Tallinn University
School of Digital Technologies

Taxonomies in DUI Design Patterns: A Systematic Approach for Removing Overlaps Among Design Patterns and Creating a Clear Hierarchy

Master Thesis

Mubashar Iqbal

Author: ".........."   ....

Supervisor: ".........."   ....

Supervisor: ".........."   ....

Head of the School: ".........."   ....

TALLINN UNIVERSITY

Non-exclusive license for reproducing the thesis and making it publicly available

I, *(name of author)* Mubashar Iqbal *(date of birth)* 06.05.1989

1. grant Tallinn University a permit (a non-exclusive license) to freely reproduce and make publicly available in the repository of Tallinn University Academic Library a piece of work created by me

   Taxonomies in DUI Design Patterns: A Systematic Approach for Removing Overlaps Among Design Patterns and Creating a Clear Hierarchy

   supervised by

   Ilja Šmorgun, David Jose Ribeiro Lamas

2. I am aware of the fact that the author also retains the rights mentioned in Clause 1.

3. I certify that granting the non-exclusive license does not infringe on the intellectual property rights of other persons or the rights arising from the Personal Data Protection Act.

Signed in Tallinn,

—————————————

(signature and date)

Declaration

I declare that apart from the work whose authors are clearly acknowledged, this thesis is the result of my own and original work.

This work has not and is not being submitted for any other comparable academic degree.

The thesis was supervised by Ilja Šmorgun (Tallinn University, Estonia) and David Lamas (Tallinn University, Estonia).

Mubashar Iqbal

. . . . (date)

. . . . (signature)

Dedication

This thesis is dedicated to my parents.

For their endless support, love and encouragement.

Abstract

Recently a library of design patterns for designing distributed user interfaces (DUIs) were created to help researchers and designers to create user interfaces and to provide an overview of solutions to common DUIs design problems without requiring a significant amount of time to be spent on reading domain-specific literature and exploring existing DUIs implementations. The current version of the DUI design patterns library need to be assessed because lot of design patterns are overlapping each other and their relationships are not clear enough to effectively find the most relevant design pattern for solving particular design problem, so the purpose of this thesis is to mature the DUI design patterns knowledge field by removing the duplicate design patterns, their description and to create a taxonomy where each design pattern should be organised in a way that will reduce redundancy, possibly leading to grouping or eventually merging similar patterns and allow to navigate to related patterns.

To achieve the defined goals, first target was to investigates the possible overlaps among design patterns and their relevancy with each other, in order to get these insights natural language processing tool was built for extracting and analysing each design pattern research paper to find potential codes. Later in this study thematic analysis was done with domain experts to get themes, their description and higher level categories from generated codes to organize all related design patterns in a clear hierarchy.

The outcomes of this thesis were includes the clarification of the relationships among design patterns by creating a taxonomy, clarified the description of individual design pattern, overlaps and duplicate design patterns were removed and merged similar design patterns.

Keywords: Distributed user interfaces, design patterns, taxonomy design, cross device interaction, thematic analysis, natural language processing

Acknowledgements

The success and completion of this thesis was required a lot of guidance, support and assistance from many people and all that I have done is only due to such supervision and assistance and I would not forget to thank them.

First of all I would like to express my special thanks of gratitude to my respected supervisors Ilja Šmorgun and David Lamas who gave me this opportunity to work on this project, steered me in the right direction and gave me all the required support and succour that made me to complete this thesis duly.

Secondly, I would like to thank the experts who participated in my research and for their invaluable contribution.

Finally I would like to thank my family, teachers and friends who supported, motivated and encouraged me throughout the study years. Also, I would like to extend my sincere esteems to all the staff members of the School of Digital Technologies and IDLAB.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The rapid evolution of technology, popularity and diversity of devices has completely changed the way of interaction with interactive systems like multi device environments those support distributed user interfaces to make interaction possible across devices (de la Guía, Lozano, & Penichet, 2014).

Such devices can classify as multi purpose devices and specific devices, whereas multi purpose devices refers to the mobile devices and stationary devices. Mobile devices like laptops, tablets, smartphones and smartwatches those are easily available and could be controlled according to the peripheral devices i.e. laptops employ keyboards, mouse and trackpads; and devices like smartphones, tablets, smartwatches employ touchscreens, accelerometers, gyroscopes, GPSs etc; and stationary devices such as Smart TVs, projectors categorized those connects with desktop computers, game consoles and control by novel interaction devices like Microsoft Kinect, the PlayStation Move jointly with the PlayStation Eye or the Wii-mote. Whereas specific purpose devices such as RFID-based panels, plane cockpits, advertisement panels etc are also a part of the available interaction environments to users (Tesoriero, 2014).

In the beginning, all these devices were used separately, researchers and designers were unaware of the existence and connection with other devices. In order to carrying out interactive tasks Distributed User Interfaces (DUIs) have been imagined those completely share the user interface between different interactive devices not just only the information between two different user interfaces (Melchior, 2011).

Recently Shmorgun, Lamas, and Mercer (2016), gathered the set of design patterns for aiding the design of distributed user interfaces those are listed at Semantic MediaWiki[1] (see Figure 1.1). Christopher Alexander originally introduced the term design pattern back in 70's, as he defined like this: "Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice". Even though this definition was defined for patterns in buildings and towns but what he says is true about object-oriented design patterns (Gamma, Helm, Ralph, & Vlissides, 1994). And the same way it is applicable on DUI design patterns. According to Borchers (2001), design patterns provide solution to recurring design problem for aiding to understand problem easily and generate new ideas.
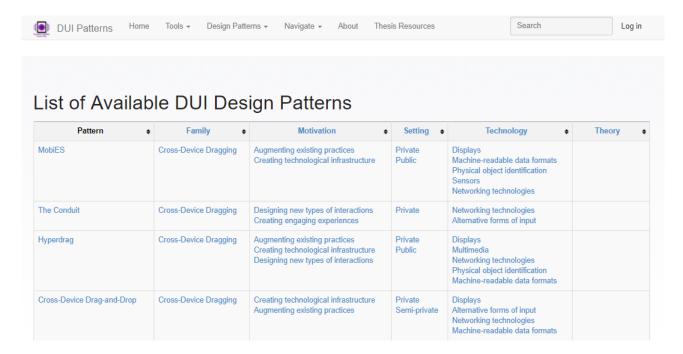
Figure 1.1: Library of DUI design patterns at semantic MediaWiki

A library of DUI design patterns has been developed to support the process of analysing existing and specifying new DUI design patterns. Currently total 47 design patterns are listed at Semantic MediaWiki, where each pattern is categorised by its name, family, motivation, setting, enabling technology, and supporting theory. The design pattern library as a whole aims to introduce common ways to HCI researchers and practitioners for designing cross-device interactions and to provide an overview of solutions to common DUIs design problems, without requiring a significant amount of time to be

---

[1] `http://idlab.tlu.ee/patterns/index.php/Main_Page`

spent on reading domain-specific literature and exploring existing DUIs implementations.

1.1 Research Problem

In current version of DUI design patterns library an issue has been identified that significant number of the patterns are overlapping and their descriptions are not clear enough to be sufficiently useful for design projects. The issue is there because of no clear classification or taxonomy was created to "mature the knowledge field in a way to allows for the description of terms and their relationships in the context of a knowledge area" (Usman, Britto, Börstler, & Mendes, 2017). According to Niu and Issa (2015), taxonomy is a knowledge organization system serving as the backbone of the domain knowledge for organizing concepts and applying the findings of a knowledge field (Vessey, Ramesh, & Glass, 2005).

To advance the knowledge and understanding of DUI design patterns there is a need to develop a clear taxonomy where each design pattern should be organised in a way that will reduce redundancy, possibly leading to grouping or eventually merging similar patterns and allow to navigate to related patterns. In another research related to DUI design patterns where Shmorgun et al. (2016) said, there is a need to provide better ways of navigating the patterns collection that would give ease of identification and selection of specific design pattern to researchers and designers for designing DUIs.

Fincher and Windsor (2000) also focused on taxonomy on theirs four principles of pattern language, where they mentioned; design pattern should have a taxonomy so a reader can find pattern(s) they need, as well as taxonomy descriptions should be clear enough and understandable for people who want to use design patterns for building DUIs in a way that they can distinguish and differentiate among patterns and use them effectively.

According to Usman et al. (2017), knowledge classification has supported the maturation of different knowledge fields mainly in four ways:

1. Classification of a knowledge field eases the sharing of knowledge (Natalia & Basil, 2009); and (Wohlin, 2014)

2. It provides a better way to understand the interrelationships between the objects of a knowledge field (Natalia & Basil, 2009)

3. Also helpful to identify knowledge field gaps (Natalia & Basil, 2009); and (Wohlin, 2014)

4. Support to make decision process(es) easier (Natalia & Basil, 2009)

The research problem is that design patterns from DUI pattern library have significant overlaps, their relationships and descriptions are not clear enough to identify and select most relevant design pattern for designing DUIs, so the current version of DUI design patterns library need to be assessed to classify design patterns their knowledge and relationships in a clear structure.

1.2   Research Questions

To achieve the defined research goals first step is to identify the overlaps among design patterns, those overlaps would be removed by performing research strategy and remaining design patterns would be organized by creating a clear hierarchy.

The research questions are as follows:

- [RQ1] Which patterns are overlapping in design patterns library?

- [RQ2] How to remove those overlaps among design patterns?

- [RQ3] How to organize remaining design patterns in a clear hierarchy?

1.3   Research Goals

Following research goals were formulated based on the above identified research problem:

- [G1] Remove the overlaps among design patterns

- [G2] Clarify the relationships among design patterns

- [G3] Clarify the description of individual design pattern

1.4 Foreseen Outcomes

Following are the expected outcomes based on the above research problem and goals:

- Identify the overlaps among different design patterns and provides systematic approach to remove those overlaps. By removing the overlaps it will reduce the redundancy and knowledge of design patterns would be organized based on the similarities those would give clear understanding and ease of identification to researchers.

- Clarify the relationships among design patterns those will allow researchers to navigate to related design patterns.

- Clarify the description of individual design pattern that would be useful to advance the understanding and knowledge of design pattern.

Chapter 2

Background

This chapter discusses important aspects related to distributed user interfaces, design patterns and taxonomy design that serve as a motivation for the research questions those are described in above chapter.

2.1 Distributed User Interfaces

The distribution of user interfaces is a reality (Tesoriero, 2014). A distributed user interfaces whose components distribute across several displays of different computing platforms that are used by different users, whether they are working at the same place or remote collaboration. Elmqvist (2011) stated the following five distribution dimensions input, output, platform, space, and time where distributed user interface components are distributed.

- **Input (I) -** also called input redirection where either managing input on a single device or distributed across several devices (Myers, Stiel, & Gargiulo, 1998); (Johanson, Hutchins, Winograd, & Stone, 2002); (Wallace, Mandryk, & Inkpen, 2008).

- **Output (O) -** also called display or content redirection where graphical output tied to a single display device, or distributed across several devices (Tan, Meyers, & Czerwinski, 2004); (Biehl et al., 2008); (Wallace et al., 2008).

- **Platform (P) -** the interface executes on a single computing platform, or distributed across different platforms (Elmqvist, 2011)

- **Space (S) -** the interface is restricted to the co-located or remote interactive spaces (Baecker, 1993).

- **Time (T) -** interface elements execute either simultaneously or distributed (Elmqvist, 2011)

Rädle et al. (2013) presented TwisterSearch an interactive prototype (see Figure 2.1) that was related to a collaborative search system, the main goal of this research was to achieve natural collaboration, collaborative web search and for supporting different working styles on a Microsoft Surface tabletop, Apple iPad tablets, and Anoto digital pen and paper. TwisterSearch was distributed on a shared and private displays where shared display was used to show collaborative search results and private displays were used to publish individual search results and findings on the shared display.

Figure 2.1: TwisterSearch: DUI supporting collaborative web search (Rädle et al., 2013)

In an another practical work, Tesoriero (2014) created the proxy work system as an example of distributable user interface where he presented the implementation of a set of distribution primitives those are applied to web application environments. Simple HTML based website was created with the navigation bar that was distributed at smartphone and application was distributed on desktop or projected wall, so users were able to navigate through website pages by using the smartphone.

Another interesting work was done by Noh et al. (2016) where a drawing application TakeOut (see Figure 2.2) was created by using distributed user interface. TakeOut drawing application is based on distributed cross device interaction between smartphone and the smartwatch, smartphone drawing application interface distributes on the connected smartwatch where smartphone interface becomes a workspace and smartwatch interface brings tool menus.

Figure 2.2: TakeOut: Drawing application using DUI (Noh et al., 2016)

2.2   Design Patterns

Design patterns provides reusable solution to a commonly occurring problem, speed up the design and development process, and an effective approach to solve particular problem. This term is mostly familiarly in software engineering, but it exists almost in every field like to address web user interface design, interactive exhibits, user interface related programming, hypermedia applications, or ubiquitous computing, web accessibility (Iacob & Damiani, 2011).

In the same context, Shmorgun et al. (2016) recently created and introduced a library of design patterns in the field of HCI to aid the researchers and designers to support the process of designing DUIs.

DUIs Design patterns are listed at semantic MediaWiki those were collected based on previous studies

and from design patterns research papers. Each design pattern is categorised by its name, family, motivation, setting, enabling technology, and supporting theory parameters. Table 2.1 is showing the detail of design patterns parameters:

Parameter	Description
Summary	Each design pattern has summary parameter that extracted from associated research paper that describes about how the design pattern works.
Description	A detailed explanation of the pattern that also extracted from associated research paper as it describes what type of interaction technique(s) are involved to make cross device / user interface sharing possible.
Design motivation	Describes the primary motivations guiding DUI design pattern.
Design goal	In design goal parameter field the main goals are listed those defined to achieve through specific design pattern.
Device type	Device type show the context of use either it is private, semi-private or public.
Enabling technology	List of technologies that are used for enabling the design pattern.
Reference	Reference to the original article where the interaction technique was first described.
Pattern family	Each design pattern is grouped with design pattern family.
Cites	All articles cited by the one where the pattern is described.
Cited by	Where one particular design pattern referred or used as an example to describe an interaction technique.
Related to	Show the information of design pattern association with another design patterns either in the context of similar interaction technique, gesture or approach.
Examples	A graphical representation of each design pattern to show how it work in real life.

Table 2.1: Parameters of design patterns

These parameters are also accounted and studied in this study except summary and description parameters because these parameters were generated from design patterns research papers and these both parameters were already extracted from research papers during the text processing activity.

2.3   Taxonomy Design

Information and knowledge have been classified for centuries (Malafsky, 2010). Nowadays, taxonomies are part of our daily life and this is particularly apparent today (Pellini & Jones, 2011). The concept of taxonomy was originally proposed by Carolus Linnaeus to group and classify organisms by using a fixed number of hierarchical levels. The same concept is adopted now in different knowledge fields, such as education, psychology and computer science etc and different classification structures have been used to construct taxonomies for these knowledge fields (Usman et al., 2017). Taxonomies are a useful and ubiquitous way of organizing information (Chilton, Little, Edge, Weld, & Landay, 2013).

Tobias (2014) illustrated in the diagram (see Figure 2.3) where he showed taxonomy as a core component of information architecture that guides the visual design of information navigation and interrelates with other components.
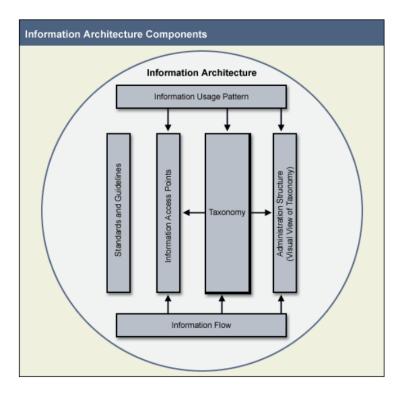
Figure 2.3: Information architecture components
Source of image: Tobias (2014)

The advent of the internet in the 1990s contributed an explosion of information dissemination and highlighted the need to develop new tools and skills to organise and retrieve such information. In this

context, taxonomies have become necessary (Pellini & Jones, 2011).

In science and engineering, a systematic description and organization of the investigated subjects helps to advance the knowledge field (Usman et al., 2017). However, little work has been done on taxonomy development in design patterns domain as it is a new but emerging field in the designing of distributed user interfaces. Based on a literature review this study proposes a systematic approach for taxonomy development in the domain of DUI design patterns.

Based on research and literature review in this study, no systematic mapping or systematic literature review has been conducted to identify and analyse the state-of-the-art of taxonomies in DUI design patterns knowledge field.

This section reviews the definition and purpose of taxonomy, it also includes various set of taxonomies and knowledge classification procedure.

2.3.1 Taxonomy Definition and Purpose

By the definition of Oxford English Dictionary[1], a taxonomy is "a scheme of classification". Taxonomy word originally came from Greek word taxis, which means 'order' and 'arrangement'. Mostly taxonomy term relates to biology field but in fact everyone use taxonomies in daily life. Every time we enter a modern supermarket, we navigate a carefully studied taxonomy of goods and products located along its aisles (Pellini & Jones, 2011).

Researchers define taxonomy as a set of structured names and descriptions (Lambe, 2007) or controlled vocabulary (Niu & Issa, 2015) that aid to organise data, information and flow of information in a consistent way.

In daily life knowledge workers spend lot of time for searching and analysing information where large repositories of digital data exists and individuals request to extract exact information what they want. Serrat (2010) said, taxonomy plays an important role to enrich the researchers experience and leverage their expertise, and it only possible when information is well organised and consistent so searching

---

[1]http://www.dictionary.com/browse/taxonomy

and browsing of information takes less time. Pincher (2010) posits that, all types of management systems are useless without designing a taxonomy for organizing or managing information.

2.3.2   Taxonomy Types

Lambe (2007) defined the below structures of taxonomies, specific purposes and when to use.

Lists (see Figure 2.4) are the more simple form of taxonomy and considered as a good for non-complex issues. Ideally, a list should contain between 12 and 15 elements, but when it becomes longer or more complicated, it is advisable to adopt a different taxonomy form, such as a tree structure (Pellini & Jones, 2011).

Figure 2.4: List taxonomy

Tree hierarchies (see Figure 2.5) are powerful to display cause-effect relationships in the taxonomy and usually use to distinguish broader and generic categories (Pellini & Jones, 2011). Tree structures are the most used taxonomies and particularly useful when concepts need to be divided into subcategories. They can be represented as pyramidal structure, as this approach mostly used for biological classification.
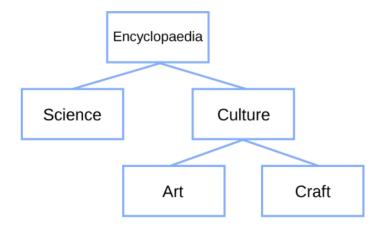
Figure 2.5: Tree / Hierarchical structure taxonomy
Source of image: Wikimedia

Matrices (see Figure 2.6) work best when required to organise information along two or three dimensions. Matrices could be helpful to categories and highlight gaps or missing categories. Mendeleev periodic table of elements is a well known example of two dimensional matrix.
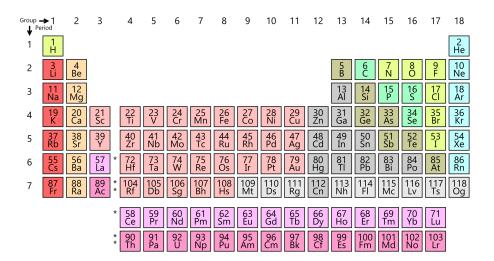
Figure 2.6: Periodic table: Matrix based taxonomy
Source of image: Wikimedia

Facets (see Figure 2.7) were introduced first in 1932 by S.R. Ranganathan for classifying books. The basic principle in faceted analysis is that there are more than one perspectives to view and classify a complex entity (Usman et al., 2017). Facets work well with the large content, frequent use of metadata and tags on digital documents (Lambe, 2007). Pellini and Jones (2011) said, facets taxonomy is an effective approach when tree structures become too large and complex.

According to Pellini and Jones (2011), facets mostly use by e-commerce organisations where large publication libraries exists, so the customers can access specific resources and information from different directions easily and effectively. The following screen shot from www.amazon.com where facets taxonomy structure allows the user to find a book by searching through books, audio books, authors, themes, editors, etc.
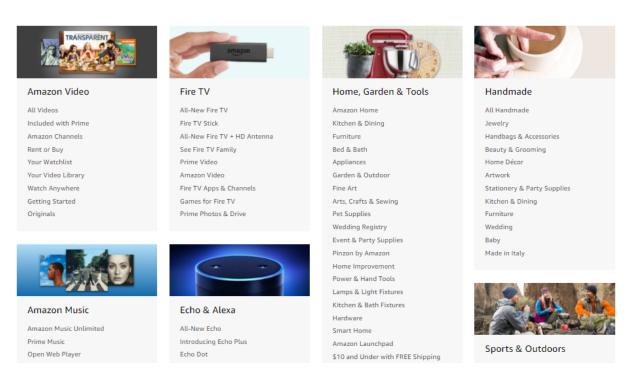
Figure 2.7: Facets taxonomy
Source of image: Amazon.com

System Maps (see Figure 2.8) are visual representations of proximity and connection among categories of a knowledge domain. They are useful when there is a coherent system of knowledge that can be communicated visually (Pellini & Jones, 2011). System maps are similar to mind maps, and provide a visual representation to show relationships among core ideas (Denham, 2006).

Figure 2.8: System map
Source of image: Wikimedia

2.3.3   Subject Matter Expert

The first step in the design of a new taxonomy is to clearly define the unit(s) of classification (Usman et al., 2017), in this study is a DUI design patterns.

Lack of subject matter expertise is a disadvantage in taxonomy development. This is understandable and makes sense to assume that the more domain expertise, the better the final information product will be (Earley, 2008). Interviewing and conducting analysis with the experts helps to understand where to further explore content in support of taxonomy development.

This is also an opportunity to prompt them for more specifics about audience needs, asking for example:

- Who are your main audiences?

- Would they understand this term in your opinion?

- What are typical user tasks?

2.3.4   Methodologies for Building Taxonomy

Content analysis or textual analysis is often used in finding a taxonomy from a large amount of textual materials (Niu & Issa, 2015), these methodologies are from social sciences for studying the content of communication, it gained popularity in the 1960s (Niu & Issa, 2015) also Krippendorff (2004) defined content analysis as "a research technique for making replicable and valid inferences from texts to the contexts of their use." By the content analysis approach taxonomy studies mostly focus on the existence of certain words, concept or concepts, later those words and concepts are analysed to find the meaning and relationship with the set of text(s) to speculate the order / hierarchy to classify the information.

Yao, Huang, and Cui (2009) built a tags extraction algorithms for extracting the tags from the tag space of Del.icio.us, where they collected a large set of tags from Del.icio.us to demonstrate the performance of taxonomy construction and evolution. Approximately the collected dataset contains more than 270 million tagging actions from almost 200,000 users in the dated range from 2007.01 to 2008.10. The constructed taxonomy by using this approach reflected the evolution of user interests the organization of online resources and web content.

In an another study, Chilton et al. (2013) presented cascade algorithm technique, a novel crowd algorithm that produces a global understanding of large datasets. Cascade algorithm technique produces human readable category labels and associated items with each category in its decision making algorithm.

Niu and Issa (2015) focused on the involvement of domain experts in intensive interviews, iterative development and competency questions use to formulate a taxonomy, these approaches develop taxonomy in a more theoretical sense, those makes the results more convincing and solid as compared to solely using content analysis. Niu and Issa (2015) used the ontology annotation tool (OAT) in their study which is widely used for natural language processing (NLP) tasks. After performing data processing, concepts were identified from the text and then each concept was put into the appropriate root concept class.

Chapter 3

Text Processing Tool

This chapter is briefly describing about the software tool that was built to analyse research papers. In order to perform text processing activity web based application was developed that includes different components where each component serving a specific function, those components could be processes, software, hardware, or any other part that helps to built different features of this system.

The technology stack is mostly open source to build this application where I used PHP (recursive abbreviation is Hypertext Preprocessor) an open source web side scripting language that easily embed in HTML to create web pages along with MySQL an open source relational database management system (RDBMS), relational database was created to save required and generated data and information in database for further analysis.MySQL database is easy to install, maintain and integrate with PHP based applications. XAMPP another open source platform that was used as a cross-platform web server solution, consisting apache hypertext transfer protocol server, database and interpreters for PHP programming language.

Hypertext Markup Language (HTML) was used with Cascading Style Sheets (CSS) and JavaScript to create visual representation for showing data in an organized tabular form to understand and interpret easily. In order to parse and extract text from the PDF based research papers a PHP based 3rd party library name is PDFParser was used.

3.1   Natural Language Processing

Natural language processing (NLP) mostly used for processing large natural language corpora, with the natural language processing it is an easy and quick way to do text processing and gain insights into what content is being published and how it resonates with the audience, similar like natural language understanding for analysing semantic features like categories, concepts, emotion, entities, keywords, meta data and relations from provided text input. TextRazor, a natural language processing APIs were used to perform text processing activity, it performs deep analysis on content to extract topics, topics relevancy score, entity extraction, concepts extraction and word relations etc.

3.2   Selection of Text Processing Service

A brief systematic review of existing and available language processing services was done before selecting the TextRazor natural language processing services. Spreadsheet was created where entries were listed based on the name of the available service, website URL, short description, capabilities, business model either is it free, trial or paid, supported programming languages, cloud-based or self-hosted or some additional features (see Table 3.1 as a reference).

Name	TextRazor
URL	https://www.textrazor.com/
Short Description	The TextRazor API uses natural language processing to extract and understand the concepts and semantics from documents, research, surveys, emails etc.
Capabilities	Entity Extraction, Disambiguation and Linking. Keyphrase Extraction. Automatic Topic Tagging and Classification.
Business Model	Charge based on daily requests >500
Supported Programming Languages	REST API, PHP, Java, Python
Environment	Cloud hosted, Self-hosted

Additional Features	Deep analysis,typed dependencies between words and Synonyms. Also offers to increase free limits and special pricing for qualifying academic users.
Limitations	Paid, no custom model but can create dictionaries and classifications.

Table 3.1: TextRazor natural language processing API

Following six different services were reviewed based on the above parameters where TextRazor services were selected based on its business model, supported programming languages, environment and importantly concepts extraction capability, whereas other services were either paid, self hosted or did not support concepts extraction.

Google natural language APIs, derive insights from unstructured text using Google machine learning. It was paid, no custom model and each service charge separately.

Microsoft cognitive services, easily evaluate sentiment and topics from the text to understand what users want. Trial version was available only for 30 days, no custom model and charge by storage and requested services separately.

IBM AlchemyLanguage, is a collection of APIs that offer text processing through natural language processing. It was paid and also required IBM watson studio that cost 150 USD per month..

Python NLTK, is a leading platform for building Python programs to work with human language data. To write own software by using this library that could take several months to develop and also it was self hosting.

NlpTools, same as NLTK, NlpTools is a library for natural language processing written in PHP and to write own software by using this library that could take several months to develop, it was self hosting and NlpTools is still under development and there could be plenty of features missing.

Text processing tool was created as a web tool and it is available on the Github[1] along with their

---

[1]`https://github.com/mubashariq/textanalysistool`

technical documentation at Github wiki[2]. This system is only accessible by using an internet browser and it also built to performs various actions like exclusion criteria, for helping to perform refinement activity, cross comparisons among design patterns to find commonalities and relevancy with each other.

3.3   Limitations

During building text processing tool some limitations were faced those are mostly related to software side.

Secure PDF files

Some PDF files were secured as mentioned above PDFParser does not support secured PDF files to parse and extract data, so manually data was extracted and created .txt files.

Invalid PDF structure and character set

Few research papers PDF contains invalid structure and character set so extra work was done in software tool to handle this situation.

---

[2]`https://github.com/mubashariq/textanalysistool/wiki/Text-Processing-Tool`

Chapter 4

Research Methodology

The overall research methodology consisted of three different phases (see Table 4.1). Phase one was mainly focused on the development of text processing tool and data gathering. In second phase of the study was thematic analysis by conducting interview sessions with domain experts. In the last phase of this study was analysis and defining the main category by using potential themes and later organized design patterns in hierarchical structure.

Stages	Activities	Additional Information
1) Data collection	Generating initial codes	Text processing to find out the initial codes from each research paper by using natural language processing APIs of TextRazor and collected design patterns parameters from semantic MediaWiki.
2) Data Analysis	Common codes (to answer RQ1)	Initial codes of each design pattern were compared with other design patterns to find the possible duplicates, related terms used for presenting particular design pattern, overlaps among design patterns.
	Potential themes (to answer RQ2)	Thematic analysis was done with the domain experts and also social network analysis was performed. Potential themes were identified those helped to formulate root concepts, remove unnecessary overlaps and merging design patterns based on similarities.

3) Categorising & Findings	Organizing in hierarchical structure (to answer RQ3)	Main categories were identified for each design pattern and design patterns were organized in a clear hierarchy.

Table 4.1: Research methodology phases and different activities

4.1   Data Collection

In order to complete the data collection phase, text processing activity was done and design parameters were collected from semantic MediaWiki.

4.1.1   Text Processing

TextRazor were found to be an efficient tool for completing the text processing task by using its natural language processing APIs. In this thesis the main focus was to develop the taxonomies so the first goal was to scout concepts from research papers, in order to achieve the this goal TextRazor API calls were prepared by using parsed text from each design pattern PDF file and sent requests one by one to TextRazor where it processed and response sent back. TextRazor extracted all the information in one request (like entities, topics, topics relevancy score, categories, synonyms, words and words relation and sentences) from where only two parameters "topics" and their "relevancy scores" were selected.

Topics

According to TextRazor, it provides an automatic understanding of hundreds of thousands of different topics at different levels of abstraction and useful for tagging to a finite set of categories, or customizing the classification process. The tagging system of TextRazor provides an easy way to add semantic metadata, boost discoverability and textual labels based on provided data text that dramatically reduces the customization effort.

Relevancy Score

The relevance of this topic to the processed document. This score ranges from 0 to 1, with 1 representing the highest relevance of the topic to the processed document. Relevancy score helped to order codes in descending order during the refinement and thematic analysis activities where most relevant codes were listed on top.

4.1.2   Initial Codes

This phase involves two different stages to gather initial codes, first stage of this phase involves generating initial codes by doing text processing on design patterns research papers text, total 13,834 initial codes were generated and saved in database along with their relevancy scores.

Second stage involves the extraction of design patterns parameters from semantic MediaWiki, parameters were downloaded in JSON (JavaScript Object Notation) format from semantic MediaWiki by using its API commands and later saved in database by using programming, total 1,022 parameters were collected for all 47 design patterns.

4.2   Sampling

DUI design patterns provide a helping source/tool-kit to researchers and designer who want to build interactive user interfaces and displays, so based on the nature of this field and my thesis research goals, the population sample is limited to the domain experts. More specifically domain experts includes from HCI field those participated in different design and development projects, software/quality assurance engineers and designers. Borchers (2001) said, to create successful interactive systems, user interface designers need to cooperate with developers and application domain experts in an interdisciplinary team. Experts profiles were created for clarifying an expert selection criteria to identify eligible participants.

The purposive sampling (known as selective, or subjective sampling) technique used that involves identifying and selecting individuals or groups of individuals that are especially knowledgeable about or experienced with a phenomenon of interest (Creswell, 2011).

In order to get to rigour, trustworthiness and rich results total 18 participants in the selected sample
based on the below created experts profiles (see Table 4.2). To ease the process of thematic analysis
and discussions it was an important to know the participant is living in Tallinn city but there was no
limitation or restriction to anyone who is willing to participate in this study.

Expertise	Status	Education & Experience	Location
HCI Experts	Currently doing research, studying or working in the field of HCI or interaction design.	Interaction design, human computer or machine interaction, interactive interfaces, distributed interfaces.	Based in Tallinn city or available remotely at Skype.
Software & Quality Assurance Engineers	Currently studying or working in the field of software or quality assurance engineering.	Development of distributed interfaces, interactive displays or building mobile or desktop applications.	Based in Tallinn city or available remotely at Skype.
Designers	Currently studying or working in the field of system designing and development.	Designing or development of distributed interfaces, interactive displays or building mobile or desktop applications.	Based in Tallinn city or available remotely at Skype.

Table 4.2: Experts profile

Total 18 domain experts (see Table 4.3) participated in thematic analysis sessions where 3 thematic
analysis sessions were conducted in group of 2, 2 and 4 experts. Multiple options were provided
to approach participants to participate in this study where 3 sessions were conducted at participants
home, 3 via Skype, 6 were conducted at participants office and remaining sessions were done in
Tallinn University IDLAB.

As per the requirements of thesis, participants age or gender was not important but the educational
and professional background was important as it described in above domain experts profiles.

13 out of 18 participants had already knowledge of either distributed user interfaces or design patterns;
whereas 10 participants were completely familiar with both terms, and remaining participants were
from the similar knowledge field so it was easy to describe and introducing this study to them.

Background	# of participants
HCI students and professionals	7
Software engineers	6
Quality assurance engineers	2
Designers	3

Table 4.3: Participants statistics

4.3 Study Protocol

Study protocols would help me to follow the plan of action, ensure that the research activities on the track and to protect from the damaging of research activities.

1. Prior to the arrival of the participant for a session, it is necessary to turn on the computer and set up the design pattern web page (before I created slides but changed after pilot study to show only design pattern MediaWiki Page) and web page would be open in Google chrome browser.

2. Greeting the participant by informal discussions.

3. Hand over the consent form for reading about the sessions and ethics those would be followed during the session.

4. For this step, I created a separate page where I explained about study, goals and its procedure, this page would be hand over to each participant on his/her arrival.

5. Now I would give him/her some time to read about design pattern to understand it and its techniques and will say; ask me if you have any question and let me know when you will finish

6. Participant will read and understand design pattern for 6 minutes (before it was 3 minutes but after pilot study it changed to 6 minutes), but would not tell the exact duration.

7. On this step I will provide web page that contains common codes for one single design pattern, user will read and understand codes. Then I will ask to generate themes by using listed codes

based on his understanding, knowledge and interests. (required time for this step was also changed, before it was 7 minutes and later it changed to 10 minutes)

8. Based on the amount of design patterns (35) I planned to perform total 18 thematic analysis sessions in first iteration of this study where each participant will perform above steps (6-8) two times, so each participant will get two design patterns in each session to find themes from selected codes.

I would inform participant if time goes up to 45 minutes and he can stop any time or finish the task in hand if participant wish to.

The completion of the session is followed by an informal discussion and thanking the user for participating and answering any questions if might they have. And the same procedure will be followed for the second and remaining sessions.

Special Case - In case if I would go to the participant home or working place then step 1 and 2 would not be applicable.

Based on the above activities in study protocol section approximately 40-45 minutes are required to complete one session with two design patterns.

4.3.1   Pilot Study

Research plan and all research instruments for this study were thoroughly checked and pre-tested in a pilot study, that was performed on two participants where one study was completed in 75 minutes and another was completed in 68 minutes.

Small modifications were done in the time frame for steps 6 and 7 (as mentioned in study protocol section) also participants discouraged to use slides but they preferred to use semantic MediaWiki web page for reading and getting knowledge of design patterns.

Two simple questions were asked only in pilot study to get response from participants based on the employed approach if any further improvements are required.

1. Is this procedure is time taking, made you tired or bored?

2. Is there any better approach to perform these steps?

One major change was happened based on the responses of above two questions from both participants. Two activities in thematic analysis (reviewing themes, defining and naming themes) were suggested to be completed at home and submit later in defined period of time, as both activities are required some amount of time to think and define. This change was included as an option to ask from the participant either he/she want to finish both sessions completely or wish to complete at home and submit later in defined period.

4.3.2 Main Study

After finishing the pilot study phase with the two domain experts, suggested changes were implemented and session time frame was updated. The study has been carried out in the same way with remaining experts as the pilot study has been conducted. In order to find the right participants, domain experts profile was created as described above and contacted via LinkedIn those were in my connection and also asked from friends to refer your friends those are matched with experts profile.

The time to carry out the study differed for each participant based on individual levels of understanding. Some of the participants had to read the information about the DUIs and patterns multiple times. Also, some of the participants used to ask questions during the study as they encountered misunderstandings. The average duration of the session with each participant was about 45 minutes as planned (excluding two activities, that time in not accounted in this study).

After completing the session I printed out the codes and created themes and handed over to the participant for performing remaining two activities also time to submit remaining work was defined based on participant wish but was asked to not exceed to more than three days. Also participants were asked to verbally provide any additional feedback about the pattern language and this study.

Chapter 5

Data Analysis

By NLP text processing and design patterns parameters I collected a large corpora of codes, dataset contains 14,856 codes including both initial codes and parameters those were shrunk to 3,406 by applying exclusion and refinement criteria, removed duplicated design patterns codes and accounted relevant codes. Total 18 participants were participated in this study to perform thematic analysis for 35 design patterns.

5.1   Common Codes

This phase involves generating common codes by comparing entire dataset and extracted only relevant codes. After applying below mentioned exclusion criteria on initial codes approximately 13,140 codes were left and 1,716 codes were trimmed off, these trimmed off codes were not removed from the database but status was updated to "deleted".

In this stage design patterns were compared with each other design patterns to identify possible duplicates, relevancy and overlaps percentage of each design pattern, total 1081 comparisons were performed for 47 design patterns.

Relevancy percentages were computed based on the similar initial codes, in this activity duplicate design patterns those had 100% similarity in an initial codes were removed because codes were gener-

ated from same design pattern research paper but different techniques were described like HeadLaser, HeadMouse, HandLaser and HandMouse are four different interaction techniques but described in the same research paper. In order to not repeat duplicate codes list for such techniques in thematic analysis only one design pattern technique was used like in this case only HeadLaser was selected.

5.1.1 Exclusion Criteria

In order to clean up the initial list of codes from each design pattern, exclusion criteria was defined to remove the abbreviations, unnecessary and interchangeable terms. Below are the points for exclusion criteria:

- All those codes were removed who had relevancy score less than 0.078 out of 1.0, by dynamically crawling topics of 47 design patterns research papers where I found least related topic score was 0.079, so this score was defined as a least relevancy limit score. Mostly codes after these were unrelated like abbreviations, words from references etc and has no reason to keep and later use in design patterns topics comparison

- All those codes were removed from the list by doing manually lookup who had no relevancy either with the topic or design pattern but ranked above 0.079 score (for example topics like Fee, OQO, truth etc).

- All those codes those were not written in English

- Mathematical notations, formulas and dates

5.1.2 Refinement

All general terms those were not suitable for the further development because of already quite abstract and generalize form (i.e. business, belief and culture etc as showed in Figure 5.1) or interchangeable terms like (interface or interfaces, network or networks etc.). All of such unqualified and generalized terms were trimmed off from the initial codes of each design pattern.

#	Topic	Score	Operation
1	Business	0.660	Delete
2	Science	0.348	Delete
3	Belief	0.149	Delete
4	Culture	0.111	Delete
5	Computer network	0.941	Delete
6	Computers	0.897	Delete
7	Electronics	0.885	Delete
8	Human–computer interaction	0.879	Delete
9	Android (operating system)	0.872	Delete
10	Web application	0.866	Delete
11	Google Nexus	0.823	Delete

Figure 5.1: Web page of refinement activity

Refinement activity was done by reviewing one by one each design pattern codes, in order to perform this activity simple web page was created where all identified codes were listed in tabular form (see Figure 5.1) along with "Delete" button to remove such terms, whereas trimmed off codes were not removed from the database but status was updated to "deleted".

After performing the refinement activity and comparison among design patterns total 3,406 common codes were identified from 35 unique design patterns including design patterns parameters. On this stage approximately each design pattern includes 70-130 common codes (see Table 5.1), those were finalized codes for thematic analysis.

Following Table 5.1 showing an overall statistics of initial codes, common codes and percentage of excluded codes those are related to each individual design pattern.

- **Initial codes –** showing the total number of codes those were generated by text processing and gathered design parameters from semantic MediaWiki.

- **Common codes –** common codes where collected by performing two different activities, first was exclusion and refinement activity to clean up the codes and second was cross comparisons among design patterns to find the related codes, it means each design pattern was compared with another design patterns to get common codes.

- **Excluded % –** excluded percentage column in Table 5.1 is giving an information about how much codes were trimmed off by applying exclusion criteria, refinement activity and cross

comparisons to consider only related codes.

Name	Initial Codes	Common Codes	Excluded %
MobiES	258	89	66 %
The Conduit	460	124	70 %
Hyperdrag	318	128	60 %
Cross-Device Drag-and-Drop	233	89	62 %
VisPorter	375	98	74 %
MultiSpace	217	99	54 %
Conductor	302	96	68 %
Cross-Device Pinch-to-Zoom	312	83	73 %
Stitching	327	128	61 %
EasyGroups	244	88	64 %
Bumping	354	110	69 %
PaperVideo	347	89	75 %
DisplayStacks	334	92	73 %
ConnecTable	327	98	70 %
Pinch	261	91	65 %
Codex	331	102	65 %
Interface Currents	172	58	61 %
Drag-and-Pop	259	105	60 %
Vacuum	275	79	71 %
HeadLaser	288	97	66 %
Perspective-Aware Interfaces	337	106	69 %
Pick-and-Drop	305	129	58 %

Lift-and-Drop	269	98	58 %
Slurp	327	105	68 %
Ubiquitous Graphics	309	89	71 %
SharedViews	430	98	77 %
Video Wall	295	87	71 %
Chucking	281	98	65 %
Voting	335	98	71 %
Shuffling	174	84	52 %
TranSticks	338	93	73 %
Touch-and-Connect	273	83	70 %
SyncTap	275	94	66 %
That One there!	315	92	71 %
Select-and-Point	235	99	71 %

Table 5.1: Individual design pattern statistics

5.2   Thematic Analysis

Thematic analysis mostly use in qualitative research, in this study it was perform to examine the codes and collated data to identify significant broader patterns also knows as potential themes. The main reason was behind thematic analysis to find out the users point of view, understandings about the design pattern and how they relate the design pattern information.

Following activities were performed during the thematic analysis session and in order to follow these defined activities, sample output structure was created (see Appendix A.1).

5.2.1   Searching for Themes

On this stage I had long list of codes for each design pattern those were identified during common codes analysis, this activity was performed on common codes with domain experts to find the broader

level of themes based on the experts point of view, understanding and knowledge. Codes were collapsed to broader level labels and themes, data were visually represented in tabular form to understand the relationship between codes and themes.

5.2.2   Reviewing and Naming Themes

This stage was bind with searching theme stage to refine the themes if possible, whereas some themes were collapsed into other themes and some were break down into smaller components. Later naming themes activity was done in the same stage to get the essence of what each theme is about, effectiveness in the context of relation with particular design pattern and what type of aspects each theme captures.

5.2.3   Defining Category

Third stage was also performed with the domain experts for defining a main category of each design pattern, it was derived based on the shared information like design pattern example at semantic MediaWiki, codes, created themes and their descriptions.

After performing the above activities for all total 35 unique design patterns with domain experts data was collected in the format of Appendix A.1, where created themes section representing the 'Searching Themes' activity, reviewed, defining and naming themes output was generated by 'Reviewing and Naming Themes' activity and category section is related to the last activity 'Defining Category'.

5.3   Data Distribution

Several steps were performed to arrange the gathered data, first gathered data was read and well understood and then themes were further analysed for cleansing and transforming similar themes, removed unnecessary and irrelevant themes from the list those were not compatible to design pattern knowledge but created by experts based on theirs understanding. In order to perform this process

data was extracted from the participants output files and distributed into following three different documents based on the different data.

Themes collection document was created for gathering all design patterns themes and categories from thematic analysis output files. Document structure includes design pattern name, themes and category (see Appendix B.1), by creating this document it became easy to compare design patterns themes and categories for cleansing and transforming similar themes (i.e. Auxiliary and helper, Mediator and medium etc.), removed unnecessary and irrelevant themes (i.e. Electronic engineering, Hardware etc.); and also all interchangeable themes (i.e. gestural control, gesture control and gesture recognition etc) named to one descriptive term like 'Gesture control', the selection of descriptive term was based on the weighted degree, it means that theme named it which term repeated more.

Arranging descriptions, another document was created for arranging themes and categories descriptions those helped to understand the context and relation of different themes and categories with design patterns.

Interaction variations, third document was created for listing different design patterns interaction variations (see Table 6.3), these interaction variations were gathered from design patterns research paper by reading and skimming, these techniques were used by researchers to accomplish one particular design problem in different ways, some techniques showing the different approaches in the design pattern, some shows the variations of different type of devices and displays for making interaction, and some follows the different gesture control.

5.4   Themes Social Network Analysis

Themes social network analysis (see Figure 5.2) was done by using Gephi, an open-source network analysis and visualization software. In order to check how the themes are grouped and to visualize the relationships of different themes towards design pattern. For performing social network analysis data was prepared beforehand by using Google sheet where design patterns theme were listed as a source and design patterns became target. The data was imported in Gephi software and graph was generated by using Yifan Hu force-directed graph algorithm and in-degree partition, Yifan Hu technique

considered as a quite easy to understand and effectively shows the weighted nodes of different in-coming relationships with edges. Labels were assigned and weighted nodes and their relational edges were coloured those later helped to visualize and understand the relationships of different themes with design pattern that led to generate higher level categories of design patterns.

Graph nodes were scaled according to the in-degree partition and different colours were added to vi-sualize how the themes are grouped together and connections with different design patterns. In graph, 'Ubiquitous interaction' theme and its incoming connections were highlighted in yellow, 'Gesture control' theme and its connections coloured in blue, 'Proxy' themes and its connections in red and 'Connector' theme and its connections represented in light-green colour. These highlighted parts of the graph are clearly giving the clues about the root themes.
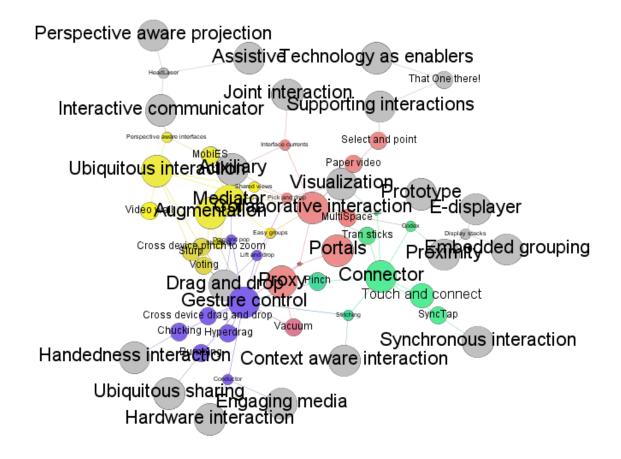
Figure 5.2: Themes social network analysis

Chapter 6

Findings

The previous chapter provided the details of the first two stages of this study where it described how the data was collected and processed. This chapter is covering the third stage for providing the detail information related to findings.

6.1   Findings from Common Codes Analysis

By common codes analysis total 3,406 codes were identified those were later used for thematic analysis. On the same stage design patterns overlaps and duplications were also identified, duplicate design patterns were removed based on the 100% similarities in initial codes as those were generated from the same research paper. Below Table 6.1 is showing the list of 12 different techniques those were excluded based on the 100% similarities and only one technique was included for thematic analysis. Total 35 unique design patterns were remained after removing such duplicate techniques.

Included Design Pattern	Excluded Design Patterns
Cross-Device Pinch-to-Zoom	Tilt-to-Preview Face-to-Mirror the Full Screen Portals

HeadLaser	HandMouse HandLaser HeadMouse
Drag-and-Pop	Drag-and-Pick
Voting	Throwing - MobiComics Send to me Retrieving
Shuffling	Taking Throwing

Table 6.1: List of duplicate design patterns

6.2 Findings from Thematic Analysis

By the thematic analysis of 35 design patterns with 18 domain experts, total 89 themes were created along with their description that contains 2-4 sentences and 35 categories one for each design pattern. On this phase various activities were performed to successfully get the desired results from thematic analysis.

Searching for themes, on this stage common codes of each design pattern were analysed and formed either in broader level themes or sub-themes and some codes were discarded those do not belong anywhere. At the end of this stage domain experts created an organized set of themes, approximately 5-10 themes were created for each design pattern.

Reviewing and naming themes, after this activity created themes were concise and immediately give the sense of what the theme is about, also on this step domain experts described each theme in couple of sentences that was helpful to understand the user perspective towards each design pattern that later was used to clarify design pattern description and it also aided to identify the main category of design pattern. After completing this step each design pattern approximately had 2-5 themes along with their description.

Defining category, total 35 categories were created, one for each design pattern in a last activity of this phase that was performed to overall know about the root concepts by relating with design pattern,

its codes, created themes and their description.

6.3   Findings from Data Distribution

From the analysis of data distribution 31 unique themes were identified as showing in following Table 6.2, by going further with these 31 themes analysis 12 design patterns were emerged, selection and naming of design patterns were based on the themes, related research paper, interaction technique and pattern family. As 'drag and drop' pattern was created from 'hyperdrag' and 'cross device drag and drop' themes that was compared with the created themes, the approach followed in these research papers and pattern family for validating the created design pattern name transformation and association with design pattern. In some cases themes were not matching with the design pattern but named after pattern family and implemented approach in research paper like in the case of 'MobiES' and 'The conduit' where the created themes were totally different as compared to pattern family and approach implemented in associated research papers.

Acquaintance	Engaging media	Prototype
Assistive	Gesture control	Proximity
Augmentation	Handedness interaction	Proxy
Auxiliary	Hardware interaction	Supporting interactions
Collaborative interaction	Interactive communicator	Synchronous interaction
Connector	Joint interaction	Technology as enablers
Context aware interaction	Mediator	Ubiquitous interaction
Drag and drop	Personalization	Ubiquitous sharing
E-displayer	Perspective aware projection	Visualization
Electronic engineering	Pick and drop	
Embedded grouping	Portals	

Table 6.2: Unique list of design patterns themes

Generated themes became keywords those would add an extra information in design patterns library to easily navigate among different through them based on their similarities and also it give the essence to understand what design pattern is about.

6.3.1 Design Patterns

Analysis of data distribution brought the following list of design patterns, once the design patterns themes, categories and their relationships have been mapped, it became easy to identify the connected design patterns, 12 unique design patterns were emerged those were grouped based on their similarities in the sense of pattern family and implemented approach in research paper and connected themes.

1. Drag and drop

2. Drag and pick

3. Lift and drop

4. Bumping

5. Throwing

6. Shuffling

7. Pinch

8. Portals

9. Connector

10. Perspective aware

11. Assistive

12. Proximity

6.3.2   Design Patterns Interaction Variations

The following Table 6.3 is showing a list of interaction variations those occurred in one or more design patterns research papers to solve design problem in a different way or sometime under different names, the list was gathered by reading and skimming the associated research paper of design pattern. These variations were collected to use in taxonomic structures to show the possible variations of particular design pattern or interaction approach.

Design patterns & Associated Research Paper	Interaction variations

Paper Video	- Spatial Location input
	- Gesture displays
	    - Shake
	    - Touch
	    - Quick upward move
	- Display proximity input
	    - Pile
	    - Side as pointer
	    - Corner as pointer

Interface Currents	- Pools - Streams
Display Stacks	- Pile - Stack and Bend - Fan - Liner loop - Collocation

Codex	- Concave
	    - Closed book
	    - Book in hand
	    - Standing book
	    - Lectern
	    - Laptop
	    - Back to back
	- Neutral
	    - Flat
	- Convex
	    - Corner to corner
	    - Face to face
	    - Battleship

Lift and drop	- Singleshot - Pick and drop

Table 6.3: Design patterns interaction variations

6.4 Findings from Social Network Analysis

Social network analysis was done to check how the themes are grouped and their relationships towards design patterns. Four weighted nodes were identified by running Yifan Hu force-directed graph algorithm along with in-degree partition.

A grouping structure by using Gephi clearly showing root themes and patterns those are gravitated around them. This network analysis approach led to the identification of four new higher level categories by visualizing the central themes and connected design patterns, these central themes became the top part of the taxonomic structure (see Figure 6.1).

Gesture Control – Patterns listed under this theme use gestures technique to make cross device connection and moving objects from one interface to another interface, gesture techniques like bumping, pinch, drag and drop etc.

Connectable – all those patterns who use some control like Stylus, TranSticks to connect two different devices.

Proxy – all those patterns are listed here who provide a placeholder on another connected device for sharing information.

Ubiquitous Interaction – design patterns those enable cross device communication without any physical barrier (also called transparent interaction technique) and by adaptive interfaces.

6.5   Taxonomic Structure

Based on the above research findings four hierarchical taxonomy structures were generated where design patterns inherits with a higher level pattern category. Taxonomy structures are divided into four different levels where it structured by category, families, design patterns and interaction variations. Following are the taxonomic structures levels, where:

- **Level 1**, is shaded in blue colour where main category is listed.

- **Level 2**, grey colour boxes are design patterns families and representing level 2 those were generated in one previous study of Mercer (2015), total 9 families are listed at semantic MediaWiki and patterns were initially grouped in these families.

- **Level 3**, is highlighted in pink colour and it represents design patterns those are newly created in this study.

- **Level 4**, red colour boxes representing different design patterns interaction variations those were implemented in different research papers and collected in this study.

Listed below taxonomic structure (see Figure 6.1) contains 3 different families and 4 design patterns those are categorised based on the category and families. Like in the example of 'Cross device portals' family where 'Portals' design pattern is listed based on the their shared similarity and interaction technique. Each design pattern has different implementations or interaction variations those are listed in below design pattern in red colour boxes.
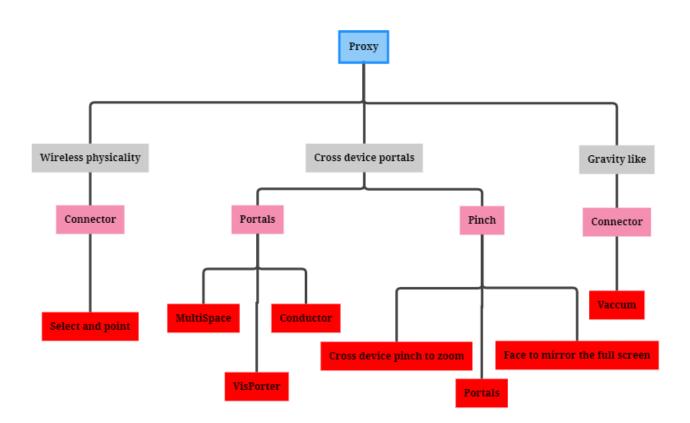
Figure 6.1: Proxy taxonomy structure

Chapter 7

Discussion

The aim of this study was to remove the overlaps among design patterns and clarifying their relationships that led to structure design patterns library in a clear hierarchy. The taxonomic structure was necessary as it is a backbone of any knowledge filed (Niu & Issa, 2015), as well as it helped to understand the relationships among design patterns, so the researchers can easily navigate through design patterns library.

Systematic approach was followed to advance the knowledge and understanding of DUI design patterns by creating a clear taxonomy where each design pattern is organised in a way that reduces the redundancy, led to grouping and merged similar patterns. In one study Shmorgun et al. (2016) said, there is a need to provide better ways of navigating the patterns collection that would give ease of identification and selection of specific design pattern to researchers and designers for designing DUIs, so now this study clarified the relationships of design patterns those allowed to navigate to related patterns easily.

Following research questions were set to answer the identified research problem and achieving the desired goals.

- [RQ1] Which patterns are overlapping in design patterns library?

- [RQ2] How to remove those overlaps among design patterns?

- [RQ3] How to organize remaining design patterns in a clear hierarchy?

In order to answer these research questions, various research activities were performed (see Table 4.1), where first activity was performed for data collection, second activity was done by doing thematic analysis & finding potential themes and last activity was related to categorization and findings.

To answer the [RQ1], initial codes were generated by using text processing tool (see Chapter 3), the tool was built by myself in this study to extract core concepts from design patterns research papers, initial codes were compared to identify possible duplicates and overlaps of design patterns with each other, total 12 duplicate patterns were identified based on the 100% similarities in initial codes.

The aim of the second research question was to remove the overlaps so to answer the [RQ2], thematic analysis sessions was conducted with domain experts (see Table 4.2), total 18 experts were participated in this study to perform thematic analysis for 35 design patterns. 31 unique potential themes were identified for each design pattern based on the experts output those helped to formulate root concepts, remove unnecessary overlaps and merging design patterns based on their similarities.

To answer the last research question [RQ3], main categories were identified for each design pattern and design patterns were organized in a taxonomic structure. Only four new higher level categories were created and 12 design patterns were identified, 9 design patterns families were collected from one previous study of Mercer (2015)), on the same stage different interaction variations (see Table 6.3) were collected as well from design patterns research papers.

Research goals and expected outcomes were achieved effectively, duplicate were removed, systematic approach was followed to remove overlaps from design patterns, reduced the redundancy and knowledge of design patterns is now organized in a taxonomic structure. Also description and relationships of design patterns now clarified, no duplication exist and also added new knowledge in the design patterns library by adding themes as a keywords those give an essence to understand what design pattern is about.

Current study has few limitations as well those could be improved in future. One possible limitation was that only two participants were from designing distributed user interfaces field, also there was no working prototype or implemented example of design patterns to educate participants. Due to these

reasons it was difficult to explain and educate participants particularly in the DUIs design patterns field.

Second limitation is that newly created taxonomies are not validated and without this I could not claim these provide better understanding, ease of identification and selection of design patterns. Another limitation was that taxonomies were developed without taxonomy development expert(s), so by overcoming this shortcoming would reveal more stronger and interesting results.

Chapter 8

Conclusion

The aim of this research was to classify DUI design patterns library and clarify the relationships of design patterns by removing the duplicates and overlaps among them. The proposed systematic taxonomy development approach was doable to guide the taxonomy development process, and constructed taxonomy reflected the evolution of experts understanding and knowledge, design patterns content, and facilitate the organization of design patterns in taxonomic structure.

This paper presents a text processing tool that was built by using an open source technology stack. Text processing tool used the TextRazor natural language processing services that efficiently generated the core concepts / initial codes from the design patterns research papers. Common codes were identified by comparing design patterns with each other, on this stage duplicate design patterns techniques were identified and excluded them before moving to further analysis. Unique design patterns common codes were further analysed by thematic analysis where 31 themes were identified those were later added in design patterns library as a additional knowledge to easily navigate among different design patterns based on their similarities and give the essence to understand what design pattern is about.

It was also important to see how the themes are grouped and connected with each other and design patterns, social network analysis was done by using Gephi software and graph was generated by using Yifan Hu force-directed algorithm and in-degree partition. By performing this data analysis activity four new higher level categories were created. Data distribution analysis was also done to

understand the meaning of the collected data and formulate the information from thematic analysis data, unique design patterns list were identified and later design patterns were organized in clear taxonomic structures.

The results produced this approach were more competitive in quality and time effective, as it uses NLP text processing services to extract core concepts from the provided text so work efforts reduces significantly.

Furthermore, although as noted previously, the methodology has some limitations and areas to be explored in future, it can be concluded that this study successfully presented the classified taxonomic structures of design patterns library by following the systematic taxonomy development approach that could also be helpful in future to classify new design patterns or related knowledge in a particular taxonomy.

Chapter 9

Future Work

Outcomes of this study has opened several new research areas to advance the DUI design patterns knowledge field. One possibility to conduct a research with end users to validate newly created taxonomies, as Usman et al. (2017) said; validation strengthens reliability and usefulness of taxonomies; and it would also be informative to investigate about how users navigate to the design patterns library after implementing new taxonomic structure without encountering excessive cognitive load to find the desired design pattern for specific design problem solution.

Another possibility for the future work could be describing and detailing the design patterns keywords those added newly in design patterns library to give more thorough overview of the patterns, identify and map alternate examples and their connections so the researchers would easily understand the patterns better and use them more effectively. As Pellini and Jones (2011) said, keywords represents as folksonomies, folksonomies are not an alternative to taxonomies but help to enhance agility, awareness, shared understanding and linked meanings.

Chapter 10

Kokkuvõte

Hajutatud kasutajaliideste disainimustrite taksonoomia: süstemaatiline lähenemine kattuvuste eemaldamiseks ja selge hierarhia loomiseks.

Hiljuti loodud hajutatud kasutajaliideste disainimustrite kogu aitab teadlastel ja disaineritel luua kasutajaliideseid ning annab ülevaate lahendustest tavapärastele hajutatud kasutajaliideste disainiprobleemidele, ilma et oleks vaja kulutada palju aega selleks, et lugeda valdkonna kirjandust ja uurida olemasolevaid hajutatud kasutajaliideste teostusi. Olemasolevas hajutatud kasutajaliideste disainimustrite kogu versioonis on kerkinud probleem, et märkimisväärne hulk disainimustreid kattub ja nende kirjeldused ei ole piisavalt selged, et olla küllaltki nii kasulikud disainiprojekti kõige olulisemate disainimustrite tuvastamiseks ja valimiseks. Probleem seisneb selles, et puudub selge klassifikatsioon, mis võimaldaks disainimustrite konkreetsesse järjekorda seadmist.

Käesoleva magistritöö eesmärgiks on edendada hajutatud kasutajaliideste disainimustrite teadmiste valdkond, kõrvaldades korduvaid ja kattuvaid disainimustreid. Selgitades nende kirjeldust ja luues taksonoomia, kus iga disainimuster on liiasusi vähendavalt korraldatud, tuua kaasa samalaadsete mustrite rühmitamise ja ühendamise, et liikuda seotud mudeliteni.

Lõputöös esitatud eesmärkideni jõudmiseks jaotus töö etappideks, millest esimene oli uurida võimalike kattumisi disainimustrites ja seotust nende vahel. Uuringu raames oli võetud aluseks tekstianalüüsi vahend, võtmaks välja ja analüüsimaks iga disainimustri kirjeldust ja leidmaks võimalike märksõnu,

mis kõige paremini seda disainimustrit kirjeldavad. Keelepõhise tekstianalüüsi vahendi loomiseks kasutati TextRazor nimelist teenust ja vahend ehitati, kasutades avatud lähtekoodiga tarkvara tehnoloogiat.

Teine etapp eeldas tekkinud märksõnade puhastamise ja selleks, et asjatud ja tähtsusetud märksõnad eemaldada, määratleti välistamiskriteeriume. Samuti oli teostatud viimistlus abstraktsete ja üldiste märksõnade kõrvaldamiseks. Viidi läbi temaatiline analüüs domeeniekspertidega, et saada kõrgema tasemelisi ja informatiivseid teemasid. Nende kirjeldustest ja kategooriatest loodud märksõnu ning teemasid omakorda paigutati ning analüüsiti, kujundamaks lõpuks kõrgetasemeline kategooria, mis on vajalik kõikide seonduvate disainimustrite korraldamiseks selge hierarhiana.

Käesolev lõputöö täitis kõik eelnevalt püstitatud eesmärgid ja hõlmab iga disainimustri hierarhilist struktuuri. Selgitades nende omavahelisi suhteid, sisaldab see iga konkreetse disainimustri jaoks seonduvaid märksõnu, mida loodi inimese ja arvuti interaktsiooni ekspertide arusaamal. Kattuvad ja korduvad disainimustrid eemaldati, sarnased disainimustrid ühendati ning liigitati ühe termini alla.

Appendix A

Thematic Analysis Output Structure

Created themes

- Interactivity
- Digital media
- Personal assistant
- Data manager
- Drag and Drop
- Cross communication/communicator
- Information sharing
- Artificial interfaces

Reviewed themes

1. Personal assistant
   a. Digital media
   b. Data manager
2. Communicator
   a. Interactivity
   b. Cross communication
   c. Information sharing
   d. Drag and drop
   e. Artificial interfaces

Defining and naming themes

Auxiliary/Helper
Providing additional help and support to enable communication between different devices and sharing information from one interface to another interface.

Mediator
Simplify communication between two different devices for sharing information.

Category

Auxiliary pattern
Both auxiliary and mediator themes are clearly indicating additional help and support to accomplish cross device information sharing task.

Figure A.1: Sample output structure of thematic analysis session

Appendix B

Themes collection

Design patterns	Themes	Categories
MobiES	- Auxiliary/Helper - Mediator	Auxiliary pattern
The Conduit	- Acquaintance - Personalization	Acquaintance/Ally pattern
Hyperdrag	- Drag and Drop - Controller	Drag and Drop
Cross-Device Drag-and-Drop	- Drag and Drop - Gesture Control - AI Messenger	Gesture Control
VisPorter	- Collaborative interaction - Gesture Control - Proxy	Collaborative interaction
MultiSpace	- Portals - Collaborative interaction - Interactiveness	Portals Interactiveness
Conductor	- Engaging media - Gesture Control	Interactor
Stitching	- Connector - Context aware interaction - Gesture Control	Connector

Table B.1: Document structure of themes collection

References

Baecker, R. (1993). *Readings in Groupware and Computer-Supported Cooperative Work.* Morgan Kaufmann Publishers, San Francisco, CA.

Biehl, J. T., Baker, W. T., Bailey, B. P., Tan, D. S., Inkpen, K. M., & Czerwinski, M. (2008). Impromptu - A New Interaction Framework for Supporting Collaboration in Multiple Display Environments and Its Field Evaluation for Co-located Software Development. *Proceedings of the International Conference on Human Factors in Computing Systems (CHI'08)*, 939–948. Retrieved from `http://doi.acm.org/10.1145/1357054.1357200` doi: 10.1145/1357054.1357200

Borchers, J. O. (2001). A pattern approach to interaction design. *AI & Society*, *15*(4), 359–376. doi: 10.1007/BF01206115

Chilton, L. B., Little, G., Edge, D., Weld, D. S., & Landay, J. A. (2013). Cascade: Crowdsourcing taxonomy creation. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1999–2008. Retrieved from `http://dl.acm.org/citation.cfm?id=2466265{%}5Cnhttp://dl.acm.org/citation.cfm?doid=2470654.2466265` doi: 10.1145/2470654.2466265

Creswell, J. W. L. P. C. (2011). *Designing and Conducting Mixed Methods Research Electronic Version.*

de la Guía, E., Lozano, M. D., & Penichet, V. M. R. (2014). Interacting with Tangible Objects in Distributed Settings. *Proceedings of the 2014 Workshop on Distributed User Interfaces and Multimodal Interaction*, 15–18. Retrieved from `http://doi.acm.org/10.1145/2677356.2677659` doi: 10.1145/2677356.2677659

Denham. (2006). *Knowledge sharing - a re-think.* Retrieved from `http://denham.typepad`

`.com/km/2006/12/knowledge{_}shari.html`

Earley.   (2008).   *Subject Matter Experts and Taxonomy Development.*   Retrieved from
`http://www.earley.com/blog/subject-matter-experts-and-taxonomy`
`-development{%}0A`

Elmqvist, N.   (2011).   Distributed User Interfaces. , 1–12.   Retrieved from `http://link`
`.springer.com/10.1007/978-1-4471-2271-5` doi: 10.1007/978-1-4471-2271-5

Fincher, S., & Windsor, P.   (2000).   Why patterns are not enough: some suggestions concern-
ing an organising principle for patterns of UI design.   *CHI'2000 Workshop on Pattern
Languages for Interaction Design: Building Momentum*, 1–6.   Retrieved from `http://`
`scholar.google.com/scholar?hl=en{&}btnG=Search{&}q=intitle:`
`Why+patterns+are+not+enough:+some+suggestions+concerning+an+`
`organising+principle+for+patterns+of+UI+design{#}0`

Gamma, E., Helm, R., Ralph, J., & Vlissides, J.   (1994).   Creational Patterns.   *Design Patterns -
Elements of Reusable Object-Oriented Software*, 94–155.

Iacob, C., & Damiani, E.   (2011).   On the use of design patterns in collaborative design processes.
Procedings of the Second Conference on Creativity and Innovation in Design - DESIRE '11,
245.   Retrieved from `http://dl.acm.org/citation.cfm?{%}5Cnid=2079216`
`.2079252{&}coll=DL{&}dl=ACM{&}CFID=109209162{&}CFTOKEN=14131771`
doi: 10.1145/2079216.2079252

Johanson, B., Hutchins, G., Winograd, T., & Stone, M. (2002). PointRight: Experience with Flexible
Input Redirection in Interactive Workspaces. *Proceedings of the 15th International Symposium
on User Interface Software and Technology (UIST'02)*, 227–234.   Retrieved from `http://`
`doi.acm.org/10.1145/571985.572019` doi: 10.1145/571985.572019

Krippendorff, K.   (2004).   *Content Analysis: An Introduction to its Methodology.*   Retrieved from
`http://www.jstor.org/stable/2288384?origin=crossref`   doi: 10.2307/
2288384

Lambe, P.   (2007).   *Patrick Lambe, Organising Knowledge: Taxonomies, Knowledge and Organisa-
tional Effectiveness.*

Malafsky, G. P.   (2010).   Organizing Knowledge with Ontologies and Taxonomies Mr . Brian D

Newman 1 Introduction 2 What are Ontologies and Taxonomies.

Melchior, J. (2011). Distributed user interfaces in space and time. *Proceedings of the 3rd ACM SIGCHI symposium on Engineering interactive computing systems - EICS '11*, 311. Retrieved from `http://dl.acm.org/citation.cfm?id=1996461 .1996544{%}5Cnhttp://portal.acm.org/citation.cfm?doid=1996461 .1996544` doi: 10.1145/1996461.1996544

Mercer, E. (2015). Towards a Pattern Language for Describing Distributed Interactions Hajutatud interaktsioone kirjeldav mustrikeel. , 48. Retrieved from `http://idlab.tlu.ee/ patterns/`

Myers, B. A., Stiel, H., & Gargiulo, R. (1998). Collaboration using multiple PDAs connected to a PC. *Proceedings of the 1998 ACM conference on Computer supported cooperative work - CSCW '98*, 285–294. Retrieved from `http://portal.acm.org/citation.cfm ?doid=289444.289503` doi: 10.1145/289444.289503

Natalia, J. S. V., & Basil, V. R. (2009). Maturing Software Engineering Knowledge through Classifications: A Case Study on Unit Testing Techniques. , *35*(4), 551–565. Retrieved from `http://ieeexplore.ieee.org/document/4775907/`

Niu, J., & Issa, R. R. (2015). Developing taxonomy for the domain ontology of construction contractual semantics: A case study on the AIA A201 document. *Advanced Engineering Informatics*, *29*(3), 472–482. Retrieved from `http://dx.doi.org/10.1016/j.aei.2015 .03.009` doi: 10.1016/j.aei.2015.03.009

Noh, W., Lee, M., Cheon, H., Kim, J., Lee, K., & Cho, J. (2016). TakeOut: Drawing Application Using Distributed User Interface for Being Close to Real Experience. *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, 173–176. doi: 10.1145/2968219.2971439

Pellini, A., & Jones, H. (2011). Knowledge taxonomies - A literature review. (May), 24. Retrieved from `http://www.odi.org.uk/ resources/details.asp?id=5753{&}title=knowledge-taxonomies -literature-review{&}utm{_}source=ODI{_}Update{&}utm{_}medium= feed{&}utm{_}campaign=Google+Reader`

Pincher, M. (2010). *A guide to developing taxonomies for effective data management.* Retrieved from `http://www.computerweekly.com/feature/A-guide-to-developing -taxonomies-for-effective-data-management`

Rädle, R., Jetter, H.-C., & Reiterer, H. (2013). TwisterSearch: A Distributed User Interfaces for Collaborative Web Search. *To appear in Distributed User Interfaces: Collaboration and Usability*, 1–17. Retrieved from `http://link.springer.com/10.1007/ 978-1-4471-5499-0{_}5{%}5Cnhttp://link.springer.com/chapter/ 10.1007/978-1-4471-5499-0{_}5` doi: 10.1007/978-1-4471-5499-0_5

Serrat, O. (2010). Taxonomies for Development. *Knowledge Solutions*(September), 1–7. Retrieved from `http://www.asiandevbank.org/documents/information/knowledge -solutions/taxonomies-for-development.pdf`

Shmorgun, I., Lamas, D., & Mercer, E. (2016). Towards a Pattern Language for Distributed User Interfaces. *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '16*, 2712–2718. Retrieved from `http://dl.acm.org/ citation.cfm?doid=2851581.2892304` doi: 10.1145/2851581.2892304

Tan, D. S., Meyers, B., & Czerwinski, M. (2004). WinCuts : Manipulating Arbitrary Window Regions for More Effective Use of Screen Space. *Chi '04 Ea*(January), 1525–1528. doi: 10.1145/985921.986106

Tesoriero, R. (2014). Distributing User Interfaces. (02071), 1–10. doi: 10.1145/2677356.2677669

Tobias, T. (2014). *Learning about Taxonomy in Web Design.* Retrieved from `http://www .tobiasandtobias.com/learning-about-taxonomy-in-web-design/`

Usman, M., Britto, R., Börstler, J., & Mendes, E. (2017). Taxonomies in software engineering: A Systematic mapping study and a revised taxonomy development method. *Information and Software Technology*, *85*, 43–59. Retrieved from `http://dx.doi.org/10.1016/ j.infsof.2017.01.006` doi: 10.1016/j.infsof.2017.01.006

Vessey, I., Ramesh, V., & Glass, R. L. (2005). A unified classification system for research in the computing disciplines. *Information and Software Technology*, *47*(4), 245–255. doi: 10.1016/ j.infsof.2004.08.006

Wallace, J. R., Mandryk, R. L., & Inkpen, K. M. (2008). Comparing content and input redirection

in MDEs. *Proceedings of the ACM 2008 conference on Computer supported cooperative work - CSCW '08*, 157. Retrieved from `http://portal.acm.org/citation.cfm?doid=1460563.1460588` doi: 10.1145/1460563.1460588

Wohlin, C. (2014). Writing for synthesis of evidence in empirical software engineering. *Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement - ESEM '14*, 1–4. Retrieved from `http://dl.acm.org/citation.cfm?doid=2652524.2652559` doi: 10.1145/2652524.2652559

Yao, J., Huang, Y., & Cui, B. (2009). Constructing evolutionary taxonomy of collaborative tagging systems. *Proceeding of the 18th ACM conference on Information and knowledge management - CIKM '09*, 2085. Retrieved from `http://portal.acm.org/citation.cfm?doid=1645953.1646314` doi: 10.1145/1645953.1646314