Poster Link -
https://github.com/fantastic-riddles/URL-Shortner/blob/develop/documents/Group_21_Poster.pdf

Implementation video -
https://drive.google.com/file/d/1qF8dR9zTcg4T-tR3HXbiGumfku__pmNj/view?usp=sharing
https://github.com/fantastic-riddles/URL-Shortner/tree/develop?tab=readme-ov-file#-about-the-project

Github Repository - https://github.com/fantastic-riddles/URL-Shortner

Notes	Self-Assessmen, 298/318	Evidence
Workload is spread over the whole team (one team member is often Xtimes more productive than the others…	3	Everyone was assigned a set of features and tasks based on the tickets.
but nevertheless, here is a track record that everyone is contributing a lot)	3	Commit history
Number of commits	3	50, Link to commit history
Number of commits: by different people	3	in GH
Issues reports: there are **many**	3	Project Board
Issues are being closed	3	evidence in GH - Project Board Sample Pull Request
Docs: doco generated, format not ugly	3	Documentation

Docs: what: point descriptions of each class/function (in isolation)	3	Documentation
Docs: how: for common use cases X,Y,Z mini-tutorials showing worked examples on how to do X,Y,Z	3	Documentation
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	Documentation
Docs: short video, animated, hosted on your repo. That convinces people why they want to work on your code.	3	Animation Video
Use of version control tools	3	Used GitHub
Test cases exist	3	Yes, apart from the existing test cases, we have added 71 new test cases.
Test cases are routinely executed	2	Yes
Issues are discussed before they are closed	3	Issues were created as tickets on GitHub Kanban board and were discussed in person before closing.
Chat channel: exists	3	SE-project-group-screenshot.png

Test cases: a large proportion of the issues related to handling failing cases.	3	Yes, test cases covered all type of scenarios and the failing cases were fixed.
Evidence that the whole team is using the same tools: everyone can get to all tools and files	3	VS Code, Github
Evidence that the whole team is using the same tools (e.g. config files in the repo, updated by lots of different people)	3	Setup files used in VS Code
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	VS Code, Github
Evidence that the members of the team are working across multiple places in the code base	3	VS Code and Github
Short release cycles	3	(hard to see in short projects) project members are committing often enough so that everyone can get your work
The file .gitignore lists what files should not be saved to the repo. See [examples]i(https://github.com/github/gitignore)	3	Gitignore
The file INSTALL.md lists how to install the code	3	Install file

The file LICENSE.md lists rules of usage for this repo	3	License
The file CODE-OF-CONDUCT.md lists rules of behavior for this repo; e.g. see example	3	Code of Conduct
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	3	Contribution
The file README.md contains all the following	3	Link to README
Video	3	Yes, Video
DOI badge: exists. To get a Digitial Object Indentifier, regiser the project at Zenodo. DOI badges look like this:	3	Yes, the badge has been attached in the README.md file as well. Link
Badges showing your style checkers	3	Yes
Badges showing your code formatters.	3	Yes
Badges showing your syntax checkers.	3	Yes
Badges showing your code coverage tools	3	Yes, using codecov3

Badges showing any other Other automated analysis tools	3	config files in GH, badges in README
Does your website and documentation provide a clear, high-level overview of your software?	3	Yes, Refer Website and Documentation
Does your website and documentation clearly describe the type of user who should use your software?	3	Yes, Refer Website and Documentation
Do you publish case studies to show how your software has been used by yourself and others?	3	Yes, refer Documentation and user testimonials on Website.
Is the name of your project/software unique? :Yes	3	Yes, Our forked Repository is named Txtly
Is your project/software name free from trademark violations?	3	No results were found in Trademark Electronic Search System
Is your software available as a package that can be deployed without building it?	0	No
Is your software available for free?	3	Yes
Is your source code publicly available to download, either as a downloadable bundle or via access to a source code repository? : Yes	3	Yes, Refer the GitHub Link

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)? : Yes	3	Refer the GitHub Link
Is your documentation clearly available on your website or within your software?	3	Yes refer Documentation
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?	3	Yes, refer Video
If you provide more extensive documentation, does this provide clear, step-by-step instructions on how to deploy and use your software?	3	Yes
Do you provide a comprehensive guide to all your software's commands, functions and options?	2	Yes
Do you provide troubleshooting information that describes the symptoms and step-by-step solutions for	2	Yes

problems and error messages?		
If your software can be used as a library, package or service by other software, do you provide comprehensive API documentation?	2	Yes, link
Do you store your documentation under revision control with your source code?	3	Yes, as doc strings in python
Do you publish your release history e.g. release date, version numbers, key features of each release etc. on your web site or in your documentation?	3	Yes and also can be found in Zenodo.
Does your software describe how a user can get help with using your software?	3	Yes
Does your website and documentation describe what support, if any, you provide to users and developers?	3	Yes refer README
Does your project have an e-mail address or forum that is solely for supporting users?	3	Yes
Are e-mails to your support e-mail address received by more than one person?	3	Yes, Refer ReadMe.md

Does your project have a ticketing system to manage bug reports and feature requests?	3	Yes
Is your project's ticketing system publicly visible to your users, so they can view bug reports and feature requests?	3	Yes
Is your software's architecture and design modular?	3	Yes, we use CRM, different environment for testing
Does your software use an accepted coding standard or convention?	3	Yes. flake8 is a popular style checker that is being used. Also pylint has been used.
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	3	Yes. API Database Export, CSV Export, refer Documentation
Does your software allow communications using open communications protocols? e.g. HTTP, FTP, XMPP, SOAP over HTTP, or domain-specific ones	3	Yes. HTTP is used for communication
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	3	Yes

Internet Explorer, Chrome, Firefox and Safari?		
Does your software adhere to appropriate accessibility conventions or standards?	3	Yes. Angular Material ensures accessibility standards are followed
Does your documentation adhere to appropriate accessibility conventions or standards?	3	Yes. pdoc and compodoc ensure that accessibility standards are followed
Is your source code stored in a repository under revision control?	3	Yes. Repository on GitHub
Is each source code release a snapshot of the repository?	3	Yes. Link to Releases
Are releases tagged in the repository?	3	Yes. Releases also have tags on GitHub
Is there a branch of the repository that is always stable? (i.e. tests always pass, code always builds successfully)	3	Yes. `develop` branch is always stable
Do you back-up your repository?	3	Yes. Backups are made and stored on local machines of each developer.
Do you provide publicly-available instructions for building your software from the source code?	3	Yes. Refer README. These instructions build the software locally.

Can you build, or package, your software using an automated tool? 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)	3	Yes. Shell Script
Do you provide publicly-available instructions for deploying your software?	3	Yes. refer README
Does your documentation list all third-party dependencies?	3	Yes. Refer Requirements
Does your documentation list the version number for all third-party dependencies?	3	Yes. Refer Requirements
Does your software list the web address, and licences for all third-party dependencies and say whether the dependencies are mandatory or optional?	3	Yes .GitHub insights provides the details of the dependencies
Can you download dependencies using a	3	Yes. By pip

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 (https://getcomposer.org/), Ruby gems (https://rubygems.org), or R PackRat (https://rstudio.github.io/packrat/)		
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?	0	No, no test case has been written to check deployment success.
Do you have an automated test suite for your software?	3	Yes, refer workflows.
Do you have a framework to periodically (e.g. nightly) run your tests on the latest version of the source code?	3	Yes, refer workflows.
Do you use continuous integration, automatically running tests whenever changes are made to your source code?	3	Yes, refer workflows.

Are your test results publicly visible?	3	Yes, refer workflows.
Are all manually-run tests documented?	3	Not Applicable. All test cases are automated.
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? e.g. release announcements, publications, workshops, conference presentations	0	No.
Does your website state how many projects and users are associated with your project?	3	Yes, refer README
Do you provide success stories on your website?	3	Yes, Refer Homepage on the website
Do you list your important partners and collaborators on your website?	3	Yes, refer README
Do you list your project's publications on your website or link to a resource where these are available?	0	There are no publications for the project
Do you list third-party publications that refer to your software on your website or link to a resource where these are available?	0	No third-party publications

Can users subscribe to notifications to changes to your source code repository?	3	Yes users can watch and star the repo
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?	3	Model can be found in CONTRIBUTING.md, we have the dev branch at the top of development and main is stable
Do you accept contributions (e.g. bug fixes, enhancements, documentation updates, tutorials) from people who are not part of your project?	3	Yes. CONTRIBUTING.md has instructions to do so
Do you have a contributions policy?	3	Yes. CONTRIBUTING.md has the policies
Is your contributions' policy publicly available?	3	Yes. CONTRIBUTING.md within the repository
Do contributors keep the copyright/IP of their contributions?	3	Yes. As per MIT License
Does your website and documentation clearly state the copyright owners of your software and documentation?	3	Yes. As per MIT License
Does each of your source code files include a copyright statement?	3	Yes. As per MIT License

Does your website and documentation clearly state the license of your software?	3	Yes, MIT License
Is your software released under an open source license?	3	Yes, MIT License
Is your software released under an OSI-approved open-source license?	3	Yes, MIT License
Does each of your source code files include a license header?	3	Yes
Do you have a recommended citation for your software?	3	Yes, citation
Does your website or documentation include a project roadmap (a list of project and development milestones for the next 3, 6 and 12 months)?	2	Refer Documents for roadmap
Does your website or documentation describe how your project is funded, and the period over which funding is guaranteed?	3	It is not funded
Do you make timely announcements of the deprecation of components, APIs, etc.?	3	There haven't been any deprecations yet