v tabulkové podobě, kde první sloupec obsahuje textová tvrzení q-setu a každý další sloupec představuje výsledek jednoho q-řazení, kde je každému tvrzení přiřazena číselná váha podle toho, do kterého sloupce výsledkové mřížky q-řazení uživatel tvrzení umístil. Záhlaví (neboli první řádek) tabulky obsahuje identifikátory jednotlivých q-řazení.

V databázi jsou výsledky průzkumu uloženy tak, že každý jeden výsledný q-set je zakódovaný v JSON řetězci v samostatném řádku tabulky výsledků spolu s identifikátorem průzkumu, ke kterému náleží. SQL dotazem na databázi tak získám pole, kde každý jeho prvek představuje jeden řádek v tabulce s výsledky.

Prvním krokem k extrahování dat bylo zjištění čísla pro nejzáporněji ohodnocený sloupec výsledkové mřížky a jeho uložení do proměnné. Tento údaj byl vypočítán z rozměru použité mřížky, který je uložen v databázi u příslušného q-setu. Druhým krokem byl sběr jednotlivých tvrzení q-setu. Ty jsem získal z prvního seřazeného q-setu a umístil do vícerozměrného pole uchovávacího již zpracované výsledky. Každé tvrzení je v něm zastoupeno svým text a dalším polem, v němž jsou uloženy ohodnocení daného tvrzení v jednotlivých vypracovaných q-řazeních.

Pro převod dat řetězce ve formátu JSON jsem použil PHP funkci *json_decode()*. Jejím výsledkem je pole složené z prvků předaného JSON řetězce. To následně v cyklu procházím a kontroluji obsah každého prvku. Pokud narazím na identifikátor sloupce výsledkové mřížky q-řazení, inkrementuji hodnotu proměnné uchovávací váhu aktuálně zpracovávaného sloupce. Pokud prvek obsahuje text s tvrzením, postupně procházím pole se zpracovanými výsledky, dokud nenarazím na pozici, ve které je uloženo dané tvrzení. K němu pak přidám váhu aktuálně zpracovávaného sloupce.

Tímto postupem v podstatě do pole se zpracovanými tvrzeními vytvářím reprezentaci jednotlivých řádků výsledné CSV tabulky. Tu po zpracování všech výsledků průzkumu vytisknu do CSV souboru pomocí PHP funkce *fwrite()*. Posledním krokem je nabídnutí vy-

tvořeného CSV souboru uživateli ke stažení. To se v PHP zprostředkuje zasláním příslušné HTTP hlavičky, k čemuž slouží funkce *header()*.

5.3.2 Zajištění funkčnosti uživatelského rozhraní

Nejpoužívanější funkcionalitou aplikace je bezpochyby drag&drop. Proto je potřeba práci s ním perfektně odladit, aby nedocházelo k žádným rušivým událostem během třídění karet s tvrzeními. Pro realizaci jsem využil jednu z existujících drag&drop knihoven, kterých je v JavaScriptu a jQuery na výběr mnoho. Má volba padla na knihovnu Dragula, která nabízí dobře odladěné provedení s příjemným API zaměřeným na snadné napojení vlastních funkcionalit. Právě to byl hlavní důvod k jejímu zvolení. Karty s tvrzeními totiž ve druhém kroku řazení vyžadují přidání efektu zvětšení karty po najetí myši, omezení počtu karet v některých drag&drop kontejnerech a implementaci alternativního způsobu jejich přemísťování při zachování drag&drop funkčnosti.

Zvětšování prvku karty po najetí myši

Pro zvětšování elementu webových stránek existuje několik způsobů, ze kterých jsem vybral metodu *animate()* JavaScriptové knihovny jQuery. Ta umožňuje provádění jednoduchých animací s použitím CSS vlastností, přičemž animace jsou plynulé i na pomalejších zařízeních. Metodu *animate()* jsem zvolil především proto, že disponuje možností snadného napojení navazujících JavaScriptových funkcí.

Této výhody jsem však při finální implementaci plně nevyužil, a proto bych do budoucna způsob zvětšování vyměnil za CSS3 vlastnost transformace a její metodu *scale()*. JavaScriptová animace totiž pracuje s elementem karty a textu v ní zvlášť, což má za následek nehezky efekt přeskakování slov v řádcích při jejím zvětšování. CSS3 *scale()* využívá parametru poměru zvětšení, díky kterému roste text v kartě rovnoměrně s její velikostí. To odstraňuje rušivý efekt „uskakování textu“.

Protože ve druhém kroku q-řazení jsou rozmístěny karty s funkcností zvětšování se po většině plochy obrazovky, bylo potřeba implementovat zpoždění mezi najetím myši na kartu a jejím zvětšením. Pro simulaci zpoždění nabízí JavaScript metodu *setTimeout()*, která po předaném časovém intervalu zavolá požadovanou funkci. Při jejím použití však vznikaly problémy s nulováním počítadla času a proto jsem hledal alternativní řešení tohoto problému. Finální volbou byl jQuery plugin *hoverIntent*. Ten nabízí rozhraní pro správu událostí po najetí kurzoru myši a přidává možnosti jako nastavitelné zpoždění, prevence události při posunu kurzoru pryč z cílové plochy, nulování časovače v případě pohybu myši přes větší plochy a delegování události.

Zajištění plynulosti drag&drop v navrhovaném rozhraní

Ačkoli nabízí knihovna Dragula možnost snadného přidání vlastních funkcností, neobešla se jejich implementace bez problémů. API knihovny Dragula generuje během práce události a poskytuje rozhraní k jejich zachycení. Na události je pak snadné navázat vlastní naprogramovanou funkcionalitu. Ne ve všech případech užití však knihovna Dragula reagovala tak, jak by se dalo očekávat. Šlo vždy jen o drobné opravy v komunikaci s knihovnou, zato jejich množství bylo značné. Nezřídka se však ukázalo, že problém nevznikal v API knihovny, ale kvůli mnou opomenutým důsledkům prováděného úkonu. Významnost těchto problémů nespočívala v komplikovanosti jejich oprav, ale v tom, že většina z nich vznikala sporadicky, a proto bylo časově náročné jejich odhalování.

Při implementaci druhého kroku q-řazení jsem narazil na potřebu zabránit umístění karty do již plného sloupce. Knihovna Dragula nabízí možnost definovat uživatelskou funkci která rozhodne, zda může být zahájen pohyb s elementem. Nenabízí však žádné možnosti toho jak za určitých podmínek zamezit uvolnění elementu na dané pozici. K tomu účelu jsem byl nucen zasáhnout do zdrojového kódu knihovny a upravit příslušné funkce tak, že v případě kdy je element kontejneru pro drag&drop větší než obalový element, nedojde k události uvolnění tažené karty.

Dynamické generování výsledkové mřížky q-řazení

Podoba výsledkové mřížky q-řazení je podmíněna lichým počtem sloupců a rozložením odpovídajícím Gaussově křivce. Metoda však umožňuje různé velikosti q-setu s tvrzeními, a proto je potřeba jejich počtu přizpůsobit počet míst ve výsledkové mřížce.

Této skutečnosti bylo možné využít pro implementaci jejich dynamicky se měnící šířky a výšky podle počtu karet s tvrzeními. Tím jsem dosáhl toho, že karty ve sloupcích mohou nabývat větších rozměrů a tak jsou při menším q-setu všechny průměrně dlouhá tvrzení čitelná i bez nutnosti zvětšování karty.

Šířka sloupců v generované výsledkové mřížce odpovídá dostupnému prostoru pro mřížku poděleného počtem sloupců v ní. Od výsledku je ještě třeba odečíst šířku rozteče mezi sloupci pro každý z nich. Počet sloupců výsledkové mřížky je přiřazen každému q-setu podle počtu tvrzení v něm. Výška jednotlivých sloupců se stanoví pomocí výšky karty s tvrzením a jejich počtu v jednotlivých sloupcích.

Výška karty je vypočítána na základě maximálního počtu karet v nejvyšším sloupci výsledkové mřížky. Pro q-sety podporované navrženým uživatelským rozhraním je to maximálně devět karet. Při devíti kartách tedy můžu stanovit fixní výšku karty v procentech obrazovky tak, aby se vše vešlo na plochu zobrazovacího zařízení.

Pro menší q-sety, které v nejdelším sloupci neobsahují všech devět karet je potřeba rozdělit uvolněný prostor tak, aby bylo plně využito výšky zobrazovacího prostoru. Toho jsem dosáhl odečtením maximálního počtu karet v nejvyšším sloupci daného q-setu od jeho maximálního zaplnění, tedy devíti karet. Po vynásobení výsledku výše stanovenou fixní výškou karty získám velikost neobsazeného prostoru sloupce v procentech. Spočítanou uvolněnou plochu pak v každém sloupci rozdělím mezi jednotlivé karty.

Pro rozdílné rozměry výsledkové mřížky bylo také potřeba zajistit rozdílný počet odstínovaných sloupců. K tomu účelu se v CSS hodí způsob definování pozadí pomocí *HSLA*. To definuje barvu za pomoci čtyř atributů [11]. Podle její pozice na kruhové paletě barev, saturace, světlosti a složky průhlednosti (v rozmezí 0–1). Tím zprostředkovává intuitivnější manipulaci s barvou při ladění vzhledu a umožňuje definovat číselnou proměnnou obsahující světlost barvy, která se v cyklu generování sloupců výsledkové mřížky přepočítává.

Kapitola 6

Testování a ověření použitelnosti vytvořené aplikace pro q-řazení

Aby byla vytvořená aplikace použitelná jako ekvivalentní varianta výzkumu metodou q-řazení, bylo nutné ji vyvíjet v souladu s potřebami potencionálního uživatele. Tím může být prakticky kdokoli, kromě starších osob, které neumí využívat ani základní možnosti informačních technologií. K realizaci takové aplikace je tak potřeba důkladné otestování její použitelnosti. Na velké testování aplikace však nedisponuji dostatečnými prostředky, proto jsem hledal způsob jak otestovat aplikaci s menším počtem lidí. Řešení této situace jsem našel v internetovém článku pojednávajícím o tom, že s každým dalším testujícím se množství nových zjištění snižuje, protože různí uživatelé pracují do jisté míry podobně [18].

Našel jsem tedy 20 dobrovolníků, kteří svolili s otestováním mnou navrhovaného rozhraní. Cílem aplikace bylo navrhnout co nejintuitivnější uživatelské rozhraní, aby i osoba, která nezná metodu q-řazení, jej sama zvládla správně dokončit. Testování takového návrhu uživatelského rozhraní vyžaduje, aby nebylo testováno víckrát na stejných osobách. Proto jsem respondenty rozdělil do několika kol testů v průměru po pěti osobách na jedno kolo. V některých případech je však nutné testovat jak na nových uživateli, tak na těch, kteří již q-řazení vypracovali. Detaily provádění takových testů obsahuje tato kapitola.

6.1 Iterativní testování prototypů aplikace v průběhu vývoje

Možnosti vývoje informačních technologií prošly s rozšířením internetu velkými změnami, především v oblasti UX designu. Je totiž tak snadné, levné a rychlé provádět testování, jako tomu nebylo nikdy dříve [13]. O to více platí tento poznatek u aplikací a webů v prostředí internetu.

Při návrhu dobré aplikace je potřeba postupovat tak, aby co nejlépe vyhovovala těm, kteří ji budou používat. Nejjednodušeji toho lze dosáhnout neustálým přizpůsobováním navrženého rozhraní a jeho funkcí potřebám uživatele [8]. Ty je sice možné do jisté míry odhadnout, vždy je však důležité otestovat, že jejich řešení je použitelné tak, jak bylo při návrhu předpokládáno. Testování zároveň napoví, co by dále mohl uživatel při své práci využít nebo jak lépe přizpůsobit prvky, se kterými pracuje jinak, než jak bylo navrženo.

Krom samotného otestování aplikace je důležitým přínosem tohoto přístupu možnost interakce s širokým množstvím uživatelů. Každý z nich vidí aplikaci unikátním pohledem. Pomocí přátelského dialogu na téma chyb a nedokonalostí uživatelského rozhraní je tak možné odhalit skutečnosti, na které by sám vývojář nebo jiní uživatelé nepřišli. Během

testování jsem proto vždy naslouchal radám a nápadům testujících na opravu nebo vylepšení aplikace. Stejně tak jsem využil možnosti konzultací s vedoucím mé bakalářské práce. Ten mi poskytl množství rad a postřehů nabitých zkušenostmi v oboru uživatelských rozhraní. Podněty získané od vedoucího nebo uživatelů jsem vždy podrobil dalšímu uživatelskému testování a ověřil tak jejich využitelnost.

První část této podkapitoly popisuje testování během vývoje aplikace a poznatky, které přineslo. Jednotlivé funkce rozhraní byly testovány prototypově. Nejdříve jsem tedy implementoval jejich základní provedení a bez dlouhého ladění je nechal otestovat uživateli. Z jejich reakce bylo možné rozpoznat, zda jim vyhovuje základní myšlenka provedených inovací. Pokud ano, mohl jsem začít s implementací jejich detailů. V případě, že uživatelé nezareagovali tak kladně, jak bylo předpokládáno, bylo potřeba nové prvky nahradit nebo přepracovat.

V druhé a třetí části podkapitoly je popsán proces testování vizuální podoby aplikace. Ten je rozdělený na testování procesu q-řazení a testování nápovědy objasňující principy této metody a možnosti uživatelského rozhraní. Tyto aspekty byly testovány především formou dialogu s uživateli, založeného na podnětech od nich obdržných a na poznatcích získaných pozorování způsobu jejich práce s aplikací.

6.1.1 Testování intuitivního pochopení práce s uživatelským rozhraním

Pro snadnou a intuitivní práci s uživatelským rozhraním je výhodné od začátku vývoje sledovat způsob, jakým uživatelé aplikaci používají. Tím jsem získal nejen informace o chybách v návrhu nebo implementaci, ale především jsem ze způsobu práce mohl odvodit vylepšení, která by uživatelé během q-řazení využili.

Větší celky testování jsem proto prováděl vždy po dokončení úprav, které jsem navrhl na základě výsledků z předchozích testování a požadavků testujících uživatelů. Takto jsem měl možnost pozorovat, které z realizovaných změn dobře fungují a co je ještě potřeba doladit nebo vyměnit.

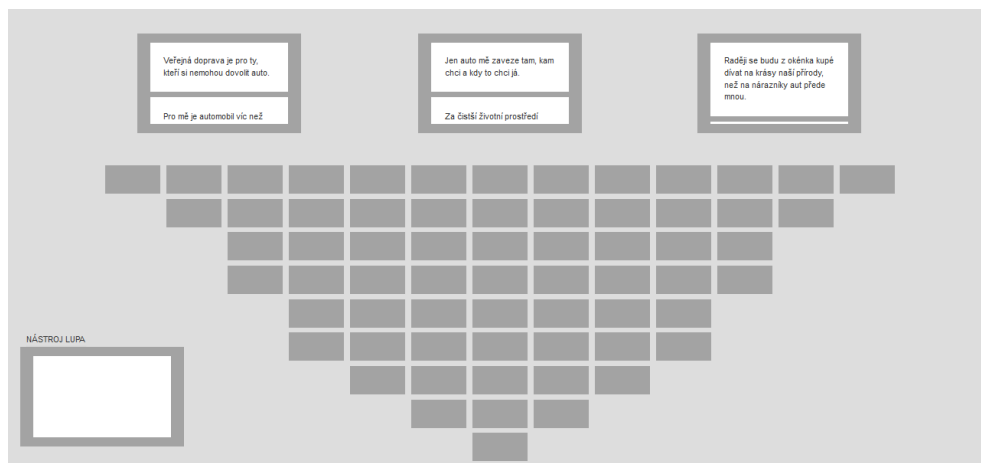
První kolo testování

Prvním kolem testování jsem se zaměřil na ověření správnosti rozložení základních prvků q-řazení na zobrazovací ploše. Bylo potřeba ověřit, zda jsou často používané prvky snadno dosažitelné a také, jestli při práci s nimi nevznikají žádné komplikace týkající se jejich rozložení.

Na základě myšlenky iterativního vývoje aplikace proběhlo první testování na jejím základním prototypu. Ten obsahoval pouze několik šedých obdélníků reprezentujících výsledkové mřížky jednotlivých kroků q-řazení a bílé obdélníky s textem reprezentující karty s tvrzeními. V aplikaci také fungovalo základní provedení drag&drop a jednoduchá lupa, bez které by nebyl druhý krok q-řazení realizovatelný. Ta zobrazovala text elementu, na kterém je umístěn kurzor do pevně ukotveného prvku lupy.

Testování bylo provedeno pěti účastníky, na q-setu o padesáti tvrzeních týkajících se způsobů dopravy. Takto velký q-set jsem zvolil proto, abych viděl reakci uživatelů v případě, kdy navržená výsledková mřížka zabere většinu plochy zobrazovacího zařízení.

Před samotným provedením q-řazení jsem každému účastníkovi vysvětlil způsob, jakým se provádí a vyzval jej, aby mi v průběhu testování sdělil každý dotaz nebo připomínku. Během jeho práce jsem si dělal poznámky na základě jeho připomínek a pozorováním způsobu, jakým jednotliví účastníci pracovali s uživatelským rozhraním.



Obrázek 6.1: Snímek testování prvního prototypu aplikace v průběhu druhého kroku q-řazení

Výsledky takto provedeného testování přinesly několik poznatků o drobných rušivých nedostatcích aplikace. Mezi ně se řadil například kurzor myši v podobě šipky použitý během přetahování karet, namísto kterého by uživatelé očekávali kurzor v podobě ruky. Dále pak formulace „Nevím jistě“ nad sloupcem pro neutrálně zařazená tvrzení nebo neestetický vzhled dočasně použitého nástroje lupy.

Testování přineslo také dvě důležitější zjištění. Jedním z nich byla skutečnost, že testující uživatelé často zapomínali, který sloupec slouží pro kladná a který pro záporná vyjádření. Sloupce by tedy bylo potřeba odlišit i jinak než textovým popiskem nad nimi. Druhým důležitým poznatkem byla značná nepřehlednost výsledkové mřížky q-řazení ve druhém kroku, na kterou mně upozornili čtyři z pěti testujících uživatelů.

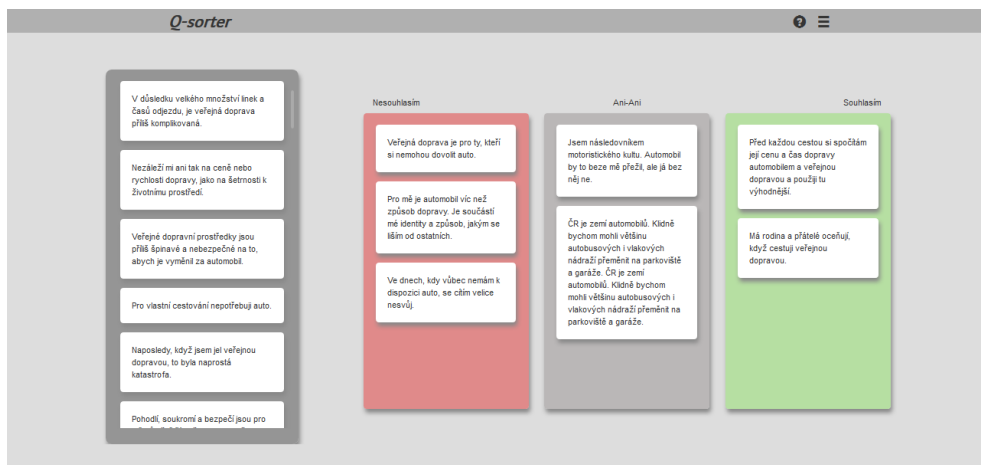
Druhé kolo testování

V tomto kole testování jsem se zaměřil na funkčnost úprav rozhraní provedených na základě výsledků z prvního kola, na zjištění nedostatků nově přidávaných prvků a funkcí rozhraní a na pochopení principu q-řazení bez nutnosti jeho vysvětlování. Průběh testování byl stejný jako v jeho prvním kole s tím rozdílem, že uživatelé dostali pouze pokyny k seřazení tvrzení podle toho, jaký na něj mají názor. To, že jim nebyl objasněn princip druhého kroku q-řazení způsobilo, že testující museli sami pochopit jak karty v něm správně organizovat. Takto bylo nejlépe možné otestovat intuitivní pochopení principů q-řazení na základě navrženého uživatelského rozhraní.

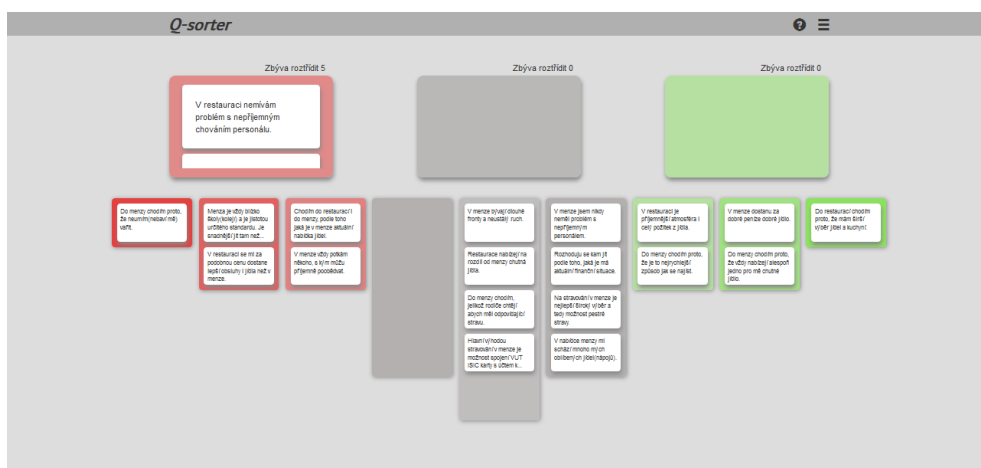
Tři z testujících byli stejní jako v prvním kole testování, další tři byly osoby, které mnou navržené rozhraní předem neviděli a stejně tak předtím neznali metodu q-řazení. Testování proběhlo na q-setu o čtyřiceti a jednom tvrzení, protože vysoký počet tvrzení v prvním kole testování se pro respondenty ukázal jako faktor odrazující od dokončení q-řazení.

Podobu aplikace během tohoto kola testování můžete vidět na obrázcích 6.2 a 6.3. Elementy uživatelského rozhraní již dostaly příjemnější grafickou podobu než v prvním kole. Byly přidány barvy pro odlišení kategorií výsledkových sloupců a výsledková mřížka q-řazení ve druhém kroku dostala novou podobu. Také přibyla funkce zvětšování karty po přejetí kurzoru a počítadla množství karet, které zbývají ve druhém kroku q-řazení rozřadit.

Rozdělení karet v prvním kroku q-řazení pochopili testující rychle a bez problémů. Bě-



Obrázek 6.2: Podoba aplikace v druhém kole testování. Snímek z prvního kroku q-řazení



Obrázek 6.3: Zachycení druhého kroku q-řazení v momentě, kdy uživatel naplnil všechny sloupce stejné barvy a neví co se zbylými kartami

hem třídění si všichni počínali rychleji než v předešlém kole testování. Důvodem bylo přidání barevného odlišení kategorií ve výsledkových sloupcích, díky kterému nemuseli hledat textový popis objasňující význam sloupce.

Ve druhém kroku postupovala ze začátku většina testujících stejnou rychlostí. Dva z uživatelů, kteří prováděli q-řazení poprvé, však neřadila karty do sloupců podle míry nesouhlasu a souhlasu s tvrzením, ale jen na základě barevného odlišení. Tito uživatelé se zastavili v momentě, kdy jim došel prostor pro umísťování karet do stejně zabarvené kategorie. Po krátkém přemýšlení však pochopili význam různých odstínů jednotlivých sloupců stejné barvy a byli nuceni přeorganizovat již rozdělené karty s tvrzeními.

Funkci zvětšování malé karty výsledkové mřížky ve druhém kroku objevili všichni testující uživatelé intuitivně. Vzniklo však množství problémů způsobených jejím špatně odladěným propojením s drag&drop knihovnou. Posledním významným poznatkem bylo zjištění, že uživatelé, kteří nejsou zvyklí na každodenní práci s myší, trpěli při přesouvání velkého počtu karet křečemi nebo bolestmi zápěstí. Proto bylo potřeba přidat další způsob přemisťování karet.

Třetí kolo testování

V tomto závěrečném kole testování jsem pěti osobám, které předem neznali metodu q-řazení ani podobu mnou navrženého uživatelského rozhraní, otevřel URL adresu aplikace a bez jakéhokoli vysvětlování je nechal provést q-řazení. Účastníci řadili q-set o třiatvaceti tvrzeních na téma stravování na vysoké škole.

Aplikace již měla finální podobu, do které od minula přibyla číselná osa pro znázornění rozdílného ohodnocení sloupců v jednotlivých kategoriích a vrchní panel s logem aplikace, nápovědou a drobným vysouvacím menu. Přibyla také funkce přesouvání karet do výsledkových sloupců na kliknutí myši a k ní se vázající označení aktivní sady ve druhém kroku q-řazení. Rovněž byly opraveny nedostatky způsobu zvětšení malé karty ve výsledkové mřížce q-setu.

V průběhu testování jsem nezaznamenal žádné problémy s funkčností jednotlivých prvků ani podněty od testujících uživatelů směrem k jejich rušivému nebo nepříjemně působícímu vlivu. Tři z pěti uživatelů ihned správně třídili karty v prvním i ve druhém kroku q-řazení. Zbylí dva měli ve druhém kroku řazení stejný problém jako ve druhém kole testování, tedy neřadili od začátku karty do sloupců podle míry nesouhlasu (souhlasu) s tvrzeními. Jeden z nich si své chybné počínání uvědomil již během prvních pár zařazených karet, když si všiml číselné osy. Druhý přesně odpovídal scénáři z minulého kola testování. Důvodem bylo podle jeho slov to, že nad dalším tříděním nepřemýšlel, když už jej v prvním kole jednou provedl.

Výsledky testování

Iterativním testováním v průběhu vývoje jsem dosáhl toho, že špatně fungující prvky uživatelského rozhraní jsem nahradil nebo upravil již na začátku jejich vývoje. Díky tomu jsem nemusel investovat mnoho času do vytváření něčeho, co bych pak musel kvůli výsledkům testů odstranit. Další výhodou bylo, že jsem měl již od začátku možnost sledovat, jak různí uživatelé pracovali s navrženým uživatelským rozhraním. Tyto poznatky jsem mohl rovnou uplatnit při vývoji aplikace, čímž se urychlil celý jeho proces.

Hlavním výsledkem testování bylo zjištění, že naprostá většina uživatelů dokáže bez pomoci správně vyhotovit q-řazení a to dokonce v čase rychlejším než jeho provedení s fyzicky uchopitelnými kartičkami. Dalším výsledkem testování pak bylo především ověření toho, že všechna navržená funkcionalita je bez problémů intuitivně použitelná.

6.1.2 Testování grafické podoby uživatelského rozhraní

Vzhled aplikace byl testován společně s intuitivním pochopením práce s uživatelským rozhraním. To z toho důvodu, že podoba navržených prvků úzce souvisí s pohodlností a intuitivností jejich používání [22]. Po každém kole byly opraveny zjištěné funkční nedostatky prvků aplikace a spolu s tím provedeno jejich grafické odladění. V dalších kolech testování pak bylo ověřeno, že podoba je uživatelsky přívětivá a že zapadá do celkového dojmu uživatele z aplikace.

Testování grafické podoby probíhalo tak, že jsem nejprve nechal účastníky provést kompletní q-řazení, a poté jsem jim položil následující otázky:

- Jak na vás působí vzhled aplikace?
- Byly všechny texty v aplikaci dostatečně čitelné?

- Odpovídalo podle vás provedení prvků aplikace jejich účelu?
- Máte jakékoliv další připomínky ke vzhledu aplikace?

Pro hodnocení vzhledu bylo zajímavé jen druhé a třetí kolo testování. To bylo způsobeno tím, že v prvním kole bylo testováno na prototypu aplikace bez vyladění jeho grafické podoby. Vzhled tak uživatelé hodnotili jako nedokončený a nepromyšlený. To zejména ve druhém kroku q-řazení, kdy všechny testující od práce s aplikací ihned odradila podoba výsledkové mřížky. Na jedné obrovce bylo podle testujících uživatelů příliš mnoho prvků

Odpovědi v následujících kolech testování se z velké části shodovaly. Shrnuje je následující přehled:

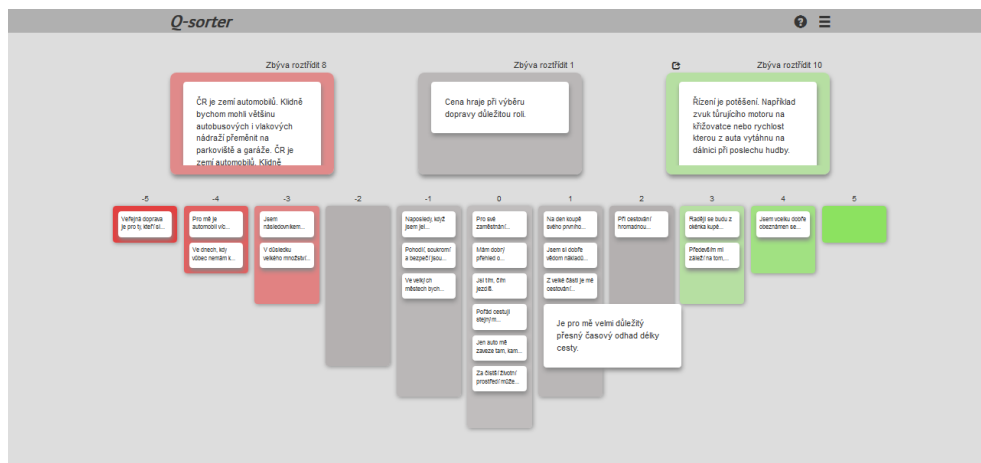
- Vzhled aplikace na mně působil neutrálním až monotónním dojmem.
- Vzhledově mě aplikace neuchvátila, ale ani jsem na ní nenašel něco, co by se mi nelíbilo.
- Jediným graficky výrazným prvkem je pro mě blikající ikonka otazníku.
- Texty v kartách byly bez problému čitelné. Až na ty, které byly umístěné ve výsledkové mřížce q-řazení.
- Popisky prvků v netextové části nápovědy se mi zdají málo výrazné.
- Aktivní sloupec ve druhém kroku řazení je špatně označený.
- Práci s aplikací jsem bez problému pochopil a působila na mně spíš jako hra, než provádění průzkumu.

Reakce uživatelů na grafickou podobu aplikace byly veskrze pozitivní. Většinou ji sice nehodnotili jako graficky propracovanou, jednoduchý vzhled však byl cílem návrhu. Někteří uživatelé ocenili to, že provádění q-řazení v ní působí spíš dojmem hry, než vypracovávání průzkumu. Jediným nedostatkem zůstalo pro některé uživatele až moc pohodlné použití různých odstínů barev kategorií ve sloupcích výsledkové mřížky druhého kroku q-řazení.

6.1.3 Testování použitelnosti nápovědy

Nápověda byla testována dvěma respondenty, kteří již dříve vypracovávali q-řazení v prvním kole testování intuitivního pochopení uživatelského rozhraní a třemi novými uživateli. Před vstupem do aplikace nedostali uživatelé žádné pokyny k tomu jak během třídění postupovat. Na začátku každého kroku q-řazení jsem uživatele požádal o to, aby před tříděním prostudovali nápovědu a vyjádřili se k její formě a obsahu. Přesto, že uživatelé k otevření nápovědy nebyli příliš ochotní, po jejím prostudování konstatovali, že její použití v prvním kroku q-řazení je rychlé a snadné.

Nápověda ve druhém kroku q-řazení se již většině testujících tak přívětivá nezdála, protože obsahovala na jedné obrazovce příliš mnoho informací. To podle jejich reakcí způsobilo nepřehlednost a velmi tím snížilo její informační hodnotu.



Obrázek 6.4: Finální podoba druhého kroku aplikace pro q-řazení

6.2 Ověření relevantnosti výsledků q-řazení s použitím vytvořeného uživatelského rozhraní

Abych dokázal správnost navrženého uživatelského rozhraní, bylo potřeba provést experiment k dokázání použitelnosti aplikace i pro větší průzkumy. Také bylo nutné zjistit, jestli výsledky získané q-řazením v aplikaci odpovídají těm, provedeným v papírové podobě s uchopitelnými kartami. Nejlépe by se funkčnost rozhraní ověřila během dvojího provedení jednoho většího průzkumu. Bohužel jsem pro takové testování nenašel dostatek dobrovolníků a tak jsem oba aspekty porovnal ve dvou nezávislých experimentech.

Provedení experimentu q-řazení online, oproti řazení s uchopitelnými prvky

Pro ověření předpokladu, že s použitím mnou navrženého uživatelského rozhraní je možné dosáhnout obdobných výsledků, jako při použití fyzicky uchopitelných prvků, jsem provedl následující experiment. Požádal jsem čtyři osoby, které předem metodu q-řazení neznaly, aby vypracovaly jeho elektronickou i papírovou podobu se stejným tématem. Výsledky dosažené v obou způsobech jsem poté porovnal a ověřil, že navzájem korespondují. Pro tento experiment jsem připravil papírovou výsledkovou mřížku q-řazení a vytiskl třiadvacet karet s tvrzeními týkajícími se názorů na dopravu automobilem a veřejnou hromadnou dopravou.

Účastníci byli nejprve postaveni před online aplikaci bez toho, abych jim vysvětlil způsob práce s ní. V průběhu třídění kartiček jsem pozoroval, jak pracují, a zapisoval si poznámky o způsobu, kterým každý z nich prováděl třídění. Výsledky jimi seřazených q-setů jsem uložil pro finální vyhodnocení.

Po třech dnech jsem stejným osobám dal stejná tvrzení na vytištěných kartách a požádal je o provedení q-řazení v papírové podobě. Průběh jejich práce jsem si opět zapisoval a nakonec jsem vyfotil seřazenou výsledkovou mřížku pro následné porovnávání.

Výsledkem bylo, že u všech účastníků byly v obou q-setech zařazeny téměř stejně karty v extrémech, zatímco zařazení karet ve středu mřížky bylo značně odlišné. Abych se ujistil, že takto rozdílně mají výsledky vypadat, požádal jsem po dalších 2 dnech tyto účastníky o opětovné seřazení q-setu v papírové podobě. Porovnáním všech výsledků jsem zjistil, že q-sety byly ve všech třech verzích vypracovány s podobnou chybou. V extrémech se

objevovaly vždy stejné karty, zatímco střed q-setu se více lišil, průměrně v devíti jinak zařazených kartách.

Ze zápisů o rozdílu během provádění dvou variant q-řazení jsem zjistil, že všichni účastníci experimentu pracovali s aplikací trochu jinak, než s uchopitelnými kartami. To se projevilo zejména během druhého kroku q-řazení. Při třídění fyzicky uchopitelných karet respondenti ihned každou svrchní kartu z balíčku umístili do výsledkové mřížky. Teprve v momentě kdy jim došlo místo v tom sloupci, do kterého chtěli umístit další tvrzení, odložili balíček a některé již umístěné karty přeorganizovali. Tím vytvořili prostor v požadovaném sloupci, aby mohli odloženou kartu zařadit. Při použití aplikace naopak nejdříve rolovali hromádkami a vybírali ty karty, u kterých věděli kam je zařadit. Mnohem méně tak měnili pozice již umístěných tvrzení. Z této skutečnosti můžeme vyvodit, že práce s papírovými kartami byla pro uživatele příjemnější. Nebáli se, že karta nebude ihned na správném místě, protože je snadné ji vzít a položit jinam. Řazení v aplikaci pro ně tedy nebylo stejně přirozené jako použití papírového q-setu. Z rozdílného průběhu jejich řazení však můžeme usuzovat, že si uživatelé intuitivně našli způsob jak vyrovnat nejistotu ohledně umístění karty pomocí jejího snazšího výběru.

Provedení průzkumu metodou q-řazení s použitím navržené online aplikace

Prvním předpokladem ke správnému provedení průzkumu metodou q-řazení je optimálně navržený q-set s dostatečně širokou škálou názorově různých tvrzení relevantních k danému tématu. Tento q-set se sestavuje oslovením velkého množství osob a sběrem všech jejich názorů. Z nich je nakonec vybrán optimální počet těch, které se objevovaly nejčastěji nebo které jsou nejvíce přínosné pro daný průzkum.

Protože jsem k provedení takto rozsáhlého sběru dat neměl prostředky, využil jsem diskusních skupin, studentských fór a komentářů na Facebooku. Na nich jsem vyhledal všechny diskuse týkající se stravování v menzách VUT v porovnání s restaurací. Z těch jsem poté vybral co nejvíce různých názorů na to, proč studenti menzy využívají, proč se jim raději vyhnou, proč preferují restaurace, nebo co shledávají dobrého a co špatného na stravování v menzách a restauracích. Takto jsem získal přes padesát názorů na toto téma, a ty jsem následně zredukoval na dvacet tři tvrzení, která co nejuvýstižněji shrnují celou problematiku.

1. *Restaurace nabízejí na rozdíl od menzy chutná jídla.*
2. *Do menzy chodím proto, že vždy nabízejí alespoň jedno pro mě chutné jídlo.*
3. *V nabídce menzy mi schází mnoho mých oblíbených jídel (nápojů).*
4. *V menze jsem nikdy neměl problém s nepříjemným personálem.*
5. *Na stravování v menze je nejlepší široký výběr a tedy možnost pestré stravy.*
6. *V restauraci nemívám problém s nepříjemným chováním personálu.*
7. *Chodím do restaurací i do menzy, podle toho jaká je v menze aktuální nabídka jídel.*
8. *Raději se nechám obsloužit v restauraci, než abych stál dlouhé fronty na jídlo a u pokladny.*
9. *Rozhoduji se kam jít podle toho, jaká je má aktuální finanční situace.*
10. *V menze vždy potkám někoho, s kým můžu příjemně poobědvat.*
11. *V menze vždy najdou nějaké jídlo, které odpovídá mé dietě (mému jídelníčku).*

12. *V restauraci je příjemnější atmosféra i celý požitek z jídla.*
13. *V menze bývají dlouhé fronty a neustálý ruch.*
14. *Do menzy chodím, jelikož rodiče chtějí abych měl odpovídající stravu.*
15. *Menza je vždy blízko školy (kolejí) a je jistotou určitého standardu. Je snadnější jít tam než hledat v každé lokalitě jinou restauraci.*
16. *V restauraci se mi za podobnou cenu dostane lepší obsluhy i jídla než v menze.*
17. *Menza je snadná volba, protože je poblíž všech fakult i kolejí.*
18. *Do restaurací chodím proto, že mám širší výběr jídel a kuchyní.*
19. *V menze dostanu za dobré peníze dobré jídlo.*
20. *Do menzy chodím proto, že je to nejrychlejší způsob jak se najíst.*
21. *Jídla v menze jsou předražena.*
22. *Do menzy chodím proto, že neumím (nebovím mě) vařit.*
23. *Hlavní výhodou stravování v menze je možnost spojení VUT ISIC karty s účtem k platbě kolejí*

Počet tvrzení bylo potřeba snížit, protože následný průzkum byl prováděn dobrovolníky. Při příliš rozsáhlém q-setu by šance na množství dokončených q-řazení prudce klesala. Tuto skutečnost lze předpokládat na základě předešlého provádění testování aplikace, kdy testující vyjádřili nelibost při řazení příliš mnoha karet. A také z vlastní zkušenosti několika nedokončených průzkumů k diplomovým pracím, jejichž dotazníky byly příliš dlouhé.

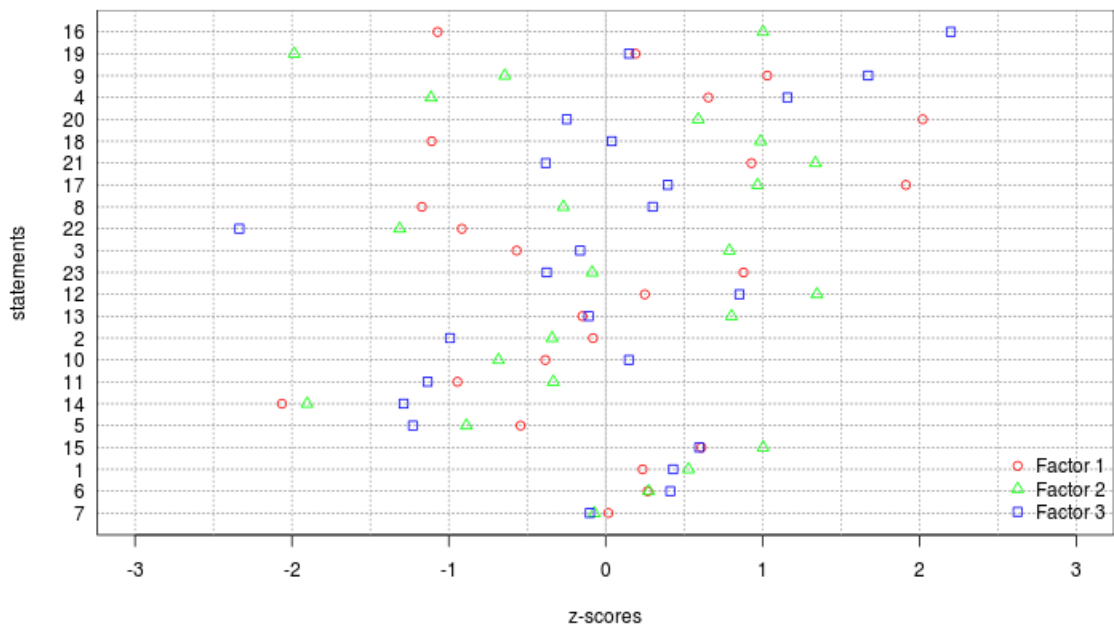
Pro získání co nejvíce výsledků se zaměřením na menzy VUT jsem využil sociální síť Facebook, kde jsem do skupiny Purkyňových kolejí a jedné ze skupin Fakulty informačních technologií uvedl příspěvek s prosbou a deset minut času na zhotovení průzkumu týkajícího se stravování v menzách a restauracích. Také jsem uživatele požádal o zpětnou vazbu k navrženému uživatelskému rozhraní.

Od účastníků tohoto průzkumu se mi nedostalo téměř žádné zpětné vazby. Tři z nich se vyjádřili k jedné objevené funkční vadě, kterou jsem při předešlém testování neodhalil. Jeden z nich uvedl, že druhý krok q-řazení byl pro něj z počátku nesrozumitelný a na správný způsob organizace karet přišel až v průběhu řazení. Tento problém byl známý již od první verze navrhované aplikace. I když se jej nepovedlo úplně eliminovat, povedlo se značně snížit počet uživatelů, kteří by nebyli schopni pro nepochopení q-řazení dokončit.

K analýze výsledků průzkumu jsem použil jediný volně dostupný online nástroj [25]. Jedná se o adaptaci balíčku qmethod, který vyvíjen Aiora Zabala jako jeden ze statistických nástrojů r-projektu. Tento online nástroj nabízí provedení analýzy z dat uložených ve formátu CSV a jejich přehledné grafické zpracování.

Z výsledků faktorové analýzy je možné vyčíst mnoho informací. Například počet faktorů, jejich charakteristiky, korelační skóre mezi nimi, hodnoty standardní chyby faktorů a mnoho dalších. Pro interpretaci výzkumu je zajímavé především faktorové skóre. To představuje pozice umístění jednotlivých tvrzení ve výsledkové mřížce pro každý z faktorů. K získání konkrétních informací se porovnávají pozice tvrzení v jednotlivých faktorech. Zejména těch tvrzení, která jsou umístěna v některém z extrémů výsledku.

Analýza dat z průzkumu odhalila tři faktory. První z nich zahrnoval účastníky preferující menzu. Jako její hlavní výhodu vidí snadnou dostupnost a to, že se jedná o nejrychlejší způsob stravování. Studenti, které sdružují další dva faktory, navštěvují menzy i restaurace. Za největší výhodu menz považují jejich snadnou dostupnost. Nevýhodou je pro ně kvalita

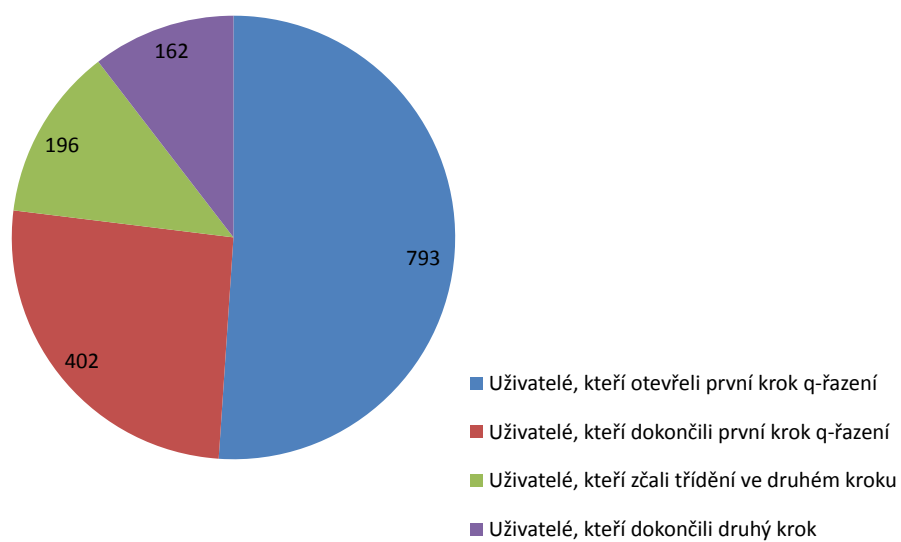


Obrázek 6.5: Graf hodnot z-skóre každého tvrzení v jednotlivých faktorech, získaný z online nástroje pro analýzu dat q-řazení [25]. (Textová tvrzení jsou nahrazena číselnými odkazy podle jejich seznamu uvedeného v předchozím textu.)

jídla za danou cenu, atmosféra během stravování a omezená nabídka jídel. Do prvního faktoru spadá 45 z vypracovaných q-setů. Druhému a třetímu faktoru odpovídá 42 a 29 výsledkových q-setů. Zbylé výsledky nelze zařadit.

Toto jsou pouze základní informace. Nyní záleží na cíli prováděného průzkumu, jak budou interpretovány. Uvažujme, že by cílem byla snaha o přilákání studentů do menzy. Pak by bylo z výsledných faktorů těchto skupin možné vyčíst, jaké důvody je od návštěvy menzy odrazují nebo proč je láká více stravování v restauracích. Z provedeného průzkumu vyplývá, že jako největší nedostatky menz vidí studenti kvalitu jídla a jeho cenu. Často se také objevovalo tvrzení, že menza nenabízí dostatečně pestrou nabídku jídel. Hlavní výhodou menz je pro všechny účastníky průzkumu snadná dostupnost.

Bohužel nemám prostředky k ověření získaných výsledků. Důležitým zjištěním však bylo především to, že ze sedmi set devadesáti tří lidí téměř polovina dobrovolně dokončila první krok q-řazení. Z té začalo sto devadesát šest lidí třídít karty ve druhém kroku q-řazení, přičemž sto šedesát dva z nich q-řazení dokončilo. Na těchto datech se do značné míry podepisuje dobrovolnost účasti v průzkumu. Téměř 83% uživatelů, kteří začali třídění ve druhém kroku q-řazení, jej však dokončilo. Z těchto údajů lze vyvodit, že aplikace může být pokládána za funkční a použitelný ekvivalent klasického provedení metody q-řazení.



Obrázek 6.6: Graf závislosti dokončených třídění na postupu uživatele aplikací

Kapitola 7

Závěr

V rámci této bakalářské práce jsem vytvořil aplikaci k provádění metody q-řazení a experimentálně ověřil její použitelnost. Vytvořená aplikace je online dostupná na <http://q-sorter.vojtam.cz/>

Abych dosáhl co nejlépe použitelného řešení, bylo nutné pochopit principy a význam metody q-řazení, nastudovat její průběh a prostředky k vypracování průzkumu na ní založeném. Před samotným návrhem aplikace jsem vyhledal existující řešení a provedl jejich analýzu, abych se vyvaroval chyb a získal co nejvíce poznatků o způsobu provádění q-řazení elektronickou formou. Pro implementaci jsem zvolil technologie, které poskytují řešení některých častých funkcionalit. Nastudoval jsem teorii návrhu uživatelských rozhraní a příjemné vizuální podoby webu a začal s implementací. Během ní jsem provedl několik uživatelských testování, která korigovala návrh a provedení aplikace k nejlépe vypadající a fungující formě. Po dokončení a otestování funkčnosti uživatelského rozhraní jsem vypracoval serverovou část aplikace. Navrhl jsem databázi, způsob organizace aplikace a vytváření vlastních průzkumů v administrátorské sekci. Také jsem implementoval export sesbíraných dat do CSV a převedl je do podoby vhodné pro analýzu výsledků q-řazení. Na závěr jsem navrhl experimenty k ověření použitelnosti vytvořeného nástroje.

Nejzajímavějším experimentem bylo provedení rozsáhlého q-řazení od sběru dat po interpretaci výsledků průzkumu. Tématem výzkumu bylo porovnání stravování v menzách VUT oproti restauracím. Průzkum dobrovolně dokončilo 162 lidí a pouze necelých 18% z uživatelů, kteří začali třídít tvrzení ve druhém kroku q-řazení, jej nedokončilo. Výsledky průzkumu ukázaly, že názorově se studenti dělí do tří skupin. Dvě z nich vidí na menze spíše záporně, jako vysoká cena jídel v poměru kvality a omezená nabídka pokrmů. Třetí skupina tyto závěry nevyvrací, preferují však menzu pro její snadnou dostupnost a rychlost, se kterou se zde můžou najíst. Třetí skupina je zároveň tou nejpočetnější.

V kontextu mé práce je nejdůležitějším zjištěním to, že 162 lidí dobrovolně dokončilo q-řazení v navrženém nástroji. Nikdo z nich navíc nepotřeboval znalost principu provádění metody q-řazení. Z toho můžeme vyvodit, že navržené uživatelské rozhraní je intuitivní a použitelné pro provedení i rozsáhlejšího průzkumu metodou q-řazení.

V případě další práce na projektu by bylo potřeba v první řadě přepracovat serverovou část aplikace. Ta je navržená příliš jednoduše na to, aby dostačoval při rozšíření aplikace o možnosti tvorby uživatelských účtů nebo správu založených průzkumů.

Při vypracovávání jsem si osvojil techniky implementace webových aplikací, postavených z velké části na programování v JavaScriptu. Také jsem se naučil jak navrhnout a testovat uživatelské rozhraní zaměřené na intuitivní pochopení práce s ním. V neposlední řadě jsem se také dozvěděl, jak navrhnout serverovou část aplikace a využít PHP k její realizaci.

Literatura

- [1] Alintech. [online]. [cit. 2016-05-05].
URL <http://www.alintech.com.au/>
- [2] Bootstrap. [online]. [cit. 2016-05-06].
URL <http://getbootstrap.com/>
- [3] HtmlQ. [online]. [cit. 2016-05-05].
URL <https://github.com/aproxima/htmlq>
- [4] JavaScript HTML DOM. [online]. [cit. 2016-05-11].
URL http://www.w3schools.com/js/js_htmlDOM.asp
- [5] PHP Data Objects. [online]. [cit. 2016-05-06].
URL <http://php.net/manual/en/book.pdo.php>
- [6] Trello. [online]. [cit. 2016-05-08].
URL <https://trello.com/>
- [7] BEVACQUA, N.: Dragula. [online]. [cit. 2016-05-06].
URL <https://github.com/bevacqua/dragula>
- [8] CHISNELL, D.: Usability Testing Demystified. 2009-10-06, [online]. [cit. 2016-05-12].
URL <http://alistapart.com/article/usability-testing-demystified>
- [9] COHENOVÁ, J.: *Neobyčejně užitečná kniha o webu*. SoftPress, 2004, ISBN 80-86497-63-1.
- [10] COOGAN, J.; HERRINGTON, N.: Q methodology: an overview. *Research in Teacher Education*, Vol 1(No.2), 24–28. Dostupné z www.uel.ac.uk/rite/issues/vol1/2/pp24-28/, 2011, ISSN 2047-3818.
- [11] COYIER, C.: Yay for HSLa. 2010-06-15, [online]. [cit. 2016-05-07].
URL <https://css-tricks.com/yay-for-hsla/>
- [12] van EXEL, J.; de GREEF, G.: Q methodology: A sneak preview. 2005, [online]. [cit. 2016-05-06].
URL <https://fcis.vdu.lt/~n.klebanskaja@evf.vdu.lt/FOV1-000B33EC/Q-methodology%2520-%2520A%2520sneak%2520preview.pdf>
- [13] GOTHELF, J.: *Lean UX*. O'Reilly Media, 2013, ISBN 978-1-449-31165-0.
- [14] HUNT, B.: Make Your Web Pages Easy to Read. 2006, [online]. [cit. 2016-05-07].
URL <https://webdesignfromscratch.com/basics/readability/>

- [15] JEFFARES, S.: POETQ. 2007, [online]. [cit. 2016-05-05].
URL <http://poetqblog.blogspot.cz/>
- [16] KENNEDY, E. D.: 7 Rules for Creating Gorgeous UI. 2014, [online]. [cit. 2016-05-07].
URL <https://medium.com/@erikdkennedy/7-rules-for-creating-gorgeous-ui-part-1-559d4e805cda>
- [17] KRUG, S.: *Web design, Nenutte uživatele přemýšlet*. Computer Press, 2006, ISBN 80-251-12918.
- [18] NIELSEN, J.: Why You Only Need to Test with 5 Users. 2000-03-19, [online]. [cit. 2016-05-12].
URL <https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>
- [19] Peter SCHMOLCK: WebQ. 2002, [online]. [cit. 2016-05-05].
URL <http://schmolck.userweb.mwn.de/qmethod/webq/>
- [20] POLECHOVÁ, P.: Q-třídění jako nástroj zpětné vazby ve škole. 2006, [online]. [cit. 2016-05-06].
URL <http://clanky.rvp.cz/clanek/o/z/920/Q-TRIDENI-JAKO-NASTROJ-ZPETNE-VAZBY-VE-SKOLE.html/>
- [21] SPOLSKY, J.: Designing for People Who Have Better Things To Do With Their Lives. 2000, [online]. [cit. 2016-05-10].
URL <http://www.joelonsoftware.com/uibook/chapters/fog0000000062.html>
- [22] TRIGG, R.: How are UX Principles Influencing Visual Design? 2016, [online]. [cit. 2016-05-08].
URL <https://uxmag.com/articles/how-are-ux-principles-influencing-visual-design>
- [23] Wikipedia: Comma-separated values. [online]. [cit. 2016-05-11].
URL https://en.wikipedia.org/wiki/Comma-separated_values
- [24] Wikipedia: CSS3. [online]. [cit. 2016-05-06].
URL <https://cs.wikipedia.org/wiki/CSS3>
- [25] ZABALA, A.: Demo: 'qmethod' package to analyse Q methodology data in R. [online]. [cit. 2016-05-01].
URL <https://azabala.shinyapps.io/qmethod-gui/>
- [26] ZABALA, A.: qmethod Cookbook. [online]. [cit. 2016-05-12].
URL <https://github.com/aiorazabala/qmethod/wiki/Cookbook>
- [27] ZABALA, A.: qmethod: A Package to Explore Human Perspectives Using Q Methodology. *The R Journal*, Vol. 6/2, 163-173. Dostupné z <http://journal.r-project.org/archive/2014-2/zabala.pdf>, 2014, ISSN 2073-4859.