National College of Ireland

Using the top-k frequent item set for mining non-overlapping patterns Configuration Manual

Shivam Gulve
Student ID: x20181400

School of Computing
National College of Ireland

Supervisor: Dr. Hicham Rafai

National College of Ireland
Project Submission Sheet
School of Computing

National College of Ireland

Student Name:	Shivam Gulve
Student ID:	x20181400
Programme:	Data Analytics
Year:	2022
Module:	MSc Research Project
Supervisor:	Dr. Hicham Rafai
Submission Due Date:	15/12/2022
Project Title:	Using the top-k frequent item set for mining non-overlapping patterns Configuration Manual
Word Count:	730
Page Count:	11

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Shivam Gulve
Date:	14th December 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Using the top-k frequent item set for mining non-overlapping patterns Configuration Manual

Shivam Gulve
x20181400

Developer Configuration Environment

1 Overview

The objective of the study is to use top k counts to find maximal sequential patterns that are non-overlapping. The project is created in Apache NetBeans 12.4, and Java 15.0.1 JDK 11 is used to implement the improved NMSP algorithm. A step-by-step process for running the program and creating the necessary environment will be provided in the setup manual Babich (1986).

2 Hardware Requirement

Processor Intel(R) Core(TM) i5-7700HQ CPU @ 2.80GHz 2.80 GHz
RAM 8 to 16 GB
Internal Storage 60 GB
System Type 64-bit-based processor
Operating System Windows 10 or 11

3 Prerequisite

The prerequisites that must be met in order to start this project are described in this section. Prerequisites could include downloading an IDE in order to run software, and this section will explain the type of environment that must be set up in order to execute an algorithm.

3.1 Setting up IDE

An integrated development environment (IDE) is a software program that assists inside the successful creation of software code by developers. By merging functions like software editing, building, testing, and packaging in a user-friendly program, it increases software quality.

3.1.1 Apache NetBeans Software

Java has such an integrated development environment named NetBeans. That used a collection of application software functional units called are modules, NetBeans allows the development of projects. Linux, Mac OS, Windows, and Solaris can accept NetBeans. For this project minimum requirement for IDE is Apache NetBeans version 12.4 and that can be download from this link `https://netbeans.apache.org/download/nb124/nb124.html`. The installation process is shown in fig 1 and fig 2

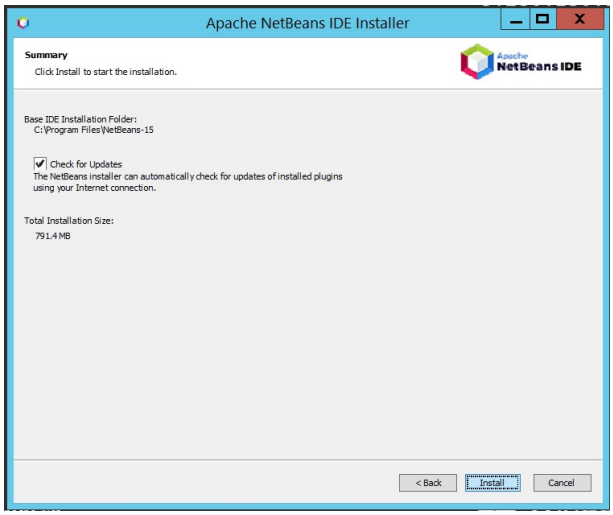

Figure 1: NetBeans Installation

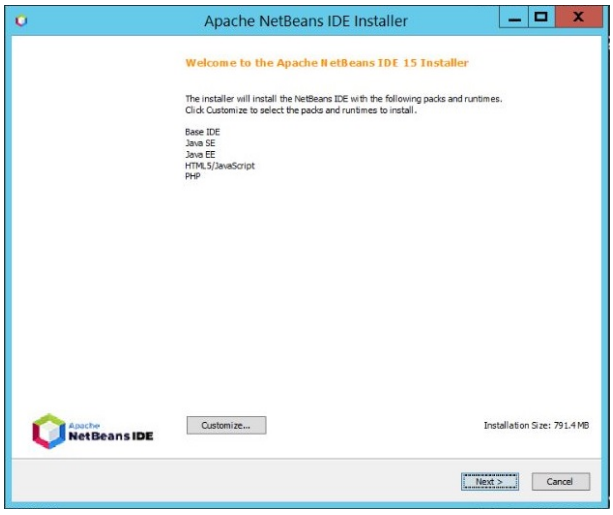

Figure 2: NetBeans Installation 2

3.1.2 Java JDK

Oracle provides Java Technology through the Java Development Kit. This offers the Standard Edition of a Java Interface for Application Programming and includes the Java Language Specifications and Java Virtual Machine Specifications. The minimum requirement for JDK is 32 bit processor. For this project minimum requirement for

JDK is JDK 11 and that can be download from here `https://www.oracle.com/ie/java/technologies/javase/jdk11-archive-downloads.html` The installation of JDK is shown in fig 3 and to check if jdk is installed or not we can use a command i.e., javac in command prompt show in fig 4

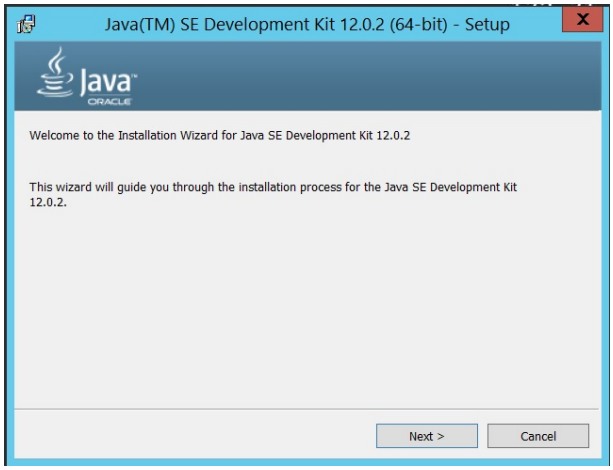

Figure 3: JDK Installation

Figure 4: JDK Installation Validation

3.1.3 Data Set

The data set we will be using for this project is about DNA, protein, and viral sequences, baby products and much more. All type of data set can be download from this link `https://github.com/shivamgulve4/research-project-data-set-/raw/main/DataSet.zip`

4 Project Configuration

After completing all the above section now is the time to configure the project. In this section we will be configuring the project i.e., how to setup the project for execution we will see here.

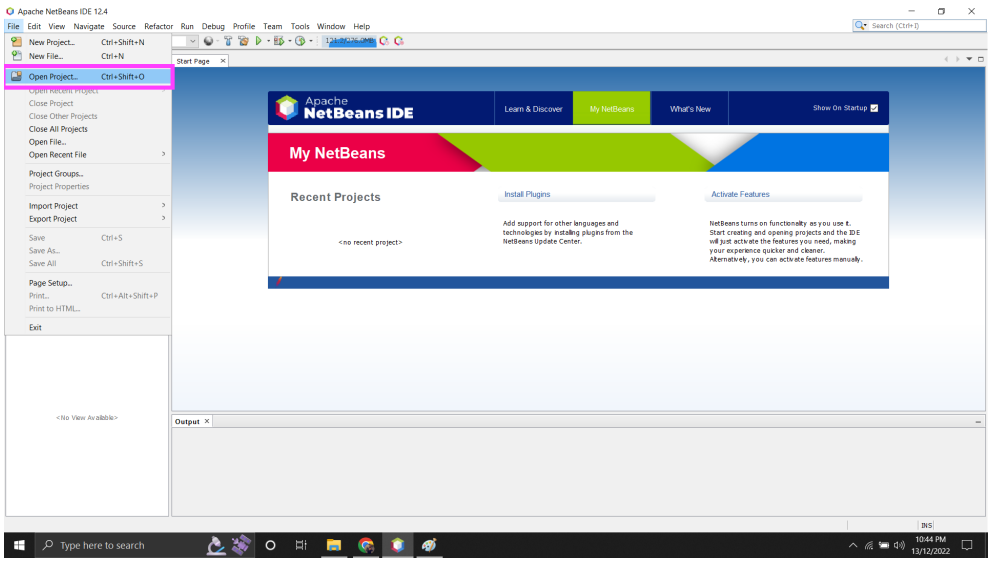

Figure 5: Opening Project

In above fig 5 shows how to locate and in next fig 6 it shows how can we open a project.

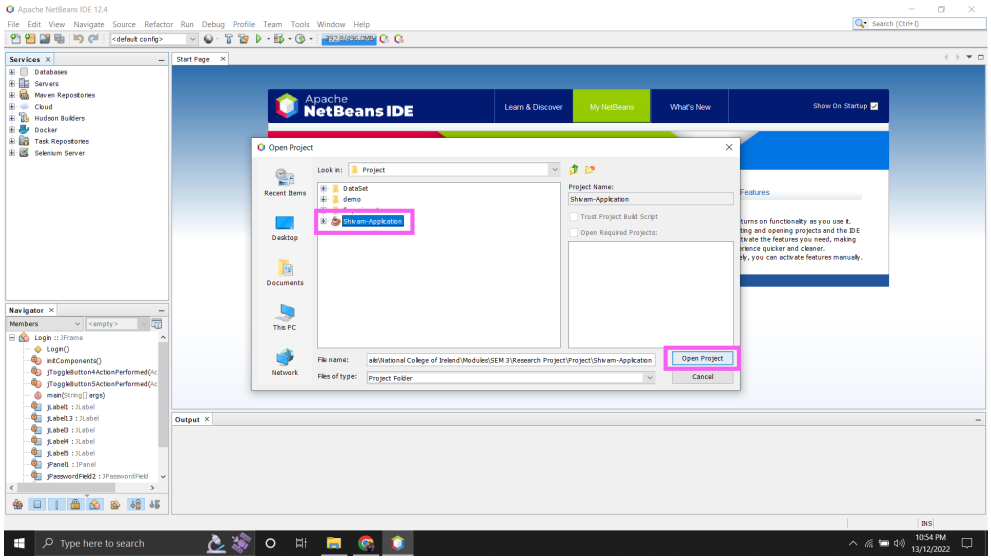

Figure 6: Locating Project

The next fig 7 shows which type of GUI are build and we can see that there are total five GUI build.

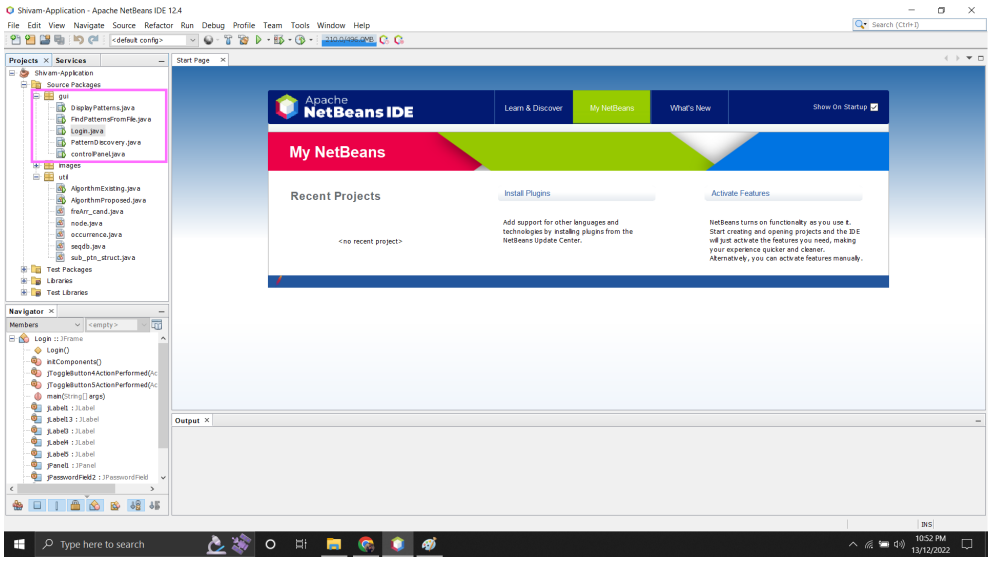

Figure 7: Implemented GUI's

So, how to run this page is shown in fig 8.

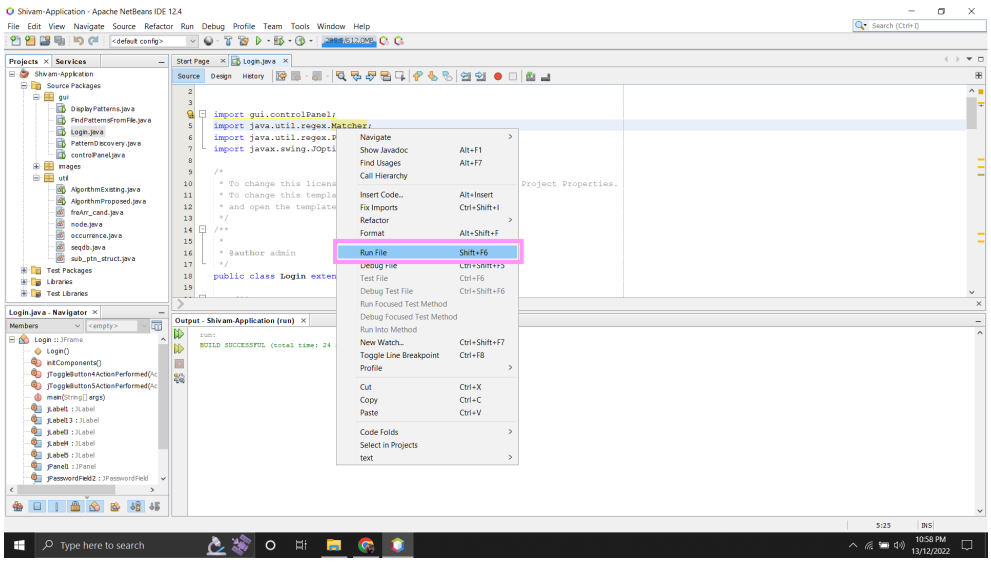

Figure 8: Execution

Execution Configuration Environment

The first one is Login page that consist of user authentication and for temporary the credentials are Username: ”admin” Password : ”123456”. shown in fig 9. For now there is no back-end to this so, the login data is static.

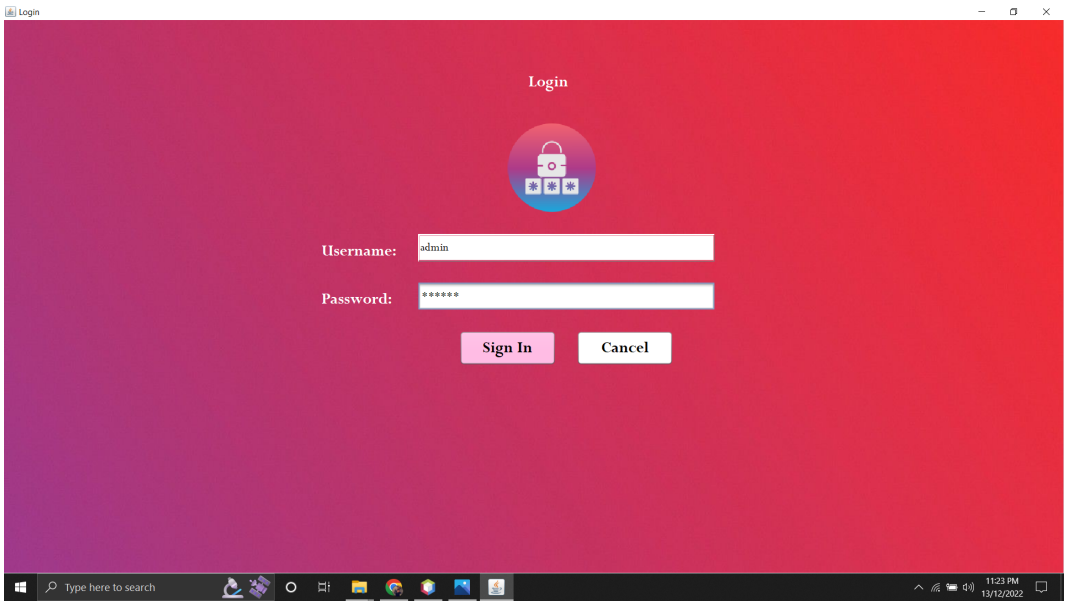

Figure 9: Login Page

After user have successfully login we will get on next page i.e., Control Panel. The control panel have three option shown in fig 10 and those are Pattern discovery from Data Set, Pattern Discovery from Input and Sign out.

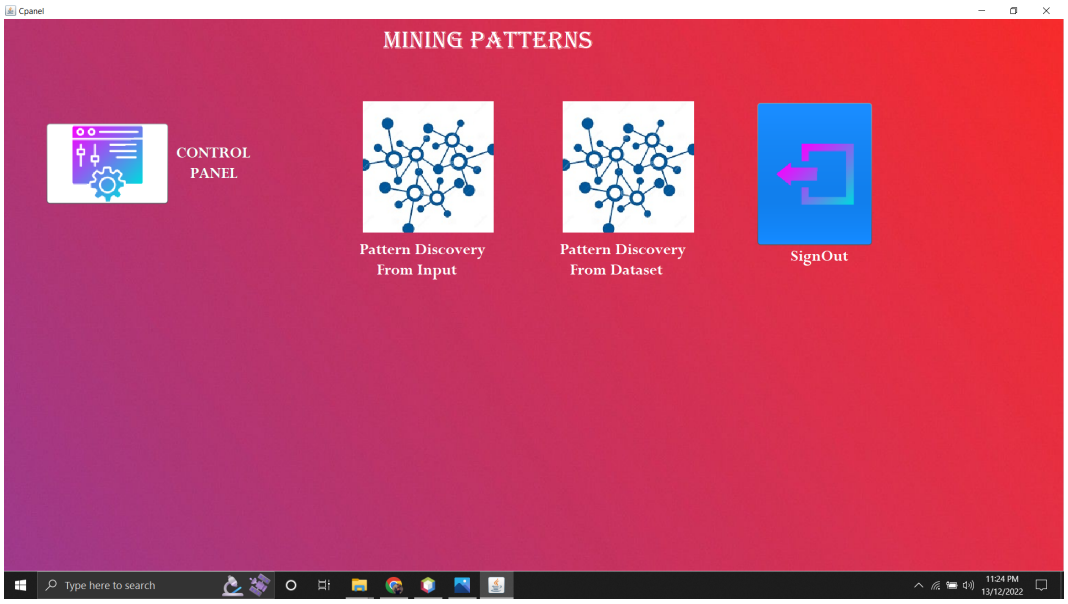

Figure 10: Control Panel

After select Pattern discovery from Data Set, it goes to next page 11. User have to browse the data set from our Local Drive 12 and then just provide all the details given below.

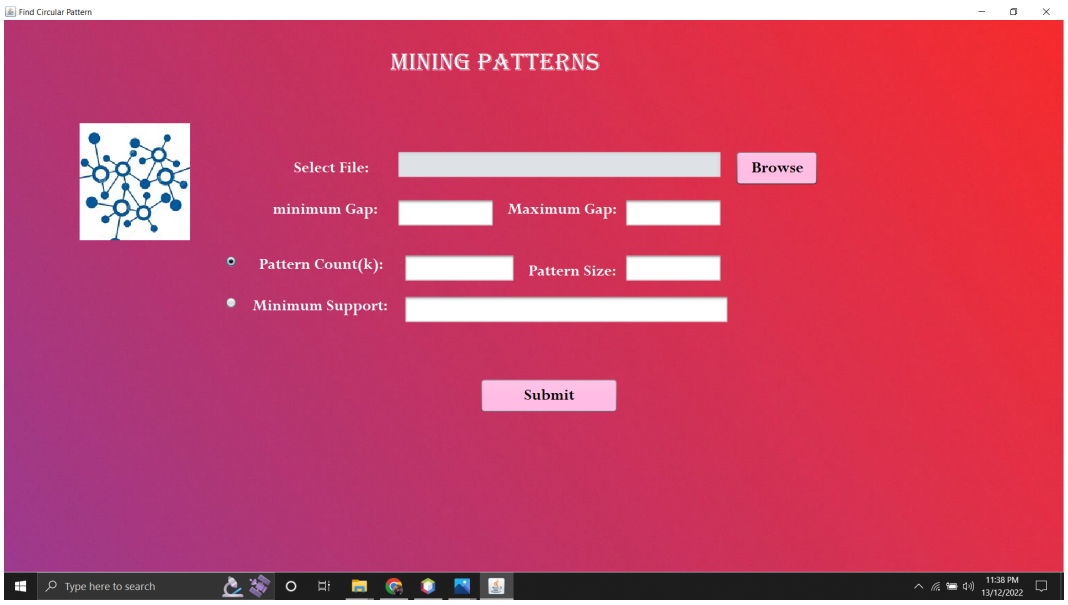

Figure 11: Pattern discovery from Data Set

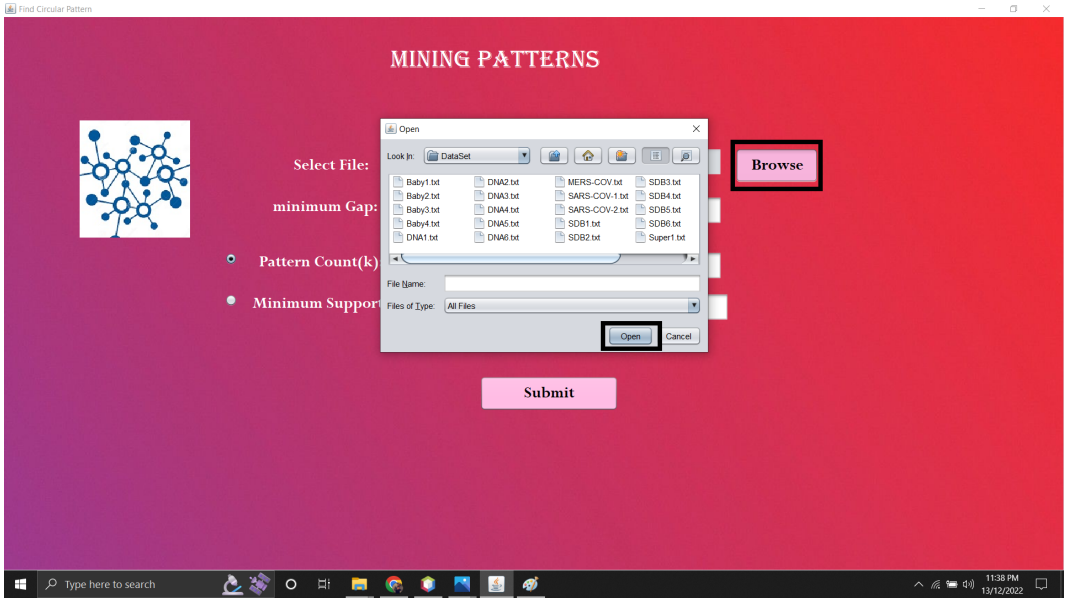

Figure 12: Browse Data Set

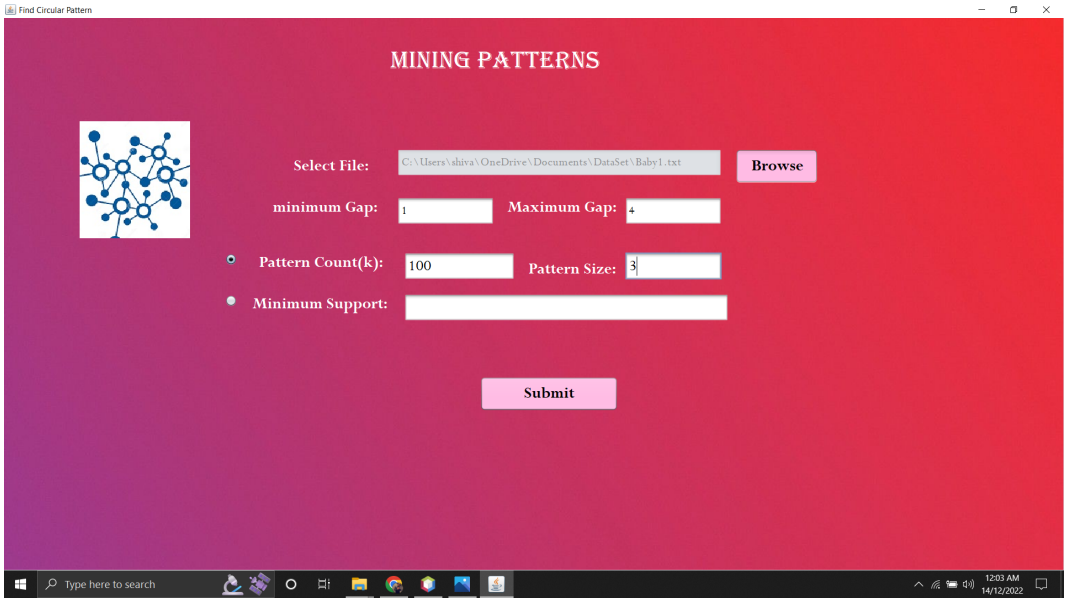

Figure 13: Filling Parameters

Now, just adding the parameters we will get NMSP patterns shown in fig 13 and when we submit we will get a processing window fig 14 and then after getting out patterns shown in fig 15

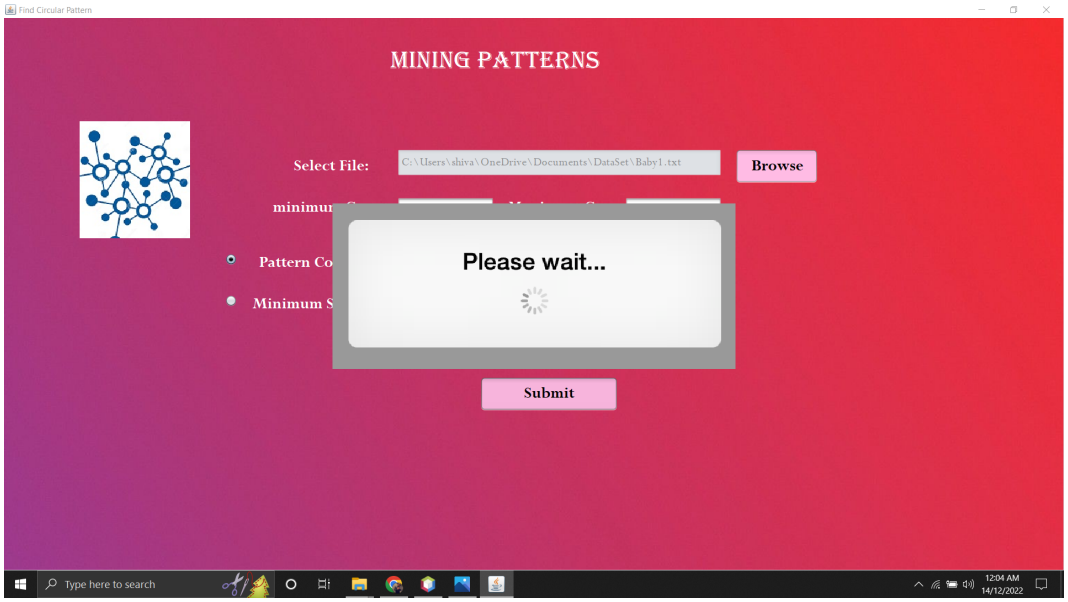

Figure 14: Waiting Window

we can also save our output and we can see how to do that in fig 16. It means we can save the output as a notepad

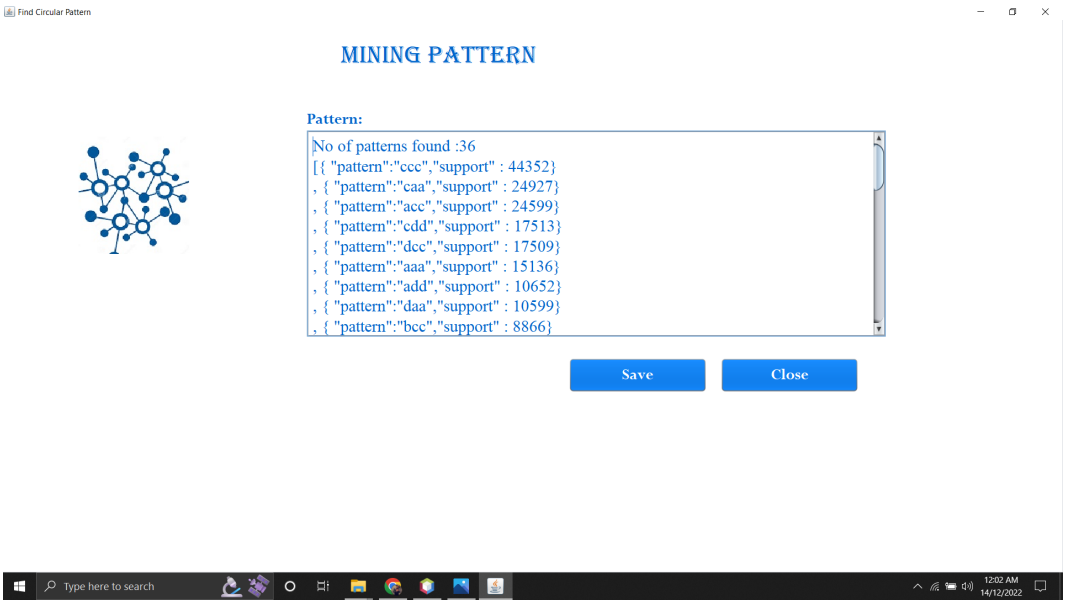

Figure 15: Pattern Output

So, if we want to compare our output we would be able to do that and just to get the preview of the last output we don't have to re-run the whole program again.

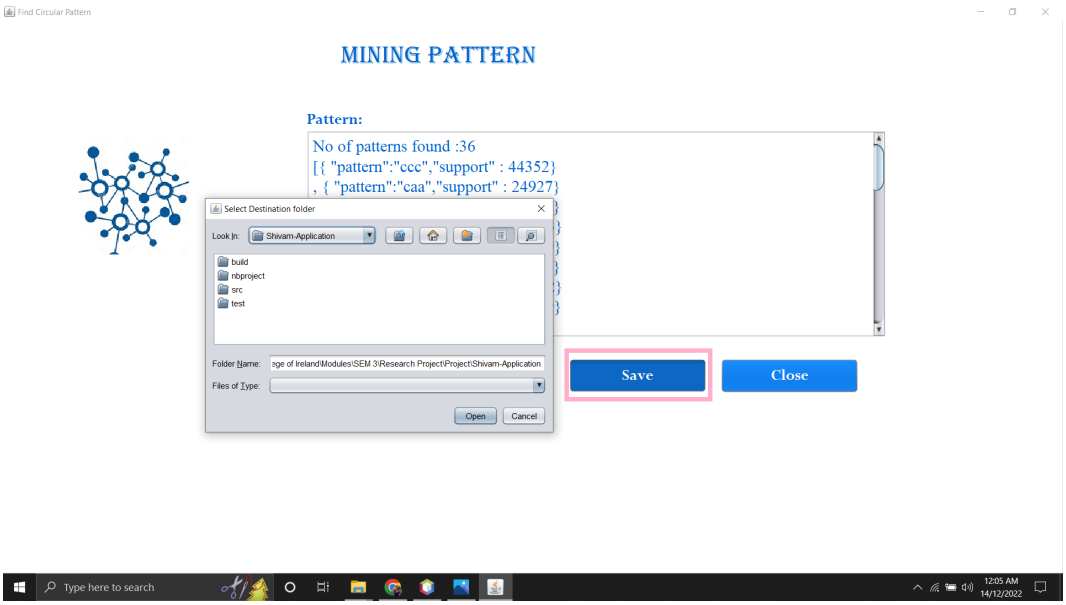

Figure 16: Save Output

we can also check different experiments cast study i.e., processing time, memory used by program shown in fig 17

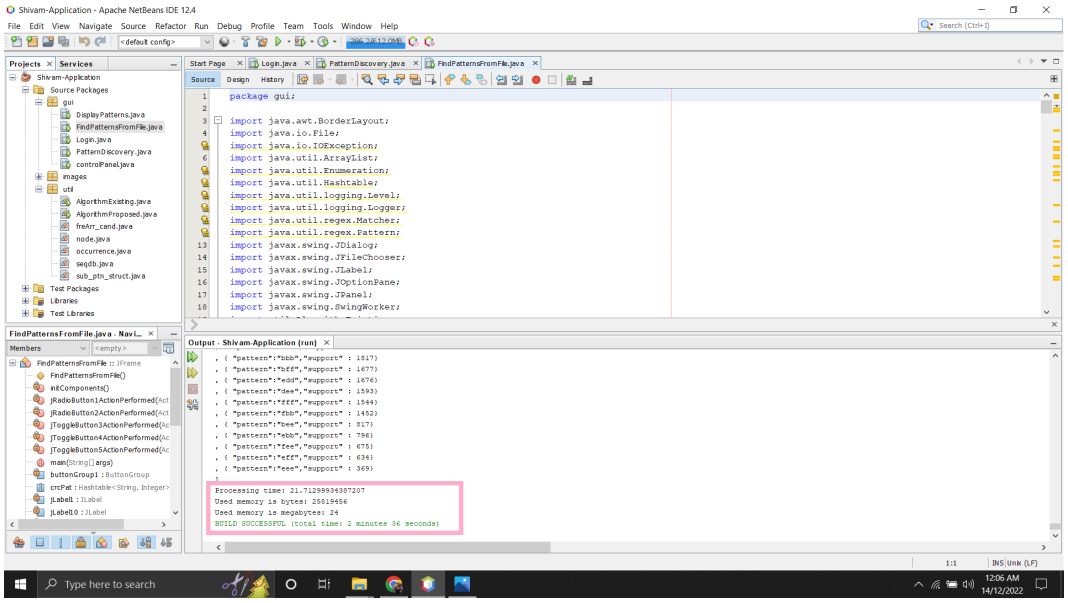

Figure 17: Experiments

Now, if we select another option i.e., Pattern Discovery from Input 18. In this we have to provide the pattern as an input and in remaining parameters we have to give same details as we gave in fig 13 So, after giving the input we get the output as shown in fig 19

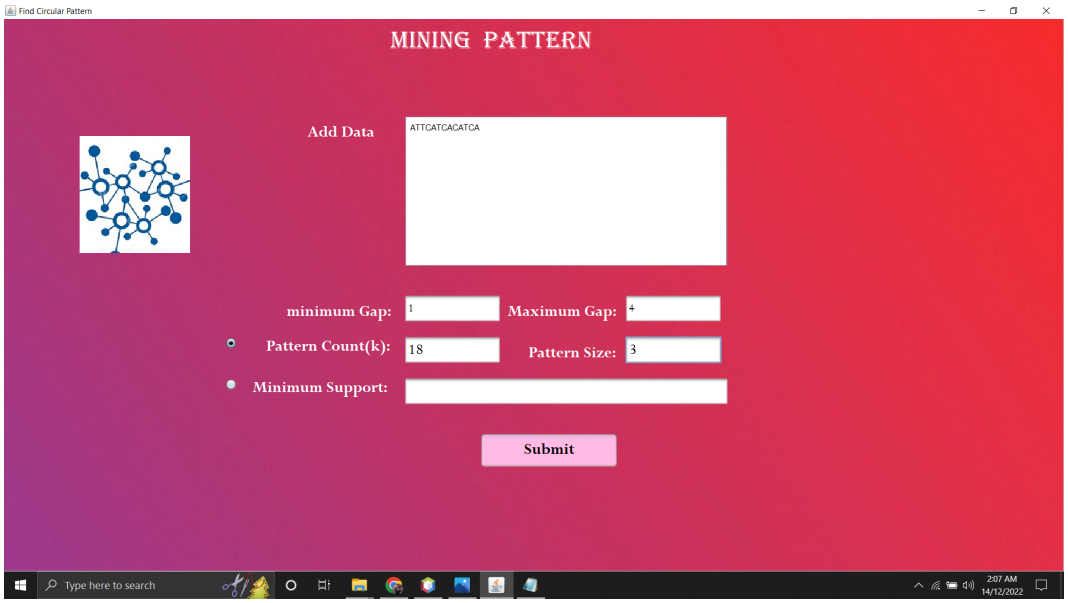

Figure 18: Pattern Discovery from Input

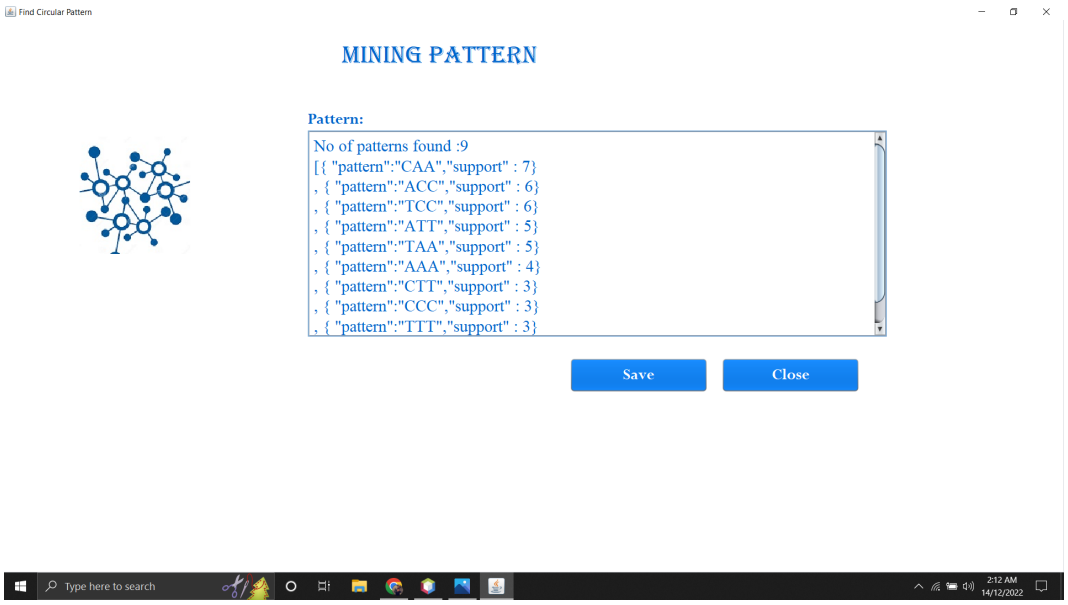

Figure 19: Pattern Discovery Output

To check the experiments like what was the processing time and memory used can be seen in fig 20

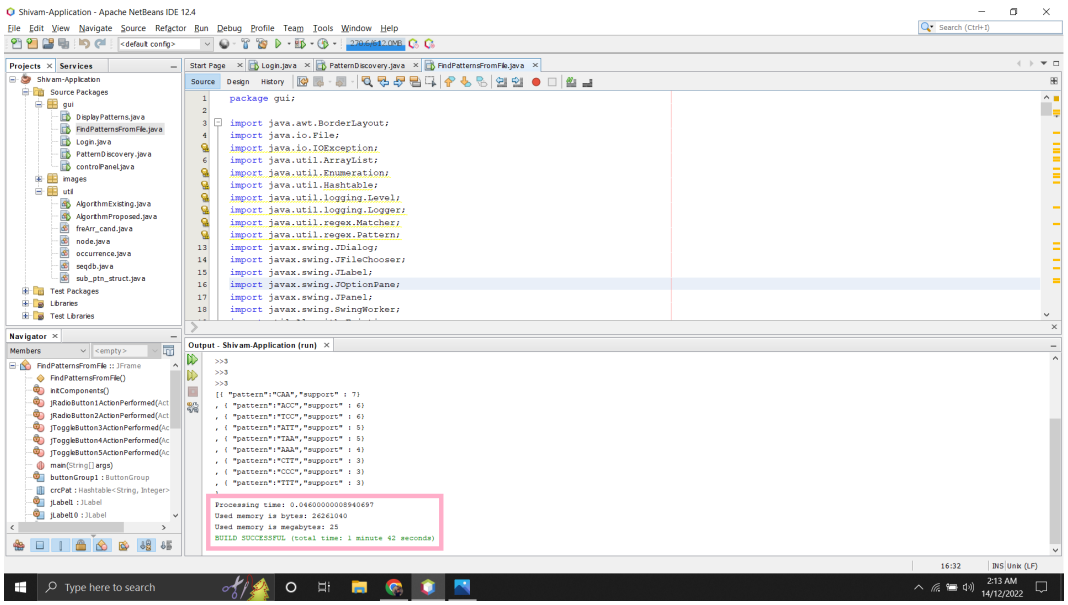

Figure 20: Pattern Discovery Experiments

References

Babich, W. A. (1986). *Software configuration management: coordination for team productivity*, Addison-Wesley Longman Publishing Co., Inc.