UNIVERSITY OF BUEA

Faculty of Science

Department of Computer Science

BACHELOR OF SCIENCE IN COMPUTER SCIENCE

Project Report

Towards Graph Relabeling as per Vantage Points

Taffo Nguemaleu Ange Kenza

SC16A732

Supervisor

William S. Shu, PhD

September 2019

DECLARATION

I hereby declare on my honor that this project report has been written by me, that to the best of my knowledge all borrowed ideas and materials have been duly acknowledged, and that it has not received any previous academic credit at this or any other institution.

_____

Taffo Nguemaleu Ange Kenza

SC16A732

Department of Computer Science

Faculty of Science

CERTIFICATION

This is to certify that this report entitled TOWARDS GRAPH RELABELING AS PER VANTAGE POINTS is the original work of TAFFO NGUEMALEU ANGE KENZA with Registration Number SC16A732, student of the Department Computer Science at the University of Buea. All borrowed ideas and materials have been duly acknowledged by means of references and citations. The report was supervised in accordance with the procedures laid down by the University of Buea. It has been read and approved by:

_____ _____

William S. Shu, PhD CEng CITP MBCS Date

_____ _____

Dr. Denis L. Nkweteyim Date

Head of Department of Computer Science

DEDICATION

This work is dedicated to my lovely and hard-working mother, Mrs. Moutcheu Tiako Honorine, and to the Almighty God.

ACKNOWLEDGMENT

I would like to thank my mother, Mrs Moutcheu Tiako and my sister, Miss Taffo Meryl for the support and care they gave me through out the production and realization of this project.

I owe everything to God Almighty, for His inspiration and guidance during the development o this project.

ABSTRACT

Graphs are a common methods to visually illustrate relationships in the data. The purpose of a graph is to present data that are too numerous or complicated to be described adequately in the text and in less space. Graphs evolve from the field of mathematics, which consist of sets of nodes and sets of edges. The main use of graphs is to describe the route from location to another.

A vantage point also called point of view is a place or position from which something is viewed or considered. Generally, a point can be viewed or perceived at different angles or positions and the observations made depending on the position of the points, by the other points involved or used must differ in most cases. Vantage points come in place when we want to have a reference given a number of points or objects used to solve a particular problem or get a particular output, just to name a few cases, and so the representation of these vantage points differ according to the problem or results you want to solve or obtain at the end.

In this project, I constructed a graph made up nodes and edges, which are labeled in the program. The nodes and edges have attributes which I displayed in the application. Depending on the point of view, the nodes and edges in the graph can be seen differently by an observer position. The other points in the graph also have their own way of seeing the vantage point, therefore we use the vantage point as a reference to analyze how things are view, or perceived in an organization. On the information, and details on the specifics of general graph representations, I have carried through the idea incorporating and formulating them to produce a more general construct.

I have developed a software (A Graph Relabeling as per Vantage points) that displays set of nodes and set of edges which connect a pair of nodes. The software allows the user to displace the nodes and edges on the screen, and place it where-ever he/she wants and the data displayed on the software can be save, from which it can be used by different institutions or organization for whatever purpose that suits them.

Table of Contents

CHAPTER 1

Introduction

In mathematics, a graph is a non-linear data structure consisting of nodes and edges. The nodes are sometimes also referred to as vertices and the edges are lines or arcs or arrows that connect any two nodes in the graph. A node is a basic unit of a data structure such as linked list or tree or graph data structure. A node can contain data and an edge can be a function or relationship.

For this project nodes and edges are labeled. Labeling or using a label is describing someone or something in a word or a short phrase.

A vantage point also called point of view is a place or position from which something is viewed or considered. The points are the nodes also referred to objects in this project. Depending on the point of view, the nodes and edges in the graph can be seen differently by an observer position. The other objects have different perception of the reference point which is the vantage point.

Purpose of the Project

The purpose of the project is to identify the way change is realized in an organization depending on how it is perceived. Depending on the reference point (vantage point) of an object, the other points contribute to what can be seen and observed from that vantage point. This generally means that a given vantage point sees the changes it makes and their effects on other points, and also has its own perception of the changes from other vantage points. So, a given vantage point has a way of seeing other points depending on the relationships with the points, and also the other points have a way of seeing the vantage point too. And perception differ depending on what is observed and which point is observing the change.

Its important to note that different viewpoints filter perception of the change differently, and perceive the perception from viewpoints differently. That is, a viewpoint may not necessarily exposed/show all its features or attributes for the other points to see, so it will filter what others points can see, same

things apply for the other viewpoints in the sense that the perception that the viewpoints display can be filtered.
The project also tries to connect standpoints external to the system of vantage points in a way the will view things. These standpoints see things differently, that is, the way they observed/perceived the given vantage point in the external point of view is completely different from the other perceptions observed by the other points, though they may require what are seen are essentially the same (they can used other points to see the vantage point in another way). Each vantage point is actually a collection of activities and outcomes of change, which are also labeled. Thus, each vantage is the point where the changes occur, the activities are the things you do to cause the change and the results of these change is the outcome of change that is labeled and stored.

After representing the graph, the user can then manipulate and save the relationship obtained from the various connections between the objects or nodes for various purposes. It will be also possible to record, manipulate information recorded in the graph so that it always remain consistent and present the information on-demand, but this project is not focused on recording and manipulation.

1.1 Background

Some of the motivation for doing this project include the following;

- Graph data structures are used in systems worldwide. When doing calculations in everyday life we need the basic knowledge of making use of graphs. Its not just for those that excel in math, but for every student to use according to their needs.
- Graphs can be used to model many types of relations and processes in physical, biological, social and information systems. Many practical problems can be represented by graphs. Emphasizing their real-world systems, the term ‘network’ is sometimes define to mean a graph in which attributes (e.g names) are associated with the vertices and edges, and the subject that expresses and understands the real-world systems as a network is called network science.
- In computer science, graphs are used to represent networks of communication, data organization, computational devices, the flow of computation, etc. So, to conclude a graph is a very useful tool that simplify the work of humans by providing quick data, help you keep track of things and allow humans to always be on the top.

- Vantage point on the other hand is very useful to keep a reference or have different points of view on different topics, problems and you can therefore obtain different results. When you have different point of views on a topic or problem, then it helps you to have different choices on which steps to make to forwards making you more effective, solve problems faster and obtain results faster as well.

Some of my personal motives for doing this project includes:

- Giving my interest in building Graphical User Interfaces (GUI), this project provides a strong usage of a lot of components in a GUI.
- Analysis of graph data structures to use, combined skills from courses learned at Level 200, Level 300 and Level 400 like csc301, csc405, csc207 and csc208.

We have talked about motives for the project, let's talk about a brief background on some of the areas relevant to our work.

Graphs can be used to model many types of relations and processes in physical, biological, social and information systems. This makes it possible for graphs to be used in many subjects such as Computer Science, Linguistics, Mathematics, Biology, Social Science, Physics and Chemistry. Therefore, Graphs can be highly used to solve real-world problems as this project specifies and provides solutions. Graphs are used as the basis of algorithms to solve problems. Representing problems in the form of a graph provide a simpler and faster way to humans to come out or obtain the data and hence be more effective in finding a solution to particular problems.

1.2 Project Objective

The objective of this project is to allow an observer to be able to record or save the changes observed in an organization depending on how he/she perceive the changes. It is important to do the project because, people have different views of things for different objects so they will like to be able to use the system in way that suits best their interest in order to be comfortable and get the results they want at the end.

The project must be diversified to include all different types of users or observer, and situations because the system must satisfy all users and situations. And also the system must not only be useful and usable but it must also be used by the users.

1.3 Parameters of the Project

I was unable to make the GUI beautiful because at the beginning, I had really had difficulties finding a GUI tool that has the features I needed for the project. I was also unable to properly implement the drag and drop functionality for the user to displace the nodes and arcs on the screen but what I tried to do write the code for it.

1.4 Report Structure

In Chapter 2, we are going to describe the analysis and design of the system, and then give the requirements gathered, methodology and finally the design. Then in Chapter 3, we will describe the implementation and findings of the project, and further give a discussion on the results obtained by stating what happened and why. Finally in Chapter 4, we will give a conclusion and also recommendations and suggestions for future research/work.

CHAPTER 2

Analysis and Design

2.1 Problem Statement

The way things are realized in an organization depends on how it is perceived. Basically in a real world situation the vantage point which is the reference point in the organization, which will contain internal items or attributes. This project seeks to provide a platform for representing nodes and arcs that are labeled based on the vantage point of the observer. The platform is implemented as a GUI (Graphical User Interface) where the vantage points, the nodes, the arcs and the attributes of the nodes are displayed. This GUI easily shows the different ways in which the nodes can be observed and perceived by observer and the relationships between nodes that is, the arcs between nodes are perceived depending on the vantage point and the results or observations obtained can then be used for various purposes in an organization or institution.
The project uses GUI tools and features to easily program and implement graph representation on the platform.

2.2 Project Aim

The aim of the project is to produce a graphical user interface with relevant text user interface where nodes and arcs are labeled based on the vantage points of their observer, which vantage point is defined by one of the graph nodes. Thus, a graph is used to represent nodes and arcs/arrows, one of the nodes is a vantage point depending on which node the observer chooses to be the vantage point, and the nodes and arcs are then labeled. The graphical user interface is used to display the relationships that exist between the nodes and arcs still depending on the vantage points of the observer.

The project demonstrates the vantage point labeling, and the change management that occur don’t need to be illustrated. The observer is the one who monitors the process of choosing the vantage points, so as the observer changes the vantage point, the arcs between nodes, and the labels to the arcs and the other

nodes change as if observed through filters as explained. That is the relationships between nodes depend on the vantage points the user chooses, if the vantage point changes the relationships changes together with the labels of the arcs and all the other nodes in accordance with the filters it is used to observe the vantage point.
The node of each vantage point has a substructure which, in its most general form is a graph that may also need labeling. This means that the project also give the possibility for the node of each vantage point to have a substructure embedded in it without affecting the whole graph containing the other nodes still related to that vantage point. And this substructure is also a graph so it will require labeling of the arcs between its other nodes, it will also contain a vantage point determined by the observer.

2.3 Methodology

For the analysis, I used the program development approach, which consists of stating the aim and objective of the project from the beginning before proceeding to give the interacting entities and the activities involved. I used this approach because it provides an easy way to grasp the purpose of the project from the beginning which helps you to plan well the process of developing the program.

The functional (algebraic) approach was used for the design which basically involves seeing most activities as operations, since some of the design issues to address, which was actually given in the csc208 notes, were automatically selected. I also used the bottom-up approach, which involves breaking the problem into smallest possible (and practical) parts. This approach is important because it helps you break down complex tasks, permitting you to start solving simple tasks and at the end merging the solutions obtained iteratively that is, again and again, until you have merged all the solutions obtained to get the final solution to the complex tasks.

In order to implement my design, I did a lot of research on different GUI tools to used that contains the features required for my project that is, finding a GUI tool that provides the widgets and facilities needed to easily implement the project. Some exploratory programming was done to in order to properly discover GUI tools that help display arrows and nodes.

Finally, the resources I used to carry out the project include; csc301 notes, csc208 notes, csc207 notes, csc306 notes, csc404 notes, the internet and all the materials given by the supervisor.

2.4 Interacting Entities and Activities

The system must be interactive for the user to use it and it must adapt to the chosen organization of the user which can be a University, a Hospital, a Pharmacy, etc… depending on the choice of the user. For the system to be interactive certain entities must be inputted by the user, activities must be carried out by the system on them and results of these activities will be the output given to the user.

a. Outline Model

The system will therefore include the following:

- Main Entities and Objects: Here we focus on the main nouns and noun phrases that seem important to description, recording their meanings, purposes and how they are used. The entities include;

  - Nodes: The nodes here are seen as objects present in the graph representations. Its purpose is to use these objects to represent the whole organization that the user chooses.

    - The nodes of the graph are objects which can contain items/variables, that is, characteristics or attributes that describe it, and this is different from the substructure.
    - Filters are just hidden items/variables that objects contain and can be made visible to other objects in the organization.

  - Arcs: The arcs are arrows showing or demonstrating the relationships between the nodes which are the objects. Its purpose is to show the relationships of each object in the organization with respect to the vantage point. It is used to show the different perceptions/changes and effects of the vantage point on the other objects, and also show the way the other points perceive the vantage point in the organization. Its attributes

    - The arcs are the arrows which shows the relationships between the nodes as functions to be carried out.
    - The results obtained from the functions are the outcomes of change, which are then labeled.

- Vantage Point: One of the graph nodes. Its the reference point in the organization, used to examine the changes with the other points.
    - If vantage point changes, the functions, the filters, the outcomes of change and the labels also change as well.

- Substructure: A vantage point containing another graph embedded in it. Its purpose is to demonstrate details of a vantage point as a graph, by showing the internal relationships of the nodes the vantage point contains. It is used for internal substructures of an organization that does not affect or change the whole structure of the organization.
    - If a vantage point contain a substructure, then it must also contains its own objects and relationships between the objects depending on the vantage point of that substructure. It will also carry out all the activities that the whole structure carries out.

- Main Activities: The focus is mostly on verbs, verbal phrases that describe main or important activities. The activities include;
    - Observer provides the objects to be used depending on the organization.
    - Observer determines the vantage point to use
    - Different vantage point shows filters to consider..
    - What the vantage point see of itself (Identity morphism/functions)
    - What other points see of the vantage point
    - What the vantage point see of the other points
    - What 2 or more points can see of the vantage point (Composite morphism/functions)
    - What the external standpoints sees of the vantage point
    - New vantage point changes the observations.
    - The outcomes of change obtained from the observations made are labeled.
    - Collection of activities is the relationships between the objects to be carried out.

b. Outline Description

An outline description of these main entities and activities [as they occur in the real world or where the final project is used].

This main entities can be seen as objects and the activities can be represented as arrows which will later be define as arrows. Basically in a real world situation the vantage point which is the reference point in the organization, will contain internal items or attributes. The vantage point will view the world then the world will see the vantage point. The vantage point will have to view itself.
An example can be a teacher and his students. The teacher can have items such as a wife/husband, children, first name and last name. The students will be the world in our case and each of them can have different attributes such as their characters in class, their performance in CA, their names, their family background, etc… So the teacher will view the students using the teacher-student relationship based on which attribute they decide to present to him in class or out of class if they meet. The students will eventually have a different view of the of the teacher using the student-teacher relationship based on what attribute the teacher will allow the students to know. So the teacher will use filters to allow only certain attributes of his to visible to the student in class. The teacher will also have a personal view about him/her self still depending on his/her attribute.

c. Requirements of the System

This will provide the description of what the system should do, the services that it provides and the constraints on its operations.

1) Functional Requirements:
   - Identify Nodes and Arcs: The user will do provide the following;
     - Input and display the objects
     - Give attributes that each object has if any
     - Determine the vantage point from one of the objects
     - Input the filters considered for the vantage point used
     - Describe substructure of the vantage point if any
     - Input external objects to the system depending on the vantage point
     - Calculate the relationships between the objects as functions (methods)
     - Change vantage point and subsequently change filters and functions
     - Label and display the outcomes obtained from the functions
     - Store the outcomes/data

- Node and Arcs (System) Requirements: The system must be able to carry out certain operations without focusing on the details of the objects (what the object is) and the description of the functions (the calculation done to implement the function) involved.
    - For the vantage point: From the vantage point determined by the user the following functions must be calculated;
        i. Function, f() what the vantage point sees of itself
        ii. Function, g() what the other objects see of the vantage point
        iii. Function, h() what the vantage point see of the other objects
        iv. Function, p o b( ) what 2 or more object can see of the vantage point
        v. Function, t() what the external standpoints sees of the vantage point

    - When the vantage point changes: The new vantage point determined by the user the following functions must be calculated;
        i. Function, f() what the vantage point sees of itself
        ii. Function, g() what the other objects see of the vantage point
        iii. Function, h() what the vantage point see of the other objects
        iv. Function, p o b( ) what 2 or more object can see of the vantage point
        v. Function, t() what the external standpoints sees of the vantage point

    - For the substructure of a vantage point: From the vantage point determined by the user the following functions must be calculated;
        i. Function, f( ) what the vantage point sees of itself
        ii. Function, g( ) what the other objects see of the vantage point
        iii. Function, h( ) what the vantage point see of the other objects
        iv. Function, p o b( ) what 2 or more object can see of the vantage point
        v. Function, t( ) what the external standpoints sees of the vantage point

2) Non-functional Requirement:

- Usability Requirements: The system shall allow the users to access the system with from their desktop. The system should be user friendly such that no prior training is needed.
- Availability Requirements: It should be available for different types of users at all times.

- Efficiency Requirements: Even if the system fails ,the system should be recovered back within an hour.
- Accuracy Requirements: The system should accurately provide real time relations depending on the situations the user chooses to address.
- Performance Requirements: The relationship are refreshed depending on the new vantage point the user chooses. The system shall display the new vantage points of the observer.
- Reliability Requirements: The system has to be reliable for the user to depend on it, in order to solve a particular situation.

3) System Requirement:

Client Platform: Client system must run all operating system of his/her computer.

Server Platform: The implementation is mainly abstract, no details required for the objects and relationships between these objects.

4) Hardware Requirement: Hardware is the backbone of any up and running software. The system demands a good hardware to support its functions and features.

For Client Platform:

- 256 MB of RAM
- 500 MHz CPU, 1 GHz preferred for 60 FPS velvet-smooth User Interface (UI)
- Python 2 and Python 3 can be used but this project uses Python 3
- Operating System: Both Linux and Windows can be used for both 64-bit and 32-bit machines.

5) Software Requirements: Here we focus on what the GUI we need to run. The aim is to produce an interactive software.

Tkinter module (“Tk interface”) is the standard Python interface to the Tk GUI toolkit from Scriptics formerly developed by Sun Labs. It is a GUI tool that runs on both Linux and windows operating systems. It is being used because of its canvas widgets which is typically used to display and edit graphs and other drawings.

The canvas widget is commonly used to implement various kinds of custom widgets. Therefore the arcs and nodes of the project were implemented using the canvas widget due to the graphics facilities it provides using tkinter.

2.5 Design

This section outlines and demonstrates the structure of the GUI and its sub-component. It further gives algorithm of the system. The design phase will be done by seeing most activities as operations. So each operation would be given in the design as an algorithm.

- Operation Name:

  The canvas widgets was used in this project. It provides structured graphics facilities for Tkinter. Its a highly versatile widget which can be used to draw graphs and plots, create graphic editors and various kinds of custom widgets.

  Label, Pen, Color, Eraser, and Save Buttons to permit the user to use well the software.

  ListBox to display the attributes of the canvas items on the screen.
- Input: Circles, arrows, and attributes represented on the screen.
- Output: A graph representation is the results of the output being displayed on the screen where the nodes are the circles and the arcs are the arrows. Attributes are also displayed on the screen.
- Post-condition: Here we give conditions that must be true if the operation successfully executes.

  For the buttons to work, define functions to be executed each time the buttons are clicked

  To display things on the canvas very well, use the create methods to add one or more canvas items which are placed in a stack. By default, new items are drawn on top of items already on the canvas.
- Algorithm: Here we are going to develop an algorithm for the operation, typically in English Language Prose.

from tkinter import *

define a class

create the various buttons that is, The pen, label, color, eraser and save buttons

define functions for the various buttons

create a canvas

create canvas items to be displayed on the canvas screen

create text to label the canvas item on the screen

create a listbox to be displayed on the canvas

make the canvas items to be active each time the mouse hover on it

enable the drag and drop functionality to displace the canvas items the canvas

define the data structure to hold the data displayed and save it using the save button

A snapshot of the structure of the Graphical User Interface is illustrate in figure 1

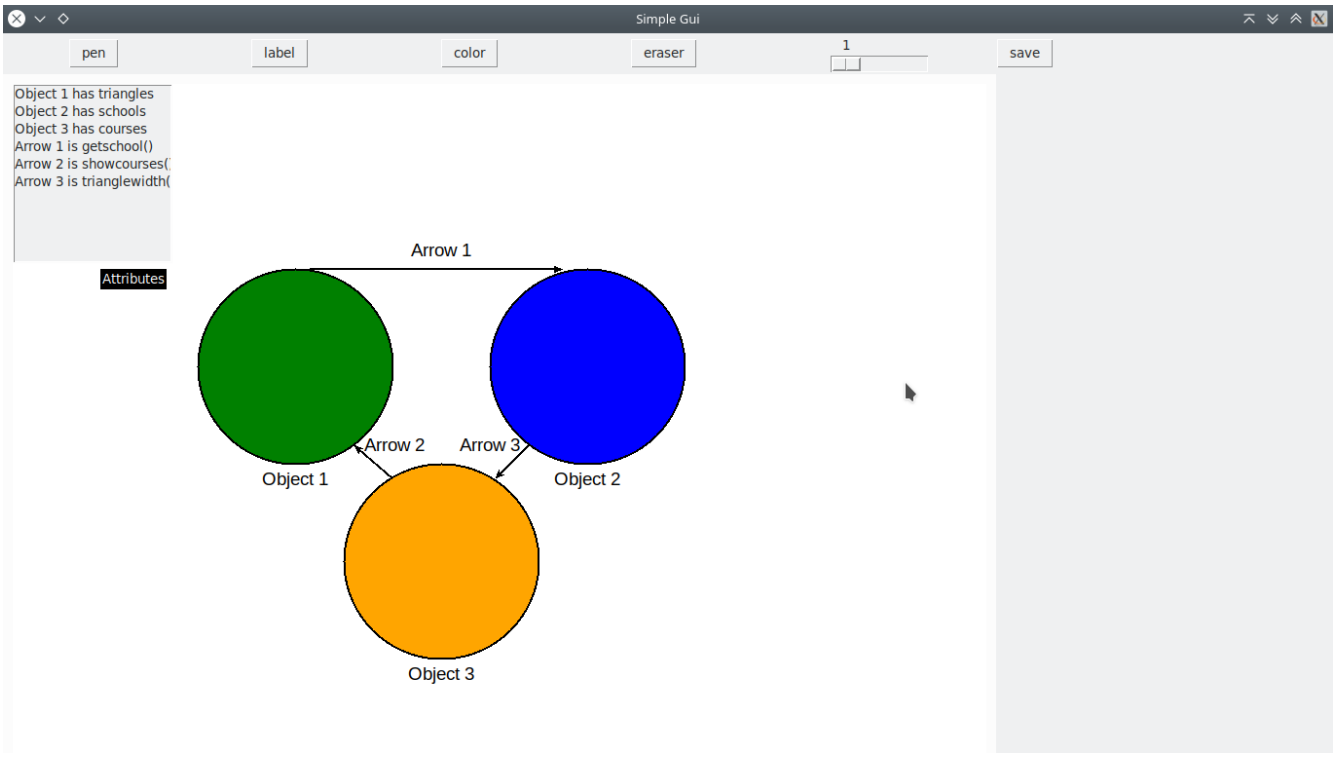


Figure 1: Snapshots of the GUI showing the canvas items and their various labeling

In designing the graphical user interface I followed a lot of steps before obtaining the final product as shown in figure 1. These steps are shown below;

- The picture below shows the objects placed on the screen;

List box for attributes.

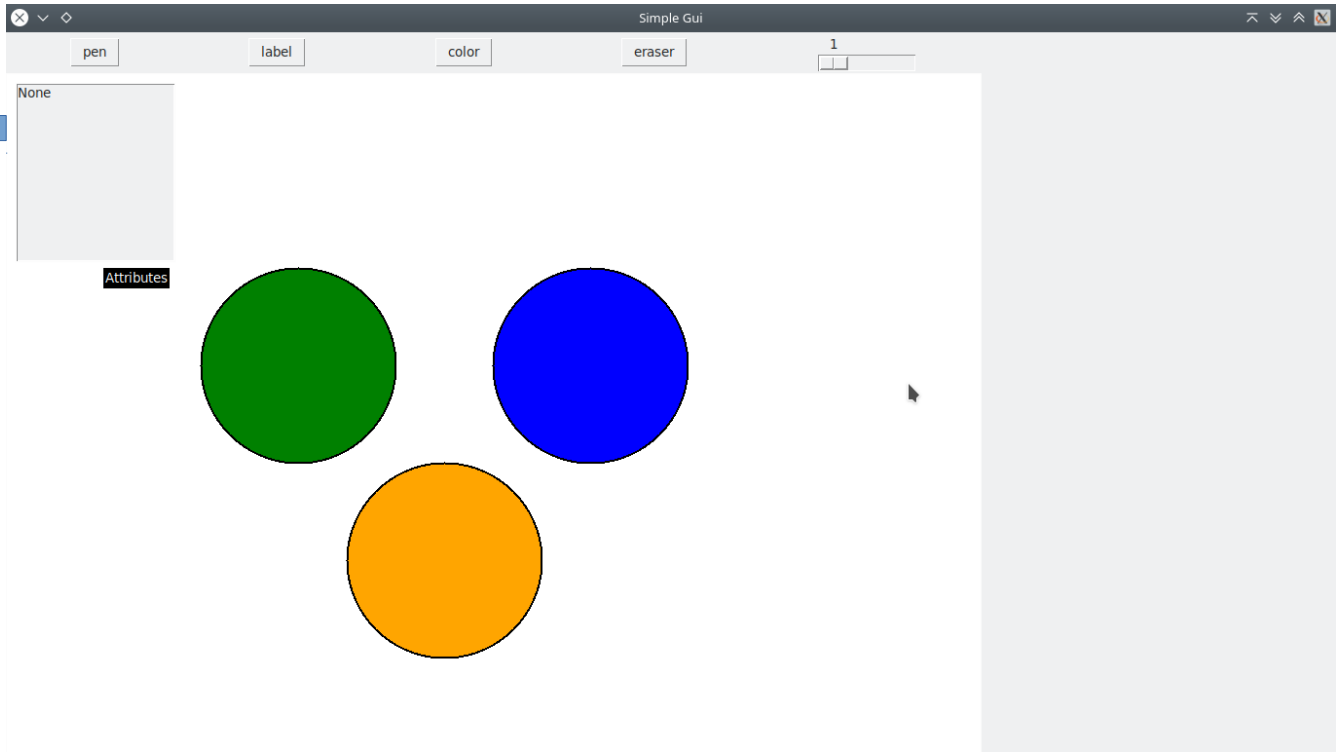


Figure 2: Objects Placed on the Screen

-The picture below show the lines connected to the objects which are circles

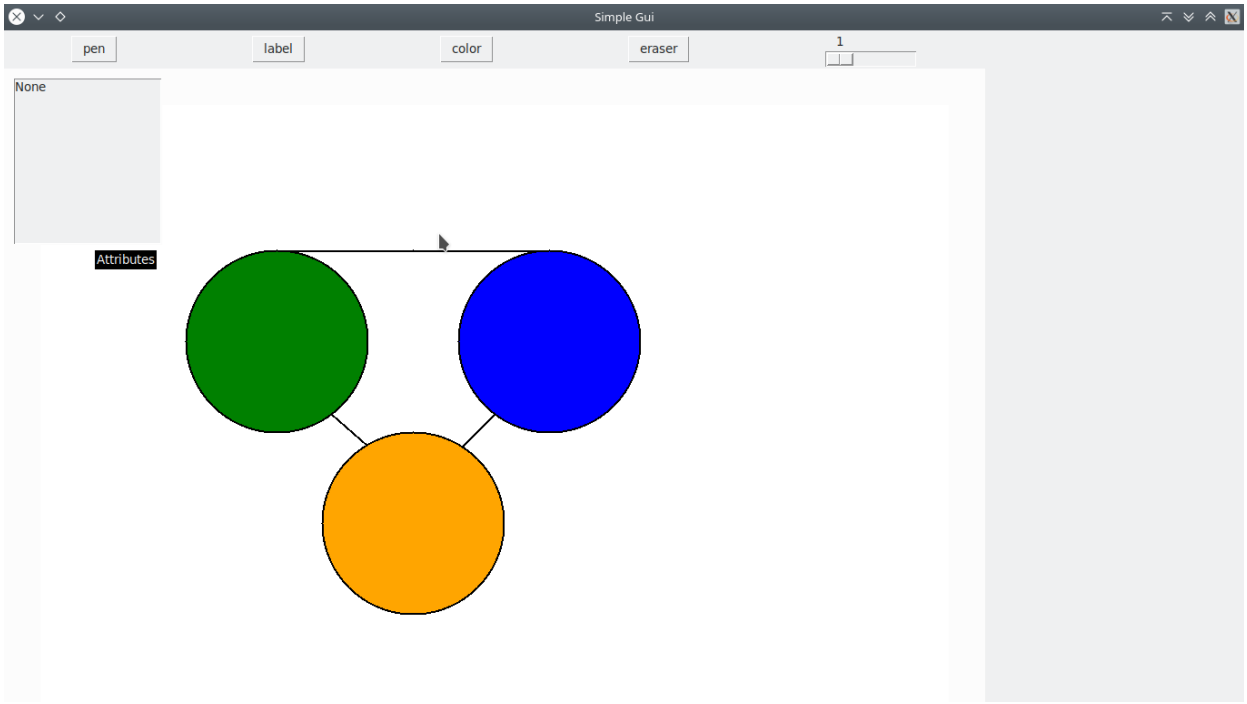


Figure 3: Showing the line connecting the objects.

- The figure below shows the label for each widgets placed on the canvas. The user can label each widgets using the label button. But the program labeled some of the widgets inside the code itself. The user can change the color of the text written with the label button using the color button on the screen.

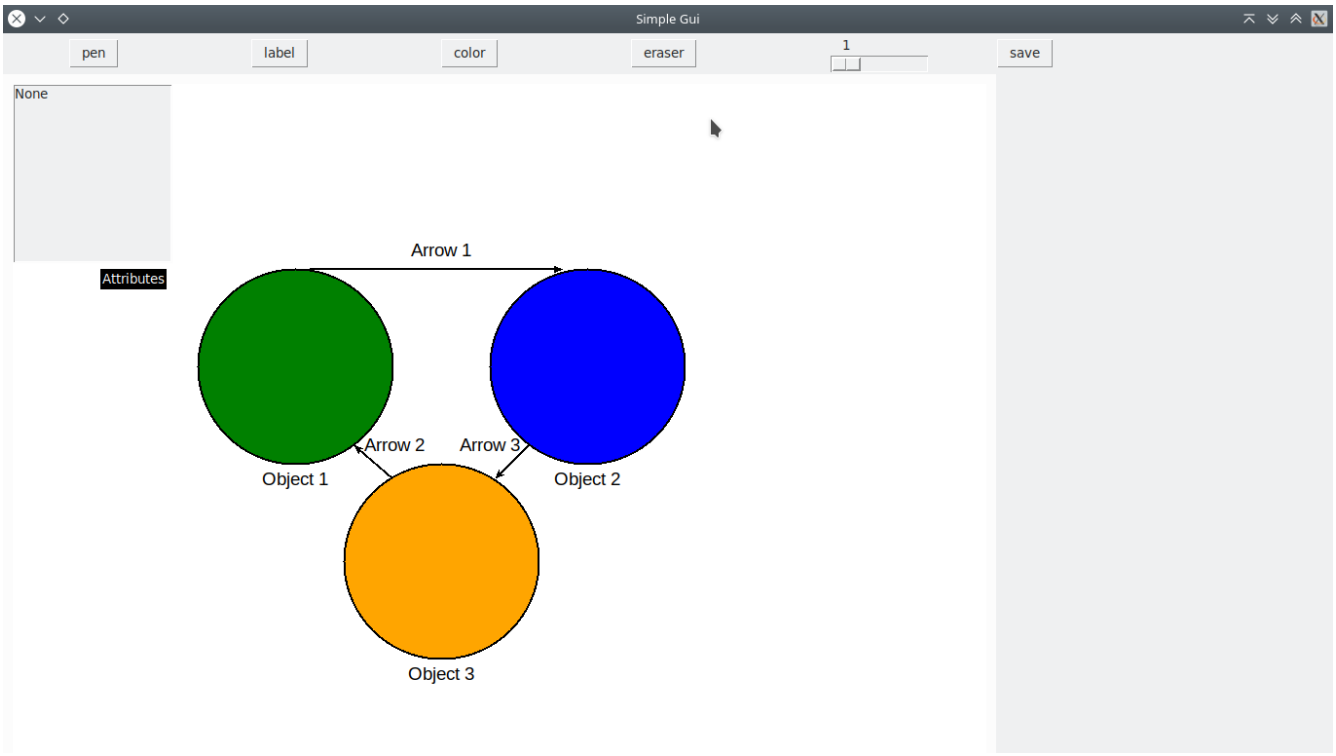
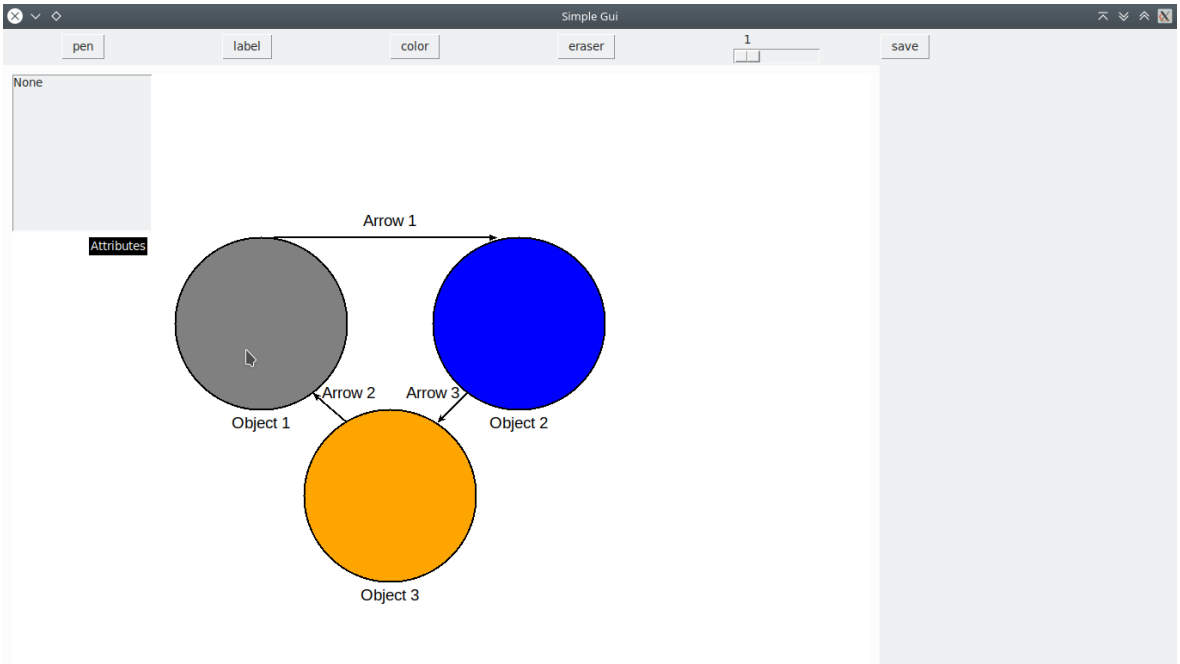


Figure 3: Labels of objects and lines.

-The figure below shows that Object 1 is highlighted and the color changed to gray, and this function applied to all the other objects and arrows. There is also a “save” button available for the user to save the canvas items as an image. The lines now have arrowheads as seen on the picture below.



CHAPTER 3

Results and Discussions

This chapter illustrate the results obtained from implementing and running the solutions gotten in the last chapter and discusses the results obtained. It also talks about some challenges encountered and how they were overcome.

3.1 Results

The software package was developed using tkinter module which is a standard Python interface to Tk GUI toolkit. It was developed under the Ubuntu 18 Linux environment but it also works well on windows. The software package is made up of python file along with a file to give details on how to run the tkinter interface. The full source code and the documentation is given in the software package which is inside a compressed file that is in a zip format.

The interface was implemented as follows: In order to run python program, PyCharm Community edition was installed first to be used as a python editor since tkinter is an inbuilt python module used to create GUI applications. Tkinter was also installed using the following command on the terminal but in some cases the tkinter module comes with python as default;

- ➔ **sudo apt-get install python3-tk**

Later on a python file was created with the extension .py, which could then make it possible for tkinter module to be imported and used in the software package. After all the installation done, the implementation was written and saved , then the code can be run properly.

The widgets used, the design done and all the implementation necessary for the project were coded in python using its inbuilt tkinter module and all the facilities and features it offers and provides. The canvas widgets was used as it provides graphics facilities and tkinter provides a lot of methods allowing you to manipulate the canvas items in various ways. Among other things, you can attach (bind) event callbacks to individual canvas items.

3.2 Discussions

In building the project and the software package, a lot of things happened, which are explained below.

Evaluation of the work

At the beginning of the project, finding a good graphical user interface tool was really a challenge. My first choices were Qt, PyQt, Javafx, Swing, Scenebuilder, Netbeans, for their drag and drop functionality, which permits you to design your project's widgets easily and very fast. But after studying well the tools I listed above, I realized that the time to learn the languages used in those tools will take all my time and at the end of the year I will not do any implementation. That's why I decided to used tkinter instead even though I had to code all the widgets and design myself since it has a much better facility feature required for my project which is the graphics facilities. So that's how I finally overcame my problem of GUI tool.

To implement the tkinter program at the beginning of the project, was really difficult in Ubuntu since I was doing everything from the python command line until I installed python editor "PyCharm" that could now easy my work and my understanding. Python programming language is very easy to understand and hence the more I could understand python, the more I could be effective in my project.

Representing the implementation at the end, showing the graph representations was quite easy after all the approach and comments I received from my supervisor.

The data structures and the drag and drop functionally did not implement as I expected. But the other project requirements and features were implemented very well.

CHAPTER 4

Conclusion

We have implemented a graph relabeling as per vantage points software that demonstrate a sample vantage point labeling and the observation of these points is determined by the observer which changes depending on the arcs between nodes and the vantage point.

Certain elements in this project leave areas for future development. I will highlight the general areas where the extra work would benefit the project.

- ➔ Recording and manipulating information recorded in the graph so that it always remains consistent, and reflected in all the arcs explicitly added. That is to record, and present on demand, what nodes see and how other nodes may see them, as well as relevant substructures and sub-activities within each node.
- ➔ Nodes without substructure are abstract entities whose details are not [yet] of interest.

REFERENCES

1) Justin Zobel, Writing for Computer Science, Third Edition, Springer, 2014.
2) William S, Shu, from csc 207 notes academic year 2016/2017
3) William S, Shu, from csc 208 notes academic year 2016/2017
4) Dr Denis Nkewteyim, from csc 306 notes academic year 2017/2018
5) Madam Wacka Ali, from csc 404 notes academic year 2018/2019
6) Dr Denis Nkewteyim, from csc 301 notes academic year 2017/2018

APPENDICES

The source code of the implementation phase is provided as a pdf document and a .py file. A text file is created to explain how the software can be executed easily.