2. https://github.com/CSC510-Team-57/To-Done

3.

|  |  |  |
| --- | --- | --- |
| Rubric | Self-Assessment SUM: 102 | Evidence |
| Workload is spread over the whole team (one team member is often Xtimes more productive than the others… | 3 | https://github.com/devyash2930/ShopSync-SE/tree/testing |
| but nevertheless, here is a track record that everyone is contributing a lot) | 3 | https://github.com/devyash2930/ShopSync-SE/commits/testing/ |
| Number of commits | 3 | <https://github.com/devyash2930/ShopSync-SE/pulse> |
| Number of commits: by different people | 3 | <https://github.com/devyash2930/ShopSync-SE/pulse> |
| Issues reports: there are **many** | 2.5 | https://github.com/devyash2930/ShopSync-SE/issues |
| Issues are being closed | 2.5 | <https://github.com/CSC510-Team-57/To-Done/issues> |
| Docs: doco generated, format not ugly | 2.5 | In GH |
| Docs: what: point descriptions of each class/function (in isolation) | 2.5 | In GH |
| Docs: how: for common use cases X,Y,Z mini-tutorials showing worked examples on how to do X,Y,Z | 2.5 | <https://github.com/devyash2930/ShopSync-SE/blob/testing/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 | 2.5 | <https://github.com/devyash2930/ShopSync-SE/blob/testing/README.md> |
| Docs: short video, animated, hosted on your repo. That convinces people why they want to work on your code. | 3 | TODO |
| Use of version control tools | 3 | We used Git |
| Test cases exist | 3 | <https://github.com/devyash2930/ShopSync-SE/tree/testing/new_tests> |
| Test cases are routinely executed | 3 | <https://github.com/devyash2930/ShopSync-SE/blob/testing/.github/workflows/unit_test.yml> |
| Issues are discussed before they are closed | 3 | <https://github.com/devyash2930/ShopSync-SE/discussions> |
| Chat channel: exists | 3 | <https://github.com/devyash2930/ShopSync-SE/discussions> |
| Test cases: a large proportion of the issues related to handling failing cases. | 3 | The issues were mostly related to the new development features we wanted to add. Additional issues were then created based on issues we discovered with either black box or white box testing |
| Evidence that the whole team is using the same tools: everyone can get to all tools and files | 3 | In GH |
| Evidence that the whole team is using the same tools (e.g. config files in the repo, updated by lots of different people) | 3 | In GH |
| 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 | In GH |
| Evidence that the members of the team are working across multiple places in the code base | 3 | In GH |
| Short release cycles | 3 | <https://github.com/devyash2930/ShopSync-SE/commits/testing/> |
| 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/devyash2930/ShopSync-SE/blob/testing/.gitignore> |
| The file INSTALL.md lists how to install the code | 3 | <https://github.com/devyash2930/ShopSync-SE/blob/testing/INSTALL.md> |
| The file LICENSE.md lists rules of usage for this repo | 3 | <https://github.com/devyash2930/ShopSync-SE/blob/testing/LICENSE> |
| The file CODE-OF-CONDUCT.md lists rules of behavior for this repo; e.g. see [example](https://github.com/probot/template/blob/master/CODE_OF_CONDUCT.md) | 3 | <https://github.com/devyash2930/ShopSync-SE/blob/testing/Code_Of_Conduct.md> |
| 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](https://github.com/probot/template/blob/master/CONTRIBUTING.md) | 3 | <https://github.com/devyash2930/ShopSync-SE/blob/testing/CONTRIBUTING.md> |
| The file README.md contains all the following | 3 | <https://github.com/devyash2930/ShopSync-SE/blob/testing/README.md> |
| Video | 3 | <https://github.com/devyash2930/ShopSync-SE/blob/testing/media/Demo.mp4> |
| DOI badge: exists. To get a Digitial Object Indentifier, regiser the project at [Zenodo](https://docs.github.com/en/repositories/archiving-a-github-repository/referencing-and-citing-content). DOI badges look like this: | 3 | <https://zenodo.org/records/14020468> |
| Badges showing your style checkers | 3 | <https://github.com/devyash2930/ShopSync-SE/blob/testing/.github/workflows/style_checker.yml> |
| Badges showing your code formatters. | 3 | <https://github.com/devyash2930/ShopSync-SE/blob/testing/.github/workflows/code_formatter.yml> |
| Badges showing your syntax checkers. | 3 | <https://github.com/devyash2930/ShopSync-SE/blob/testing/.github/workflows/main.yml> |
| Badges showing your code coverage tools | 3 | <https://github.com/devyash2930/ShopSync-SE/blob/testing/.github/workflows/code_cov.yml> |
| Badges showing any other Other automated analysis tools | 3 | In the README |
| Question 1.1: Does your website and documentation provide a clear, high-level overview of your software? | Yes |  |
| Question 1.2: Does your website and documentation clearly describe the type of user who should use your software? | Yes |  |
| Question 1.3: Do you publish case studies to show how your software has been used by yourself and others? | No |  |
| Question 2.1: Is the name of your project/software unique? | Yes |  |
| Question 2.2: Is your project/software name free from trademark violations? | Yes | We believe so |
| Question 3.1: Is your software available as a package that can be deployed without building it? | Yes | There is no need to build |
| Question 3.2: Is your software available for free? | Yes |  |
| Question 3.3: Is your source code publicly available to download, either as a downloadable bundle or via access to a source code repository? | Yes |  |
| Question 3.4: Is your software hosted in an established, third-party repository likeGitHub ([https://github.com](https://github.com/)), BitBucket ([https://bitbucket.org](https://bitbucket.org/)),LaunchPad ([https://launchpad.net](https://launchpad.net/)) orSourceForge ([https://sourceforge.net](https://sourceforge.net/))? | Yes | We use GitHub |
| Question 4.1: Is your documentation clearly available on your website or within your software? | Yes |  |
| Question 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? | Yes |  |
| Question 4.3: If you provide more extensive documentation, does this provide clear, step-by-step instructions on how to deploy and use your software? | N/A |  |
| Question 4.4: Do you provide a comprehensive guide to all your software’s commands, functions and options? | Yes |  |
| Question 4.5: Do you provide troubleshooting information that describes the symptoms and step-by-step solutions for problems and error messages? | No |  |
| Question 4.6: If your software can be used as a library, package or service by other software, do you provide comprehensive API documentation? | N/A |  |
| Question 4.7: Do you store your documentation under revision control with your source code? | Yes |  |
| Question 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? | Yes |  |
| Question 5.1: Does your software describe how a user can get help with using your software? | Yes | We provide links to contact us, or users can ask in discussions |
| Question 5.2: Does your website and documentation describe what support, if any, you provide to users and developers? | Yes | We provide an area to report bugs and ask for new features, in issues on github |
| Question 5.3: Does your project have an e-mail address or forum that is solely for supporting users? | No |  |
| Question 5.4: Are e-mails to your support e-mail address received by more than one person? | No |  |
| Question 5.5: Does your project have a ticketing system to manage bug reports and feature requests? | Yes | We have an issues page |
| Question 5.6: Is your project's ticketing system publicly visible to your users, so they can view bug reports and feature requests? | Yes | The issues page is publicly visible |
| Question 6.1: Is your software’s architecture and design modular? | Yes |  |
| Question 6.2: Does your software use an accepted coding standard or convention? | Yes |  |
| Question 7.1: Does your software allow data to be imported and exported using open data formats? | No |  |
| Question 7.2: Does your software allow communications using open communications protocols? | Yes |  |
| Question 8.1: Is your software cross-platform compatible? | Yes |  |
| Question 9.1: Does your software adhere to appropriate accessibility conventions or standards? | Yes |  |
| Question 9.2: Does your documentation adhere to appropriate accessibility conventions or standards? | Yes |  |
| Question 10.1: Is your source code stored in a repository under revision control? | Yes | Yes as we use GitHub |
| Question 10.2: Is each source code release a snapshot of the repository? | Yes |  |
| Question 10.3: Are releases tagged in the repository? | Yes |  |
| Question 10.4: Is there a branch of the repository that is always stable? (i.e. tests always pass, code always builds successfully) | Yes | testing Branch |
| Question 10.5: Do you back-up your repository? | No |  |
| Question 11.1: Do you provide publicly-available instructions for building your software from the source code? | N/A |  |
| Question 11.2: Can you build, or package, your software using an automated tool? | No |  |
| Question 11.3: Do you provide publicly-available instructions for deploying your software? | Yes |  |
| Question 11.4: Does your documentation list all third-party dependencies? | Yes | In the requirements |
| Question 11.5: Does your documentation list the version number for all third-party dependencies? | Yes |  |
| Question 11.6: Does your software list the web address, and licences for all third-party dependencies and say whether the dependencies are mandatory or optional? | Yes |  |
| Question 11.7: Can you download dependencies using a dependency management tool or package manager? | Yes |  |
| Question 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? | Yes |  |
| Question 12.1: Do you have an automated test suite for your software? | Yes |  |
| Question 12.2: Do you have a framework to periodically (e.g. nightly) run your tests on the latest version of the source code? | No |  |
| Question 12.3: Do you use continuous integration, automatically running tests whenever changes are made to your source code? | Yes |  |
| Question 12.4: Are your test results publicly visible? | Yes |  |
| Question 12.5: Are all manually-run tests documented? | No |  |
| Question 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? | No |  |
| Question 13.2: Does your website state how many projects and users are associated with your project? | Yes |  |
| Question 13.3: Do you provide success stories on your website? | No |  |
| Question 13.4: Do you list your important partners and collaborators on your website? | No |  |
| Question 13.5: Do you list your project's publications on your website or link to a resource where these are available? | No |  |
| Question 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? | No |  |
| Question 13.7: Can users subscribe to notifications to changes to your source code repository? | No |  |
| Question 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? | No |  |
| Question 14.1: Do you accept contributions (e.g. bug fixes, enhancements, documentation updates, tutorials) from people who are not part of your project? | Yes |  |
| Question 14.2: Do you have a contributions policy? | Yes |  |
| Question 14.3: Is your contributions' policy publicly available? | Yes |  |
| Question 14.4: Do contributors keep the copyright/IP of their contributions? | Yes |  |
| Question 15.1: Does your website and documentation clearly state the copyright owners of your software and documentation? | Yes |  |
| Question 15.2: Does each of your source code files include a copyright statement? | No |  |
| Question 15.3: Does your website and documentation clearly state the license of your software? | Yes |  |
| Question 15.4: Is your software released under an open source license? | Yes |  |
| Question 15.5: Is your software released under an OSI-approved open-source license? | Yes |  |
| Question 15.6: Does each of your source code files include a license header? | No |  |
| Question 15.7: Do you have a recommended citation for your software? | Yes |  |
| Question 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)? | Yes |  |
| Question 16.2: Does your website or documentation describe how your project is funded, and the period over which funding is guaranteed? | No |  |
| Question 16.3: Do you make timely announcements of the deprecation of components, APIs, etc.? | No |  |