



Borg Collective

GSoC 2023 Proposal

Mentors: [@m3nu](#) [@yfprojects](#) [@Hofer-Julian](#)
[@ThomasWaldmann](#) [@witten](#)

Personal Details:

Name:	Divyansh Singh
Nick:	Divi
Pronouns:	he/him
Github:	diivi
Email:	divyanshanandsingh@gmail.com
University:	Shiv Nadar University, India
Course:	Bachelor of Technology in Computer Science and Engineering
Year of Study:	2nd (Graduating 2025)
Location:	Curugram, India
Timezone:	Indian Standard Time (GMT +5:30)
Matrix:	@divy4n5h:matrix.org
Résumé:	CDrive
Socials:	Twitter LinkedIn



Contents

1. [Abstract](#)
2. [About Me](#)
 - a. [Background and Professional Experience.](#)
 - b. [Availability and Commitments.](#)
3. [Essential Pre-requisites](#)
4. [Previous Contributions](#)
 - a. [Vorta](#)
 - b. [Borg](#)
 - c. [Borgmatic](#)
5. [Projects](#)
 - a. [Restore a database backup to a different database or server](#)
 - b. [Enhance archive table and archive actions](#)
 - c. [Search a file in the diff and extract views](#)
 - d. [Implement Exclude GUI](#)
 - e. [Stretch Goals](#)
6. [Timeline](#)
7. [Expectations from mentors](#)
8. [Post GSoC](#)

Abstract

Borg Collective is a well-known organization that offers a range of Python-based backup tools, including Borg, Borgmatic, and Vorta. Borg is a **file backup tool** that performs tasks like **compression, encryption, authentication, and data deduplication**. Currently, Borg is preparing for the release of Version 2 and is seeking developer support to add a few additional features, including shell completions, documentation updates, and comparisons with other backup tools.

Borgmatic, on the other hand, serves as a command-line interface (**CLI**) **wrapper for Borg** and is designed to help manage Borg repos, settings, and related tasks, including monitoring.

Vorta is a **desktop GUI for Borg** that operates in the user's taskbar and conducts backups in the background. It is officially supported on Linux and MacOS platforms, with its functionality on Windows currently being tested by the community.



As a contributor to the Borg Collective for over **two months**, I have gained a concrete understanding of the organization's backup tools. Having worked on all three of these projects, I'll be a strong candidate for Google Summer of Code (GSoC) 2023. During the summer, my main focus would be Borgmatic and Vorta. I'd love to work towards the following goals to help improve these projects:

1. Make Vorta more **beginner-friendly**, which will help **retain new users** and showcase its capabilities.
2. Enhance Vorta and Borgmatic by adding additional features to **further empower users and increase their satisfaction**. I have already added some features to these projects and seek to continue doing so.
3. Work on **small tasks** to support Borgmatic and Vorta in catching up with Borg's latest developments and assist the maintainers of Borg with **the v2 release**.
4. I aim to provide the same level of excellent support and advice to developers interested in contributing to these projects as I have received from the Borg community. **Interacting with the community** is also a key objective for me.

About Me

Background and Professional Experience

I'm a software developer who is passionate about building impactful user-facing software that people use and love. I'm particularly drawn to JavaScript & Python, which I've used for over three years. I have always been impressed by their simplicity and widespread use, which made them an easy choice for me to start building my projects with.

For me, the most rewarding part of being a software developer is seeing people use and appreciate the software I've built. It is deeply fulfilling to know that something I've created is having a positive impact on people's lives. This is also why I value open-source projects like Borg a lot, because I know the features I add to them will be used by thousands of people worldwide.



Being involved in open-source development has exposed me to a multitude of new fields and best practices in the software development world that I may not have learned otherwise. Through my experiences, I got the opportunity to interact with individuals from various parts of the globe, and in doing so, I've gained valuable insights and knowledge.

Furthermore, I'm grateful for the senior engineers I've collaborated with during my previous internships. Their guidance has allowed me to develop a deeper understanding of writing production code that thousands of people use, DevOps, TDD, and many other crucial aspects of software development. I look forward to continuing this learning with the experienced mentors at Borg Collective.

My major past open source contributions have been to the [OpenFoodFacts](#) project, in which I made [several important commits](#), and the [FreeCodeCamp](#) Organization, where I was mentioned in the list of the [top code contributors](#), because of an [accessibility PR](#) I made.

I have interned as a software engineer at product-based companies like [Shipsy](#) and [Scenes By Avalon](#), and service-based companies like [Devlance](#). This has allowed me to gain hands-on experience in diverse environments and develop a broad skill set and a deeper understanding of the software development lifecycle.



Availability and Commitments

I have no other obligations during the summer. My summer break from the university will commence on **May 12th, 2023** and conclude on **August 18th, 2023**.

This will enable me to dedicate myself full-time, working 40 hours per week, to the project for at least three months. Even after my classes begin, I will not have any exams until October, permitting me to continue working on the project with a flexible schedule of around 30 hours per week.

I am confident I can complete asynchronous work within this time frame, respond promptly to my mentors, and remain engaged with the Borg community. Borg Collective is the only organization I'll be applying for this GSoC.

Date	May-2023
1	End-Semester Examination
2	End-Semester Examination
3	End-Semester Examination
4	End-Semester Examination
5	Buddha Purnima
6	End-Semester Examination
7	
8	End-Semester Examination MBA Q4/Q8 End Term exam
9	End-Semester Examination MBA Q4/Q8 End Term exam
10	End-Semester Examination MBA Q4/Q8 End Term exam
11	End-Semester Examination
12	
13	
14	
15	
16	Last Viewing Day

Current Semester ends on 11th May

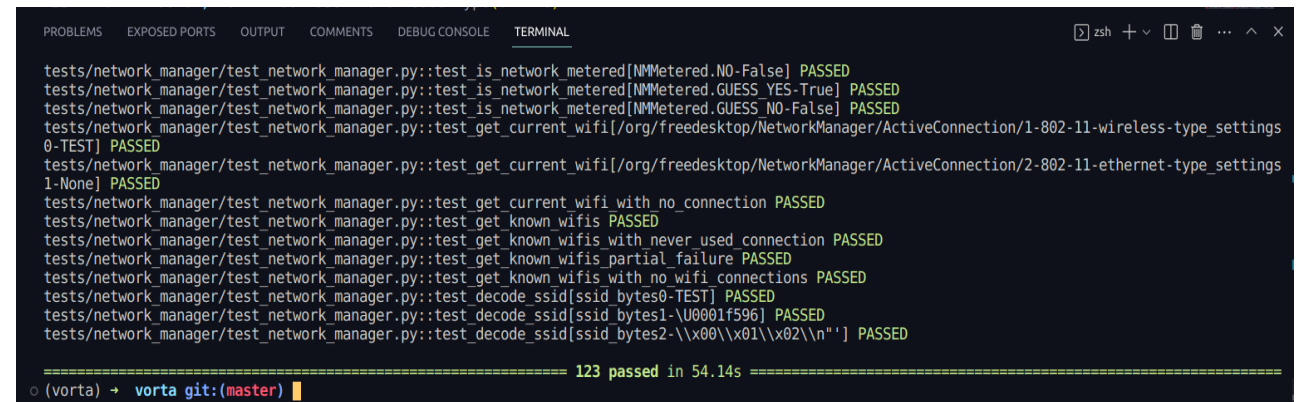
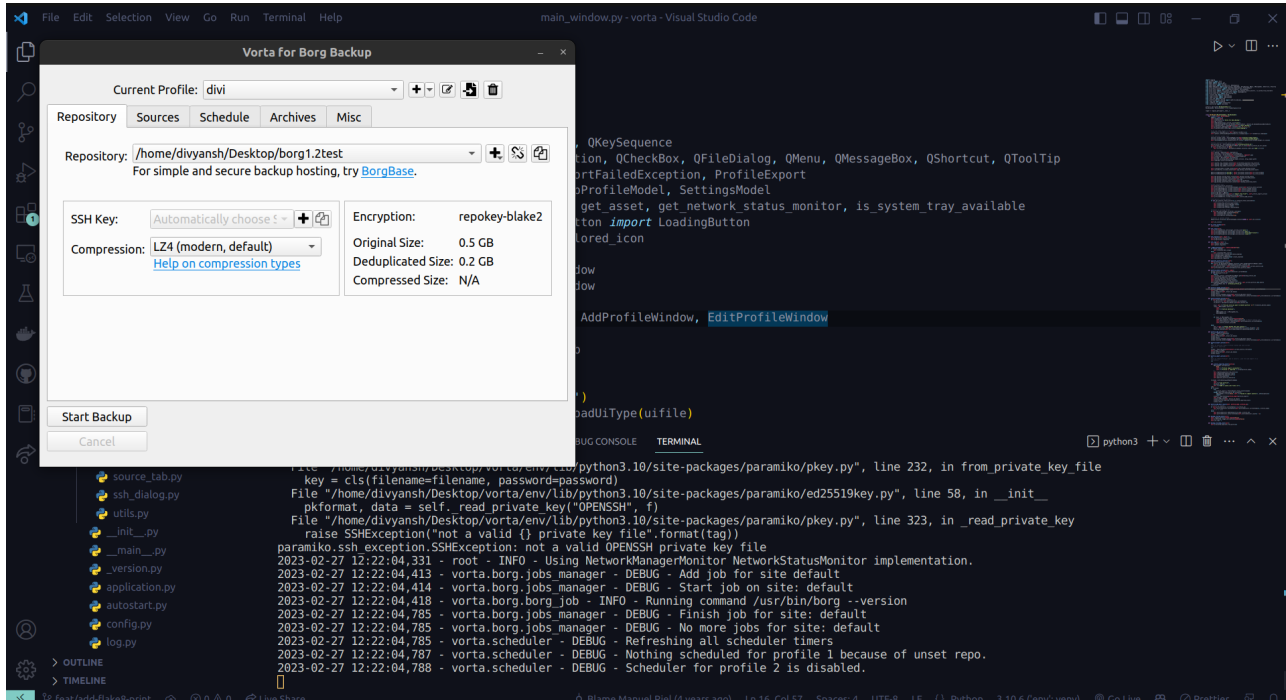
Date	Aug-22
1	
2	
3	
4	
5	
6	
7	
8	UG Course Registration begins for Monsoon 2022 (4th Year)
9	Muharram
10	UG Course Registration begins for Monsoon 2022 (2nd & 3rd Year)
11	Raksha Bandhan
12	PG/Ph.D. registration begins for Monsoon 2022
13	
14	
15	Independence Day
16	
17	
18	Start of classes for all students

Next Semester begins mid-August

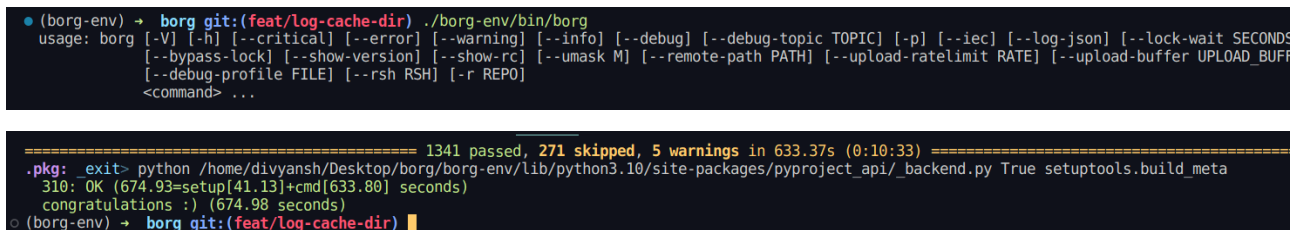
Essential Pre-requisites

I have read [Google's contributor guide](#).

I can build Vorta from source and run all tests:



I can build Borg from source and run all tests:





I can build Borgmatic from source and run all tests:

```
• → borgmatic git:(feat/add-sqlite-support) python3 -c "import borgmatic; print(borgmatic.__file__)"
/home/divyansh/Desktop/borgmatic/borgmatic/__init__.py

TOTAL    2454    0    100%
76 files skipped due to complete coverage.
===== 1041 passed, 2 warnings in 3.28s =====
py310: commands[1]> black --check .
All done! 🌟🍌🌟
```

Previous Contributions

Being one of the few contributors who have made consistent and important contributions to **two projects** from the Borg Collective (Borgmatic and Vorta) for more than two months, I have gained an in-depth understanding of their source code. I have **covered many files** with the PRs I created and understood how these tools work under the hood. Not only have I made code contributions, but I also contributed to **discussions and code reviews** whenever I could, which is also an essential aspect of open source. Additionally, I have contributed by [creating](#) and [editing designs](#) for a few Vorta interfaces and icons due to my keen interest in **UI & UX design**. I believe that these contributions make me a **robust and flexible** candidate for GSoC, who can work on **two projects and several tasks concurrently**:

Vorta

[#1606 remove paramiko](#)

[#1609 add a link to the logs folder in borg warnings](#)

[#1612 Replace print with logging in application.py](#)

[#1613 descriptive messages when no sources are selected to backup](#)

[#1621 add settings for files list views](#)

[#1637 add profile name to log messages](#)

[#1656 block after vorta --create and log status](#)

[#1658 settings to allow new networks and show notif when a network is disallowed](#)

[#1664 feat: quick mount](#)

[#1665 feat: assign names to repos](#)

[#1666 ci: add ruff for print checks](#)

Borg

[#7388 feat: add cache dir to --debug](#)



Borgmatic

[#50 feat: add dump-restore support for SQLite databases](#)

[#52 fix: remove extra dark mode styles](#)

[#53 feat: add optional check for the existence of source directories](#)

[#54 feat: file:// URLs support](#)

[#55 fix: rephrase error when running from config](#)

[#56 fix: no error on database backups without source dirs](#)

[#57 feat: tag repos](#)

[#58 docs: copy to clipboard support](#)

[#60 feat: allow defining custom variables in config file](#)

Projects

I found many ideas on the list enticing, but because of the 350-hour time span for a GSoC project, I have decided to work on the following four ideas during the summer. My primary focus will be Borgmatic and Vorta to minimize mentor and context switching, but I also intend to work on tasks related to Borg. If time permits, or I get stuck somewhere and find myself waiting for a reply from my mentors, I would also like to tackle the stretch goals listed at the end. I intend not to be involved with the Borg Collective solely for the duration of GSoC. Instead, I aspire to continue contributing to the project beyond the program's timeline.

1. Restore a database backup to a different database or server.

Borgmatic supports creating database backups and restoring those backups to the same database. However, this doesn't support use cases like:

1. **Running a test restore to another database or server** without impacting the production database.
2. In the event of a database loss, **restoring a database to a newly created replacement server with a different hostname and/or credentials**.

I will work on adding support to Borgmatic's restore action to override the restored database name, hostname, port username, password, etc. and plumb those values through to the restore commands. Additionally, I will manually test this on all database types Borgmatic supports, update the existing documentation, and add tests for these features.

I have gone through the Borgmatic database dump and restore hooks while working on [#50 feat: add dump-restore support for sqlite databases](#) and have gained



valuable experience that will help me in tackling this issue. There has already been a [discussion](#) regarding this, which I can follow. This is an important and frequently requested addition to Borgmatic, and I am excited to implement it.

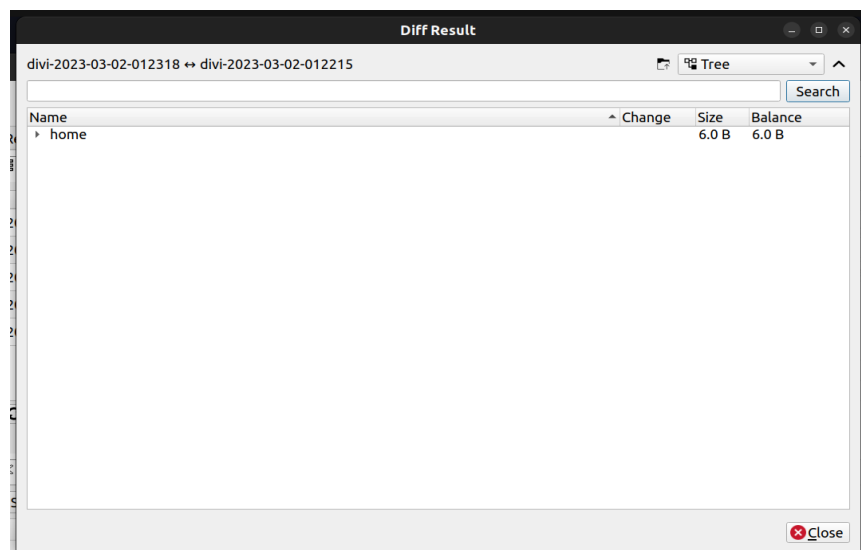
2. Enhance the archive table and archive actions.

Vorta comprises a table listing the archives in a Borg repository. The Archives tab also includes a button to rename an archive. Currently, the button opens a dialog with a text field to rename the archive and save changes. I will work on removing this button and adding functionality to **rename an archive inline** (by double-clicking on the table cell). I will also **add another column to the table that shows whether an archive was created by the scheduler or by the user manually**. This will involve **changing the Archive model** and digging deeper into the Vorta source code, which will enable me to tackle future issues confidently during GSoC. I'll also work on **hiding the compact button** from this tab for users running borg versions < 1.2. Another important UX issue I wish to tackle is the inability to **refresh multiple archives at once**.

I have looked at the source code and know which files to change. I have discussed a base implementation with mentors already and believe I'll be able to complete this task in a few weeks.

3. Search a file in the diff and extract views.

Vorta has a feature for comparing two archives (backup snapshots) and one for extracting specific files from an archive. The interface for both features is quite alike, consisting of a file list and view mode options, such as Tree, Simplified Tree, and Flat List. Nonetheless, **finding a specific file within the returned results remains a challenge for users, who must sift through all the files presented in the views**. To address this issue, my objective with this task is to **introduce a search bar to the diff and extract view** interfaces, like the one below:





This will involve editing the current **diffresult.ui** and **archivedialog.ui** files, and adding the search bar for them. Also, my work will involve working with the current **FileTreeModel** class and adding filtering capabilities to it. Once all this is done, I'll add **unit tests** to cover all the changes that I made. This is an exciting task considering the time it will save for the end user and the impact I'll get to make.

I also have an idea in mind for **advanced search capabilities** that I proposed in the discussion related to this issue [here](#).

4. Implement Exclude GUI.

While creating backups, users often want to ignore some large folders, like folders with cache files and environment-specific packages/modules. Currently, Vorta users exclude files by manually adding text rules, such as `node_modules/*`, to the sources tab.

However, **this method isn't user-friendly and makes for a bad UX**. New users have a hard time getting started with setting up exclusion patterns, and even experienced users have difficulty entering precise and powerful rules.

To address this issue, I intend to **design and implement a graphical user interface (GUI) that allows users to select files to exclude**. Additionally, I plan to **incorporate pre-defined exclude presets**, such as the ability to exclude common macOS cache files, to make the feature even more user-friendly. This capability is an essential part of Vorta and is already included in other backup software applications.

To develop this feature, I will **research which files should be excluded and how they should be grouped**. I will also collaborate with my mentors to determine the optimal **way to store these exclusions**, preferably as a file or data structure, so that other contributors can modify them as well.

I have already started work on the design and have received some community feedback to work on. The complete design can be found [here](#). I'll try to finalize the design with my mentors during the community bonding period so that I just have to work on its development during the coding period.

This task will be the longest one I will do this summer and will help me be better acquainted with Vorta source code. I'll work on adding a **new UI with QtCreator**, **design the storage structure for all presets and exclusion rules**, and **implement the logic underneath for passing these rules to Borg**. This will increase my knowledge of the Qt framework and give me more confidence to work on other tasks.

I have attached my current progress with the designs to the next page. This will keep improving as I get more suggestions on the [discussion thread](#) related to this issue, and I will have a final design ready soon.



Patterns Tab

Some instruction text here maybe to tell the user what they are doing, ignore the non SFPro font, more lorem ipsum

Search Term

- ☐ Folders with .nobackup
- ☐ Python extra folders
- ☐ Temporary Cache Files
- ☐ .-lock.*
- ☐ *.*~
- ☐ */Library/*/Logs/
- ☒ *~
- ☐ #/nix/
- ☐ /private
- ☒ /tmp/

Placeholder

Cancel Save

Raw Tab

Some instruction text here maybe to tell the user what they are doing, ignore the non SFPro font, more lorem ipsum

Search Python Cache F

```
# Temporary files
*/tmp/*
*/src/tmp/
*/home/Downloads/
tmp*
*/can/list
*/a/lot/of/paths/here/*

# Packages/Modules
node_modules/*
env/
pip*
*/can/list

# Python Cache Files
__pycache__
```

Placeholder

Cancel Save

Add Pattern

Presets:

- None
- MacOS Cache Files
- This is an alternate backgr...
- Modules/pacakges
- Python Cache
- Temporary Files

Patterns:

```
# Temporary files
*/tmp/*
*/src/tmp/
*/home/Downloads/tmp*
*/can/list
*/a/lot/of
paths/here/*
```

Cancel Add

Placeholder

Cancel Save



5. Stretch Goals

If time permits, I'd like to work on tasks that improve various Borg-related repositories. This is also something I'd like to work on post-GSoC.

- a. **Improve Vorta test coverage** by adding unit tests and optimizing current tests wherever possible.
- b. **Allow users to perform backups to multiple repos from one profile**, as discussed in [#942](#).
- c. Add support for **Bootstrapping a Borgmatic restore from nothing**.
- d. I aim to **close old issues and reduce the number of open issues** in the Borgmatic repository. I plan on contributing to code reviews and becoming a core **contributor** too. Specifically, I'd like to work on issues [#418](#), [#610](#) and [#375](#).

Timeline

I have planned a timeline that aligns with the default 12-week duration of GSoC. The timeline includes deliverables for each checkpoint, which are scheduled weekly or every two weeks. It is important to note that the timeline may be subject to changes based on my mentor's discretion, as progress is assessed after each checkpoint.

Community Bonding Period (May 04 - May 28)

- Discuss and finalize the designs for my projects with the Vorta community.
- Learn more about the features and functionality of Vorta, Borgmatic and Borg.
- Check the development environment/setup with mentors and ask for suggestions to improve my workflow.
- Discuss with my mentors a communication medium and time for weekly catch-ups and discussions related to my project.
- Get pending pull requests merged and work on a few good first issues.
- Discuss top-level approaches for all Projects with my mentors and make sure I am ready to start coding on May 29th.
- Dive deeper into the Borg community, interact with users and maintainers from all three projects, and get to know my mentors better.

Coding Period

(May 29 - Jun 12) Week 1,2

- Work on Project 1 - Restore a database backup to a different database or server.
- Manually test the restoring functionality by restoring one database's data to another.



- Write tests for all the databases supported by Borgmatic - PostgreSQL, MySQL, MongoDB and SQLite.
- Start working on Project 2 - Enhance the archive table and archive actions on the side, splitting it into multiple PRs. Solving the more straightforward issues first, like removing the Compact button for Vorta users running Borg version < 1.2.

(Jun 13 - Jun 20) Week 3

- Finish Project 2 - Enhance the archive table and archive actions.
- Work on comparatively longer issues like detecting whether an archive was created by the user or the scheduler and showing it in the UI.
- Write tests for the changes that I made.
- Get all PRs merged after a review from my mentors.

(Jun 21 - July 3) Week 4,5

- Start working on Project 3 - Search a file in the diff and extract views.
- Edit the UI file according to the outcome of my discussion with the community and mentors.
- Write code for filtering and finding files, implement direct string equality search first, and connect it to the UI via a search button.

(July 4 - July 10) Week 6

- Finish work to add basic searching functionality to the diff and extract views.
- Write/Update tests for all the functionality that I added.
- Discuss further additions, like filtering capabilities for these views, with my mentors to implement them later.

Midterm Evaluation Period (July 10 - July 14)

(July 11 - July 25) Week 7,8

- With more than half of the work done, start working on Project 4 - Implement Exclude GUI.
- Discuss if and how the task can be split into smaller tasks to accomplish it more efficiently and effectively.
- Implement the Design for the Exclude Dialog (a tabbed view) and replace the current exclude input box with a button to open it.
- Discuss the implementation details and differences between Presets and custom rules with my mentors.
- Research which presets/rules should be shipped with Vorta and how they should be grouped, and get these groups reviewed by my mentors and the community.
- Check how the exclusions are passed to Borg and implement a data structure to store all exclusion presets and rules.
- Write code to handle adding/deleting and editing exclusions.
- Concurrently, write unit tests for my additions to the codebase.



(July 26 - Aug 10) Week 9,10

- Start with the last phase of Project 4 - Implement Exclude GUI.
- Finish any tasks remaining from the previous weeks.
- Write code for passing exclusions to Borg and check manually if everything works with various folders.

(Aug 11 - Aug 28) Week 11,12

- Prepare for the final week by getting all open PRs merged.
- Write integration tests for Project 4 - Implement Exclude GUI.
- Discuss future work and any improvements that can be made to my projects, like adding filtering capabilities to the search functions in the diff and extract views.
- Write a blog post documenting my GSoC experience with the Borg collective, in addition to the mandatory weekly PSF blog, to summarize my journey and potentially attract future contributors.

Final Evaluation Period (Aug 28 - Sept 4)

September 2023 & Beyond

- Start working on Stretch Goals, fixing good first issues in the various Borg collective repositories.
- Fix any bugs that might arise from my contributions.
- Write additional tests for Vorta if needed.
- Work on other exciting issues, like [separate logs for each task](#) for Vorta.
- Work on new Borgmatic issues, especially issues tagged with the “design finalised” label and become a core contributor.



Expectations from mentors

Contributing to the Borg Collective has been a great experience so far. I couldn't have asked for better help and support to get familiar with the organization, and make my first contributions in no time! The Vorta, Borgmatic, and Borg mentors have been incredibly helpful in guiding me through technical challenges and providing me with constructive criticism to enhance my coding skills and become a better open-source contributor. I would greatly appreciate the following during the GSoC period from my mentors:

1. As a mentee, I would appreciate **assistance in comprehending existing code**, which would help me make informed decisions during the summer.
2. I would like to have **regular weekly or fortnightly meetings or check-ins** with my mentors to discuss my work progress and adjust my timeline if necessary.
3. **Constructive feedback and insightful suggestions** from my mentors would aid in my growth as a software engineer.
4. My **mentors' professional experience and knowledge** would be invaluable in enhancing my professional development.

Post GSoC

I wish to **continue contributing** to the Borg Collective projects with the same enthusiasm and zeal even after GSoC. Having worked with all three projects under the Borg Collective, I am confident in my ability to **comprehend and explain the code to other contributors**, particularly after completing GSoC. This will help bring new contributors to the project and improve their developer experience too.

My primary objective during GSoC is to become a valuable member of the Borg team and assist future candidates in their contributions! In addition to the guidance provided by my mentors, I have received a lot of support from the Borg community, which has helped me better understand the project. I aspire to **stay involved with this community**, make new connections and **advocate the use of Borg** among other communities I am involved with.



I will **raise awareness about the project** in my college and the various tech meetups I attend regularly. I will keep solving issues in all three projects and maybe even work on some **unselected GSoC tasks** post-summer, especially some of the [Stretch Goals](#) I've listed in my proposal.

I have been the **most active contributor** during the application discussion period, not only to Vorta, but Borgmatic too. Making **20+ important pull requests** to these projects has proved my dedication and seriousness about them. I strongly believe that open-source organizations should seek not only good GSoC contributors but also **valuable community members that help maintain and support the project for a long time**, and I aspire to be precisely that.