

CSC510 GROUP 24 PROJECT 2 SUBMISSION

ARCHISMITA GHOSH
DHARANI GUDA
PAUL MACNICHOL

<https://github.com/CSC510-Software-Engineering-Fall-2024/SimplyClip>

Total self evaluation score - 221 (Two Hundred Twenty One)

Workload is spread over the whole team (one team member is often X times more productive than the others.)	two	
but nevertheless, here is a track record that everyone is contributing a lot)	three	https://github.com/CSC510-Software-Engineering-Fall-2024/SimplyClip/commits/Group24DeveloperBranch/
Number of commits	two	44
Number of commits: by different people	two	Dharani Guda : 15 Archismita: 26 Paul MacNichol: 3
Issues reports: there are many	two	https://github.com/CSC510-Software-Engineering-Fall-2024/SimplyClip/issues
Issues are being closed	two	https://github.com/CSC510-Software-Engineering-Fall-2024/SimplyClip/issues?q=is%3Aissue+is%3Aclosed
Docs : Doco generated, format not ugly	three	https://docs.google.com/document/d/1qqRibAaK9KKA27T0R6YKSUtNoSLTU-2gSeFjilyCxSE/edit?tab=t.0
Docs : what : point descriptions of each class/function (in isolation)	three	https://docs.google.com/document/d/1Wmt-RiAomIBqRhwKKvRo-hxQuDnZH6tYHkLfO

		CkyA4I/edit?tab=t.0
Docs : how : for common use cases X,Y,Z mini-tutorials showing worked examples on how to do X,Y,Z	two	readme.md
Docs: why: docs tell a story, motivate the whole thing, deliver a punchline that makes you want to rush out and use the thing	three	POSTER ATTACHED ABOVE
Docs: short video, animated, hosted on your repo. That convinces people why they want to work on your code.	three	readme.md
Use of version control tools	three	
Test cases exist	two	
Test cases are routinely executed	two	https://github.com/CSC510-Software-Engineering-Fall-2024/SimplyClip/actions
Issues are discussed before they are closed	Three	
Chat channel: exists	three	https://discord.com/channels/@me/1275530116507635713
Test cases: a large proportion of the issues related to handling failing cases.	two	readme.md
Evidence that the whole team is using the same tools: everyone can get to all tools and files	three	
Evidence that the whole team is using the same tools (e.g. config files in the repo, updated by lots of different people)	three	
Evidence that the whole team	three	

is using the same tools (e.g. tutor can ask anyone to share screen, they demonstrate the system running on their computer)		
Evidence that the members of the team are working across multiple places in the code base	three	
Short Release Cycles	two	
The file .gitignore lists what files should not be saved to the repo. See [examples]i(https://github.com/github/gitignore)	three	https://github.com/CSC510-Software-Engineering-Fall-2024/SimplyClip/blob/Group24DeveloperBranch/.gitignore
The file INSTALL.md lists how to install the code	Two	
The file LICENSE.md lists rules of usage for this repo	three	
The file CODE-OF-CONDUCT.md lists rules of behavior for this repo; e.g. see example	three	
The file CONTRIBUTING.md lists coding standards and lots of tips on how to extend the system without screwing things up; e.g. see example	three	
Readme - Video	three	
Readme - DOI badge: exists.	two	
Readme - Badge showing style checkers	two	
Readme - Badge showing code formatters	two	
Readme - Badges showing syntax checkers.	two	

Readme - code coverage tools	two	
Readme - Badges showing any other Other automated analysis tools	two	
Does your website and documentation provide a clear, high-level overview of your software?	two	
Does your website and documentation clearly describe the type of user who should use your software?	three	
Do you publish case studies to show how your software has been used by yourself and others?	zero	
Is the name of your project/software unique?	three	
Is your project/software name free from trademark violations?	three	
Is your software available as a package that can be deployed without building it?	two	
Is your software available for free?	three	
Is your source code publicly available to download, either as a downloadable bundle or via access to a source code repository?	three	
Is your software hosted in	three	

<p>an established, third-party repository like GitHub (https://github.com), BitBucket (https://bitbucket.org), LaunchPad (https://launchpad.net) or SourceForge (https://sourceforge.net)?</p>		
<p>Is your documentation clearly available on your website or within your software?</p>	<p>two</p>	
<p>Does your documentation include a "quick start" guide, that provides a short overview of how to use your software with some basic examples of use?</p>	<p>three</p>	
<p>If you provide more extensive documentation, does this provide clear, step-by-step instructions on how to deploy and use your software?</p>	<p>two</p>	
<p>Do you provide a comprehensive guide to all your software's commands, functions and options?</p>	<p>two</p>	
<p>Do you provide troubleshooting information that describes the symptoms and step-by-step solutions for problems and error messages?</p>	<p>three</p>	
<p>If your software can be used as a library, package or service by other software, do you provide</p>	<p>one</p>	

comprehensive API documentation?		
Do you store your documentation under revision control with your source code?	three	
Do you publish your release history e.g. release data, version numbers, key features of each release etc. on your web site or in your documentation?	one	
Does your software describe how a user can get help with using your software?	three	all details in readme.md
Does your website and documentation describe what support, if any, you provide to users and developers?	two	all details in readme.md
Does your project have an e-mail address or forum that is solely for supporting users?	two	all details in readme.md
Are e-mails to your support e-mail address received by more than one person?	one	
Is your project's ticketing system publicly visible to your users, so they can view bug reports and feature requests?	three	https://github.com/CSC510-Software-Engineering-Fall-2024/SimplyClip/issues
Is your software's architecture and design modular?	Two	
Does your software use an accepted coding standard	two	

or convention?		
Does your software allow data to be imported and exported using open data formats? e.g. GIF, SVG, HTML, XML, tar, zip, CSV, JSON, NetCDF, or domain specific ones	Three	
Does your software allow communications using open communications protocols? e.g. HTTP, FTP, XMPP, SOAP over HTTP, or domain-specific ones	one	
Is your software cross-platform compatible? e.g. does it run under two or more of Windows, Unix/Linux and Mac OS X, or can be used from within two or more of Internet Explorer, Chrome, Firefox and Safari?	Three	
Does your software adhere	Two	

to appropriate accessibility conventions or standards?		
Does your documentation adhere to appropriate accessibility conventions or standards?	Two	
Is your source code stored in a repository under revision control?	Three	
Is each source code release a snapshot of the repository?	One	
Are releases tagged in the repository?	Two	
Is there a branch of the repository that is always stable? (i.e. tests always pass, code always builds successfully)	Three	
Do you back-up your repository?	Three	

<p>Do you provide publicly-available instructions for building your software from the source code?</p>	Three	
<p>Can you build, or package, your software using an automated tool?</p> <p>e.g. Make (https://www.gnu.org/software/make/), ANT (http://ant.apache.org/), Maven (https://maven.apache.org/), CMake (https://cmake.org/), Python setuptools (https://pypi.python.org/pypi/setuptools), or R package tools (https://cran.r-project.org/doc/manuals/r-devel/R-exts.html)</p>	One	
<p>Do you provide publicly-available instructions for deploying your software?</p>	Three	

Does your documentation list all third-party dependencies?	Two	
Does your documentation list the version number for all third-party dependencies?	Two	
Does your software list the web address, and licences for all third-party dependencies and say whether the dependencies are mandatory or optional?	One	
Can you download dependencies using a dependency management tool or package manager?* e.g. Ivy (http://ant.apache.org/ivy/), Maven (https://maven.apache.org/), Python pip (https://pypi.python.org/pypi/pip) or setuptools (https://pypi.python.org/pypi/setuptools), PHP Composer	Three	

<p>(https://getcomposer.org/), Ruby gems (https://rubygems.org), or R PackRat (https://rstudio.github.io/packrat/)</p>		
<p>Do you have tests that can be run after your software has been built or deployed to show whether the build or deployment has been successful?</p>	Three	
<p>Do you have an automated test suite for your software?</p>	Three	
<p>Do you have a framework to periodically (e.g. nightly) run your tests on the latest version of the source code?</p>	Three	
<p>Do you use continuous integration, automatically running tests whenever</p>	Zero	

changes are made to your source code?		
Are your test results publicly visible?	Two	
Are all manually-run tests documented?	Three	
Does your project have resources (e.g. blog, Twitter, RSS feed, Facebook page, wiki, mailing list) that are regularly updated with information about your software?*	Zero e.g. release announcements, publications, workshops, conference presentations	
Does your website state how many projects and users are associated with your project?	One	
Do you provide success	Zero	

stories on your website?		
Do you list your important partners and collaborators on your website?	Three	
Do you list your project's publications on your website or link to a resource where these are available?	Zero	
Do you list third-party publications that refer to your software on your website or link to a resource where these are available?	One	
Can users subscribe to notifications to changes to your source code repository?	Zero	
If your software is developed as an open source project (and, not just a project developing open source software), do you have a governance	Two	

model?		
Do you accept contributions (e.g. bug fixes, enhancements, documentation updates, tutorials) from people who are not part of your project?	Three	
Do you have a contributions policy?	Three	
Is your contributions' policy publicly available?	Three	
Do contributors keep the copyright/IP of their contributions?	Zero	
Does your website and documentation clearly state the copyright owners of your software and documentation?	Two	
Does each of your source code files include a copyright statement?	Two	

Does your website and documentation clearly state the licence of your software?	Three	
Is your software released under an open source licence?	Three	
Is your software released under an OSI-approved open-source licence?	Three	
Does each of your source code files include a licence header?	Three	
Do you have a recommended citation for your software?	One	
Does your website or documentation include a project roadmap (a list of project and development milestones for the next 3, 6 and 12 months)?	Three	
Does your website or	Zero	

documentation describe how your project is funded, and the period over which funding is guaranteed?		
Do you make timely announcements of the deprecation of components, APIs, etc.?	Zero	



SimplyClip



Scan to Learn More!

Your Ultimate Clipboard Workspace for Seamless Productivity and Focus!

ABOUT US

SimplyClip is the ultimate productivity tool for research students and power users. Effortlessly collect, organize, and prioritize text snippets from different tabs, eliminating distractions and the need to switch windows. Enjoy seamless content management with features such as text summarization, URL-saving, text search, and one-click exporting to CSV or DOC. Enhance your efficiency and maintain your focus with SimplyClip!

KEY FEATURE

Advanced machine learning model for text summarizations



Copying images into extension clipboard



Word count of each list item available



Highlighting filtered search text



UI Dark Mode



Future Enhancements

Cross-browser support

Expand compatibility to other browsers such as Firefox and Edge.

Shortcut Customization

Allow users to create custom keyboard shortcuts for frequently used actions within the extension.

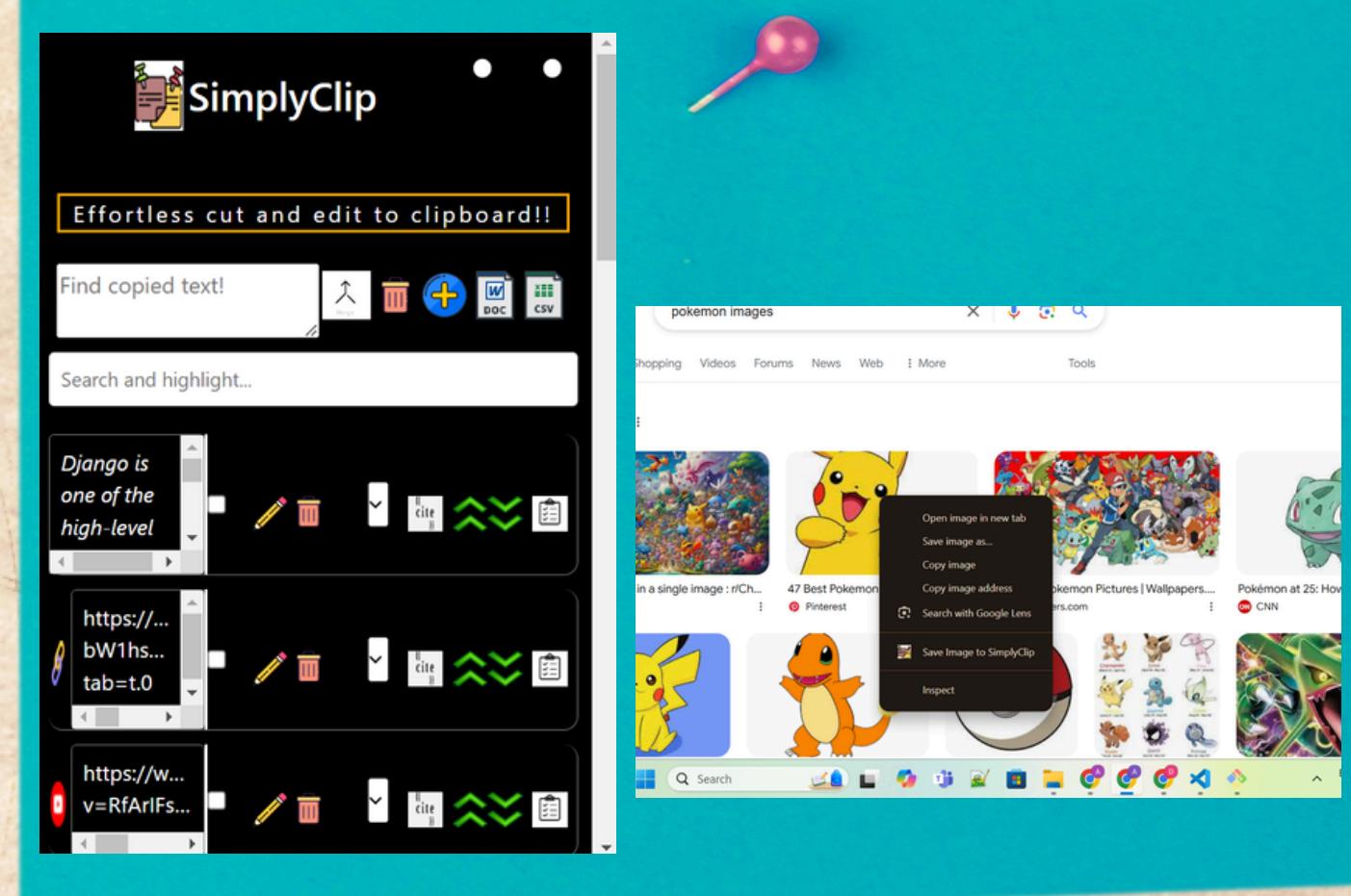
API Integration

Provide an API for other apps to access and manage clipboard data programmatically.

110+ Test Cases

100% Statement Coverage

97.41% Branch Coverage



BUILT WITH

