

**Team:**

Dan Bye

Danny Brill

James Draper

**Title:** Visual Process Manager

**Project Summary:**

Process manager that shows various usage statistics in a visually appealing manner. Additionally, the manager is able to sort processes, send signals to processes, and view process history.

---

**Project Requirements****No Business Requirements**

---

**User Requirements**

ID	Requirement	Topic Area	Priority
US-01	I want to be able to review my processes visually	Visualization	Critical
US-02	I want to manage my processes by interacting with the visualization	Process Management	High
US-03	I want to see a history of my usage	Recording	Medium
US-04	I want to be able to sort my processes by different metrics	Visualization	Medium

**Functional Requirements**

ID	Requirement	Topic Area	Priority
FR-01	Processes must be identified by PID	Process Management	Critical
FR-02	Processes must be presented visually	Visualization	High
FR-03	Visual process instances change appearance based on activity	Visualization	High
FR-04	Processes can be managed by sending signals	Management	High
FR-05	Processes can be sorted by category/metric	Management	Medium
FR-06	Process data can be stored in a database	Recording	Medium
FR-07	Database data can be shown over time	Visualization	Medium
FR-08	I don't want to be able to kill the program with the program	Management	Low

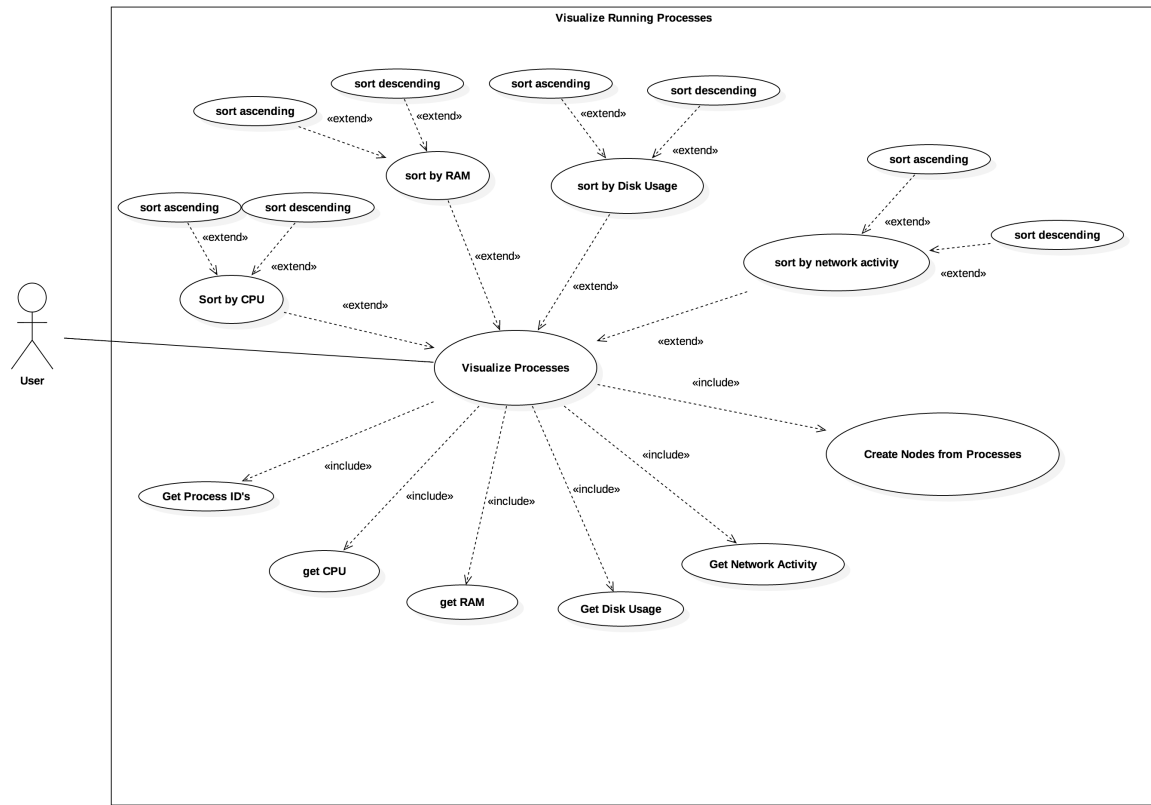
## Non-Functional Requirements

ID	Requirement	Topic Area	Priority
NF-01	I want the visualization to be sleek and intuitive	Visualization	Low
NF-02	I want to be the only one that can access my data	Security	Medium
NF-04	I want the color and locality of process nodes to be descriptive	Visualization	Medium
NF-05	I want my data to be updated in reasonable time	Performance	Low

## Use Cases

---

Use Case 01

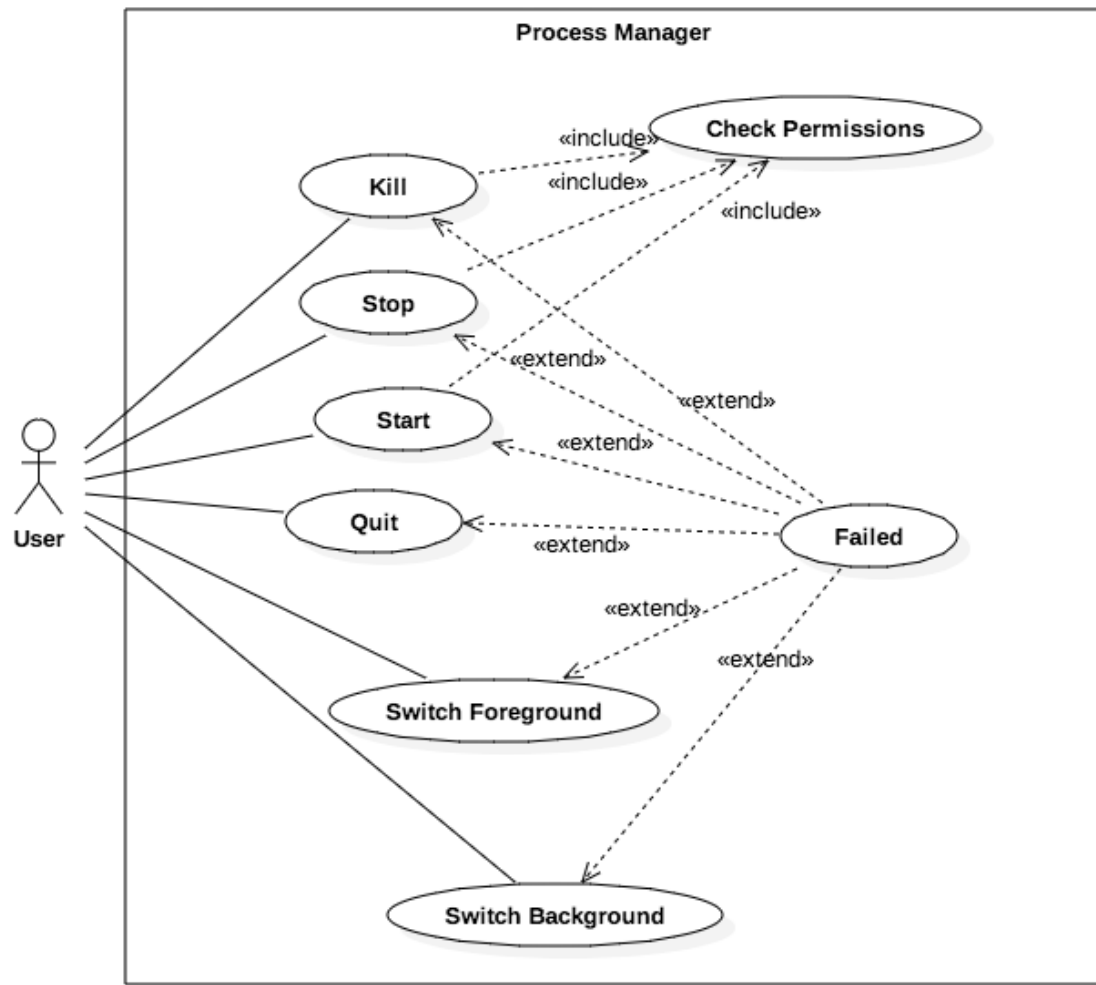


UC01 Diagram

<b>Use Case ID</b>	UC-01					
<b>Use Case Name</b>	Visualize Running Processes					
<b>Description</b>	User can visually render processes as colored nodes in a graph					
<b>Actors</b>	User					
<b>Pre-Conditions</b>	Program is running and processes have been found and displayed to the user.					
<b>Post-Conditions</b>	Nodes populated graph with position and color specific to their metrics.					
<b>Frequency of Use</b>	Every use of the program.					
<b>Flow of events</b>	Actor Action	System Response				
	1. Open Program					
	2. Choose Graph View	Process information gathered and displayed in node form.				
	3. Chose sorting preferences	Nodes re-render based on chosen preferences.				
<b>Variations</b>						
<b>Exceptions</b>	2. Graph View doesn't populate	Ask user for permission to gather system-level processes				

## UC01 Document

## Use Case 02



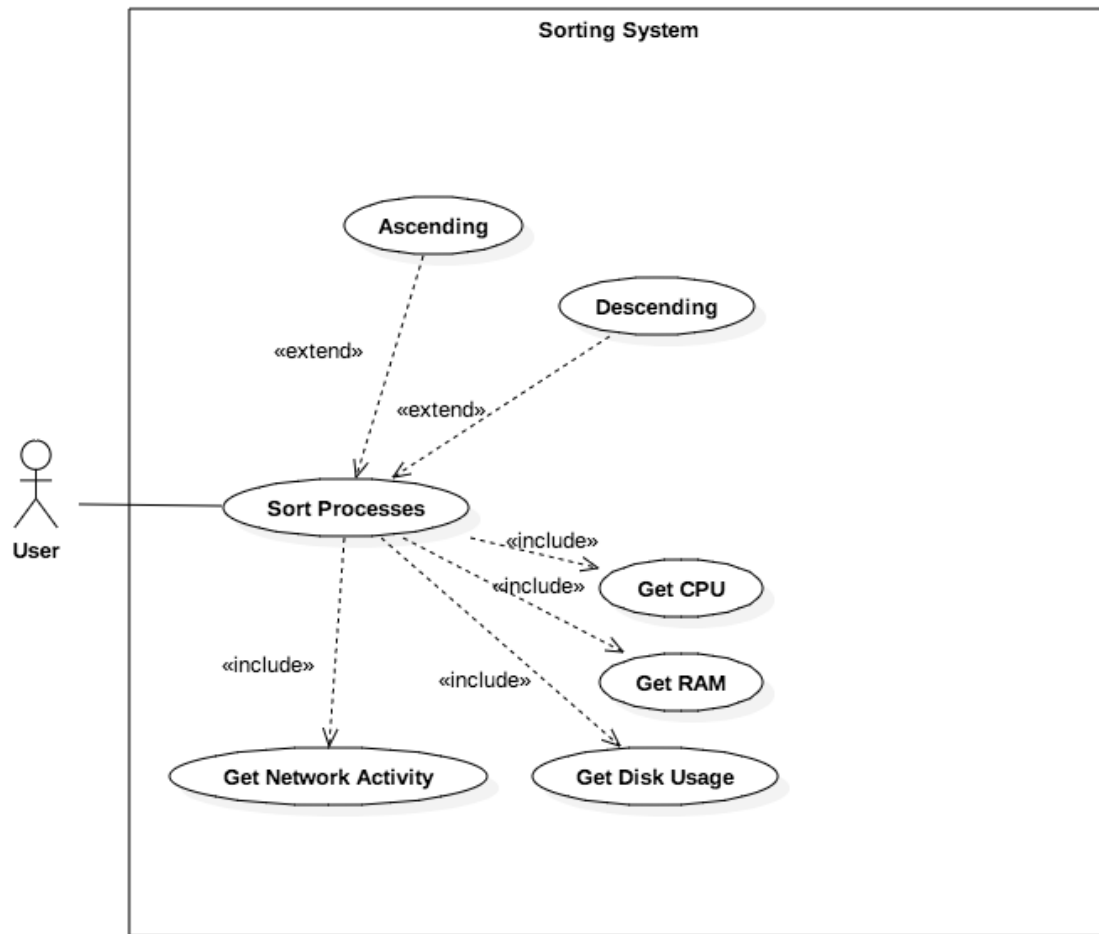
UC02 Diagram

<b>Use Case ID</b>	UC-02					
<b>Use Case Name</b>	Process Management					
<b>Description</b>	User can perform various actions involving system level processes. This includes killing, stopping, starting, quitting, etc.					
<b>Actors</b>	User					
<b>Pre-Conditions</b>	Program is running and processes have been found and displayed to the user.					
<b>Post-Conditions</b>	Process behavior is manageable from the user interface.					
<b>Frequency of Use</b>	Every use of the program.					
<b>Flow of events</b>	Actor Action	System Response				
	1. Open Program					
	2. Choose process view	Process information gathered and displayed.				
	3. Choose process to manage	Access to management actions.				
	4. Select management action	Process gets sent a signal based on action chosen.				
<b>Variations</b>						
<b>Exceptions</b>	4. Action fails	Notify user of failure				

### *UC02 Document*

---

## Use Case 03

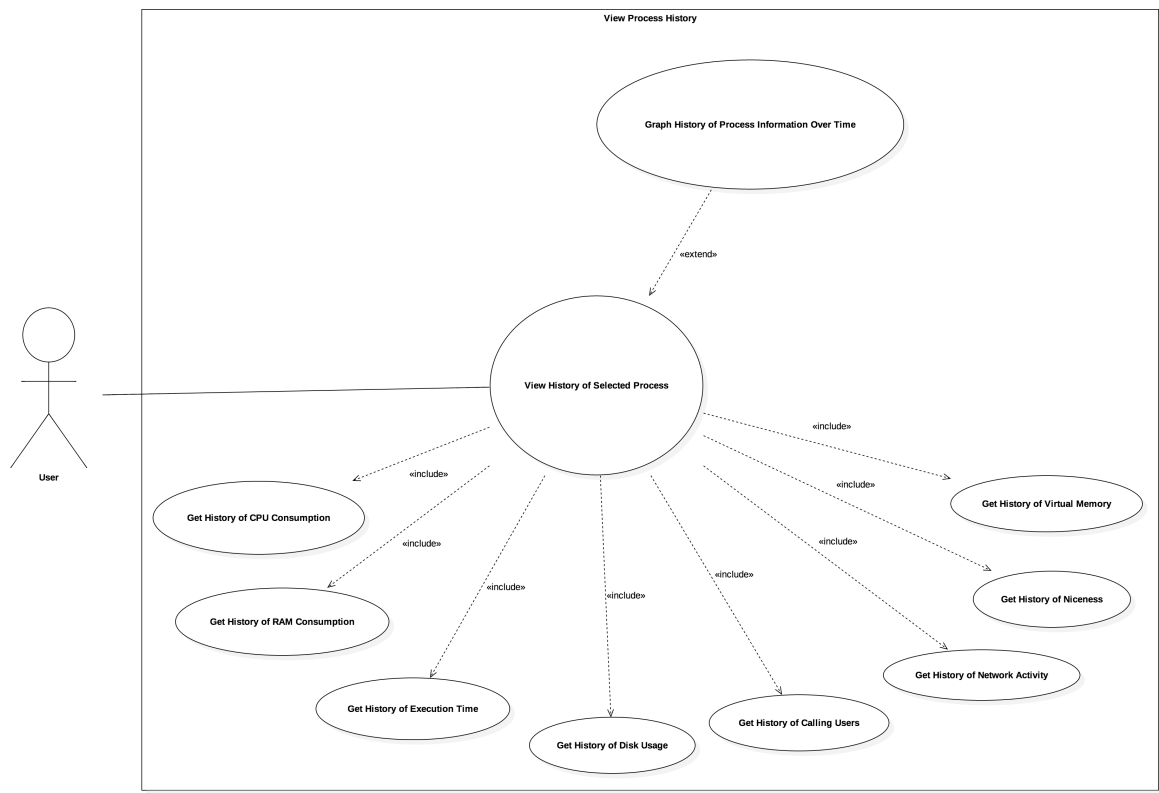


UC03 Diagram

Use Case ID	UC-03					
Use Case Name	Sort Processes					
Description	User can sort processes based on multiple system metrics.					
Actors	User					
Pre-Conditions	Program is running and information about processes has been gathered.					
Post-Conditions	Processes are displayed in order by chosen metric.					
Frequency of Use	Every use of the program.					
Flow of events	Actor Action	System Response				
	1. Open Program					
	2. Choose process sort view	Process information gathered and displayed.				
	3. Choose metric to sort by	Processes rearranged in descending order based on chosen metric				
	4. Switch to ascending order	Process sorting reversed by same chosen metric.				
Variations	4. Switch to descending order					
Exceptions						

UC03 Document

# Use Case 04



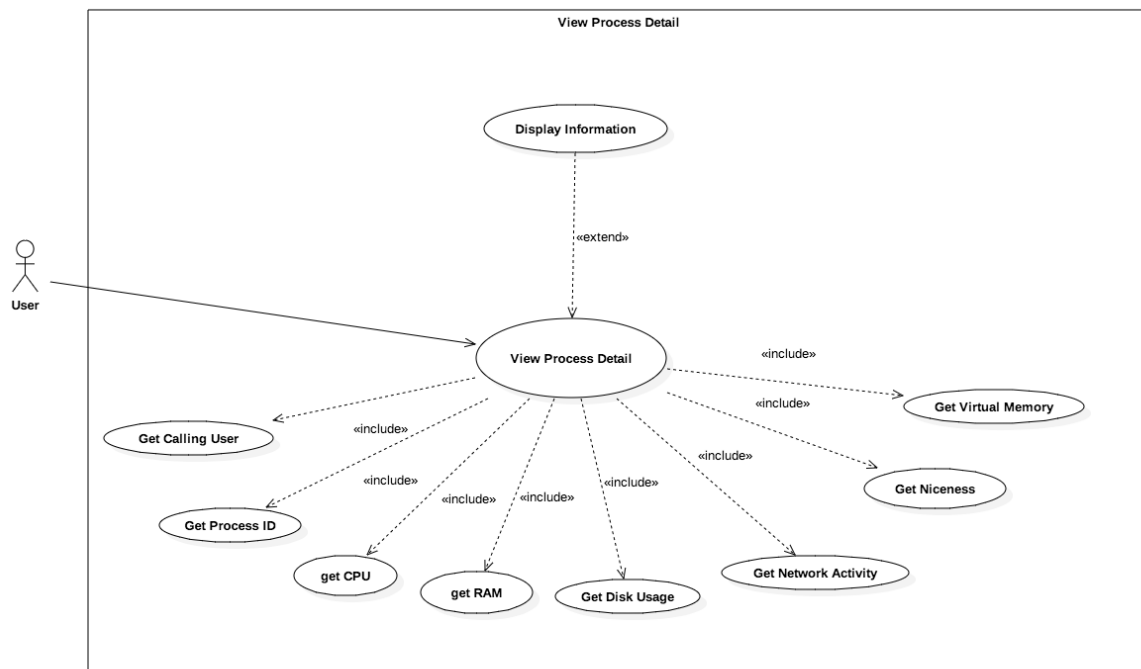
UC04 Diagram



<b>Use Case ID</b>	UC-04					
<b>Use Case Name</b>	View Process History					
<b>Description</b>	User can view process attributes over time on a scatterplot.					
<b>Actors</b>	User					
<b>Pre-Conditions</b>	Database must be ready to populate with process information.					
<b>Post-Conditions</b>	Charts are generated detailing process attributes over time.					
<b>Frequency of Use</b>	Often, though not necessarily every use of the program.					
<b>Flow of events</b>	Actor Action	System Response				
	1. Open Program					
	2. Choose Process History on a single process	Process information starts populating database at interval.				
	3. Chose metric to view.	Scatterplot of data generated and refreshed at interval.				
<b>Variations</b>	3. Metrics: CPU Usage, RAM Usage, Disk Usage, Network Usage, Energy Consumption					
<b>Exceptions</b>						

### UC04 Document

## Use Case 05

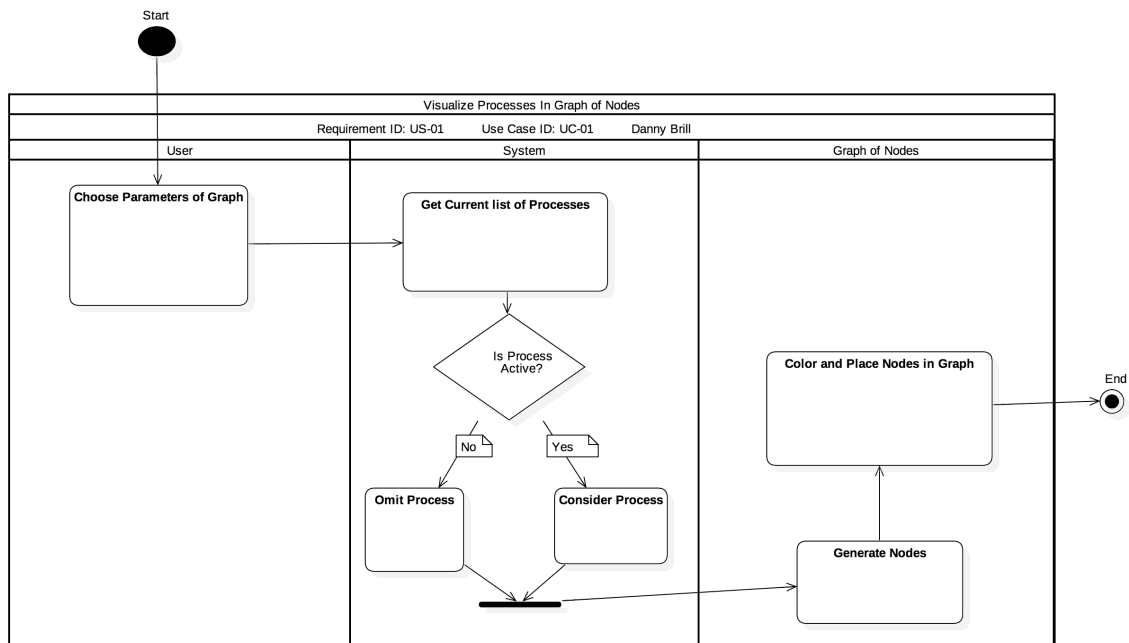


### UC05 Diagram

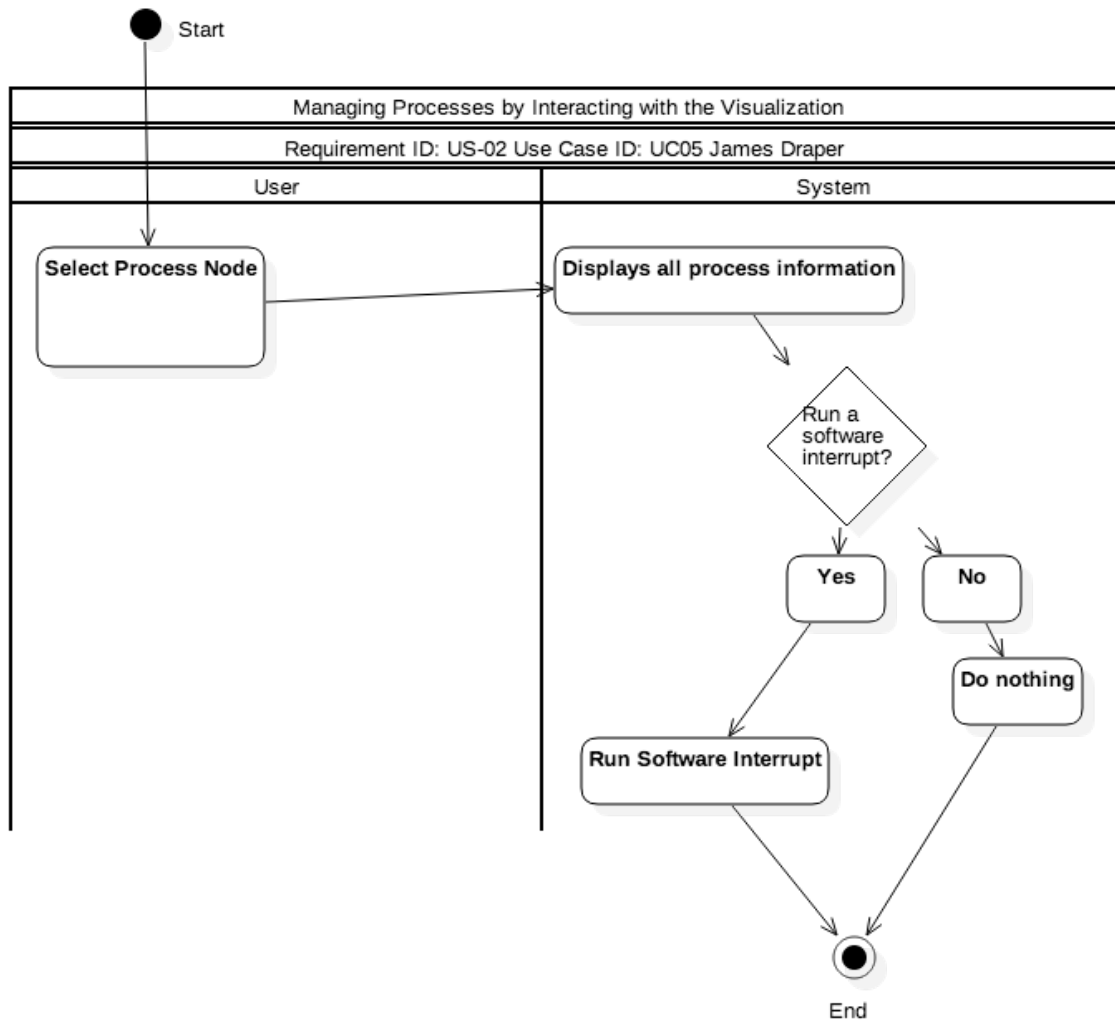
<b>Use Case ID</b>	UC-05					
<b>Use Case Name</b>	View Process Detail					
<b>Description</b>	User can see process details by clicking on a node or list item.					
<b>Actors</b>	User					
<b>Pre-Conditions</b>	User must have privileges to get process information.					
<b>Post-Conditions</b>	Details of a process are displayed when a node or list item is clicked on.					
<b>Frequency of Use</b>	Often, though not necessarily every use of the program.					
<b>Flow of events</b>	Actor Action	System Response				
	1. Open Program					
	2. Choose process node or list item	Process information is gathered and displayed.				
	3. Chose detail view.	Detailed statistics displayed.				
<b>Variations</b>						
<b>Exceptions</b>						

## UC05 Document

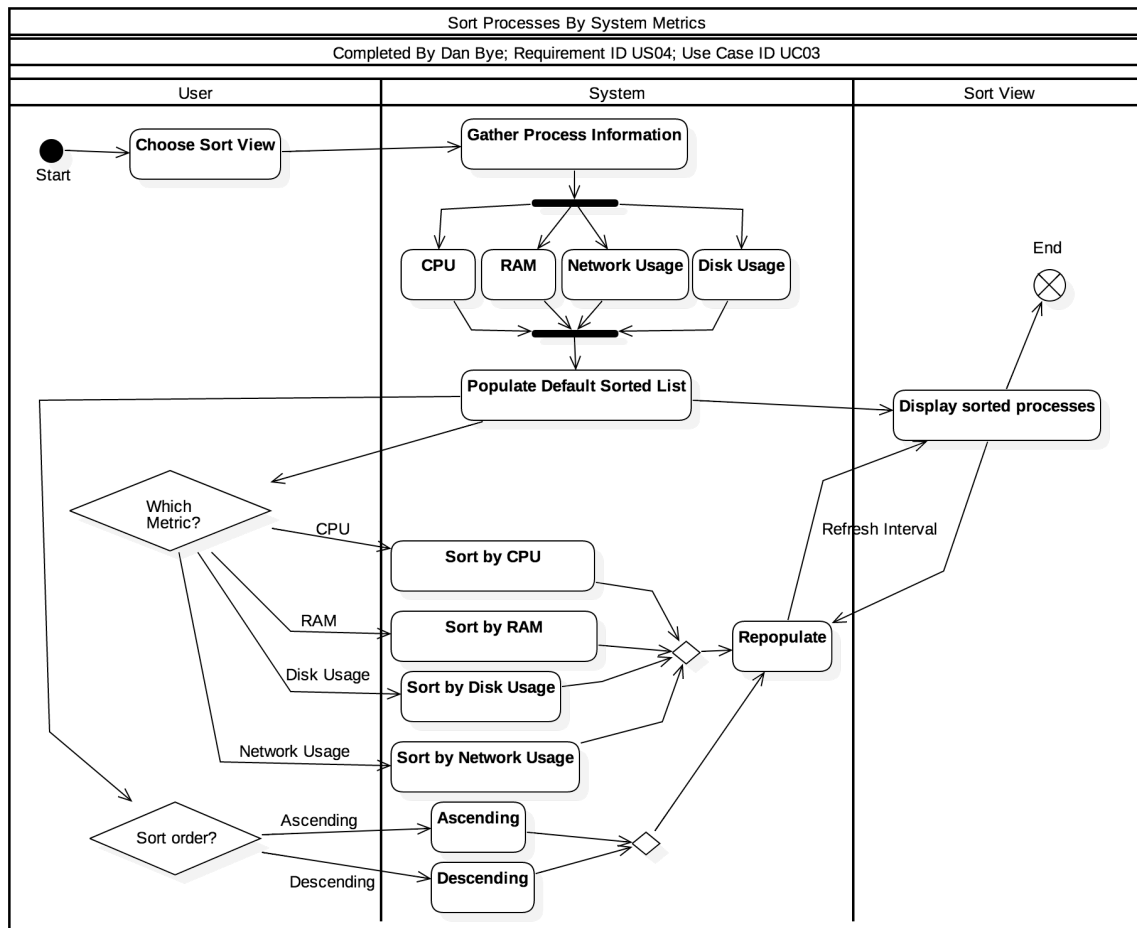
## Activity Diagrams



Danny Brill



James Draper



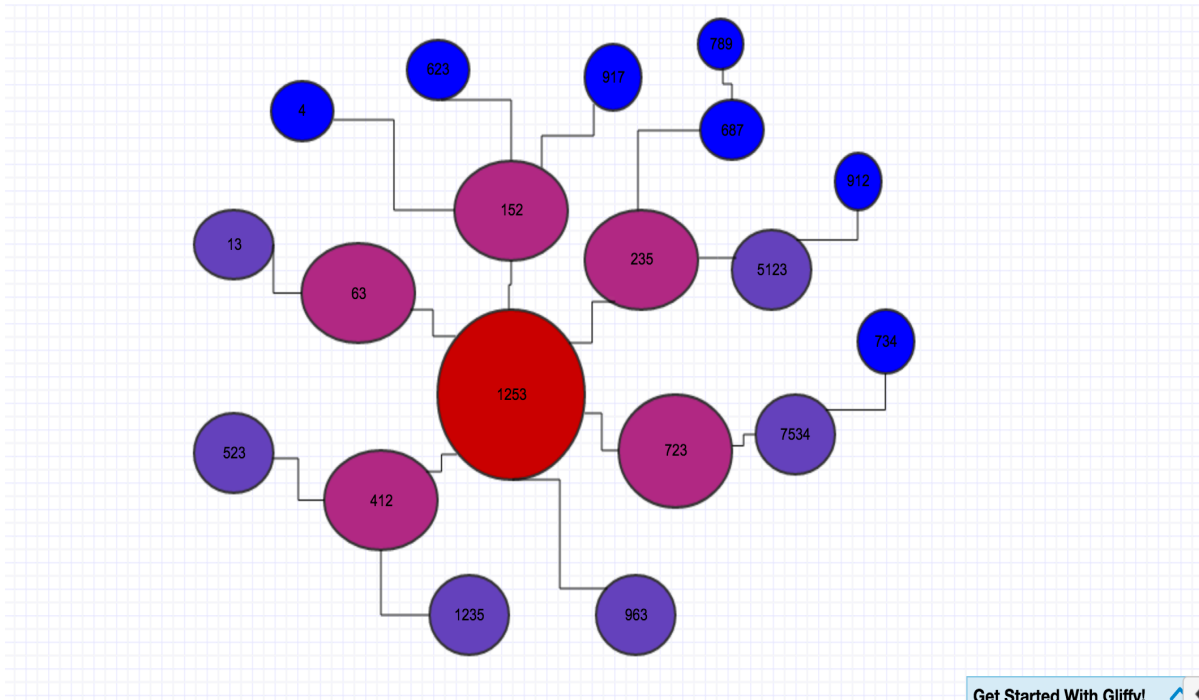
*Dan Bye*

## Data Storage

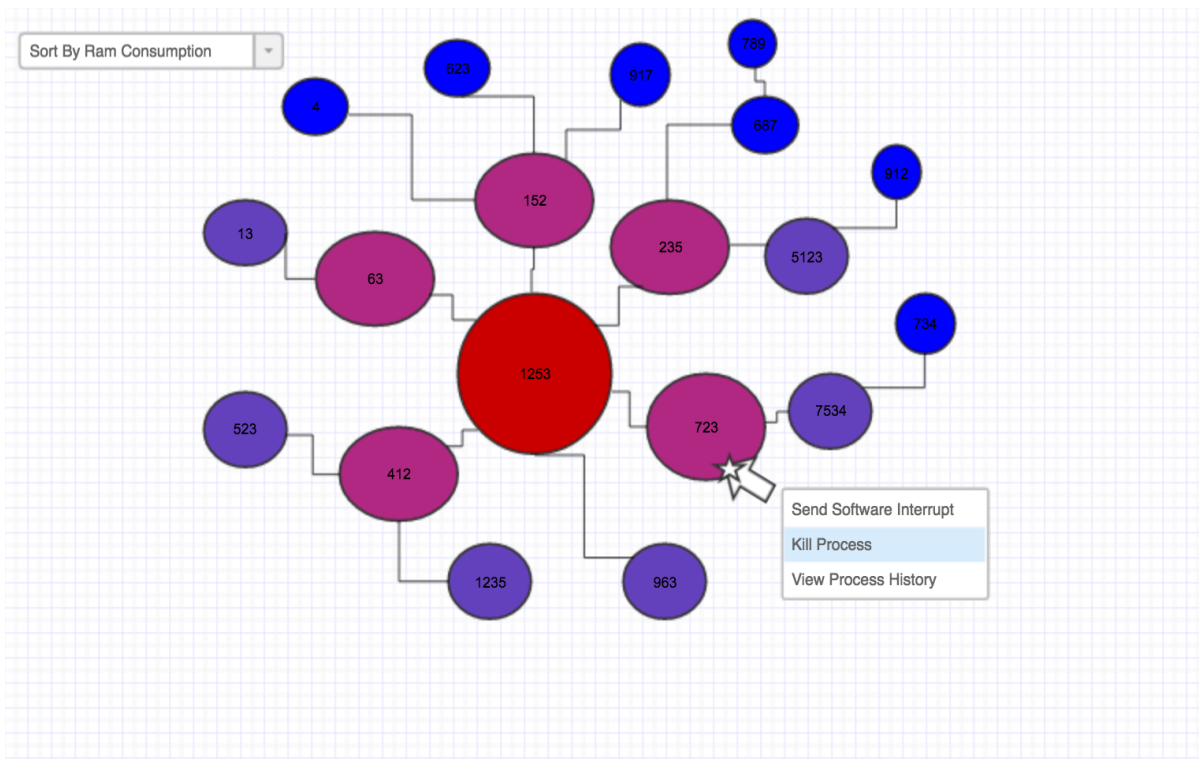
Data storage will apply to process history only. Most statistics will be viewed in near real-time. Process history will either be stored in an array, or an SQL database.

## UI Mockups

**Lists/History graphs were viewed as self explanatory**



*Visualization*



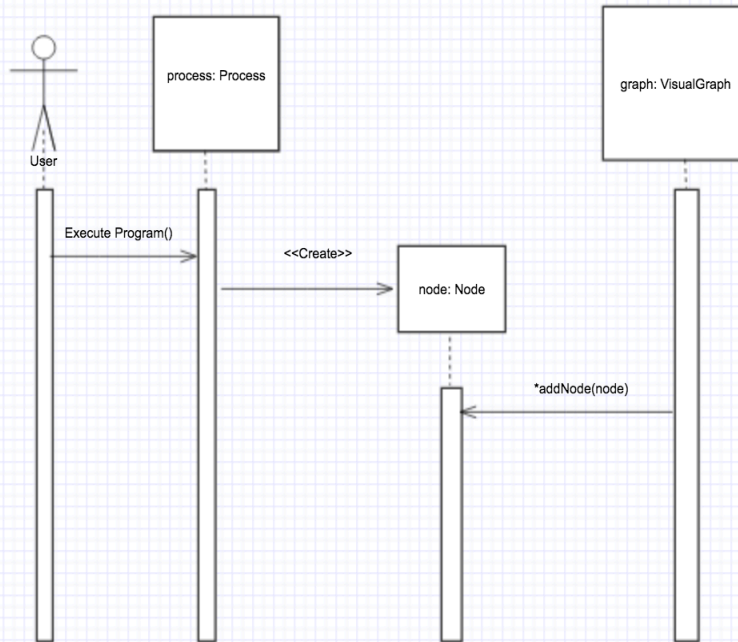
*Process Management*

---

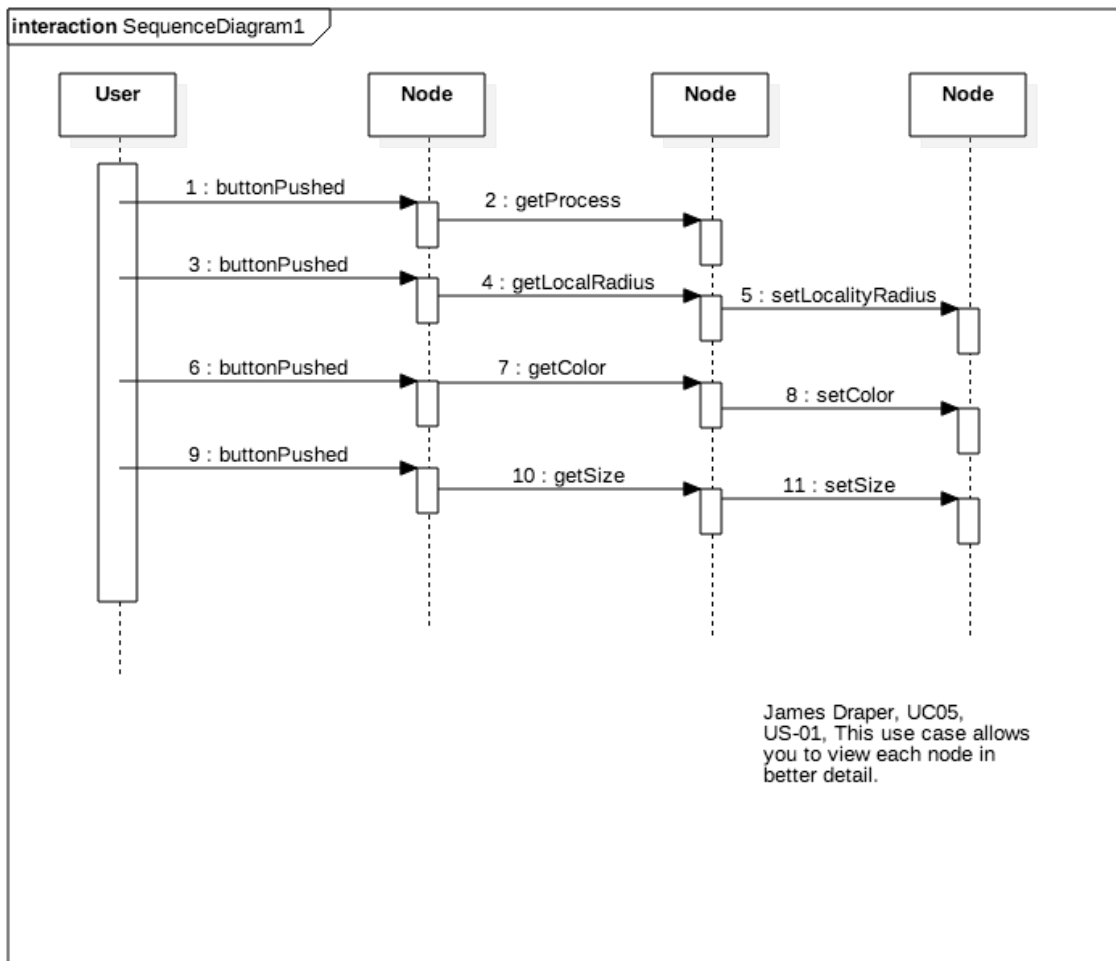
## User Interactions

---

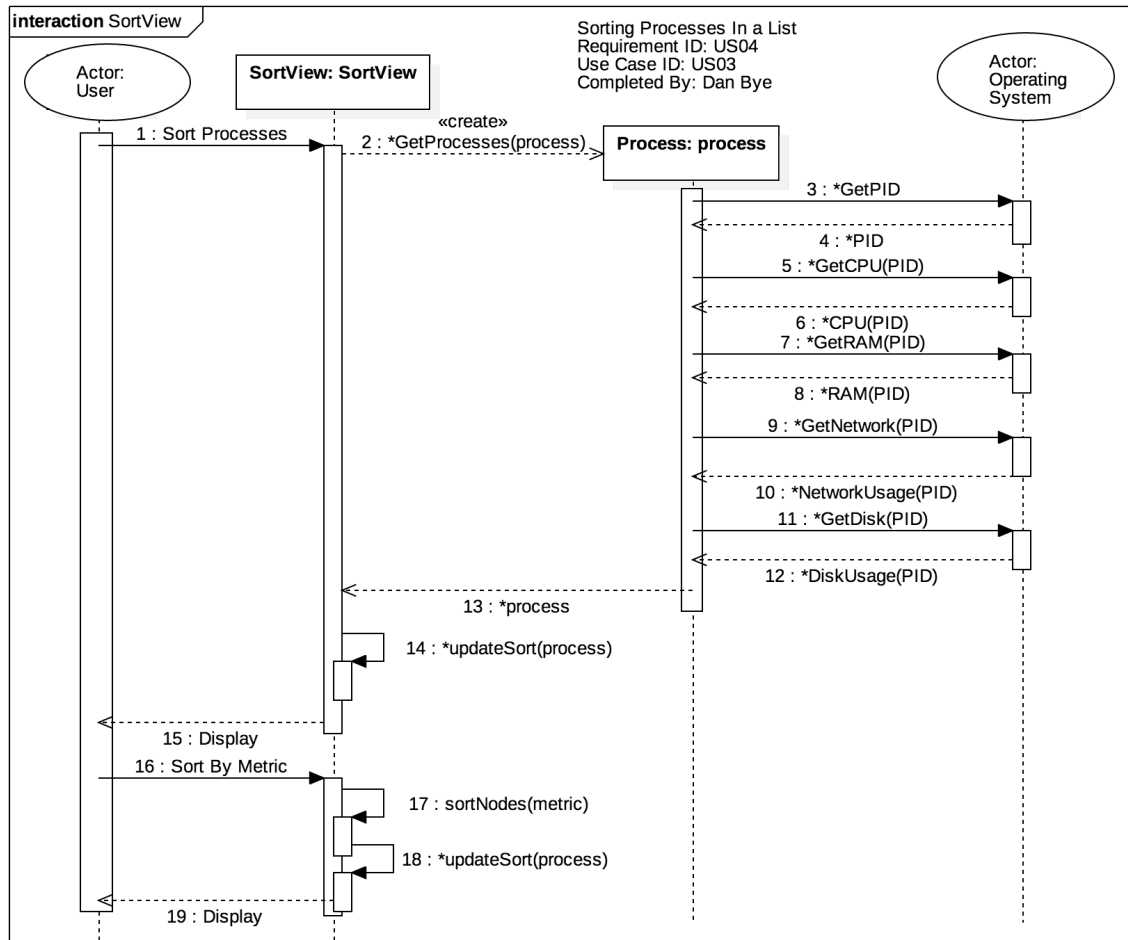
Visual Running Processes as Nodes in a Graph  
Requirement Id: US-01  
Use Case Id: UC-01  
Danny Brill



*Danny Brill*



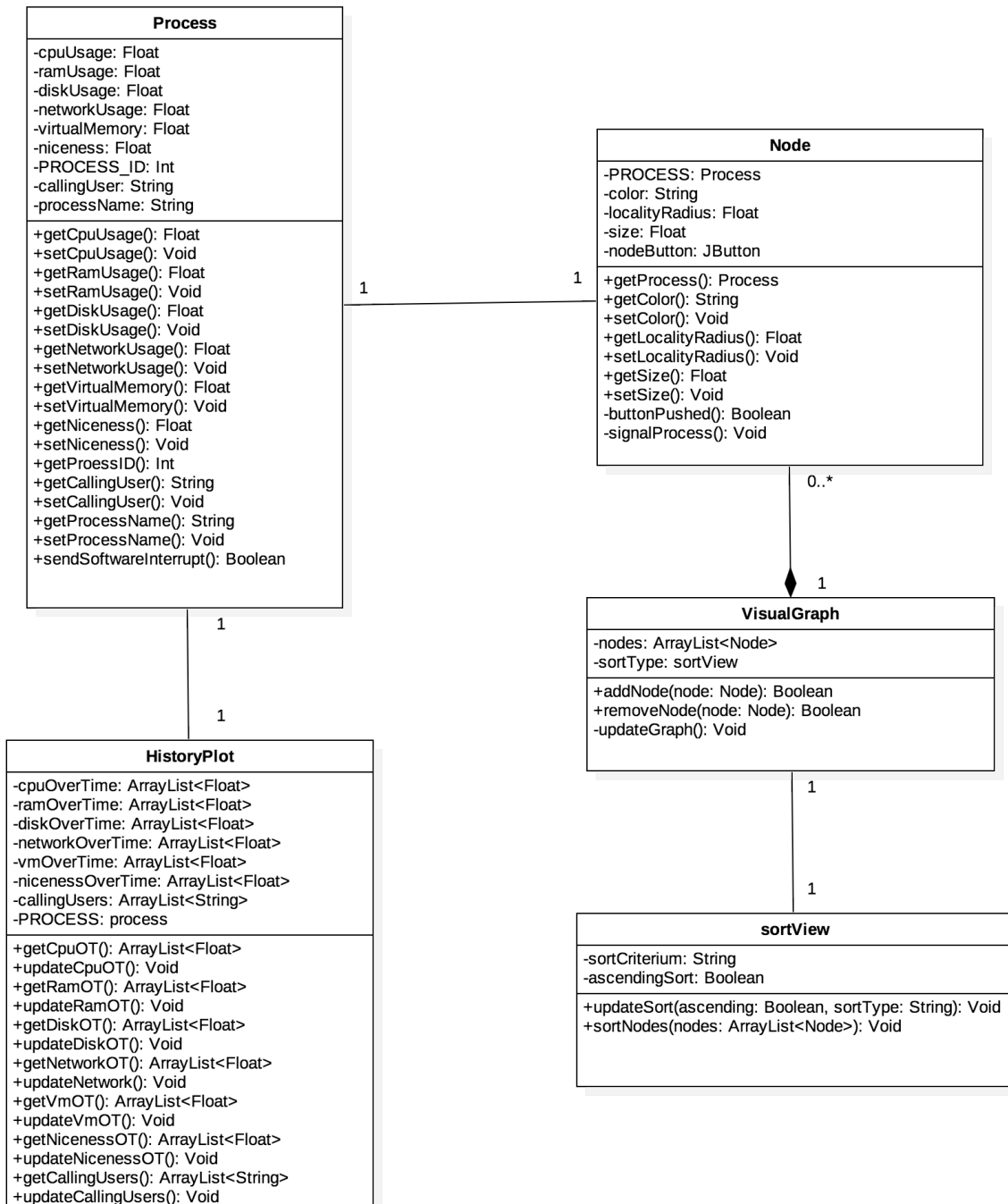
*James Draper*



*Dan Bye*

## Class Diagram





Class Diagram