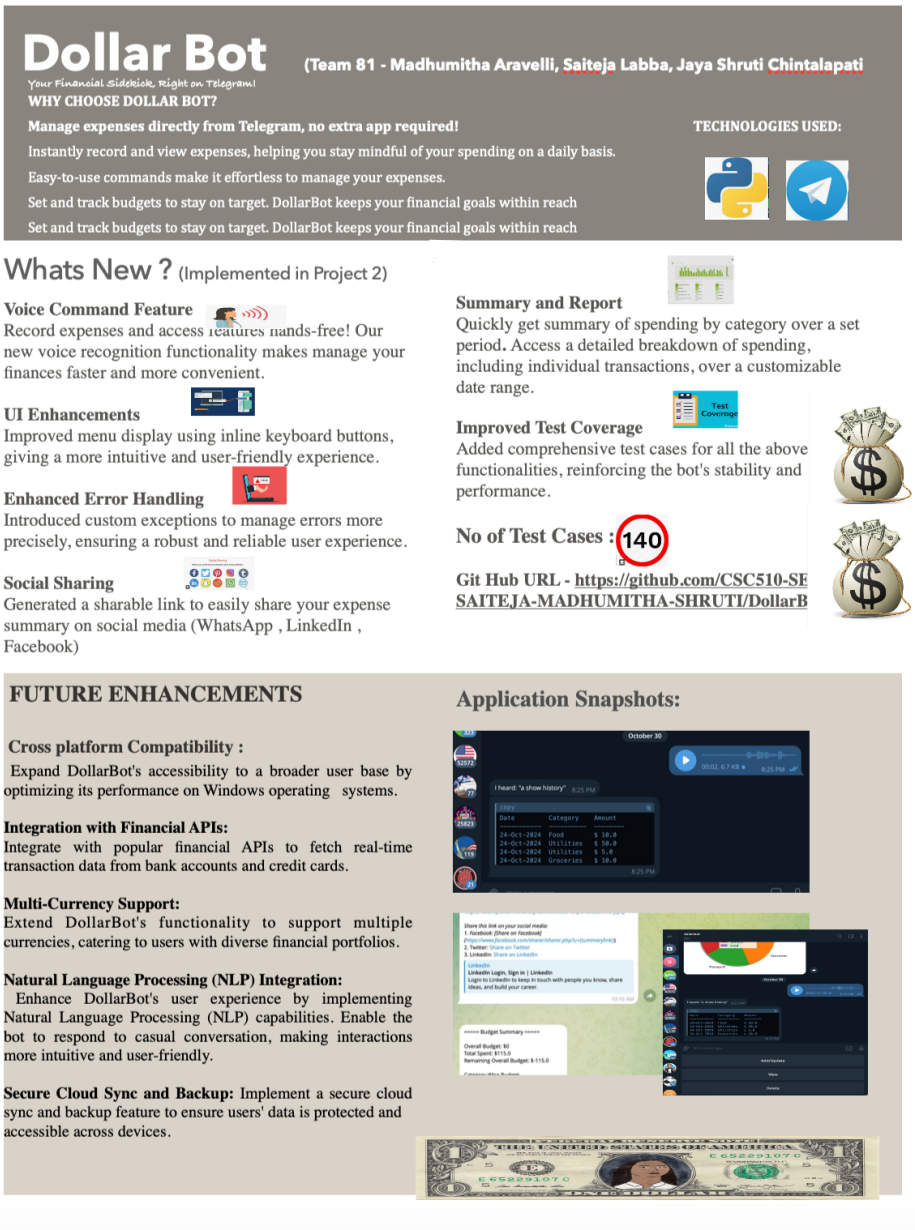
SE Project 2 :

Group No : 81

Name : Madhumitha Aravelli

Team members : Saiteja Labba , Madhumitha Aravelli , Jaya Shruti Chintalapati

Repo link - https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot



RUBRIC

Github Project Link	**https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot**		
Column 1	**Column 2(self evaluation)**	**Evidence**	Comments
	Madhumitha Score : 104 /105		
Workload is spread over the whole team	3	https://github.com/orgs/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/projects/1/views/1	
but nevertheless, here is a track record that everyone is contributing a lot)	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/pulse	
Number of commits	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/graphs/contributors?from=9%2F28%2F2024	
No.of commits: by different people	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/graphs/contributors?from=9%2F28%2F2024	

Issue Reports: Many	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/issues?q=is%3Aissue+is%3Aclosed	
Issues are closed	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/issues?q=is%3Aissue+is%3Aclosed	
Docs: doco generated, format not ugly	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/blob/main/README.md	
Docs: what: point descriptions of each class/function (in isolation)	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/tree/main/code	docstrings explaining each class and function are included in the corresponding code files (.py)
Docs: how: for common use cases X,Y,Z mini-tutorials showing worked examples on how to do X,Y,Z	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot#information_desk_person-use-cases	
Docs: why: docs tell a story, motivate the whole thing, deliver a punchline that makes you want to rush out and use the thing	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot#dollarbot---because-your-financial-future-deserves-the-best	

Docs: short video, animated, hosted on your repo. That convinces people why they want to work on your code.	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot#dollarbot---because-your-financial-future-deserves-the-best	Linked short video as a link in Readme.md
Use of version control tools	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/tree/main	
Test cases exist	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/tree/main/test	
Test cases are routinely executed	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/actions/workflows/python-app.yml	
Issues are discussed before they are closed	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/issues?q=is%3Aissue+is%3Aclosed	
Chat channel: exists	3	https://github.com/tpanati/DollarBot/blob/main/docs/workflows/whatsApp_Group%20chat_screenshot.png	
Test cases: a large proportion of the issues related to handling failing cases.	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/issues?q=is%3Aissue+is%3Aclosed For example: https://github.com/tpanati/DollarBot/issues/43	
Evidence that the whole team is using the same tools: everyone can get to all tools and files	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/pulls?q=is%3Apr+is%3Aclosed	

Evidence that the whole team is using the same tools (e.g. config files in the repo, updated by lots of different people)	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/pulls?q=is%3Apr+is%3Aclosed	
Evidence that the whole team is using the same tools (e.g. tutor can ask anyone to share screen, they demonstrate the system running on their computer)	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/pulls?q=is%3Apr+is%3Aclosed	
Evidence that the members of the team are working across multiple places in the code base	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/network	
Short release cycles	2	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/releases	
The file .gitignore lists what files should not be saved to the repo. See [examples]i(https://github.com/github/gitignore)	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/blob/main/.gitignore	
The file INSTALL.md lists how to install the code	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/blob/main/README.md	No separate file for Install.md . Instructions for installation are included in readme.MD

The file LICENSE.md lists rules of usage for this repo	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/blob/main/LICENSE#:~:text=/-,LICENSE,-CSC510%2DSE%2DSAITEJA	
The file CODE-OF-CONDUCT.md lists rules of behavior for this repo; e.g. see example	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/blob/main/CODE_OF_CONDUCT.md	
The files CONTRIBUTING.md lists coding standards and lots of tips on how to extend the system without screwing things up	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot/blob/main/CONTRIBUTING.md	
The file README.md contains all the following	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot#	
Video	3		
DOI badge: exists. To get a Digitial Object Indentifier, regiser the project at Zenodo. DOI badges look like this: Zenodo doi badge	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot#	
Badges showing your style checkers	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot#	
Badges showing your code formatters.	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot#	
Badges showing your syntax checkers.	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot#	
Badges showing your code coverage tools	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot#	

Badges showing any other Other automated analysis tools	3	https://github.com/CSC510-SE-SAITEJA-MADHUMITHA-SHRUTI/DollarBot#	
Software Sustainability Questions:			
1.1 Does your website and documentation provide a clear, high-level overview of your software?	Y		
1.2 Does your website and documentation clearly describe the type of user who should use your software?	Y		
1.3: Do you publish case studies to show how your software has been used by yourself and others?	N		
2.1: Is the name of your project/ software unique?	Y		
2.2: Is your project/ software name free from trademark violations?	Y		
3.1: Is your software available as a package that can be deployed without building it?	N		
3.2: Is your software available for free?	Y		

3.3: Is your source code publicly available to download, either as a downloadable bundle or via access to a source code repository?	Y		
3.4: Is your software hosted in an established, third-party repository likeGitHub (https://github.com), BitBucket (https://bitbucket.org),LaunchPad (https://launchpad.net) orSourceForge (https://sourceforge.net)?	Y		
4.1: Is your documentation clearly available on your website or within your software?	Y		
4.2: 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?	Y		
4.3: If you provide more extensive documentation, does this provide clear, step-by-step instructions on how to deploy and use your software?	Y		

4.4: Do you provide a comprehensive guide to all your software's commands, functions and options?	Y		
4.5: Do you provide troubleshooting information that describes the symptoms and step-by-step solutions for problems and error messages?	N		
4.6: If your software can be used as a library, package or serviceby other software, do you provide comprehensive API documentation?	N		
4.7: Do you store your documentation under revision control with your source code?	Y		
4.8: 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?	Y		
5.1: Does your software describe how a user can get help with using your software?	Y		

5.2: Does your website and documentation describe what support, if any, you provide to users and developers?	Y		
5.3: Does your project have an e-mail address or forum that is solely for supporting users?	Y		
5.4: Are e-mails to your support e-mail address received by more than one person?	N		
5.5: Does your project have a ticketing system to manage bug reports and feature requests?	Y		
5.6: Is your project's ticketing system publicly visible to your users, so they can view bug reports and feature requests?	Y		
6.1: Is your software’s architecture and design modular?	Y		
6.2: Does your software use an accepted coding standard or convention?	Y		
7.1: Does your software allow data to be imported and exported using open data formats?	Y		

7.2: Does your software allow communications using open communications protocols?	Y		
8.1: Is your software cross-platform compatible?	Y		
9.1: Does your software adhere to appropriate accessibility conventions or standards?	Y		
9.2: Does your documentation adhere to appropriate accessibility conventions or standards?	Y		
10.1: Is your source code stored in a repository under revision control?	Y		
10.2: Is each source code release a snapshot of the repository?	Y		
10.3: Are releases tagged in the repository?	Y		
10.4: Is there a branch of the repository that is always stable?	Y		
10.5: Do you back-up your repository?	Y		
11.1: Do you provide publicly-available instructions for building your software from the source code?	Y		

11.2: Can you build, or package, your software using an automated tool?	N		
11.3: Do you provide publicly-available instructions for deploying your software?	Y		
11.4: Does your documentation list all third-party dependencies?	Y		
11.5: Does your documentation list the version number for all third-party dependencies?	Y		
11.6: Does your software list the web address, and licences for all thirdparty dependencies and say whether the dependencies are mandatory or optional?	Y		
11.7: Can you download dependencies using a dependency management tool or package manager?	Y		
11.8: 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?	Y		
12.1: Do you have an automated test suite for your software?	Y		

12.2: Do you have a framework to periodically (e.g. nightly) run your tests on the latest version of the source code?	N		
12.3: Do you use continuous integration, automatically running tests whenever changes are made to your source code?	Y		
12.4: Are your test results publicly visible?	Y		
12.5: Are all manually-run tests documented?	N		
13.1: 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?	N		
13.2: Does your website state how many projects and users are associated with your project?	N		
13.3: Do you provide success stories on your website?	N		
13.4: Do you list your important partners and collaborators on your website?	Y		

13.5: Do you list your project's publications on your website or link to a resource where these are available?	N		
13.6: Do you list third-party publications that refer to your software on your website or link to a resource where these are available?	N		
13.7: Can users subscribe to notifications to changes to your source code repository?	Y		
13.8: If your software is developed as an open source project (and, not just a project developing open source software), do you have a governance model?	Y		
14.1: Do you accept contributions (e.g. bug fixes, enhancements, documentation updates, tutorials) from people who are not part of your project?	Y		
14.2: Do you have a contributions policy?	Y		
14.3: Is your contributions' policy publicly available?	Y		

14.4: Do contributors keep the copyright/IP of their contributions?	Y		
15.1: Does your website and documentation clearly state the copyright owners of your software and documentation?	Y		
15.2: Does each of your source code files include a copyright statement?	Y		
15.3: Does your website and documentation clearly state the licence of your software?	Y		
15.4: Is your software released under an open source licence?	Y		
15.5: Is your software released under an OSI-approved open-source licence?	Y		
15.6: Does each of your source code files include a licence header?	Y		
15.7: Do you have a recommended citation for your software?	N		

16.1: Does your website or documentation include a project roadmap (a list of project and development milestones for the next 3, 6 and 12 months)?	Y		
16.2: Does your website or documentation describe how your project is funded, and the period over which funding is guaranteed?	N		
16.3: Do you make timely announcements of the deprecation of components, APIs, etc.?	N		