

# Analytical Visualization Tool for Processing

Nidhin Mahesh

nidhinmahesh1@gmail.com

## Project Abstract

This project aims at making an analytical dashboard for Processing. A visually minimalistic and carefully designed dashboard is essential for effortless insights and important usage information at a glance. Proposed visualization tool consists of *top 3 most used in Processing Development Environment (PDE)*, *most used per type* and *community growth* graphs. Another feature the project will introduce is to send auto-generated emails containing monthly insights to permitted members of Processing.

## Project Description

Analytical Visualization Tool enables contributors of Processing to focus on the user interest to design and develop for those features users use or need the most at the right time. Also, this helps to reduce effort on features which are in the least interest for the users. More data inputs from the software in the future will strengthen tool.

Tool is build for contributors to carefully understand their users on how they are using Processing

## Visual Design

Analytical dashboards provide the user with at-a-glance information used for analysis and decision making and are less time-sensitive and not focused on immediate action. Using visualization to compare one or many values sets is much easier than looking at numbers in the API data. Column and line charts are probably the most used ones.

D3 is inarguably the most dominant and important programming library for creating dynamic, interactive visualizations. AnyChart is a flexible, feature-packed JavaScript (HTML5) based charting library that is free for open-source use (with licensing fees for commercial use). We use AnyChart extensively in this project as it features a wide variety of multivariate charts to kickstart development of dashboards and data visualization projects. Like d3.js library, AnyChart binds data to serialized vector graphics (SVGs) and uses JavaScript to create responsive, interactive visualizations without needing to learn d3 syntax.

## Monthly active users

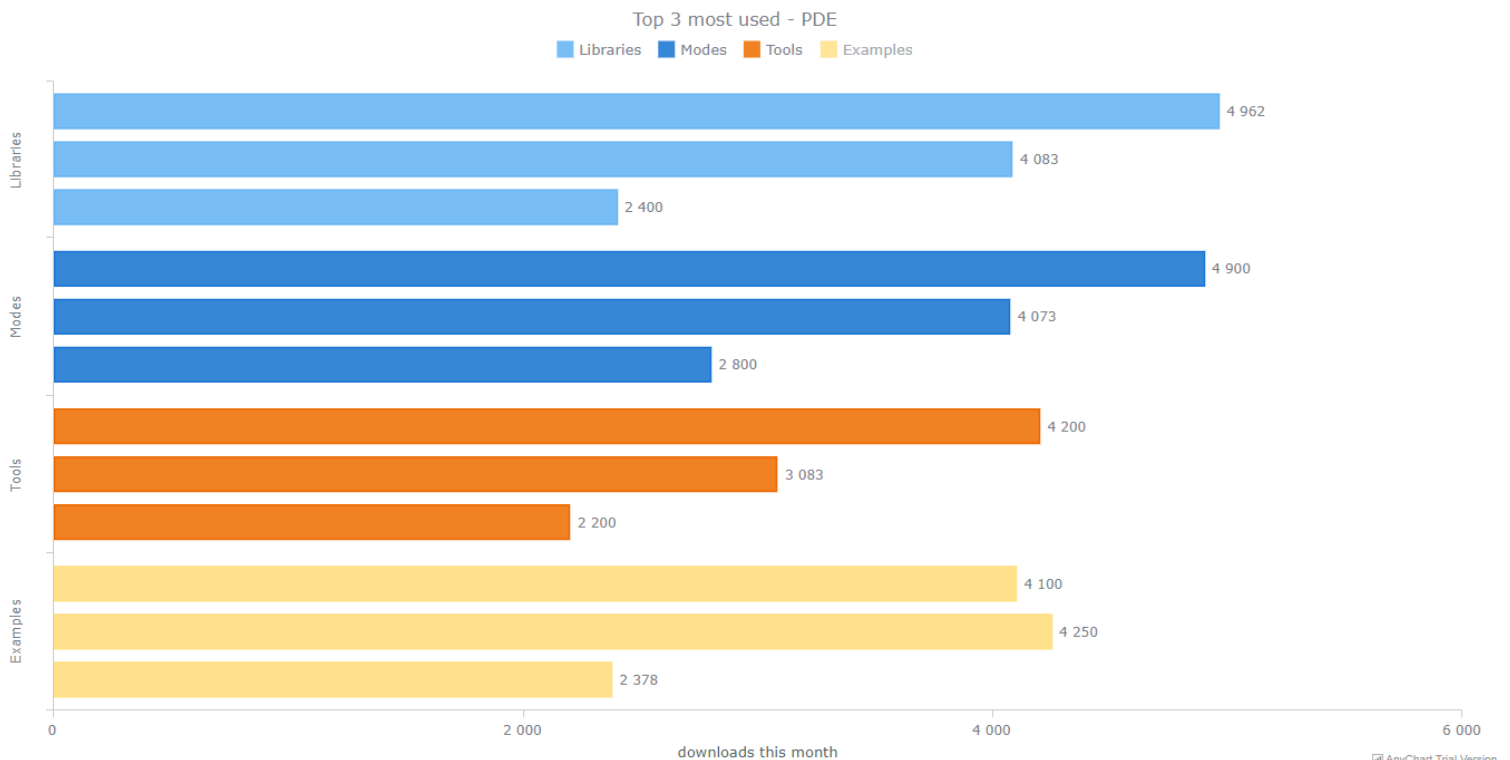
Count of monthly active users of Processing can be obtained by count of most used library in processing. This data gives a rough estimate of the number of active users. Below widget is displayed in the top left corner of the Visualization Tool.

Monthly Active  
Users  
**4962**

*Widget showing monthly active users*

## Top 3 most used in PDE

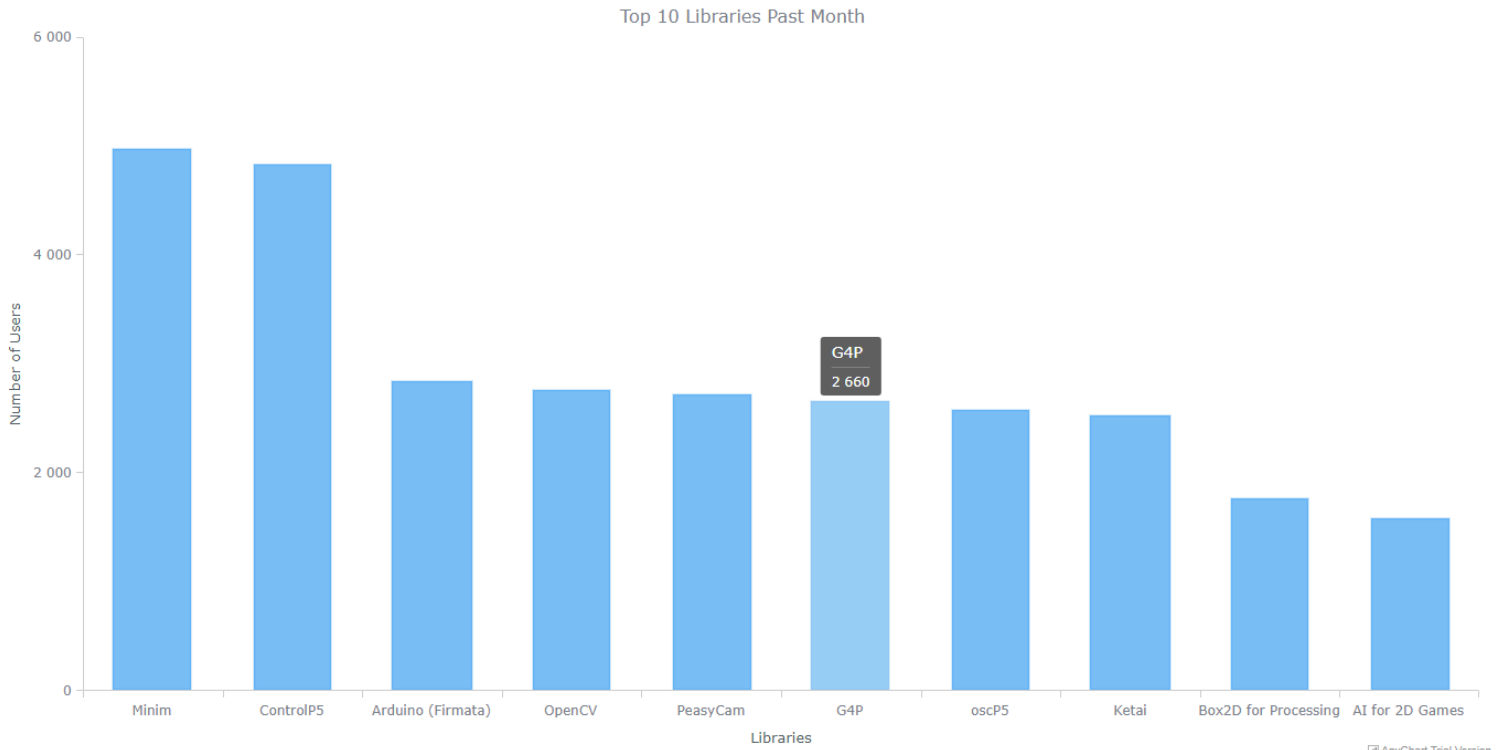
This Multi Series Bar Chart shows the top three most used libraries/modes/ tools and examples past month. This gives a quick sense of people using Processing. On clicking one of these type (Libraries / Modes / Tools / Examples) will load a Column Chart below, which shows top 10 most used of that type in the past month.



*Chart representing top 3 most used*

## Most used per type

On clicking one of these sections will load a column chart below showing top 10 most-used (of that type) with the number of users arranged in descending order. Tooltip associated with the graph shows the accurate count. (users in this case). On clicking on of the bar redirects to the website of that libraries/modes/ tools or examples.



*Selecting a type in above chart expands new chart showing the top 10 most used of that type.*

## Community Growth (Discussion and Q/A)

### Stack Overflow

Processing users, developers and enthusiasts use Stack Overflow for Q/A on programming based doubts and discussions. Keeping track of them is helpful to know the common issues faced by the users and this helps to provide an early solution. Count of the number of questions, answers, and comments on the tag Processing in Stack Overflow is helpful to keep track of the number of users actively engaging with the software.

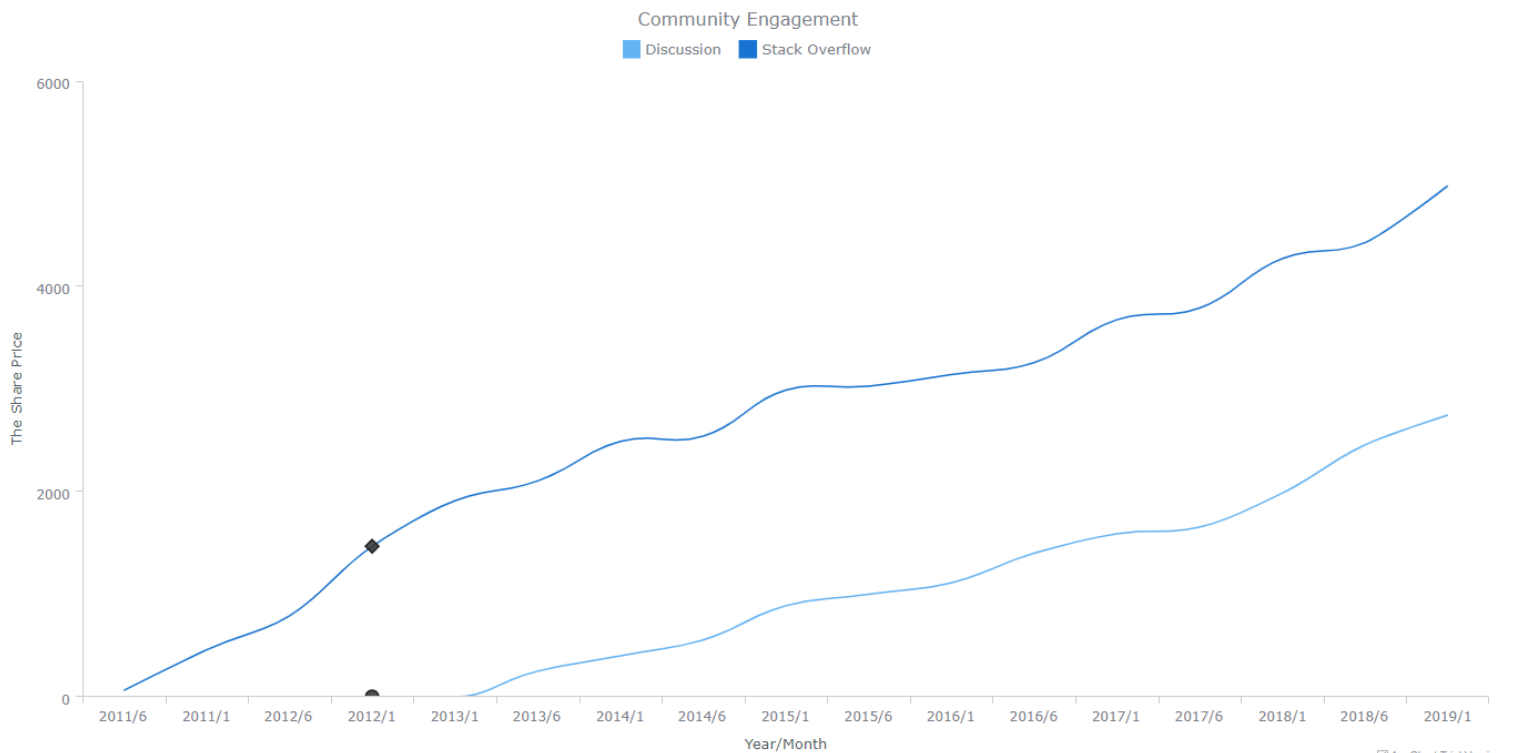
API:

[1]<https://api.stackexchange.com/2.2/tags/processing/info?order=desc&sort=popular&site=stackoverflow>

Above API which gives the count is utilized for the Project.

## Discourse

The number of users in the Processing forum hosted using Discourse can be used alongside the graph of Stack Overflow for better analysis of the community engagement. At present, the count of users (all time and new) is accessible to the admins of Processing forum hosted in Discourse at [processing.discourse.org](https://processing.discourse.org). This can be utilized to form a graph as represented below. The data from Stack Overflow and Discourse can be obtained and represented using Spline chart.



Community Growth Spline Chart

## Storage for Community Growth data.

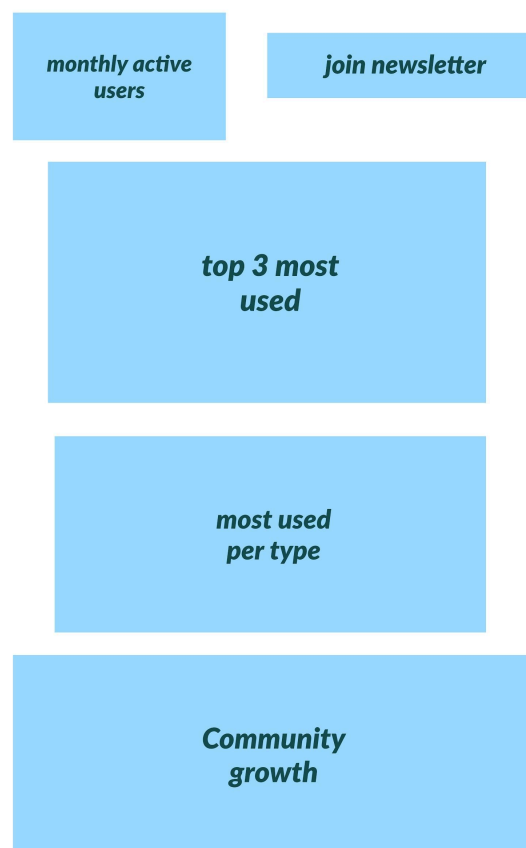
Currently, usage data is updated every 24 hours and last 30 days information are only available with the API. This data is inadequate for showing growth graphs and do not help to predict the growth.

To solve this, I introduced a simple database to store the analysis data for a longer term and provide them for building graph. In this graph, Link [1] is fetched every month and count is stored in the database for showing the graph for Stack Overflow. Database then provide this data through an API to feed this graph. This mechanism can be also expanded in future for other graphs to show usage statistics from much earlier period till the newest.

## Defining the Layout

Layouts are designed keeping in mind to help the flow of viewer and priority of the data for the user. Naturally top left corner of the screen will get more attention so try to position key info from left to right.

Grid layout shows the effective alignment and consistency to create a basic structure (skeleton of our dashboard). As the viewer finish the first row and move to the next chart below it, those data looks connected so that the dependencies that will affect decisions making on one group of information from based on info from another. So viewers do need to go back and forth, create a continuous flow for easy scanning. This also helps for overall scalability and of the design in the future as each component or chart has a consistent structure and making it easier to work with interfaces.



*Shapes represent the layout size and spacing of charts in the tool*

## Monthly insight newsletter

Summary of this dashboard is periodically sent to contributors who are subscribed to this Visualization Tool. Contributors who has access to this tool can also subscribe for monthly insight with their email address. This email is stored in the database. The database have following fields: email, join timestamp, active status (0 or 1), feedback. First day of every month, mail is generated and sent to all active address containing the summary of graphs and link to redirect to the Analysis Tool.

Reply to monthly insight email is stored as feedback for the current usage-statistics. This feedback is store in the database alongside their email address. This is to ensure only newest feedback from the subscriber of this mail is stored.

## Development Process & Timeline

I'm confident that doing this project will help me to understand processing and its user base in depth which in turn helps In narrowing my focus to critical areas for future development of Processing.

My semester exams are scheduled from 6th May 2019 till 21st May 2019. So I will be busy with my academics during that period. I will still keep in touch with my mentor and participate in discussion in the gap between exams. I don't have any other commitments that may affect the GSoC. The detailed timeline of the project is as follows:

*May 21 - 27, 2019 [ 5 days]*

1. During this period, I'll discuss my approach on how to implement the goals I have mentioned in my proposal. I'll make sure to clear all my doubts regarding the implementation of my goals so that when the official coding period begins, I can easily implement the goals mentioned in my proposal.
2. I will also give a thorough reading of the documentation, code and Project Wiki to get an idea of the coding style, a deep understanding of the project and its code. I will also keep myself active in the community and get in touch with the developers and contributors to the project.
3. More importantly, I will use this period to lay a sturdy establishment, so that when the official coding starts I will be able to give my project a kick start.

May 27th Officially coding begins for Google Summer of Code, 2018

*May 14<sup>th</sup> - June 14<sup>th</sup> [4 weeks][40+ hours per week]*

During this period of time, I'll be implementing a few basic features that I've proposed in my proposal. The week wise details of the same are given below -

Period	Task
--------	------

Week 1	<ol style="list-style-type: none"> <li>1. Configure AnyChart and set up to fetch data.</li> <li>2. Write script to fetch user count from most-used library</li> <li>3. design and code <i>Monthly Active Users</i>.</li> </ol>
Week 2	<ol style="list-style-type: none"> <li>1. Implement <i>top 3 most used</i> chart as shown in mockup from the 4 API's.</li> <li>2. Implement <i>Most Used per Type</i> from the API of type selected.</li> <li>3. Fetch link from API and configure redirect to website.</li> </ol>
Week 3	<ol style="list-style-type: none"> <li>1. Design and implement database for <i>Community Growth Graph</i>.</li> <li>2. Write script to fetch data from Stack Overflow periodically and store in database.</li> </ol>
Week 4	<ol style="list-style-type: none"> <li>1. Implement and test <i>Community Growth Graph</i> using the data stored in database.</li> </ol>

### *June 24<sup>th</sup> - June 28<sup>th</sup> Phase I evaluations*

Submit the Phase I evaluations and ask my mentor for any feedback/advice on the work I have done so far. I'll take any feedback/advice seriously and I'll follow them diligently.

### *June 28<sup>th</sup> - July 22<sup>th</sup> [3.5 weeks][40+ hours per week]*

Continue with the implementation of the proposed ideas and add more features to the dashboard as proposed. Completing all the proposed implementations during this period without fail will be given utmost priority so that adequate time for testing and debugging can be ensured.

The week wise report for the same are as follows -

Period	Tasks
--------	-------

Week 5	<ol style="list-style-type: none"> <li>1. Fetch data from Discourse admin and create API to fetch the data.</li> <li>2. Write script to fetch the data periodically (every month) from Discourse admin and update it in database.</li> </ol>
Week 6	<ol style="list-style-type: none"> <li>1. Design and create a new database for getting email address from Visualization tool.</li> <li>2. Write script to fetch the emails and store in DB.</li> </ol>
Week 7	<ol style="list-style-type: none"> <li>1. Implement monthly insight mail to subscribed users.</li> <li>2. Write test for generating and sending monthly insight mail.</li> <li>3. Add option for users to unsubscribe monthly insight.</li> </ol>

### *June 22<sup>nd</sup> - July 26<sup>th</sup> Phase II evaluations*

Give an elaborate report on the new features I've added and the enhancements I've made during the past 3.5 weeks. Ask the mentor for any advice/feedback and follow them diligently.

### *July 26<sup>th</sup> - August 19<sup>th</sup> [3 weeks][40+ hours per week]*

I'll spend time on testing and debugging to ensure superior quality of the code written and to make sure the tool is stable, reliable and user-friendly as proposed. I'll also complete the implementation of all the goals mentioned in my proposal.

Period	Tasks
Week 8	<ol style="list-style-type: none"> <li>1. Configure mail to obtain feedback as a reply from the mail.</li> <li>2. Store the feedback to the database alongside their mail id.</li> </ol>



Week 9	<ol style="list-style-type: none"> <li>1. Write test for feedback.</li> <li>2. Add smooth scrolling</li> <li>3. Fix potential bugs in the layout of web page.</li> </ol>
Week 10	<ol style="list-style-type: none"> <li>1. Ask mentor for reviews and make final changes.</li> <li>2. Document the Visualization Tool.</li> </ol>

### *August 19<sup>th</sup> - August 26<sup>th</sup> [Final Week] [Project Submission]*

The completed project along with its appropriate documentation will be submitted. I'll continue to work on the project after the submission to support the project and be part of this amazing community.

## More about me

### Interest and experience:

I'm a junior computer science undergrad from Amrita Vishwa Vidyapeetham, Amritapuri.

I had worked on projects focusing data analysing for testing apps and games and also core Android in which I had developed various apps for different Open-Source Organisations as a hobby and also as a freelancer. Earlier in 2018, I had also tried a hand with blockchain technology with bitcoin. Furthermore to mention I have previous experience in developing Android Applications with Java and Kotlin. Lately, I shifted the development from Kotlin to Flutter framework so I could deploy apps for both iOS and Android seamlessly.

My main area of interest is Android, Data analytics and Computer Networks.

### Related Projects: (open source)

1. Mobile Games Ab-testing with Cookie Cats – using Pandas. Analyze an AB-test from the popular mobile puzzle game Cookie Cats using Pandas Foundations and Statistical Thinking in Python.

2. Exploring the Bitcoin cryptocurrency market — Data Cleaning, DataFrames. Explore the market capitalization of Bitcoin and other cryptocurrencies using pandas Foundations, Manipulating DataFrames with pandas, Cleaning Data in Python.

## Contribution to Open Source:

For the past two months, I have been learning Processing and reading their codebase to get familiar with development to begin my contribution on Processing. Before that I did various project on Data Analysis and Android development. I use GitHub for all my projects so I'm well versed with git.

I did a beginner patch to Syssters community:

<https://github.com/syssters/powerup-android/pulls?utf8=%E2%9C%93&q=is%3Apr+is%3Aopen+nidhinmahesh>

## Student Information:

- Resume: [link to resume](#)
- GitHub: [@nidhinmahesh](#)
- Portfolio: <https://nidhinmahesh.github.io>
- Email: nidhinmahesh1@gmail.com
- Telephone: (+91) 8075 8646 14 (India)
- Time Zone: GMT+5:30
- Blog (Medium): <https://medium.com/@nidhinmahesh>

## References

Visual design - <http://tonsky.me/blog/github-redesign/>

Scratch - [https://en.scratch-wiki.info/wiki/Scratch\\_Statistics](https://en.scratch-wiki.info/wiki/Scratch_Statistics)