Thesis project

Is a graph-based interface more efficient than a traditional one for channel discovery?

Authors: Parastou Khosrowshahi Kamran Barakpour
Supervisor: Ola Petersson
Examiner: Mathias Hedenberg
Semester: VT 2014
Subject: Computer Science
Abstract
There are several applications which make it possible for users to search and find their favorite channels; these channels could be anything such as TV channels, radio channels, blogs, websites, etc.

Is a graph-based interface more efficient than a traditional one for channel discovery? The aim of this thesis is to answer the question by developing a new browsing method in a graphical user-friendly interface and compare it with a traditional hierarchical method. A graphical application is a graph-based application, in which the nodes (or vertices) represent channels and the edges represent the links among channels.

The graphical application is built with the newest Hypertext Markup Language Standard (HTML5) and Cascading Style Sheets (CSS3). A large part of the graphical application is built using the JavaScript library Sigma.js. The resulting graphical application is tested with users and compared with the traditional hierarchical user interface. A questionnaire is provided for end-users to gather feedback. After analyzing the participant’s feedback, the results were surprising and interesting. A majority of participants enjoyed interacting with graphical application, and they were interested to use it in the future. The final result is merged into the thesis report.

This thesis includes all the work of creating graphical and hierarchical user interfaces, testing the hierarchical and graphical user interfaces, and analyzing the user’s feedback.

Key-words: Sigma JavaScript library, channel discovery, Visualization, HTML5, CSS3, Hierarchy
## Contents

1 Introduction .................................................................................................................... 1

1.1 Background .................................................................................................................. 1

1.2 Previous research ....................................................................................................... 2

1.3 Problem definition ...................................................................................................... 3

1.4 Purpose and research question/hypothesis ................................................................ 3

1.5 Scope/Limitation .......................................................................................................... 3

1.6 Goals and Criteria ...................................................................................................... 4

1.7 Outline ......................................................................................................................... 4

2 Theory ............................................................................................................................... 4

2.1 Hierarchical ................................................................................................................ 5

2.2 Visualization ............................................................................................................... 5

2.3 Graph Theory .............................................................................................................. 5

3 Method .............................................................................................................................. 6

3.1 Scientific approach ..................................................................................................... 6

3.2 Project Requirements ................................................................................................. 6

3.2.1 Similarity in nodes ................................................................................................. 7

3.3 Data Collection .......................................................................................................... 7

3.4 Software Development ............................................................................................... 8

3.5 Test Conduction ......................................................................................................... 9

3.5.1 Usability Testing .................................................................................................... 9

3.5.2 Direct observation in a Lab Setting (lab based method) ....................................... 9

3.5.3 Hierarchical application test Scenario .................................................................. 9

3.5.4 Graphical application test Scenario ...................................................................... 11

3.6 Analysis ...................................................................................................................... 12

4 Summary of empirical data .......................................................................................... 13

4.1 Approaches for reporting results ............................................................................... 13

4.2 Analyzing data process ............................................................................................. 14

5 Analysis .......................................................................................................................... 15

6 Discussion ....................................................................................................................... 25
1 Introduction
This chapter describes the background information of the thesis work including the methods and research that have been done so far. This chapter also gives a brief summary of the techniques, methods and research that will be done during the thesis work.

1.1 Background
Although the main goal of this thesis is to make a comparison between two user interfaces and reflect the user’s ideas about this comparison, first we need to provide some explanation in order to clarify the problem to the readers.

Assume an end-user who wants to discover new things that he/she might be interested in, such as a new TV channel, a radio channel, a blog, a website, or anything else, all represented as a channel in hierarchical and graphical applications. To model such applications we could have a hierarchical user interface that includes all the channels in hierarchical categories or we could have some other interfaces like the graph-based user interface that provides functionalities for the end-user to discover new channels. The UIs (hierarchical and graphical user interfaces) are just models that could be applied to many different things, all called channels in graphical and hierarchical applications.

Navigating a hierarchy is one of the most popular methods for discovering things. Hierarchical user interface is a simple application containing channels, in which channels are included and related together based on hierarchical structure. In order to use a hierarchical interface, the user chooses a related category and goes through some hierarchical steps to find his/her favorite channel. Figure 1.1 shows an example of hierarchical application.

![Figure 1.1: Example of hierarchical user interface structure](image-url)
A Graphical user interface is a new idea for discovering channels that presents the visualized data. The end-user can simply open the application and start discovering channels by interacting with the interface. An end-user can find his/her favorite channels or discover new channels which are suggested visually by the application. Figure 1.2 shows an abstract example of graphical application.

![Figure 1.2: Example of graphical user interface](image)

In our thesis, we create both UIs, test them and analyze the result from a comparison between them.

### 1.2 Previous research

Hierarchy is one of the most popular methods for discovering things and it has been used in many different ways. “In a computing context, there are various types of hierarchical systems. Most file systems, for example, are based on a hierarchical model in which files are placed somewhere in a hierarchical tree model” [1]. In a hierarchical file system, the files, folders, and drivers are displayed in groups, which allow the user to see only the files they are interested in seeing.

Hierarchical data is another example of hierarchy. Hierarchical data is a group of data where there is a hierarchical relationship among data items. “Hierarchical data can be found in a variety of database applications, including forum and mailing list threads, business organization charts, content management categories, and product categories” [2].

A Graphical user interface is a new idea for discovering channels by visualizing the data. Data visualization has a long story. From ancient cave painting to a very modern, complicated visualization tools we have today.

Before 17th century, data visualization was used mainly in maps for displaying cities, roads, resources, etc. It is believed that Michael Florent Van Langren provided the first visual representation of statistical data in 1644 [3].

The 18th century granted us a Scottish engineer named William Playfair. He represented new ways of visualization. In his work he invented four types of the most popular graphs we use today (line, bar, circle, and pie charts).

In the early 20th century, different types of charts and graphs were heavily used in business applications, textbooks, science, and government. In the last two decades, data visualization has been a very popular field that extended into many focus areas. Today,
a big group of visualization tools are used by researchers, businesses, and individuals [3].

Graph visualization is used in various types of applications. A file hierarchy in a computer system is a common type of hierarchical navigation tool that allows the end user to navigate through the file hierarchy to find a specific file. The reference article talks about problems that could have been experienced by end users who have tried to navigate through the file hierarchy. The article also describes different types of graph visualization layouts. Navigation and interaction in information visualization is another important topic, which has been discussed in the article briefly [4].

1.3 Problem definition
Although navigating hierarchically is one of the most common methods for discovering channels, there are some restrictions associated with a hierarchical application which makes it overwhelming for the users to search and find their favorite channels. For instance, if a user tries to find a channel, he/she should follow some steps and guess an answer at each step (suggested by the application) to find a channel. This means that the user is bound to follow the application’s suggestions; in other words, the user is limited to someone else’s “frame of mind”. That being said, there is a risk of never finding the desired channel even if that channel exists. Additionally, the user may never be sure whether the channel exists or not.

To avoid such limitations, we explore other ways to find channels to make it possible for users to interact with a graphical user interface in a way that they do not have to follow the hierarchical steps to find their favorite channels. We intend to create, develop and test a graphical application to find out how useful our graphical application could be for discovering channels.

1.4 Purpose and research question/hypothesis

| RQ1. | Is a graph-based interface more efficient than a traditional one for channel discovery? |

A graphical application is a user interface based on graphs represented by nodes and links. The nodes represent the channels and the links represent the similarity between channels. In fact, the graphical application is a network of nodes (channels) that allows a user to see all the channels available in only one step. The user can easily interact with the interface to find his/her favorite channel. Dragging, zooming in, zooming out, pop-up box, color coded nodes are the functions available for a user while interacting with the application.

1.5 Scope/Limitation
Due to time limitations, we could not implement the graphical application with a large amount of nodes. Extracting data from different APIs requires a lot of time, which was not possible due to our time limit. Therefore, we used our own hypothetical data in our applications to be able to finish our investigation in time. The applications were implemented as web applications and could be used and tested on different devices.
Unfortunately there was no time to use and test the application on any mobile devices. All tests were performed on a local server.

1.6 Goals and Criteria
This section describes the practical goals pursued by this work in order to solve the problem.

- The first goal was to create a graphical application. To build the application we used Microsoft Visual Studio as a platform. HTML5, CSS3 and JavaScript are the technologies that are used for creating the application. The language used for developing the application is C#.

- The second goal was to create a simple hierarchical user interface using HTML, CSS and JavaScript technologies.

- The third goal was about testing the applications. The applications were tested by real end-users in order to get the end-users’ ideas about both applications.

- The last goal was to analyze the end-users’ feedback in order to provide a final result of the comparison between the graphical and hierarchical applications.

1.7 Outline
This thesis is divided into six chapters. Each chapter has some sections discussing subjects related to the chapter’s major subject. In chapter 2, we define the background and theory of important topics such as graph theory and visualization that are used in this thesis work. In chapter 3, we describe the process of practical work concerning choosing the data, implementing and testing the applications. In chapter 4, we analyze the test’s results extracted from the end-users’ feedback. In chapter 5, we discuss the results. In chapter 6, we make a conclusion and provide a summary of the whole work. We also outline some future work that could be done later in order to improve this work.
2 Theory

In this chapter we explore the literature related to the concepts hierarchical, visualization and graphs. This chapter describes the theories on which the practical work of this thesis is based on, in order to help the reader to understand the topics better.

2.1 Hierarchical

“A hierarchy is an organizational structure in which items are ranked according to levels of importance” [5]. Hierarchical means having a structure consisting of multiple levels. Alternatively, anything that can be broken down into a series of levels which range from broad to specific can be considered a hierarchy.

In other words, a hierarchy is a kind of structure where items are graded based on levels of belonging. For instance, a hierarchical user interface is in our thesis a navigation tool including categorized entities that makes the search easier for the user. The user can make the hierarchical search by going through the specific categories in one or several steps.

2.2 Visualization

“Data visualization is a general term used to describe any technology that lets corporate executive and other end users “see” date in order to help them better understand the information and put it in a business context” [6].

Visualizing data is an important aspect in creating a graphical application. A graphical application enables the end-user to see the data while interacting with the application. Visualization of data has many advantages. For instance, in a graphical application the nodes (channels) have different sizes depending on their popularity, which could be useful for an end-user while interacting with the graphical application. In our graphical application, similar entities are linked together and all of them are visible in order to help the user to find new channels more easily.

2.3 Graph Theory

“A graph is a way of representing connections between places. Mathematically, a graph is a collection of nodes and edges. Nodes are locations that are connected together by the edges of the graph” [7].

In computer science, graphs are used to model various things, for example networks of communication. In our thesis, a graphical application is a graph-based application in which the nodes (or vertices) represent channels and the edges represent the links among channels.
3 Method

In this chapter, we describe the process of the practical work, and explain the different steps of developing and testing the applications in detail. Since our project is primarily a practical approach, we divide this project into six main steps:

- Project Requirements
- Data collection
- Similarity formula
- Software development
- Conducting tests
- Analysis

In the first section, we describe the scientific approach of our thesis, and then the main steps of our project through the following sections are explained in detail.

3.1 Scientific approach

For our thesis work we chose an inductive approach. An inductive approach is a type of thinking that involves identifying patterns in a data set to reach conclusions and build theories [8].

Our thesis is a quantitative study that intends to reflect a practical result of comparing hierarchical and graphical applications. We use a quantitative approach in our work where experiments on real participants are conducted, and a survey is provided to be filled in by participants. We measure a result by analyzing the answers from a survey [9].

3.2 Project Requirements

There are two user interfaces (hierarchical and graphical) that needs to be designed and developed. In this subsection we provide a list of requirements for both applications. The requirements are provided by our own idea in order to develop applications in a way we aimed to be.

The hierarchical user interface is implemented in HTML. The requirements that should be fulfilled in the hierarchical user interface are:

- Total number of channels is 50.
- Sports, Music, Movies and Games are the categories that should be shown in the first page.
- The name of boxes (channels) in the hierarchical user interface should be exactly the same as the name of nodes (channels) in the graphical application.
- Whenever a user clicks on any box, he/she should be sent to a page including same categorized channels or new subcategories page.
- When the user clicks on any channel, a box should pop up giving a description of the channel.

The requirements that should be fulfilled in the graphical application are listed below:

- Total number of nodes is 50.
- Each node represents a specific channel.
- Similar nodes should be connected to each other.
• The size of a node depends on the number of subscribers to the corresponding channel.
• When a user clicks on any node, the box should pop up including the description about the channel and a subscribe button.
• The layout must include zoom in and zoom out functionality.
• If a user subscribes to a channel, the color of the corresponding node should indicate this.

3.2.1 Similarity in nodes
When we talk about similarity between nodes, there are many different aspects that could be considered. In our graphical application, we consider only one specific feature to determine similarity among nodes, namely the size of the intersection between the set of subscribers to the corresponding channels. For example, if there are two nodes (channels) with 30 and 40 subscribers respectively, and 10 subscribers of the first channel are also subscribers to the second channel, we say that these two nodes are similar since the intersection of the subscribers of these two nodes is of size 10. There should be a formula to determine a minimum overlap of the same subscribers in two nodes for different situations in order to specify similarity between these two nodes. For instance, if we have a node with 1000 subscribers and another node with 10 subscribers and the overlap is 10, it is not sufficient to say that these two nodes are similar. However, if the overlap is 100, we consider the nodes similar.

To be more precise, we need to provide a mathematical formula to determine similarity between nodes. Hans Frisk, a mathematics professor at Linnaeus University, assisted us in finding a formula. Equation 1 shows that two nodes are similar if

\[(c/a) \times 10^{(c/b)} > 0.085 \]  

In the formula above, \(a\) represents the number of subscribers of the first node, \(b\) represents the number of subscribers of the second node, and \(c\) represents the overlap of the same subscribers from the first and second node.

3.3 Data Collection
We need to provide data for both applications. This data should be exactly the same in both applications since we intend to test and compare the hierarchical and graphical applications using the same data.

Although extracting real data from social networks such as Twitter and YouTube could be a good idea, the amount of data in such APIs is big and it is not suitable for our applications since we only have 50 channels. Therefore, we create our own hypothetical data and insert exactly the same hypothetical data in both hierarchical and graphical applications.

There are four main categories that cover all 50 channels in our applications. The categories with number of related channels are listed below:
• Sports: 13 (subcategories: Football(9), Tennis(4))
• Music: 12
• Movies: 13
• Games:12

3.4 Software Development
In the hierarchical application, we only need to present channels inside different web pages and link the pages together. We use the Sublime text editor (http://www.sublimetext.com/) for writing the HTML and CSS codes. We use HTML and CSS technologies to create the user interface pages and link them together, and JavaScript for creating a pop-up box for each channel.

Figure 3.1 shows the first page of the hierarchical user interface, which includes the four main categories. The user can simply click on each box and go either through the channels related to the chosen category or other subcategorized pages.

![Categories](image.jpg)

Figure 3.1: Categories

Developing a graphical application is technically more demanding. We use Microsoft Visual Studio 2010 as a platform for developing the graphical application. We use HTML5, CSS3 and JavaScript for implementing the application. We need to use a graph visualization library for creating the nodes and edges in a graphical application. We choose Sigma.js as a JavaScript library for this.

Figure 3.2 shows the graphical application, in which the nodes in blue show the channels that user has already subscribed to, the node in orange is the node that the user clicked on, and the nodes in red show the channels that are similar and linked to the clicked channel (orange node).
3.5 Test Conduction
Test Conduction is a very important part of our thesis. The details of the techniques and methods that we use to conduct the test are mentioned below.

3.5.1 Usability Testing
Usability testing refers to evaluating a product or service by testing it with representative users. During a test participants typically interact with a product while spectators supervise and take notes. The goal is to identify any usability problems, collect data and determine the participant's satisfaction with the product [10].

Usability testing focuses on measuring a human-made product's capacity to meet its intended purpose. Usability testing perfectly matches the requirements we intend to test in our project. Therefore, we choose the usability testing method for conducting tests in our project. We test both hierarchical and graphical interfaces based on the usability testing technique. A quantitative data is collected for this part of our work.

3.5.2 Direct observation in a Lab Setting (lab based method)
Direct observation is used as a method of testing for our project [11]. Totally five students from Linnaeus university were participated in the test. The participants have been randomly chosen with different study background.

To conduct the test, we provide a test environment and present both hierarchical and graphical user interfaces to the end-users. We also give the participants a quick introduction of how to work with applications. Finally, we ask participants to work and interact with the UIs.

3.5.3 Hierarchical application test scenario
This subsection describes how an end-user could interact and test the hierarchical application. The hierarchical application lets users search and find a specific channel by going through some hierarchical navigational steps. Suppose an end-user wants to search for the Manchester United channel. The user then opens the hierarchical
application, and he/she is presented with a page containing category boxes. The user is likely to know what category belong to. In our example, the user would choose the “Sport” category. On second page, there are also boxes with different tags (or subcategories) related to the sport category, and here again an end-user has to choose a correct tag to be linked to the next page including channels with similar categories. In our example, an end-user must choose and click on “Football” box to be linked to another page. Finally the end-user can find the specific channel (Manchester United channel) in the third hierarchical step. If an end-user clicks on “Manchester United” channel, a box pops up including some information about the channel.

Figures 3.3, 3.4 and 3.5 show the flow that end-users could use to find the Manchester United channel using a hierarchical approach.

![Categories](image)

**Figure 3.3: Categories**

![Sport Channels](image)

**Sport Channels**
3.5.4 Graphical application test scenario
This subsection describes how an end-user could interact and test the graphical application. The graphical application presents all available channels to an end-user at once. In order to show how the graphical application works, we created an example to help the reader understand the differences from the hierarchical application. Assume an end-user finds the “Premier League” channel. When clicking on the channel, the color of the corresponding node becomes orange, other channels that are similar to this channel become red. The end-user can see that the clicked channel is connected to some channels. The “Manchester United” channel is in red and connected to the “Premier league” channel. Therefore, if an end-user keeps discovering and following the links, he/she can easily find the “Manchester United channel” in red color which is linked to “Premier league” channel. An end-user can click on the “Manchester United” channel, and a popup box including some information about “Manchester United channel” and a subscribe button is shown. If an end-user finds the information interesting, he/she can simply press the subscribe button to subscribe to the channel.

Figure 3.6 shows how an end-user could discover and find the “Manchester United” channel in the graphical application.
3.6 Analysis

After the participants were done working with the applications, they were asked to fill in a questionnaire. In chapter 4, the analyses of the results from the actual questionnaires are discussed. The questionnaire is included as an appendix at the end of this report.
4 Summary of empirical data
This chapter reflects the raw results and a summary of data collected from the questionnaires. This chapter also describes the process of analyzing the results.

4.1 Approaches for reporting results
Once the participants filled out the questionnaire, we transferred the responses from the questionnaire into a spreadsheet. Table 4.1 shows an overview of the final results.

<table>
<thead>
<tr>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1</td>
<td>Agree</td>
<td>Neither agree nor disagree</td>
</tr>
<tr>
<td>User 2</td>
<td>Agree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>User 3</td>
<td>Agree</td>
<td>Agree</td>
</tr>
<tr>
<td>User 4</td>
<td>Disagree</td>
<td>Neither agree nor disagree</td>
</tr>
<tr>
<td>User 5</td>
<td>Strongly agree</td>
<td>Agree</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Q4</th>
<th>Q5</th>
<th>Q6</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1</td>
<td>Neither agree nor disagree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>User 2</td>
<td>Neither agree nor disagree</td>
<td>Neither agree nor disagree</td>
</tr>
<tr>
<td>User 3</td>
<td>Agree</td>
<td>Neither agree nor disagree</td>
</tr>
<tr>
<td>User 4</td>
<td>Neither agree nor disagree</td>
<td>Disagree</td>
</tr>
<tr>
<td>User 5</td>
<td>Agree</td>
<td>Agree</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Q7</th>
<th>Q8</th>
<th>Q9</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1</td>
<td>Agree</td>
<td>Disagree</td>
</tr>
<tr>
<td>User 2</td>
<td>Agree</td>
<td>Disagree</td>
</tr>
<tr>
<td>User 3</td>
<td>Strongly agree</td>
<td>Neither agree nor disagree</td>
</tr>
<tr>
<td>User 4</td>
<td>Disagree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>User 5</td>
<td>Strongly agree</td>
<td>Agree</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Q10</th>
<th>Q11</th>
<th>Q12</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1</td>
<td>Disagree</td>
<td>Disagree</td>
</tr>
<tr>
<td>User 2</td>
<td>Agree</td>
<td>Neither agree nor disagree</td>
</tr>
<tr>
<td>User 3</td>
<td>Agree</td>
<td>Neither agree nor disagree</td>
</tr>
<tr>
<td>User 4</td>
<td>Agree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>User 5</td>
<td>Agree</td>
<td>Neither agree nor disagree</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Q13</th>
<th>Q14</th>
<th>Q15</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1</td>
<td>Agree</td>
<td>Agree</td>
</tr>
<tr>
<td>User 2</td>
<td>Strongly agree</td>
<td>Agree</td>
</tr>
<tr>
<td>User 3</td>
<td>Strongly agree</td>
<td>Agree</td>
</tr>
<tr>
<td>User 4</td>
<td>Strongly agree</td>
<td>Agree</td>
</tr>
<tr>
<td>-------</td>
<td>----------------</td>
<td>-------</td>
</tr>
<tr>
<td>User 5</td>
<td>Agree</td>
<td>Agree</td>
</tr>
</tbody>
</table>

Table 4.1: An overview of the final result

4.2 Analyzing data process
Google Forms is a free and easy to use. Google Forms offers multiple layout options that are very simple to use. After collecting the data, Google form provides some powerful ways to analyze the data [13].

In our thesis work, we used a Google form to create our survey. After editing and creating the survey, it was filled in by the participants in our study. Then, we used Google form to analyze the responses and build pie charts from the collected data.
5 Analysis
There are in total 15 questions in the questionnaire and it was answered by five participants. The answers for each question were analyzed separately. In this chapter, we present the result of the analysis of each question. The result of the analysis is presented in the pie charts below. A summary of the comparison between hierarchical and graphical applications are discussed below the pie charts.

1. This is fun working with graphical user interface.

![Pie chart showing the results of question 1]

<table>
<thead>
<tr>
<th>Response</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly agree</td>
<td>1</td>
<td>20%</td>
</tr>
<tr>
<td>Agree</td>
<td>3</td>
<td>60%</td>
</tr>
<tr>
<td>Neither agree nor disagree</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>Disagree</td>
<td>1</td>
<td>20%</td>
</tr>
<tr>
<td>Strongly disagree</td>
<td>0</td>
<td>0%</td>
</tr>
</tbody>
</table>

Figure 5.1: Question one

Figure 5.1 shows how much an end-user could enjoy interacting with the graphical application. The average result is 3.8 meaning that a majority of the participants thought it was fun working with the graphical application.
2. Graphical application is a useful user interface for discovery channels.

Figure 5.2: Question two

Figure 5.2 shows whether a graphical application could be a useful application for discovering channels. The average result is 3.8 showing that most participants thought a graphical application is a good tool for channel discovery.

3. You prefer to use graphical user interface next time.

Figure 5.3: Question three
Figure 5.3 shows how much participants is interested in using a graphical application in the future. The average result is 3.8 meaning that participants were positive about using the graphical application in the future.

4. Functions in graphical user interface were very useful for you to find channels easier.

![Pie chart showing results]

<table>
<thead>
<tr>
<th>Strongly agree</th>
<th>Agree</th>
<th>Neither agree nor disagree</th>
<th>Disagree</th>
<th>Strongly Disagree</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 5.4: Question four

Figure 5.4 shows if the functionality provided for graphical application are helpful and make it easy for the end user to use the application. The average result is 3.4 meaning that the participants found the functionality quite useful.

5. You were able to find channels very fast using graphical user interface.

![Pie chart showing results]

<table>
<thead>
<tr>
<th>Strongly agree</th>
<th>Agree</th>
<th>Neither agree nor disagree</th>
<th>Disagree</th>
<th>Strongly Disagree</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 5.5: Question five
Figure 5.5 shows how fast applicants were able to find channels using the graphical application. The average result is 3.2 meaning that the participants were not able to find channels pretty fast.

6. In graphical application, when you click on any channel, some other channels related to that clicked channel become highlighted; it was very helpful for you to find new channels close to your interests.

![Pie chart showing responses to the question.]

<table>
<thead>
<tr>
<th>Response</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly agree</td>
<td>1</td>
<td>20%</td>
</tr>
<tr>
<td>Agree</td>
<td>2</td>
<td>40%</td>
</tr>
<tr>
<td>Neither agree nor disagree</td>
<td>1</td>
<td>20%</td>
</tr>
<tr>
<td>Disagree</td>
<td>1</td>
<td>20%</td>
</tr>
<tr>
<td>Strongly Disagree</td>
<td>0</td>
<td>0%</td>
</tr>
</tbody>
</table>

Figure 5.6: Question six

Figure 5.6 shows if the functionality of coloring channels could be helpful for an end-user to make it easier to find similar channel. The average result is 3.6 meaning that the coloring functionality for similar channels is useful in the graphical application.

7. Graphical application uses blue color for channels you already subscribed to, it was very helpful for you to find those channels easily.

![Pie chart showing responses to the question.]

<table>
<thead>
<tr>
<th>Response</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly agree</td>
<td>2</td>
<td>40%</td>
</tr>
<tr>
<td>Agree</td>
<td>2</td>
<td>40%</td>
</tr>
<tr>
<td>Neither agree nor disagree</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>Disagree</td>
<td>1</td>
<td>20%</td>
</tr>
<tr>
<td>Strongly Disagree</td>
<td>0</td>
<td>0%</td>
</tr>
</tbody>
</table>
Figure 5.7: Question seven

Figure 5.7 shows if a function of coloring channel could be helpful for an end-user to see channels he/she already is subscribed to. The average result is 4 meaning that this functionality is very helpful.

8. In order to find a specific channel, working with hierarchical application was easier and faster than working with graphical application.

<table>
<thead>
<tr>
<th>Opinion</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly agree</td>
<td>1</td>
<td>20%</td>
</tr>
<tr>
<td>Agree</td>
<td>1</td>
<td>20%</td>
</tr>
<tr>
<td>Neither agree nor disagree</td>
<td>1</td>
<td>20%</td>
</tr>
<tr>
<td>Disagree</td>
<td>2</td>
<td>40%</td>
</tr>
<tr>
<td>Strongly Disagree</td>
<td>0</td>
<td>0%</td>
</tr>
</tbody>
</table>

Figure 5.8: Question eight

Figure 5.8 shows if applicants had to spend less time for finding a specific channel using the hierarchical application compared to the graphical application. The average result is 3.2 meaning that finding a specific channel takes equal time using either the hierarchical application or the graphical application.
9. This is fun working with the hierarchical user interface.

![Pie chart showing the responses to question nine]

Strongly agree: 0
Agree: 1
Neither agree nor disagree: 3
Disagree: 1
Strongly Disagree: 0

Figure 5.9: Question nine

Figure 5.9 shows how much an end-user could enjoy interacting with the hierarchical application. The average result is 3 which is neutral.

10. You found a hierarchical application very useful for finding channels.

![Pie chart showing the responses to question ten]

Strongly agree: 0 (0%)
Agree: 4 (80%)
Neither agree nor disagree: 0 (0%)
Disagree: 1 (20%)
Strongly Disagree: 0 (0%)

Figure 5.10: Question ten
Figure 5.10 shows if participants found a hierarchical application useful for finding channels. The average result is 3.6 meaning that the hierarchical is useful for finding channels.

11. you prefer to use hierarchical application for next time.

Figure 5.11: Question eleven

Figure 5.11 shows how much participants are interested in using a hierarchical application in the future. The average result is 3.2 meaning that the answer is just above average, and not very positive.
12. You were able to find channels very fast using hierarchical application.

Figure 5.12: Question twelve

Figure 5.12 shows how fast applicants were able to find channels using the hierarchical application. The average result is 4.2 meaning that a majority of the participants were able to find channels very fast.

13. Hierarchical application categorizes the subjects; it is very helpful to find a specific channel.

Figure 5.13: Question thirteen
Figure 5.13 shows how useful it is to have channels categorized in a hierarchical application. The average result is 4.6 meaning that using categories for channels in the hierarchical application is a very important and useful feature.

**14. It is a good idea to use graphical application on mobile devices.**

![Pie chart showing the results of the survey](image)

<table>
<thead>
<tr>
<th>Response</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly agree</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>Agree</td>
<td>5</td>
<td>100%</td>
</tr>
<tr>
<td>Neither agree nor disagree</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>Disagree</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>Strongly Disagree</td>
<td>0</td>
<td>0%</td>
</tr>
</tbody>
</table>

Figure 5.14: Question fourteen

Figure 5.14 shows what percentage of participants that support the idea of using the graphical user interface on mobile devices. All participants agreed that it is a good idea to use the hierarchical user interface on mobile devices.
15. It is a good idea to use hierarchical application on mobile devices.

![Pie chart showing the responses to the question](image)

<table>
<thead>
<tr>
<th>Response</th>
<th>Count</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly agree</td>
<td>1</td>
<td>20%</td>
</tr>
<tr>
<td>Agree</td>
<td>2</td>
<td>40%</td>
</tr>
<tr>
<td>Neither agree nor disagree</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>Disagree</td>
<td>2</td>
<td>40%</td>
</tr>
<tr>
<td>Strongly Disagree</td>
<td>0</td>
<td>0%</td>
</tr>
</tbody>
</table>

Figure 5.15: Question fifteen

Figure 5.15 shows what percentage of participants that support the idea of using the hierarchical user interface on mobile devices. The average result is 3.4 meaning that there is some indications that the participant agrees that the idea of using the hierarchical user interface on mobile devices would be good.

Here we discuss a summary of the comparison between the hierarchical and the graphical applications based on the results extracted from the questionnaire.

A. The comparison shows that 80% of the users had fun working with the graphical application while the overall result of the same experience for the hierarchical application is only 20%.

B. Users found both a graphical and a hierarchical application equally useful for discovering channels.

C. 80% of the users prefer to use the graphical application next time.

D. 100% of the users believe that using the graphical user interface on mobile devices would be a good idea while only 60% of them support the idea of using the hierarchical user interface on mobile devices.

E. 40% of the users were happy with the functionality provided on the graphical application and found them very helpful for finding channels.
6 Discussion
This chapter discusses the problem, the method of solving the problem and the results of the study. In this chapter we discuss our own thoughts and reflections, and compare them with the results extracted from the survey to evaluate how similar they are.

6.1 Problem
In our thesis project we created a hierarchical and a graphical application, which both were used to find new channels. Although using a hierarchy is a popular method for discovering things, there are some restrictions associated with common hierarchical applications that makes it overwhelming for users to search and find their favorite channels. For instance, in sub section 3.6.4 we showed an example of an end-user who tries to find a “Manchester United” channel via hierarchical application. As mentioned in that section, the end-user has to know the category of that specific channel and follow the right hierarchical steps to find it. Therefore, if an end-user chooses the wrong category he/she will not be able to find the specific channel. Moreover, he/she cannot be sure if a “Manchester United” channel exists or not.

6.2 Solving the problem
In sub section 3.6.5 we made the same example for an end user using the graphical application. The user was able to find the “Manchester United” channel while browsing through the graphical application. In the graphical application there are no hierarchical steps and an end-user does not need to have any background information for discovering and finding channels. In addition, all existing channels are shown to the end-user when an end-user opens a graphical application. Therefore, it reduces the risk of missing an existing channel.

6.3 Result
To share and test our idea, we created both a hierarchical and a graphical application. Five participants were asked to test both applications and fill in a survey about their experiences. After analyzing and comparing the results, we found out that the opinions of the participants are very close to what we expected. Most users found the graphical application fun, useful and would prefer to use it in the future.

The graphical application provides a new method of finding channels by visualizing the data for the end-user. We believe a graphical application could be more effective than traditional methods such as the hierarchical method for finding channels.
7 Conclusions
This chapter summarizes the results and concludes the work of this thesis. There is also a section regarding future work, where we provide some suggestions for further research within the area.

7.1 Conclusions
The research question of our thesis is about investigating if a graph-based interface is more efficient than a traditional one for channel discovery? Although there is no explicit "yes" or "no" answer for our research question, we found out some interesting results in our thesis work. According to what was previously stated, both hierarchical and graphical user interfaces are useful in their own unique ways. The hierarchical application could be more useful for a user who has information about a specific channel and tries to find that channel in some easy steps. The graphical application visualizes a set of data allowing the user to see all the channels and have fun while discovering new channels.

According to the results extracted from the users’ feedback, most of the users found the graphical application fun, useful and would prefer to use it in the future. Since channels are categorized in the hierarchical application, users spent less time finding a specific channel in the hierarchical application. All users believed that using the graphical user interface on mobile devices would be a good idea while only 60% of them supported the idea of using the hierarchical user interface on mobile devices. Although we only developed the graphical application with only 50 nodes, we believe the graphical application would be useful with much more nodes (500 or even more). We think the zoom functionality in the graphical application would help to use the application with no difficulties for the cases with smaller sizes such as on mobile devices. Developing the similar graphical application with more nodes and real data for mobile devices would be a good topic for the future work.

7.2 Future Research
This section is about the possible future improvements and some suggestions for further research related to our project.

We believe if more time and budget was spent on the graphical application, the application would have been implemented in a much more fancy way which would have fulfilled the requirements efficiently and the end user probably would have been happier with the graphical application. Interacting with the graphical application with more channels and real data would be more interesting and entertaining for the end-user. We also believe that this application will be more useful and interesting on mobile devices and based on our survey, most users agreed with us. This is something that we have not tested yet.

Although we used Sigma.js as the graph visualization tool in our graphical application, there are many more visualization libraries that satisfy the requirements and that possibly could have provided more functionality in the graphical application. The formula used for drawing edges between similar nodes could also be compared with other formulas to find the most useful one.
References


Appendix

Questionnaire

1. This is fun working with graphical user interface.
   ○ Strongly agree
   ○ Agree
   ○ Neither agree nor disagree
   ○ Disagree
   ○ Strongly disagree

2. Graphical application is a useful user interface for discovery channels.
   ○ Strongly agree
   ○ Agree
   ○ Neither agree nor disagree
   ○ Disagree
   ○ Strongly disagree

3. You prefer to use graphical user interface next time.
   ○ Strongly agree
   ○ Agree
   ○ Neither agree nor disagree
   ○ Disagree
   ○ Strongly disagree

4. Functions in graphical user interface were very useful for you to find channels easier.
   ○ Strongly agree
   ○ Agree
   ○ Neither agree nor disagree
   ○ Disagree
   ○ Strongly disagree

5. You were able to find channels very fast using graphical user interface.
   ○ Strongly agree
   ○ Agree
   ○ Neither agree nor disagree
   ○ Disagree
6. In graphical application, when you click on any channel, some other channels related to that clicked channel become highlighted; it was very helpful for you to find new channels close to your interests.

- Strongly disagree
- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

7. Graphical application uses blue color for channels you already subscribed to, it was very helpful for you to find those channels easily.

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

8. In order to find a specific channel, working with hierarchical application was easier and faster than working with graphical application.

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

9. This is fun working with the hierarchical user interface.

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

10. You found a hierarchical application very useful for finding channels.

- Strongly agree
- Agree
- Neither agree nor disagree
11. You prefer to use hierarchical application for next time.

☐ Strongly agree
☐ Agree
☐ Neither agree nor disagree
☐ Disagree
☐ Strongly disagree

12. You were able to find channels very fast using hierarchical application.

☐ Strongly agree
☐ Agree
☐ Neither agree nor disagree
☐ Disagree
☐ Strongly disagree

13. Hierarchical application categorizes the subjects; it is very helpful to find a specific channel.

☐ Strongly agree
☐ Agree
☐ Neither agree nor disagree
☐ Disagree
☐ Strongly disagree

14. It is a good idea to use graphical application on mobile devices.

☐ Strongly agree
☐ Agree
☐ Neither agree nor disagree
☐ Disagree
☐ Strongly disagree

15. It is a good idea to use hierarchical application on mobile devices.

☐ Strongly agree
☐ Agree
☐ Neither agree nor disagree
☐ Disagree
Strongly disagree