

---

# **Software Requirements Specification**

for

# **Gephi**

**Version 0.92**

**Prepared by Konstantinos Varvoutas**

**Aristotle University of Thessaloniki**

**February 2017**

# Table of Contents

<b>Table of Contents</b> .....	<b>ii</b>
<b>Revision History</b> .....	<b>iii</b>
<b>1. Introduction</b> .....	<b>1</b>
1.1 Purpose .....	1
1.2 Document Conventions .....	1
1.3 Intended Audience and Reading Suggestions .....	1
1.4 Product Scope .....	1
1.5 References .....	2
<b>2. Overall Description</b> .....	<b>3</b>
2.1 Product Perspective .....	3
2.2 Product Functions .....	3
2.3 User Classes and Characteristics .....	7
2.4 Operating Environment .....	8
2.5 Design and Implementation Constraints .....	8
2.6 User Documentation .....	8
2.7 Assumptions and Dependencies .....	9
<b>3. External Interface Requirements</b> .....	<b>10</b>
3.1 User Interfaces .....	10
3.2 Hardware Interfaces .....	18
3.3 Software Interfaces .....	19
3.4 Communications Interfaces .....	19
<b>4. System Features</b> .....	<b>19</b>
4.1 Graph Visualization .....	19
4.2 Graph Layout .....	22
4.3 Graph Metrics .....	23
4.4 Filters .....	24
4.5 Data Table .....	25
4.6 Dynamic Graphs .....	28
4.7 Graph Export .....	29

**5. Other Nonfunctional Requirements ..... 30**

5.1 Performance Requirements .....30

5.2 Safety Requirements .....30

5.3 Security Requirements .....30

5.4 Software Quality Attributes .....30

**Glossary..... 31**

## Revision History

<b>Name</b>	<b>Date</b>	<b>Version</b>
Gephi	8 February 2010	0.70 Alpha
Gephi	17 February 2010	0.70 Alpha2
Gephi	1 April 2010	0.70 Alpha3

Gephi	30 April 2010	0.70 Alpha4
Gephi	1 October 2010	0.70
Gephi	8 April 2011	0.80 Alpha
Gephi	4 October 2011	0.80
Gephi	31 March 2012	0.81
Gephi	3 Jan 2013	0.82
Gephi	20 December 2015	0.90
Gephi	14 February 2016	0.91

# **1. Introduction**

## **1.1 Purpose**

The purpose of this document is to present a detailed description of the open-source software Gephi. It will explain the purpose and features of the software, the interfaces of the software, what the software will do and the constraints under which it must operate. This document is intended for users of the software and also potential developers.

## **1.2 Document Conventions**

This Document was created based on the IEEE template for System Requirement Specification Documents.

## **1.3 Intended Audience and Reading Suggestions**

- Typical Users, such as students, who want to use Gephi for analyzing networks (Social networks, Social Media networks, Semantic networks etc.)
- Advanced/Professional Users, such as engineers or researchers, who want to use Gephi for more demanding graph analysis.
- Programmers who are interested in working on the project by further developing it or fix existing bugs.

## **1.4 Product Scope**

Gephi is a tool that people can use to analyze graphs and networks. Users can use it to represent their data as a graph and interact with them. They can alter the layout of the graph or calculate metrics of it, to understand their data in a better way and discover hidden aspects about it, such as patterns or faults during data sourcing.

This is a software for [Exploratory Data Analysis](#), a paradigm appeared in the [Visual Analytics](#) field of research.

## **1.5 References**

Gephi's website:

<https://gephi.org/>

Gephi's GitHub page:

<https://github.com/gephi>

IEEE Template for System Requirement Specification Documents:

<https://goo.gl/nsUFwy>

GNU General Public License version 3:

<http://www.gnu.org/licenses/gpl.html>

CDDL Common Development and Distribution License:

<https://opensource.org/licenses/CDDL-1.0>

## **2. Overall Description**

### **2.1 Product Perspective**

Gephi was developed for everyone who is interested in graphs and wants either to just experiment with them so that he can understand them or wants to use them as a means of analyzing data. It can handle various graph formats (GEXF, GDF, Pajek, DOT, GraphML, UCINET, CSV...) and supports directed, undirected and mixed graphs.

It is an open source project and it has a very active developer team to support it and provide feedback to users. It was developed to run on Windows, Mac OS X and Linux.

### **2.2 Product Functions**

File:

- New Project: Creation of a new project
- Open: Loads an existing project or a file of one of the supported graph formats
- Open Recent: Loads one of the displayed, recently opened files.
- Close Project: Closes the currently open project.
- Properties: Displays some properties of the project (such as the title) which can be edited.
- Import Spreadsheet: Loads a CSV file (edges table or nodes table).
- Import Database -> Edges List: Loads the edges of a graph from a Database (MySQL, SQLite, PostgreSQL)
- Generate -> Dynamic Graph Example: Creates a Dynamic Graph from scratch.
- Generate -> Multi-Graph Example: Creates a Multi-Graph from scratch.
- Generate -> Random Graph: Creates a Random type of graph from scratch.
- Save: Saves the project without changing its name or directory.

- Save as: Saves the project and gives the user the ability to change its name or directory.
- Export -> Graph file: Exports the current project's graph in one of the supported file formats
- Export -> SVG/PNG/PDF file: Exports a screenshot of the current project's graph in one of the available formats (SVG, PNG or PDF).
- Exit: Gephi shuts down

Workspace:

- New: Creates a new workspace.
- Delete: Deletes the current workspace.
- Rename: Renames the current workspace

Tools:

- Plugins: Here the user can change his plugin settings (update already installed plugins, install new plugins etc.).
- Options: Here the user can change some of Gephi's settings (key bindings etc.).
- Language: Changes the language to one of the languages displayed.
- Online docs and support: Redirects the user to <https://gephi.org/users/support/>

Window:

- Context
- Data Table: Displays the Data Table tab.
- Filters: Displays the Filters tab.
- Graph: Displays the Graph tab.
- Layout: Displays the Layout tab.
- Output: Displays the Output-Log tab.
- Preview: Displays the Preview tab.
- Preview Settings: Displays the Preview Settings tab.
- Appearance: Displays the Appearance tab.

- Welcome: Displays the Welcome window.
- Statistics: Displays the statistics tab.
- Timeline: Displays the timeline tab.
- Configure Window: Here the user can select one of the available display options for the currently selected tab.
- Close Window: Closes the currently selected tab
- Document Group -> New Document Group: Creates a new Document Group.
- Document Group -> Manage: Selects or removes one of the existing Document Groups.

#### Help

- Check for Updates: Displays the plugins that can be updated to newer versions
- About: Displays the logo of Gephi, which licenses are being used, the product version and other info.

#### Main Pages:

- Overview: Displays the Overview page, which by default includes the tabs: Appearance, Layout, Graph, Context, Filters, Statistics and Queries.
- Data Laboratory: Displays the Data Laboratory page, which by default includes the tab Data Table.
- Preview: Displays the Preview page, which by default includes the tabs: Preview and Preview Settings.

#### Overview:

- Appearance: Here the user can change the color of the edges/nodes. He can choose one color for all the edges/nodes or choose multiple colors, according to the values of a certain attribute.
- Layout: Here the user can change the layout of the graph by using one of the available layouts.

- Graph: Here the user can interact directly with the graph itself, by dragging nodes, zooming in or out etc.
- Context: Displays the number of nodes and edges and the type of the graph.
- Filters: Here the user can choose which filters to apply on the graph. (the filters are being applied by drag and drop on to the “Queries” tab).
- Statistics: Here the user can choose which of the graph statistics will be calculated (by pressing on the button “Run” of the desired statistic).
- Queries: This is the module where filters are being placed.

#### Data Laboratory:

- Nodes: Displays the Nodes table
- Edges: Displays the Edges table
- Configuration: Preferences about how the data is presented.
- Add node: Creates a new node and adds it to the graph.
- Add edge: Creates a new edge between two existing nodes of the graph
- Search/Replace: Stand Search/Replace functionality.
- Import Spreadsheet: Opens a dialog to load a csv file (edges or nodes table)
- Export Table: Exports the table as a csv file, the user can choose which attributes will be included in the file.
- More Actions -> Clear Graph: All nodes and edges of the graph will be deleted.
- More Actions -> Clear Edges: All edges of the graph will be deleted.
- More Actions -> Detect and merge node duplicates: Removes node duplicates by merging them into one node.
- More Actions -> Manage Dynamic Column Estimators:
- Filter: Used to find all the rows of the table that contain a specific value at an attribute selected by the user.
- Add column: Adds a new attribute to the table (edges/nodes)
- Merge columns: Combines the selected columns into one. The merge strategy must be selected by the user.

- Delete column: Removes all the values of a column and the column itself from the table.
- Clear column: For every row sets the selected column empty.
- Copy data to other column: For every row the value of the target column becomes identical to the value of the source column.
- Fill column with a value: For every row the value of the selected column is set to the specified one.
- Duplicate a column: Creates a duplicate of the selected column.
- Create a boolean column from regex match: Creates a new boolean column, the value of which depends on whether the value of the selected existing column matches a regular expression or not. The regular expression must be specified by the user.
- Create column with list of regex matching groups: Creates a new column. For every row the value of this column is a list about how the value of the selected existing column matches with the user's regular expression.
- Convert column to dynamic: Converts the selected static column into a dynamic one.

#### Preview

- Preview Settings: Here the user can configure rendering settings.
- Preview Ratio: When set under 100% a partial graph is displayed.
- Refresh: Applies changes made by the user.
- Preview: After changes have been applied, the updated graph will be shown here.

### **2.3 User Classes and Characteristics**

- Typical Users, such as students, who want to use Gephi for analyzing networks (Social networks, Social Media networks, Semantic networks etc.)

- Advanced/Professional Users, such as engineers or researchers, who want to use Gephi for more demanding graph analysis.
- Programmers who are interested in working on the project by further developing it or fix existing bugs

## **2.4 Operating Environment**

- Windows 2000
- Windows XP
- Windows Vista
- Windows 7
- Windows 8
- Windows 10
- Mac OS X
- Linux

## **2.5 Design and Implementation Constraints**

Gephi is developed in Java, it uses OpenGL for its visualization engine and has been built on top of the NetBeans Platform. It uses a modular design where every feature is wrapped into a separate module and the modules depend on each other through well-written APIs. There are several APIs available to make plugin development easy.

## **2.6 User Documentation**

There is a quick start guide available on the website of Gephi:

<https://gephi.org/users/quick-start/>

There are two other official tutorials, one for visualization:

<https://gephi.org/users/tutorial-visualization/>

And one for layouts:

<https://gephi.org/users/tutorial-layouts/>

In this page: <https://gephi.org/users/> users can find every available tutorial including tutorials made by the community, video tutorials, the official tutorials mentioned above, non-English tutorials etc.

Additional help and information can be found at Gephi's wiki:

<https://github.com/gephi/gephi/wiki>

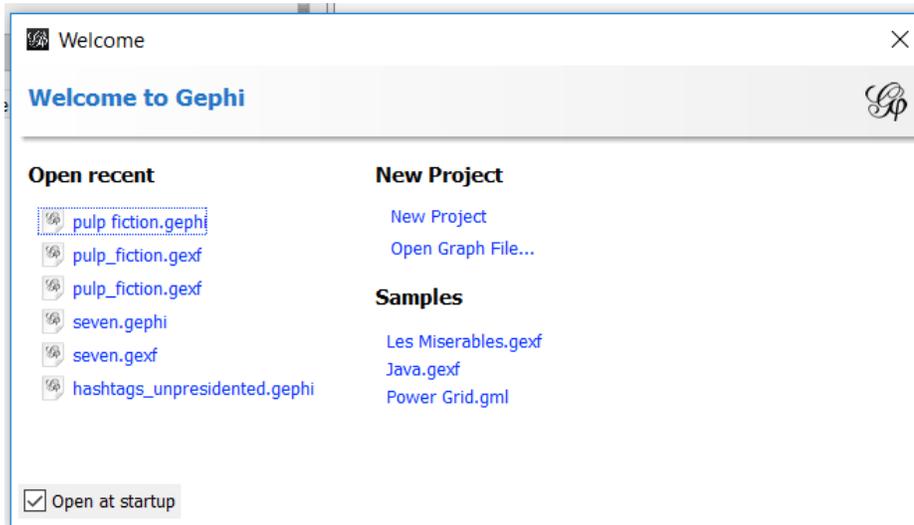
## **2.7 Assumptions and Dependencies**

Gephi is developed in Java and therefore requires Java to be installed on the user's system. The latest stable version of Gephi requires Java version 7 or higher. This applies to Windows and Linux users. On Mac OS X, Java is bundles with the application.

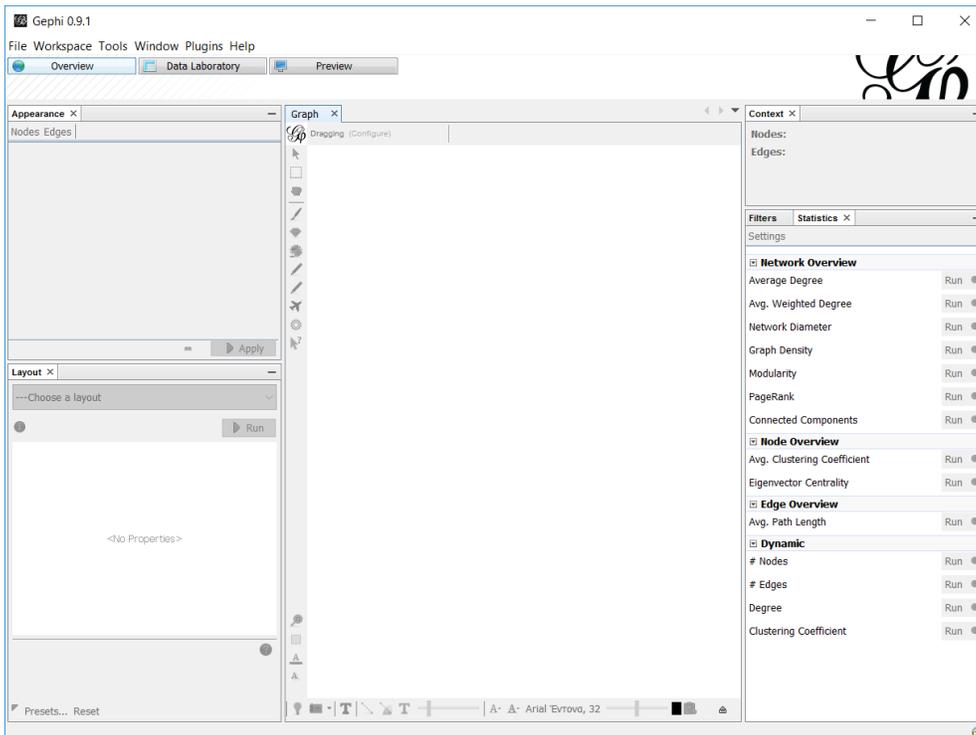
## 3. External Interface Requirements

### 3.1 User Interfaces

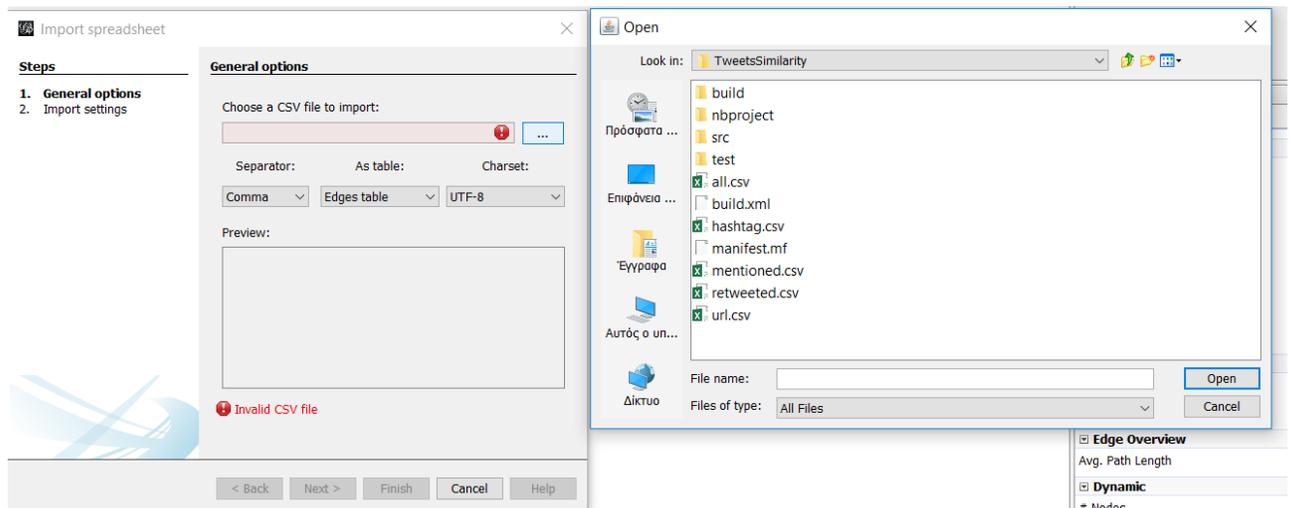
#### 1. Gephi's Welcome Screen:



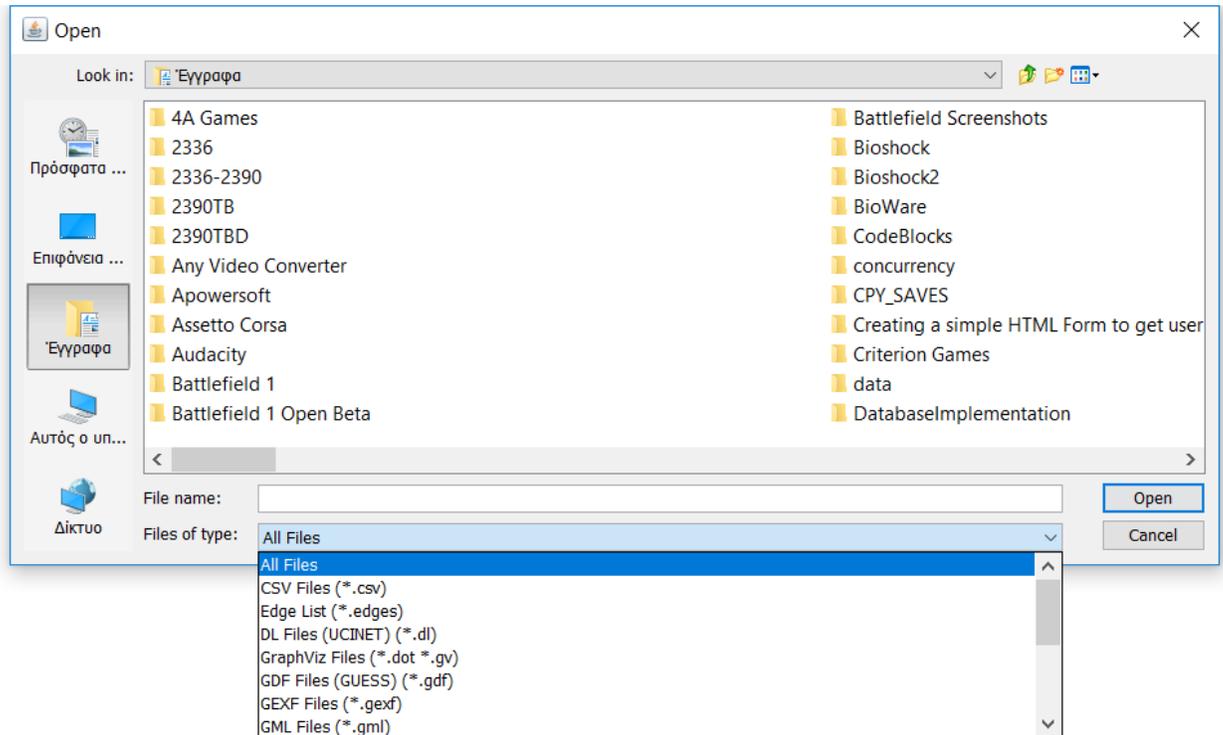
#### 2. Gephi's Main Screen:



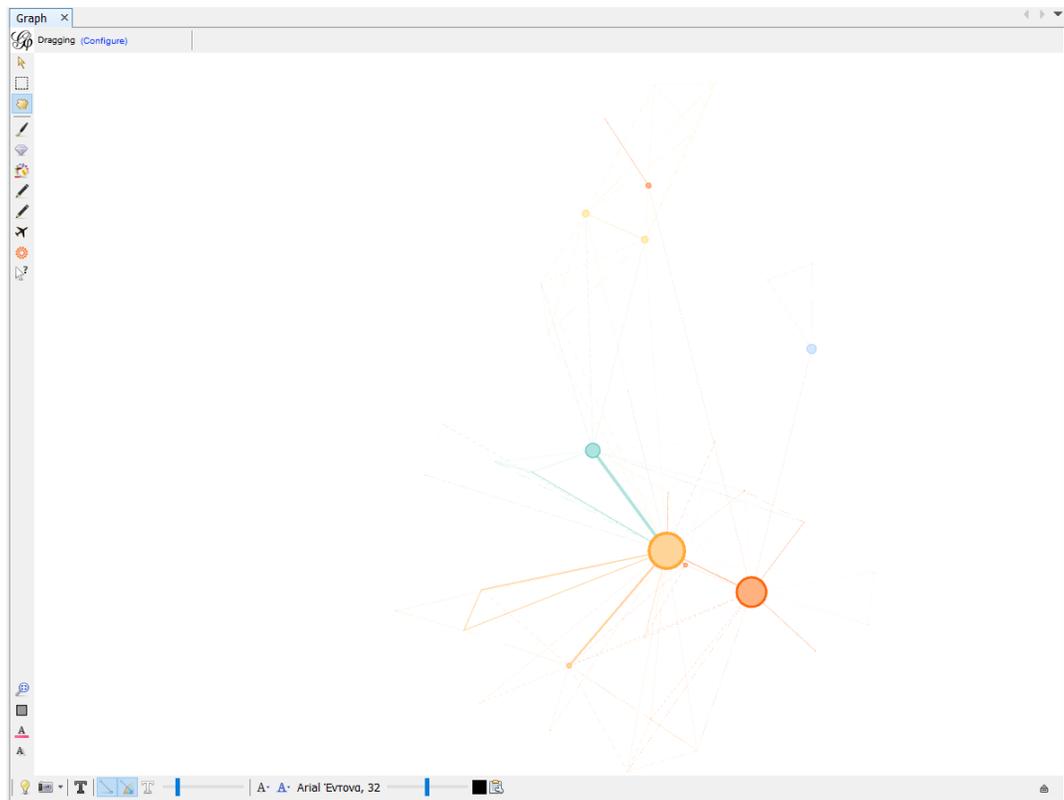
### 3. Import Graph (CSV file)



### 4. Open Graph (all supported file formats)



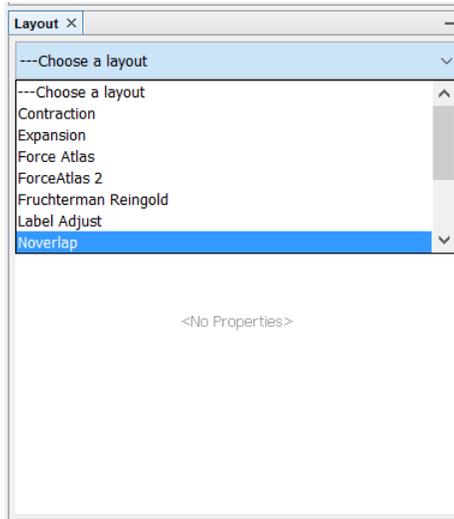
- Overview of the Graph. Left side buttons, from top to bottom: direct selection (select only one node/edge), rectangle selection (select part of the graph), Drag (drag a node using the mouse), Painter (color nodes by clicking on them), Sizer (change size of nodes by clicking on them and dragging the mouse vertically), Brush (colors the selected node and its nearest neighbour), Node Pencil (create new node where mouse is clicked), Edge Pencil (create new edge by clicking first on source node and then on target node), Shortest Path (display the shortest path between two clicked nodes), Heat Map (Set color intensity on a node neighbourhood, based on the distance from the clicked node), Edit (Click on a node to edit its attributes), Center on Graph, Reset Colors (remove color from every edge and node), Reset Label Colors( remove color from every label), Reset Label Visible( make labels invisible).



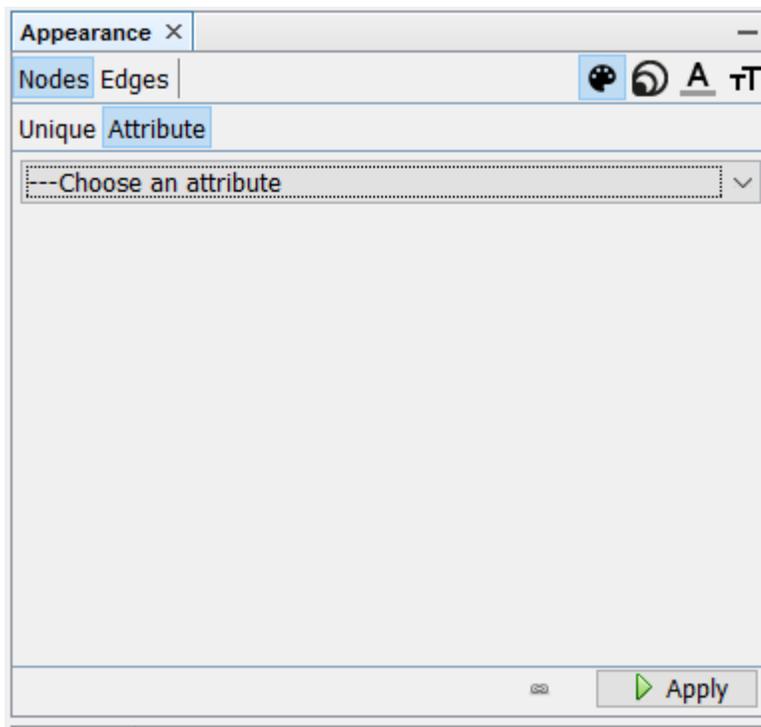
Bottom buttons, from left to right: Background color (left click to switch between black and white, right click to choose color), Take Screenshot, Show Node Labels, Show Edges, Edges have source node color, Show Edge labels, Edge Weight Scale, Size Mode( Fixed, Scaled or Node size) Color Mode( Unique, Object or Text), Font, Font

size scale, Default color, Attributes (choose which attributes are displayed on graph overview)

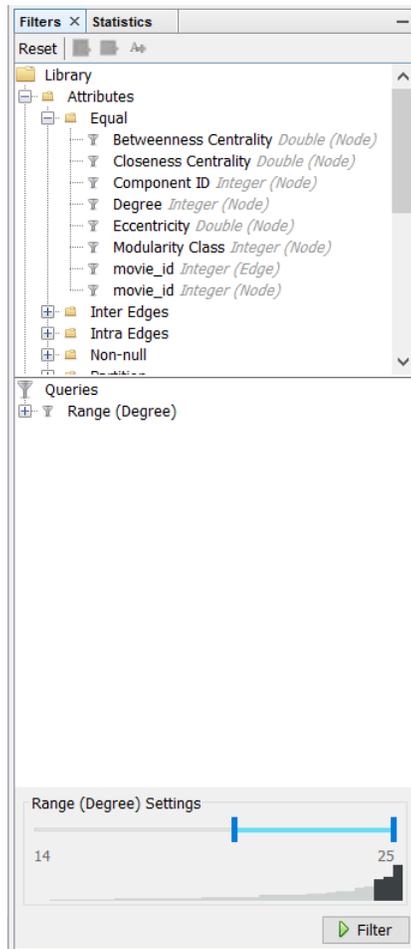
## 6. Change Graph Layout



## 7. Change Node/Edge Colour (Unique applies the exact same colour to all nodes/edges. Attribute changes the colour of each node/edge according to its value of the selected attribute)



8. Apply Filters Create filters that can hide nodes and edges ( that do not comply with the filter's settings) To apply a filter drag it from the “Filters” module to the “Queries” module and press the “Filter” button on the bottom right side. Filter settings can be modified.



### 9. Calculate Graph Metrics

The screenshot shows the 'Filters' panel in Gephi, which is used to calculate and display various graph metrics. The panel is organized into several sections:

- Network Overview:** Includes Average Degree, Avg. Weighted Degree, Network Diameter, Graph Density, Modularity, PageRank, and Connected Components.
- Node Overview:** Includes Avg. Clustering Coefficient and Eigenvector Centrality.
- Edge Overview:** Includes Avg. Path Length.
- Dynamic:** Includes # Nodes, # Edges, Degree, and Clustering Coefficient.

Each metric has a 'Run' button next to it, indicating that the calculation is performed on demand.

### 10. Data Laboratory

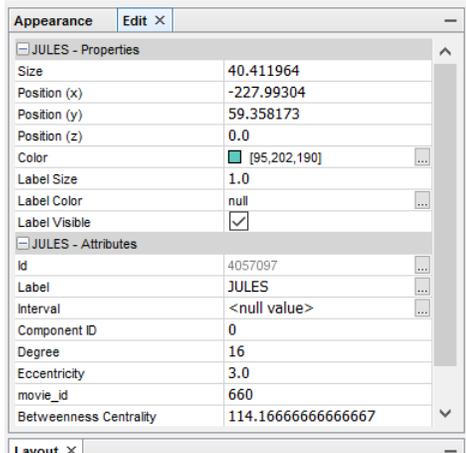
The screenshot displays the 'Data Laboratory' window in Gephi, which allows users to view and manipulate the data table. The interface includes a menu bar with options like 'Configuration', 'Add node', 'Add edge', 'Search/Replace', 'Import Spreadsheet', 'Export table', and 'More actions'. Below the menu is a toolbar with various data manipulation actions:

- Add column
- Merge columns
- Delete column
- Clear column
- Copy data to other column
- Fill column with a value
- Duplicate column
- Create a boolean column from regex match
- Create column with list of regex matching groups
- Negate boolean values

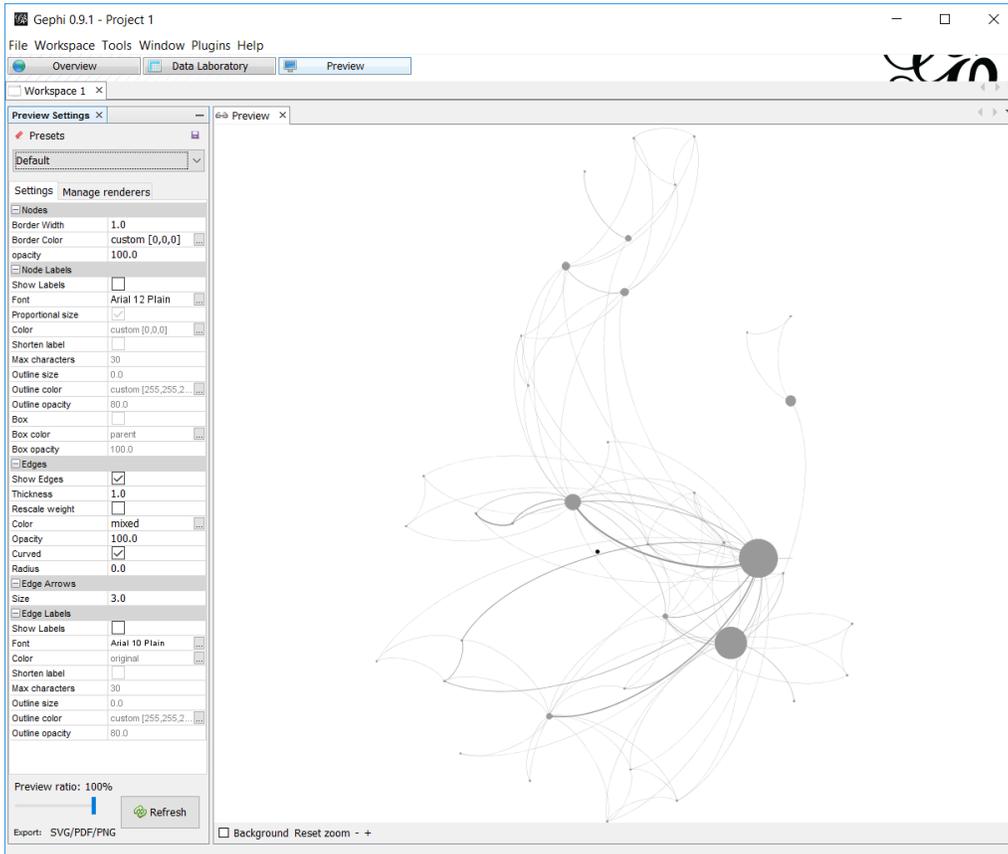
The main area of the window shows a data table with the following columns: Id, Label, Interval, Component ID, Degree, Eccentricity, movie\_id, Betweenness Centrality, Closeness Centrality, and Modularity Class. The table contains 40 rows of data, each representing a node in the network.

Id	Label	Interval	Component ID	Degree	Eccentricity	movie_id	Betweenness Centrality	Closeness Centrality	Modularity Class
4057101	VINCENT	0	25	3.0	3.0	660	301.0	1.405405	0
4058019	BUTCH	0	17	3.0	660	248.333333	1.621622	1	
4057097	JULES	0	16	3.0	660	114.166667	1.648649	2	
4061652	MAYNARD	0	3	4.0	660	70.0	2.486486	4	
4057077	HONEY BUNNY	0	8	4.0	660	49.5	2.081081	3	
4057073	PUMPKIN	0	8	4.0	660	49.5	2.081081	3	
4060209	SPORTSCASTE...	0	2	4.0	660	36.0	2.540541	1	
4058398	MIA	0	11	3.0	660	33.5	1.864865	0	
4057986	MARSELLUS	0	10	3.0	660	26.666667	1.891892	1	
4057605	BRETT	0	7	3.0	660	2.666667	1.972973	1	
4058186	JODY	0	4	4.0	660	0.333333	2.297297	0	
4058231	LANCE	0	4	4.0	660	0.333333	2.297297	0	
4058592	BUDDY	0	2	4.0	660	0.0	2.351351	0	
4060076	CAPT KOONS	0	5	3.0	660	0.0	2.027027	0	
4059110	ED SULLIVAN	0	2	4.0	660	0.0	2.351351	0	
4058053	ENGLISH DAVE	0	4	3.0	660	0.0	2.054054	0	
4060290	ESMARELDA	0	1	4.0	660	0.0	2.594595	1	
4060614	FABIENNE	0	3	3.0	660	0.0	2.216216	1	
4062356	FOURTH MAN	0	2	4.0	660	0.0	2.324324	2	
4061593	GAWKER #2	0	3	4.0	660	0.0	2.540541	1	
4062737	JIMMIE	0	3	4.0	660	0.0	2.297297	2	
4063840	MANAGER	0	5	4.0	660	0.0	2.162162	3	
4057763	MARVIN	0	6	3.0	660	0.0	2.0	1	
4060058	MOTHER	0	5	3.0	660	0.0	2.027027	0	
4063718	PATRON	0	5	4.0	660	0.0	2.162162	3	
4061572	PEDESTRIAN	0	3	4.0	660	0.0	2.540541	1	
4059417	PREACHER	0	3	4.0	660	0.0	2.324324	0	
4063490	RAQUEL	0	3	4.0	660	0.0	2.297297	2	
4057772	ROGER	0	6	3.0	660	0.0	2.0	1	
4060231	SPORTSCASTE...	0	1	5.0	660	0.0	3.513514	1	
4061832	THE GIMP	0	2	5.0	660	0.0	3.432432	4	
4062894	THE WOLF	0	3	4.0	660	0.0	2.297297	2	
4056826	WATRESS	0	4	5.0	660	0.0	2.972973	3	
4063481	WINSTON	0	3	4.0	660	0.0	2.297297	2	
4060044	WOMAN	0	5	3.0	660	0.0	2.027027	0	
4056770	YOUNG MAN	0	4	5.0	660	0.0	2.972973	3	
4056774	YOUNG WOMAN	0	4	5.0	660	0.0	2.972973	3	
4061778	ZED	0	2	5.0	660	0.0	3.432432	4	

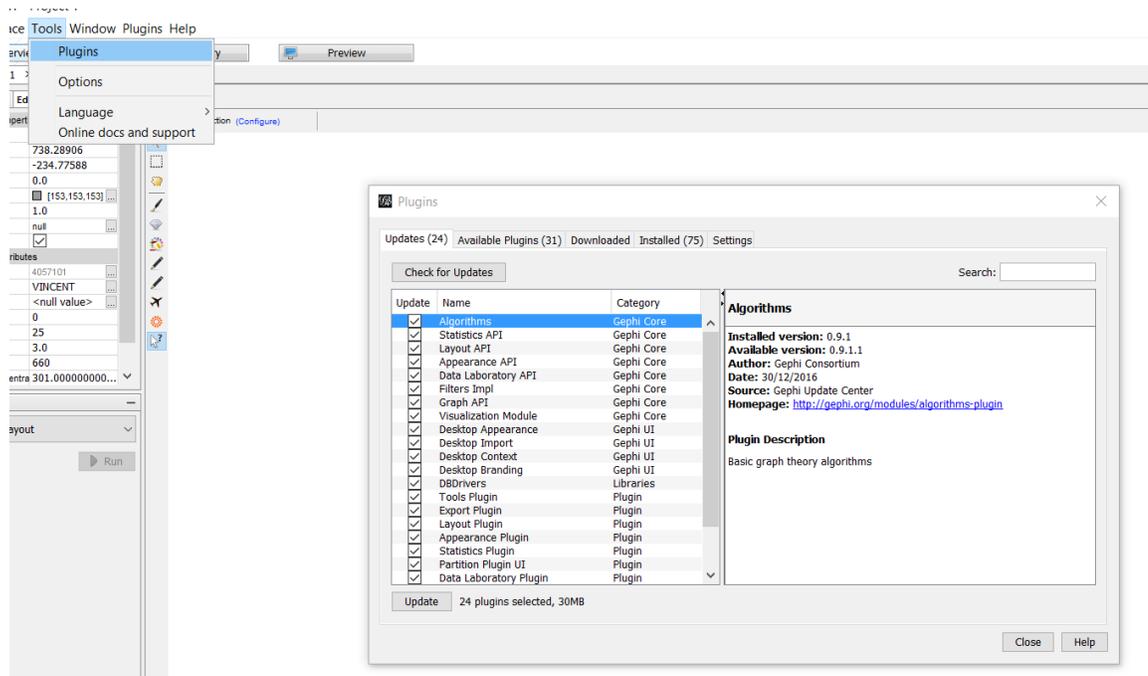
11. Edit Node (either use the “Edit” button and click on the Node in the Overview screen or left click on the Node in the Data Laboratory screen)



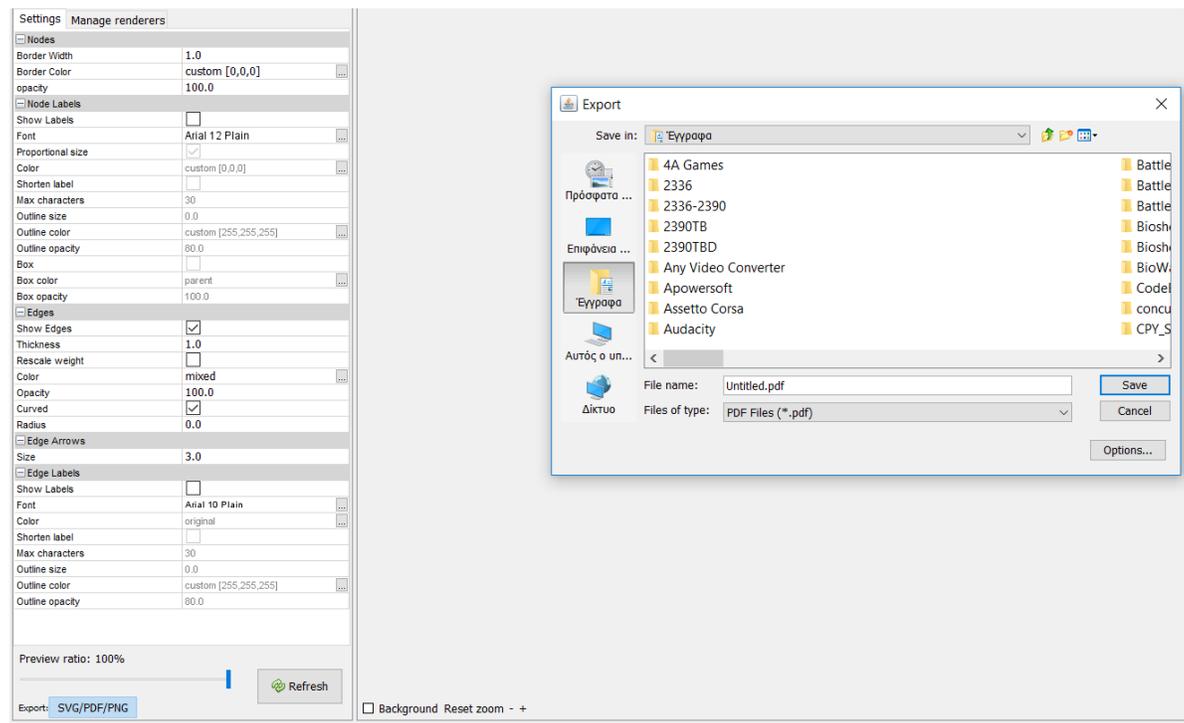
12. Preview Graph



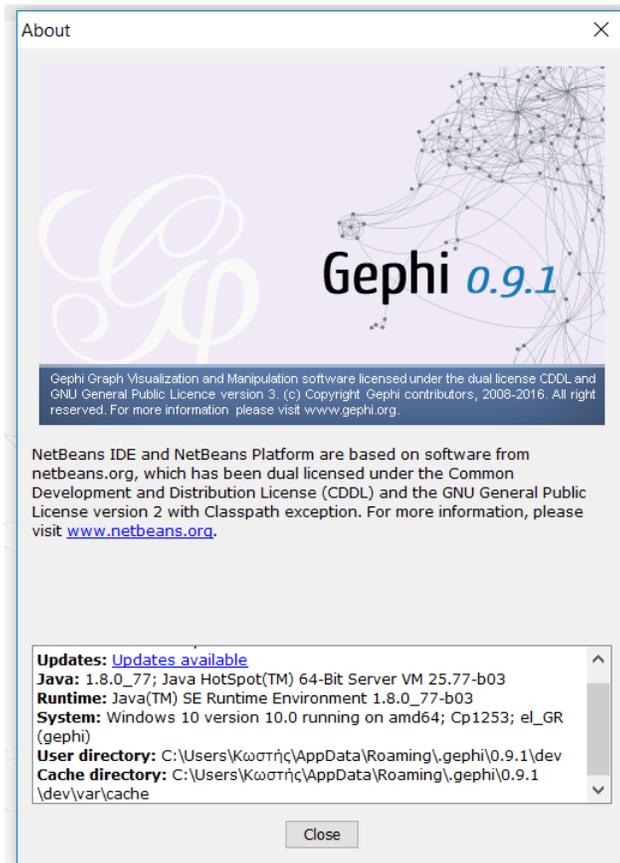
### 13. Manage Plugins



### 14. Export Graph



## 15. Help -&gt; About



## 3.2 Hardware Interfaces

The minimum hardware requirements of Gephi are a 500 Megahertz CPU and 128 megabytes of RAM. Also, because Gephi uses an OpenGL 3D engine to speed up graph visualization, a compatible graphics card is required. A system with these specifications can handle a Network of approximately 1000 edges and nodes. For bigger networks, additional memory is required (<https://gephi.org/users/requirements/>).

### **3.3 Software Interfaces**

Gephi requires Java to be installed on the system, more specifically Java version 7 or 8 for its latest release. Additional information can be found on section 2.7 of this document.

Gephi can be connected with a MySQL, SQLite or PostgreSQL database to import a graph edge list.

### **3.4 Communications Interfaces**

Gephi requires an internet connection to install new plugins, update already installed ones and update some of its components (APIs, modules etc.).

## **4. System Features**

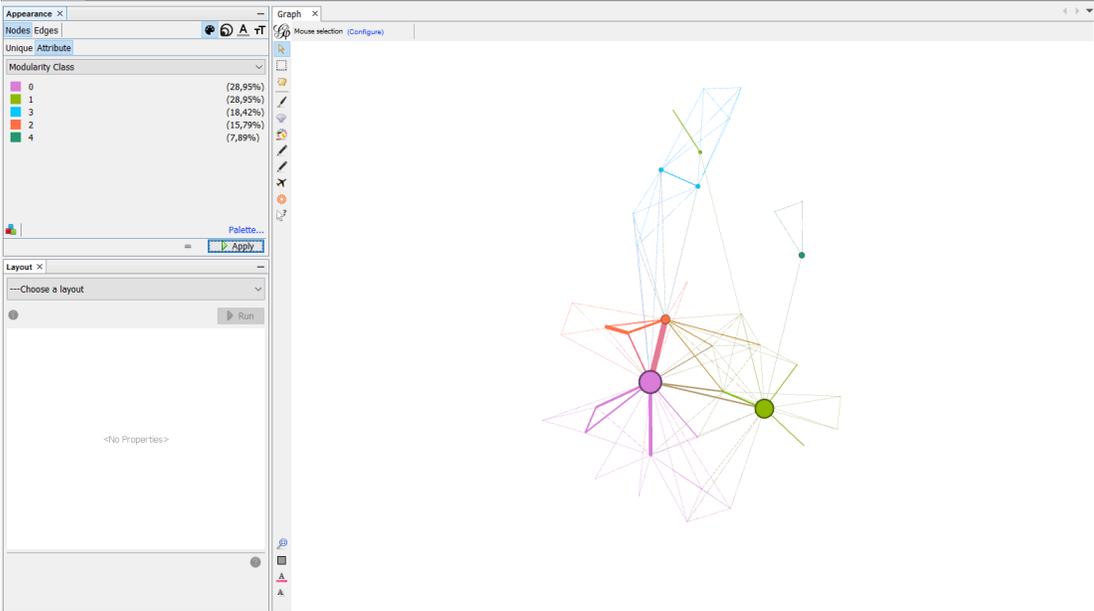
This section demonstrates Gephi's most prominent features and explains how they can be used and the results they will give back to the user.

### **4.1 Graph Visualization**

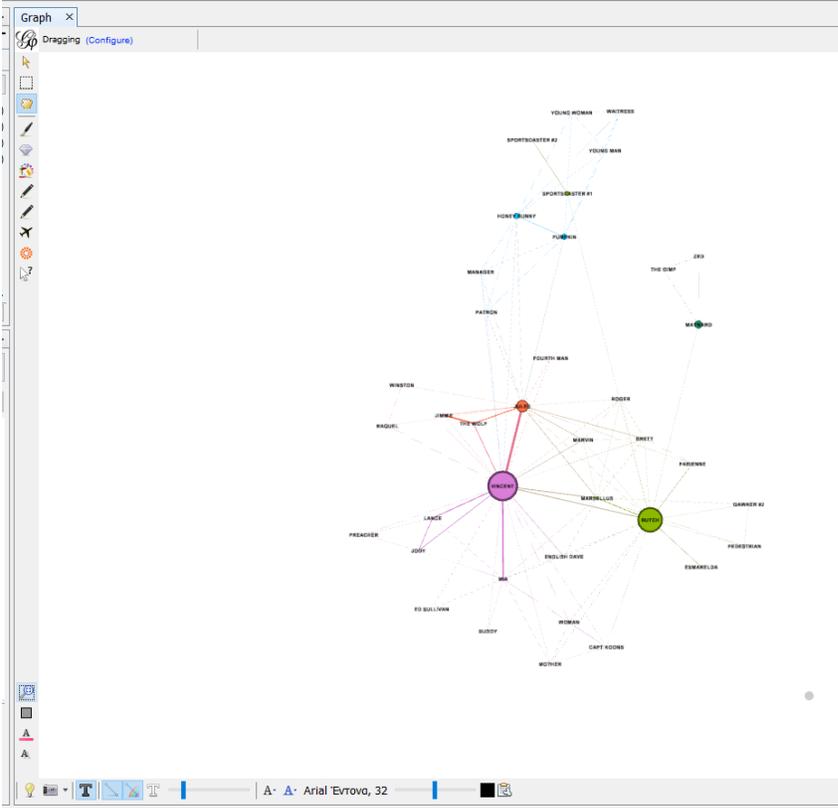
Users can directly interact with the graph by clicking on nodes to drag them, edit their attributes or color, add new nodes and edges, display shortest paths between nodes, make node labels visible/invisible etc.

Users can also use the appearance module to change the color of nodes. They can use the same color for the entire graph or use the values of a specific attribute as a reference point.

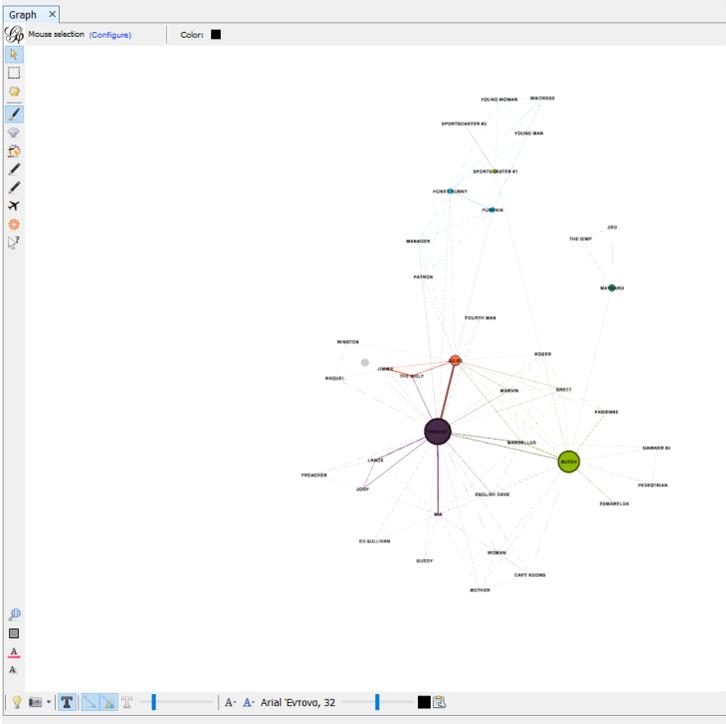
Example of changing the color of nodes based on one of their attributes (modularity class):



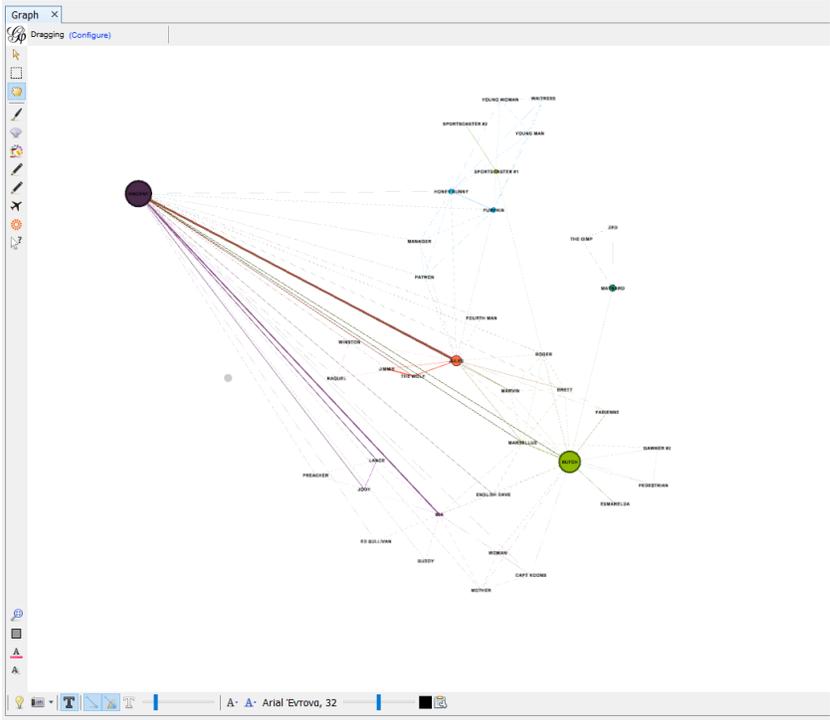
Example of making node labels visible (using the Show Node Labels – T button ):



Example of changing the color of only one node( Vincent) using mouse selection:  
(The user has to press the “Painter” button first, on the left side of the screen, then choose color, on the top of the screen, and finally click on the desired node)



Example of Node Dragging (by pressing the "Drag" button on the left side of the screen) :

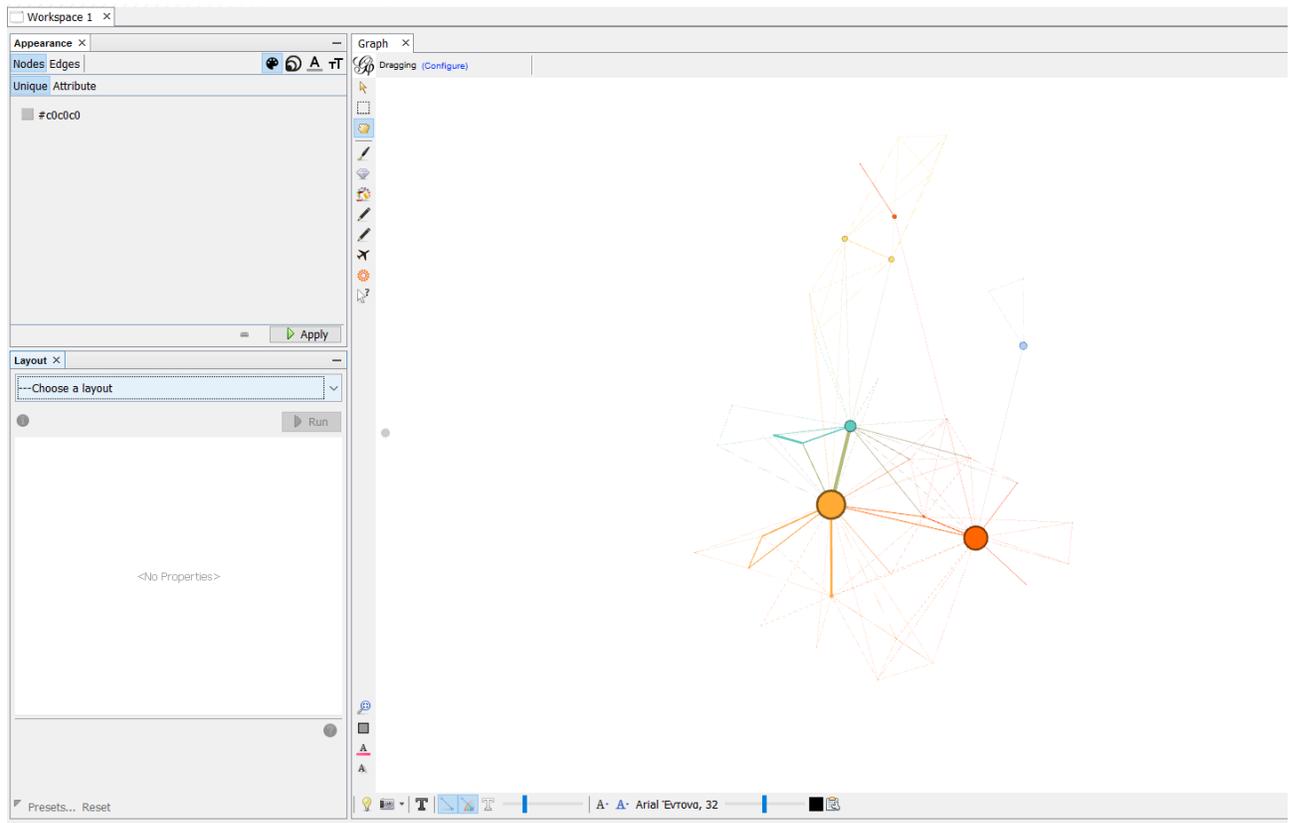


## 4.2 Graph Layout

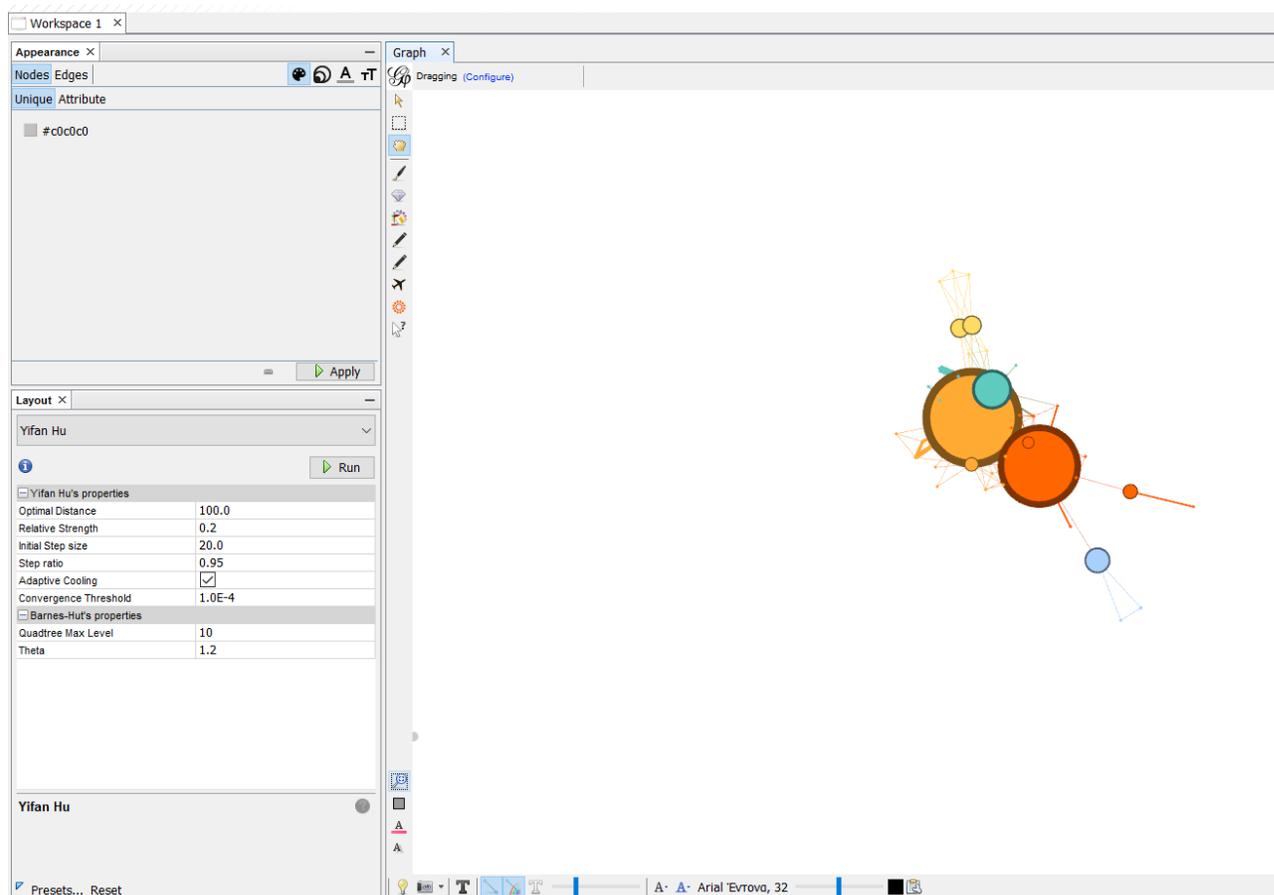
Gephi provides the following layout algorithms: Contraction, Expansion, Force Atlas, Force Atlas 2, Fruchterman Reingold, Label Adjust, OpenOrd, Random Layout, Rotate, Yifan Hu and Yifan Hu Proportional.

Example of applying a Layout algorithm to a graph:

The graph, before changing the Layout:



The graph, after applying the layout “Yifan Hu”:



### 4.3 Graph Metrics

Gephi provides the following Graph metrics:

Network Overview: Average Degree, Average Weighted Degree, Network Diameter,

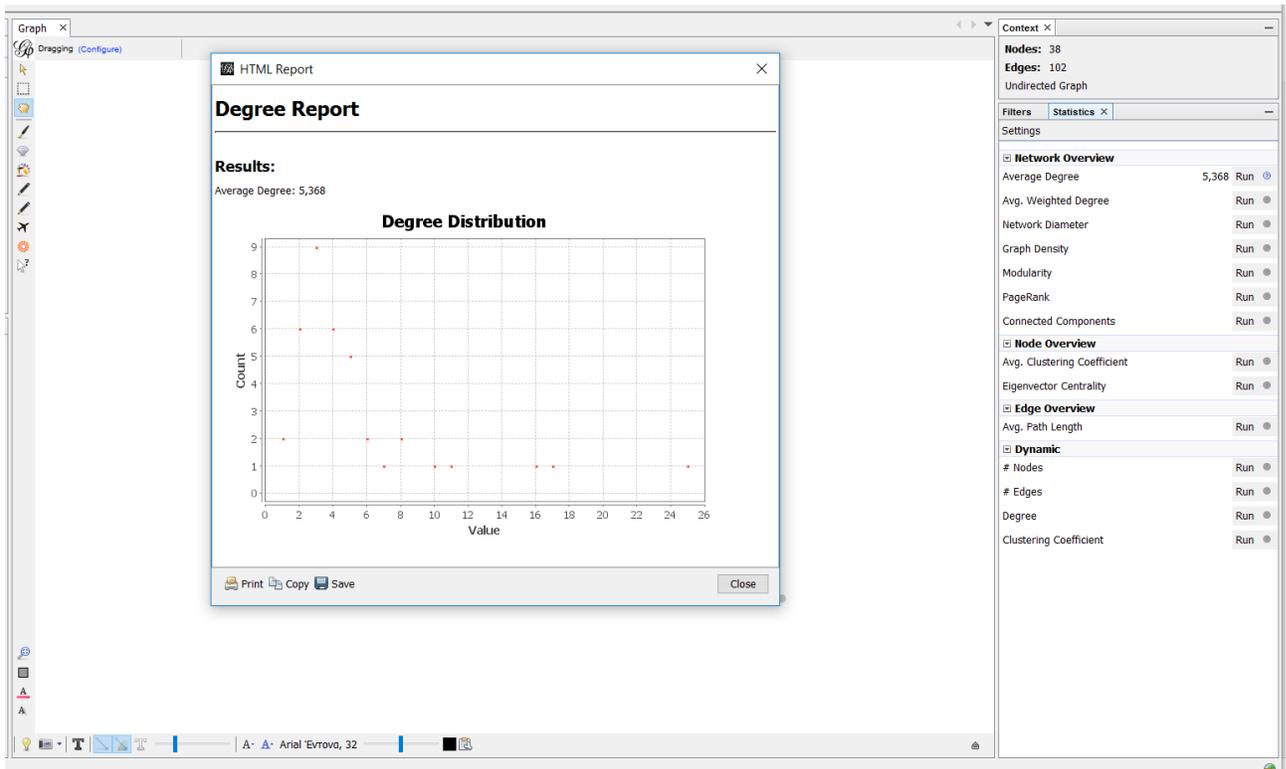
Graph Density, Modularity, PageRank, Connected Components

Node Overview: Average Clustering Coefficient, Eigenvector Centrality

Edge Overview: Average Path Length

Dynamic: #Nodes, #Edges, Degree, Clustering Coefficient

Example of calculating the “Average Degree” of the graph:



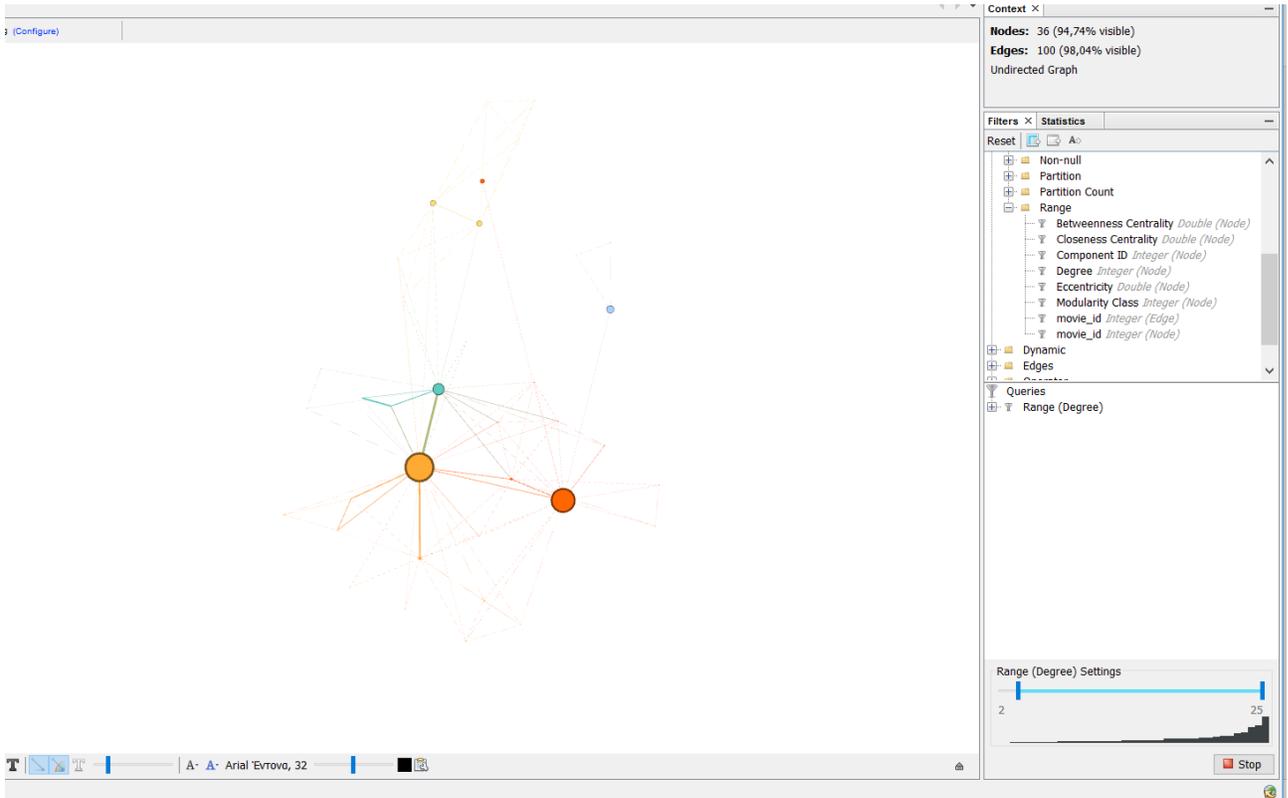
The graph which represents the distribution of the Average Degree can be printed, copied or saved.

#### 4.4 Filters

To apply a filter the user must select it and drag it to the “Queries” module.

Example of applying a filter which removes every node that has a degree outside of the range [2,25], in other words it removes every leaf node.

On the bottom right side of the screen there is the Stop/Filter button which removes/applies the selected Filters. In the context module, we can see the number and percentage of the nodes and edges that did not get removed by the filter.



### 4.5 Data Table

Here the data of the graph is presented in two tables, one for nodes and one for edges and can be edited by the user. This is the Nodes table, the user can change to

id	Label	Interval	Component ID	Degree	Eccentricity	movie_id	Betweenness Centrality	Closeness Centrality	Modularity Class
4057101	VINCENT	0	25	3.0	0.0	660	381.0	1.405405	0
4058019	BUTCH	0	17	3.0	0.0	660	248.333333	1.821622	1
4057097	JULES	0	16	3.0	0.0	660	114.166667	1.448649	2
4061552	MAYNARD	0	3	4.0	0.0	660	78.0	2.486486	4
4057077	HONEY BUNNY	0	8	4.0	0.0	660	49.5	2.881081	3
4057073	PUMPKIN	0	8	4.0	0.0	660	49.5	2.881081	3
4060209	SPORTSCASTER #1	0	2	4.0	0.0	660	36.9	2.340541	1
4058398	MEA	0	11	3.0	0.0	660	33.5	1.864865	0
4057986	MARSELLUS	0	10	3.0	0.0	660	26.666667	1.891892	1
4057905	BRETT	0	7	3.0	0.0	660	2.666667	1.972973	1
4058186	JOEY	0	4	4.0	0.0	660	0.333333	2.297297	0
4058231	LANCE	0	4	4.0	0.0	660	0.333333	2.297297	0
4058592	BUDDY	0	2	4.0	0.0	660	0.0	2.351351	0
4060076	CAPT KIDDIE	0	5	3.0	0.0	660	0.0	2.827027	0
4059110	ED SULLIVAN	0	2	4.0	0.0	660	0.0	2.351351	0
4058055	ENGLISH DAVE	0	4	3.0	0.0	660	0.0	2.654054	0
4060290	ESMARELDA	0	1	4.0	0.0	660	0.0	2.394955	1
4060614	FABEENNE	0	3	3.0	0.0	660	0.0	2.216216	1
4062356	FOURTH MAN	0	2	4.0	0.0	660	0.0	2.242424	2
4061593	GAWKER #2	0	3	4.0	0.0	660	0.0	2.340541	1
4062737	JIMMIE	0	3	4.0	0.0	660	0.0	2.297297	2
4063840	MANAGER	0	5	4.0	0.0	660	0.0	2.162162	3
4057763	MARVIN	0	6	3.0	0.0	660	0.0	2.0	1
4060058	MOTHER	0	5	3.0	0.0	660	0.0	2.827027	0
4063718	PATRON	0	5	4.0	0.0	660	0.0	2.162162	3
4061572	PREDSTEFAN	0	3	4.0	0.0	660	0.0	2.340541	1
4059417	PREACHER	0	3	4.0	0.0	660	0.0	2.242424	0
4063490	RAQUEL	0	3	4.0	0.0	660	0.0	2.297297	2
4057772	ROGER	0	6	3.0	0.0	660	0.0	2.0	1
4060231	SPORTSCASTER #2	0	1	3.0	0.0	660	0.0	3.313514	1
4061832	THE GIMP	0	2	3.0	0.0	660	0.0	3.432432	4
4062894	THE WOLF	0	3	4.0	0.0	660	0.0	2.297297	2
4058028	WAITRESS	0	4	3.0	0.0	660	0.0	1.972973	3
4063481	WINSTON	0	3	4.0	0.0	660	0.0	2.297297	2
4060044	WOMAN	0	5	3.0	0.0	660	0.0	2.827027	0
4058770	YOUNG MAN	0	4	3.0	0.0	660	0.0	2.872973	3
4058774	YOUNG WOMAN	0	4	3.0	0.0	660	0.0	2.872973	3
4061778	ZED	0	2	3.0	0.0	660	0.0	3.432432	4

the edges table by pressing the “Edges” button on the top left hand side of the screen. The user can add new nodes/edges, edit existing ones, merge columns, create duplicates of columns etc.

Example of adding a new Edge to the graph (by pressing the “Add edge” button):

The screenshot shows the Gephi interface with a table of nodes. The table has columns: Label, Interval, Component ID, Degree, Eccentricity, and movie\_id. An 'Add edge' dialog box is open, allowing the user to select the new edge type (Directed or Undirected), source and target nodes, and the edge kind.

Label	Interval	Component ID	Degree	Eccentricity	movie_id
VINCENT		0	25	3.0	660
BUTCH		0	17	3.0	660
JULES		0	16	3.0	660
MAYNARD		0	3	4.0	660
HONEY BUNNY		0	8	4.0	660
PUMPKIN		0	8	4.0	660
SPORTSCASTER #1		0	2	4.0	660
MIA		0	11	3.0	660
MARSELLUS		0	10	3.0	660
BRETT		0	7	3.0	660
JODY		0	4		
LANCE		0	4		
BUDDY		0	2		
CAPT KOONS		0	5		
ED SULLIVAN		0	2		
ENGLISH DAVE		0	4		
ESMARELDA		0	1		
FABRIENNE		0	3		
FOURTH MAN		0	2		
GAWKER #2		0	3		
JIMMIE		0	3		
MANAGER		0	5		
MARVIN		0	6		
MOTHER		0	5		
PATRON		0	5		
PEDESTRIAN		0	3	4.0	660
PREACHER		0	3	4.0	660
RAQUEL		0	3	4.0	660
ROGER		0	6	3.0	660

Example of merging two columns into one:

The screenshot shows the Gephi interface with a table of nodes. The table has columns: Interval, Component ID, Degree, Eccentricity, movie\_id, Betweenness Centrality, Closeness Centrality, and M. A 'Merge columns' dialog box is open, allowing the user to choose the columns to merge and a merge strategy.

Interval	Component ID	Degree	Eccentricity	movie_id	Betweenness Centrality	Closeness Centrality	M
	0	25	3.0	660	301.0	1.405405	0
	0	17	3.0	660	248.333333	1.621622	1
	0	16	3.0	660	114.166667	1.648649	2
	0	3	4.0	660	70.0	2.486486	4
	0	8	4.0	660	49.5	2.081081	3
	0	8	4.0	660	49.5	2.081081	3
	0	2			36.0	2.540541	1
	0	11			33.5	1.864865	0
	0	10			26.666667	1.891892	1
	0	7			2.666667	1.972973	1
	0	4			0.333333	2.297297	0
	0	4			0.333333	2.297297	0
	0	2			0.0	2.351351	0
	0	2			0.0	2.027027	0
	0	5			0.0	2.351351	0
	0	4			0.0	2.054054	0
	0	1			0.0	2.594595	1
	0	3			0.0	2.216216	1
	0	2			0.0	2.324324	2
	0	3			0.0	2.540541	1
	0	3			0.0	2.297297	2
	0	6			0.0	2.162162	3
	0	5			0.0	2.0	1
	0	5			0.0	2.027027	0
	0	5			0.0	2.162162	3
	0	3			0.0	2.540541	1
	0	3			0.0	2.324324	0
	0	3			0.0	2.297297	2
	0	6			0.0	2.0	1
	0	1			0.0	3.513514	1
	0	2			0.0	3.432432	4
	0	3			0.0	2.297297	2
	0	4			0.0	2.972973	3
	0	3			0.0	2.297297	2
	0	5			0.0	2.027027	0
	0	4			0.0	2.972973	3
	0	4			0.0	2.972973	3
	0	2			0.0	3.432432	4

Choosing the new column's title:

4	4.0	660
4	4.0	660
2	4.0	660
5		
2		
4		
1		
3		
2		
3		
3		
5		
6	3.0	660
5	3.0	660
6	4.0	660

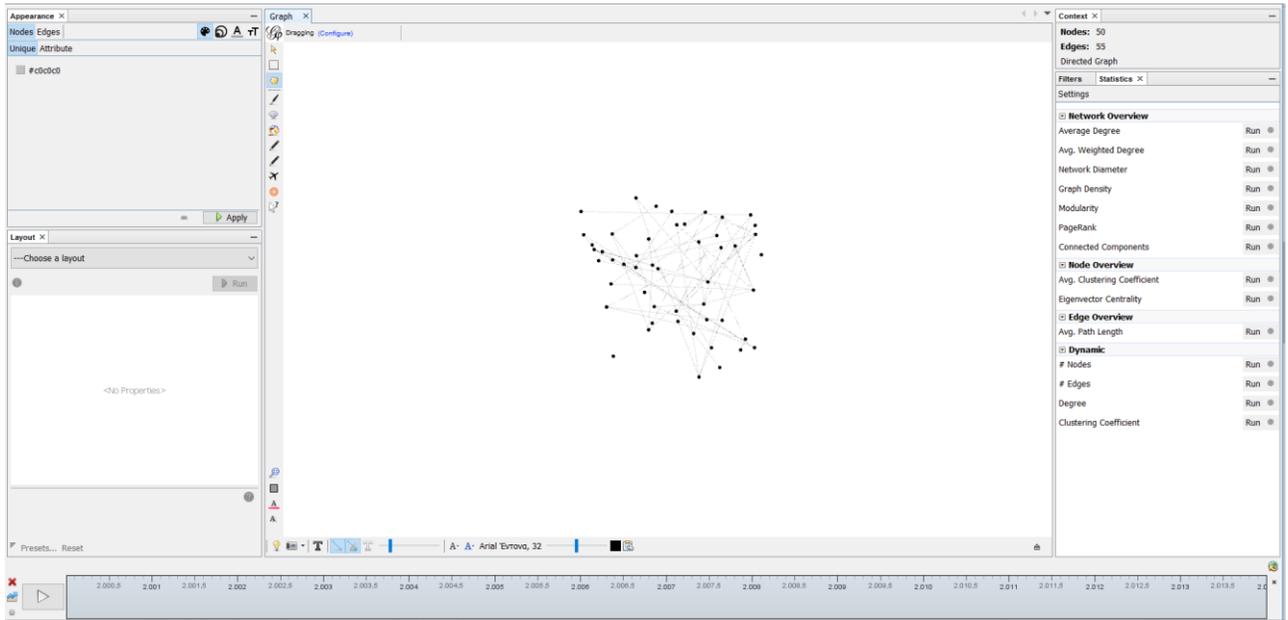
 Sum number values ✕

New column title:

The new column (last one from the left) in the table:

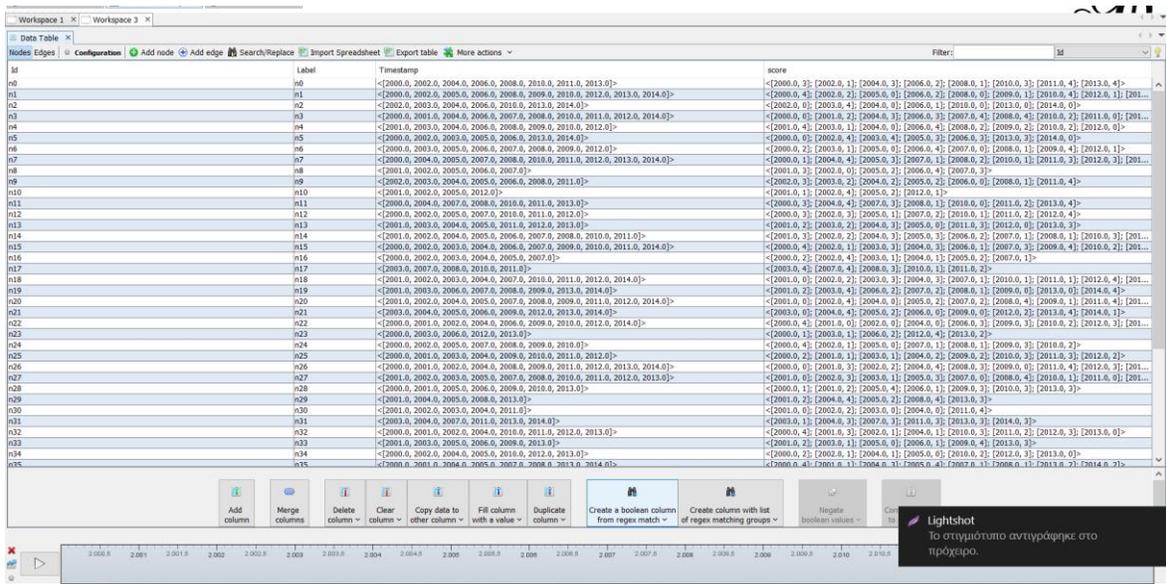
Id	Label	Interval	Component ID	Degree	Eccentricity	movie_id	Betweenness Centrality	Closeness Centrality	Modularity Class	Centrality Combination
4057101	VINCENT		0	25	3.0	660	301.0	1.405405	0	302.4054054054054055
4058019	BUTCH		0	17	3.0	660	248.333333	1.621622	1	249.95494949494949617
4057097	JULES		0	16	3.0	660	114.166667	1.648649	2	115.8153153153153187
4061652	MAYNARD		0	5	4.0	660	79.0	2.486486	4	72.4864864864864864
4057077	HONEY BUNNY		0	8	4.0	660	49.5	2.081081	3	51.581081081081081
4057073	PUMPKIN		0	8	4.0	660	49.5	2.081081	3	51.581081081081081
4060209	SPORTSCASTER #1		0	2	4.0	660	36.0	2.540541	1	38.5405405405405403
4058398	MI		0	11	3.0	660	33.5	1.864865	0	35.364864864864865
4057986	MARSELLUS		0	10	3.0	660	26.666667	1.891892	1	28.5385858585858599
4057605	BRETT		0	7	3.0	660	2.666667	1.972973	1	4.639639639639639
4058186	JODY		0	4	4.0	660	0.333333	2.297297	0	2.6306306306306307
4058231	LANCE		0	4	4.0	660	0.333333	2.297297	0	2.6306306306306307
4058592	BUDDY		0	2	4.0	660	0.0	2.351351	0	2.3513513513513513
4060076	CAPT KOONS		0	5	3.0	660	0.0	2.027027	0	2.027027027027027
4059110	ED SULLIVAN		0	2	4.0	660	0.0	2.351351	0	2.3513513513513513
4058055	ENGLISH DAVE		0	4	3.0	660	0.0	2.054054	0	2.054054054054054
4060290	ESMARELDA		0	1	4.0	660	0.0	2.594595	1	2.5945949494949497
4060614	FABIENNE		0	3	3.0	660	0.0	2.216216	1	2.2162162162162162
4062356	FOURTH MAN		0	2	4.0	660	0.0	2.324324	2	2.3243242424242424
4061593	SAWYER #2		0	3	4.0	660	0.0	2.540541	1	2.5405405405405403
4062737	JIMMIE		0	3	4.0	660	0.0	2.297297	2	2.2972972972972974
4063840	MANAGER		0	5	4.0	660	0.0	2.162162	3	2.1621621621621623
4057763	MARVIN		0	6	3.0	660	0.0	2.0	1	2.0
4060058	MOTHER		0	5	3.0	660	0.0	2.027027	0	2.027027027027027
4063718	PATRON		0	5	4.0	660	0.0	2.162162	3	2.1621621621621623
4061572	PEDESTRIAN		0	3	4.0	660	0.0	2.540541	1	2.5405405405405403
4059417	PREACHER		0	3	4.0	660	0.0	2.324324	0	2.3243242424242424
4063490	RAQUEL		0	3	4.0	660	0.0	2.297297	2	2.2972972972972974
4057722	ROGER		0	6	3.0	660	0.0	2.0	1	2.0
4060231	SPORTSCASTER #2		0	1	5.0	660	0.0	3.513514	1	3.5135135135135136
4061832	THE GMP		0	2	5.0	660	0.0	3.432432	4	3.4324324324324325
4063894	THE WOLF		0	3	4.0	660	0.0	2.297297	2	2.2972972972972974
4056826	WAITRESS		0	4	5.0	660	0.0	2.972973	3	2.972972972972973
4063481	WRISTON		0	3	4.0	660	0.0	2.297297	2	2.2972972972972974
4060044	WOMAN		0	5	3.0	660	0.0	2.027027	0	2.027027027027027
4056778	YOUNG MAN		0	4	5.0	660	0.0	2.972973	3	2.972972972972973
4056774	YOUNG WOMAN		0	4	5.0	660	0.0	2.972973	3	2.972972972972973
4061778	ZED		0	2	5.0	660	0.0	3.432432	4	3.4324324324324325

## 4.6 Dynamic Graphs



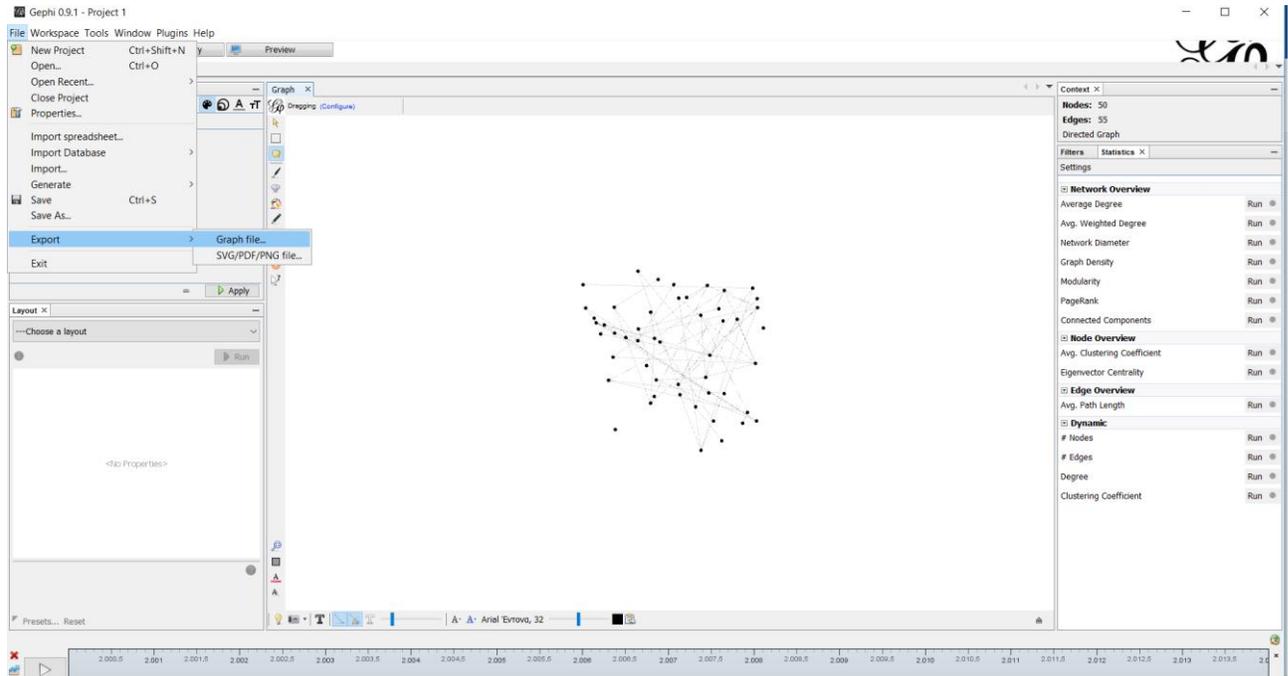
On the bottom side of the page we have the timeline bar, which can be used to demonstrate how the graph has changed at each timestamp. The user can select custom time intervals.

We can see in the data laboratory the appearance of the attribute Timestamp, which represents the time, and also the different values of the attribute score for every timestamp. Gephi also provides special metrics for dynamic graphs (for more information refer to section 4.3)

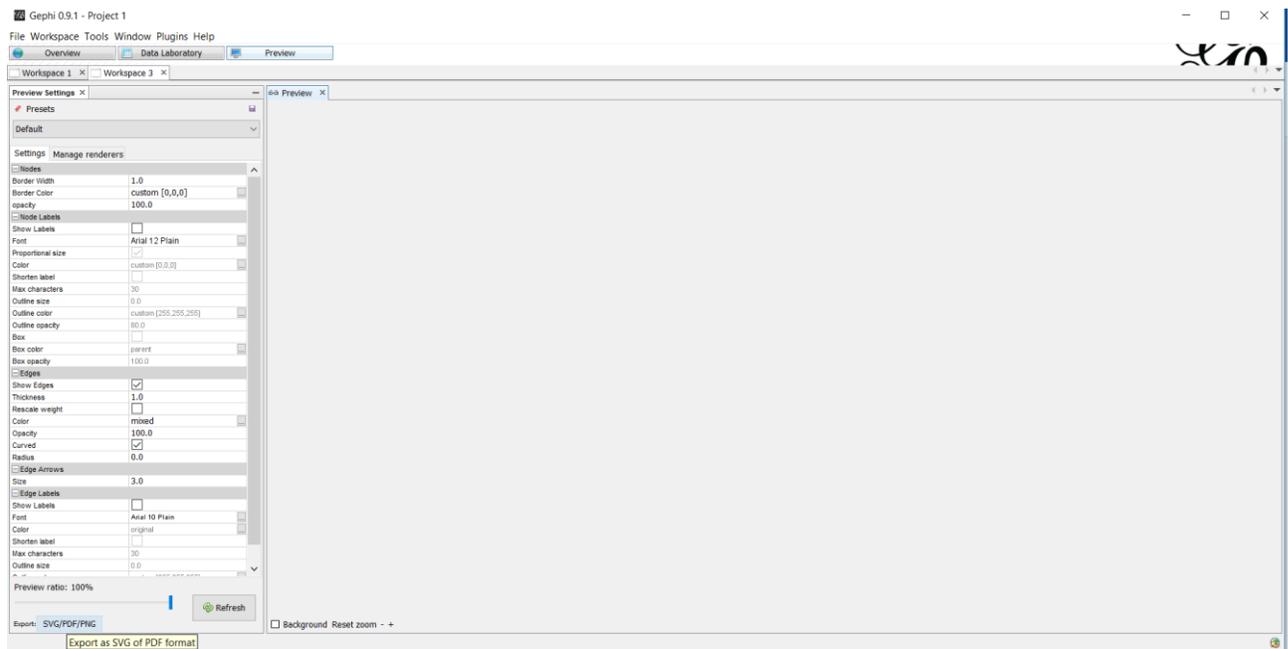


## 4.7 Graph Export

The user can export the graph, either via the main toolbar



Or via the preview page:



The available output file formats are: SVG PDF and PNG (just for the graph it self) and also every supported graph format (such as gexf , which also includes the graph data)

## **5. Other Nonfunctional Requirements**

### **5.1 Performance Requirements**

Gephi requires a system with at least a 500 megahertz CPU and 128 megabytes of RAM and an OPENGL 1.2 compatible graphics card. However, these requirements can support effectively graphs of up to 1000 nodes and edges. Performance depends on the graph size and as a result, the system requirements for bigger graphs are more demanding. (precise figures: <https://gephi.org/users/requirements/>)

### **5.2 Safety Requirements**

To ensure that no one of Gephi's users loses any data while using Gephi (due to a crash or a bug of some kind) the developer team updates Gephi regularly. There is a bug tracker available where users can report any bugs they have encountered so that the developers can fix it in the next release.

### **5.3 Security Requirements**

Gephi does not have any security requirements and thus any type of user can use it without any additional privileges.

### **5.4 Software Quality Attributes**

Gephi provides the users with both simple and advanced features. Due to its well designed and easy to use interface it can be used by both experts and typical users. However, users must already have a basic knowledge of graphs before using it.

## Glossary

References: [https://en.wikipedia.org/wiki/Main\\_Page](https://en.wikipedia.org/wiki/Main_Page)

- Graph: a system of nodes connected in pairs by edges. Often subdivided into directed graphs or undirected graphs according to whether the edges have an orientation or not. Mixed graphs include both types of edges.
- Node: A node is (together with edges) one of the two basic units out of which graphs are constructed. Nodes of graphs are often considered to be atomic objects, with no internal structure.
- Edge: An edge is (together with vertices) one of the two basic units out of which graphs are constructed. Each edge has two (or in hypergraphs, more) vertices to which it is attached, called its endpoints. Edges may be directed or undirected; undirected edges are also called lines and directed edges are also called arcs or arrows. In an undirected simple graph, an edge may be represented as the set of its vertices, and in a directed simple graph it may be represented as an ordered pair of its vertices. An edge that connects vertices  $x$  and  $y$  is sometimes written  $xy$ .
- Degree: the degree of a node of a graph is the number of edges incident to the node.
- Weight: A graph structure can be extended by assigning a weight to each edge of the graph. Graphs with weights, or weighted graphs, are used to represent structures in which pairwise connections have some numerical values. For example, if a graph represents a road network, the weights could represent the length of each road.
- Network(graph) Diameter: The diameter of a graph is the length of the shortest path between the most distanced nodes.
- Graph Density: graph density is the ratio of the number of edges and the number of possible edges.

- Modularity: Modularity is one measure of the structure of networks or graphs. It was designed to measure the strength of division of a network into modules (also called groups, clusters or communities).
- PageRank is an algorithm used by Google Search to rank websites in their search engine results. Because the web is represented as a graph when PageRank is applied to it, it can be applied to every graph.
- Connected component: a connected component (or just component) of an undirected graph is a subgraph in which any two vertices are connected to each other by paths, and which is connected to no additional vertices in the supergraph.
- Clustering coefficient: clustering coefficient is a measure of the degree to which nodes in a graph tend to cluster together.
- Centrality: In graph theory and network analysis, indicators of centrality identify the most important vertices within a graph.
- Closeness Centrality: In a connected graph, the closeness centrality (or closeness) of a node is a measure of centrality in a network, calculated as the sum of the length of the shortest paths between the node and all other nodes in the graph. Thus, the more central a node is, the closer it is to all other nodes.
- Betweenness Centrality: In graph theory, betweenness centrality is a measure of centrality in a graph based on shortest paths. For every pair of vertices in a graph, there exists a shortest path between the vertices such that either the number of edges that the path passes through (for undirected graphs) or the sum of the weights of the edges (for directed graphs) is minimized. The betweenness centrality for each vertex is the number of these shortest paths that pass through the vertex.
- Dynamic Graph: An update on a graph is an operation that inserts or deletes edges or vertices of the graph or changes attributes associated with edges or vertices, such as cost or color. A dynamic graph is a graph that is subject to a sequence of updates.

