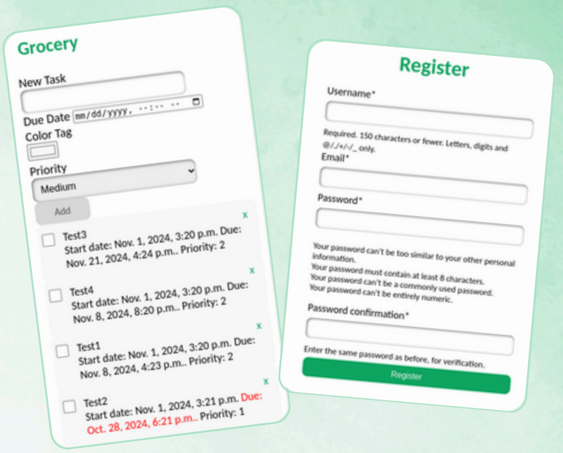
☑ To-Done ☑

Helping you get it done

Team 57: Aastha || Spencer || Charlie

About

To-Done is an easy to use web based task managing system where you can create new tasks in set lists. Tasks are assigned a start date and a due date and can have notes and colors added to them. Tasks can also be assigned priority. Lists are the organization tool used to hold tasks, and they can be organized via tags. We have over 89 tests to ensure code quality.



Grocery

New Task

Due Date mm/dd/yyyy, --:-- --

Color Tag

Priority

Medium

Add

□ Test3
Start date: Nov. 1, 2024, 3:20 p.m. Due: Nov. 21, 2024, 4:24 p.m.. Priority: 2

□ Test4
Start date: Nov. 1, 2024, 3:20 p.m. Due: Nov. 8, 2024, 8:20 p.m.. Priority: 2

□ Test1
Start date: Nov. 1, 2024, 3:20 p.m. Due: Nov. 8, 2024, 4:23 p.m.. Priority: 2

□ Test2
Start date: Nov. 1, 2024, 3:21 p.m. Due: Oct. 28, 2024, 6:21 p.m.. Priority: 1

Register

Username*

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

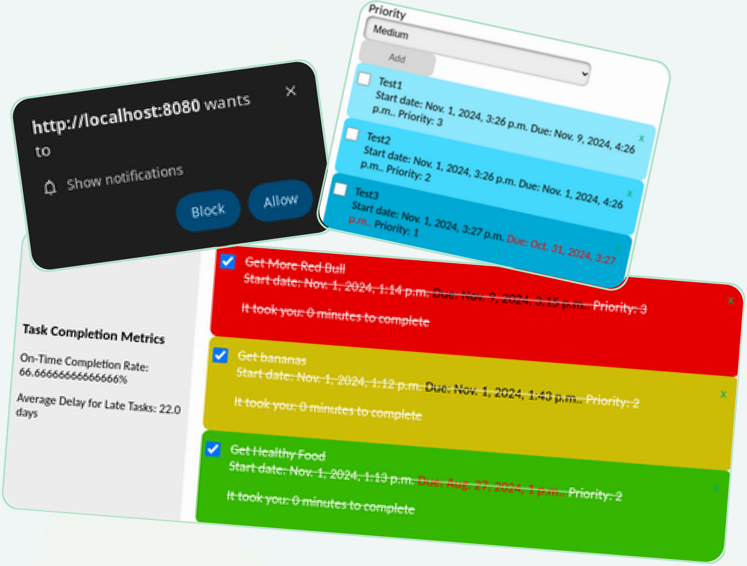
Email*

Password*

Your password can't be too similar to your other personal information.
Your password must contain at least 8 characters.
Your password can't be a commonly used password.
Your password can't be entirely numeric.

Password confirmation*

Enter the same password as before, for verification.



http://localhost:8080 wants to

Show notifications

Priority

Medium

□ Test1
Start date: Nov. 1, 2024, 3:26 p.m. Due: Nov. 9, 2024, 4:26 p.m.. Priority: 3

□ Test2
Start date: Nov. 1, 2024, 3:26 p.m. Due: Nov. 1, 2024, 4:26 p.m.. Priority: 2

□ Test3
Start date: Nov. 1, 2024, 3:27 p.m. Due: Oct. 31, 2024, 3:27 p.m.. Priority: 1

Task Completion Metrics

On-Time Completion Rate: 66.66666666666666%

Average Delay for Late Tasks: 22.0 days

☑ ~~Get More Red Bull~~
~~Start date: Nov. 1, 2024, 1:14 p.m. Due: Nov. 9, 2024, 3:15 p.m.. Priority: 3~~
~~It took you: 0 minutes to complete~~

☑ ~~Get bananas~~
~~Start date: Nov. 1, 2024, 1:12 p.m. Due: Nov. 1, 2024, 1:43 p.m.. Priority: 2~~
~~It took you: 0 minutes to complete~~

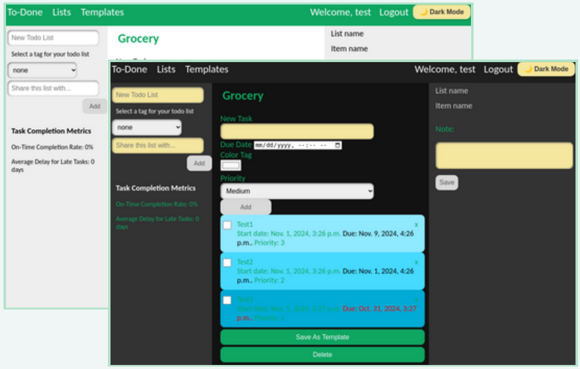
☑ ~~Get Healthy Food~~
~~Start date: Nov. 1, 2024, 1:13 p.m. Due: Aug. 27, 2024, 1 p.m.. Priority: 2~~
~~It took you: 0 minutes to complete~~

What's New

- Notifications when a task is due or nearing the due date
- On-time metrics for completing tasks
- A comprehensive list of all upcoming tasks in the order that they are due
- Implementation of a priority tracker
- Theme toggle option

Why choose us?

- Significant Growth Opportunities
  - A solid set of 89 test cases ensures high-quality standards
- Diverse Tech Stack
  - An excellent blend of frontend and backend technologies to showcase and strengthen your skills
- Practical, Real-Word Application
  - Gain valuable experience with a project thats both practical and impactful



To-Done Lists Templates Welcome, test Logout Dark Mode

New Todo List

Select a tag for your todo list

none

Share this list with...

Task Completion Metrics

On-Time Completion Rate: 0%

Average Delay for Late Tasks: 0 days

Grocery

List name

Item name

To-Done Lists Templates Welcome, test Logout Dark Mode

New Todo List

Select a tag for your todo list

none

Share this list with...

Task Completion Metrics

On-Time Completion Rate: 0%

Average Delay for Late Tasks: 0 days

Grocery

New Task

Due Date mm/dd/yyyy, --:-- --

Color Tag

Priority

Medium

□ Test1
Start date: Nov. 1, 2024, 3:26 p.m. Due: Nov. 9, 2024, 4:26 p.m.. Priority: 3

□ Test2
Start date: Nov. 1, 2024, 3:26 p.m. Due: Nov. 1, 2024, 4:26 p.m.. Priority: 2

□ Test3
Start date: Nov. 1, 2024, 3:27 p.m. Due: Oct. 31, 2024, 3:27 p.m.. Priority: 1

List name

Item name

Note:

Github

Demo

Tech Stack

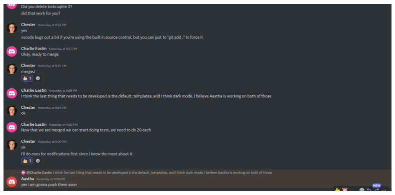
Future Scope

- Email Notifications
- Calendar and Scheduling integration
- List sharing with other users
- Reoccurring tasks real time update
- Containerization
- User authentication by OTP
- Progress Tracking

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

3.

Rubric	Self-Assessment SUM: 103	Evidence
Workload is spread over the whole team (one team member is often Xtimes more productive than the others…	3	https://github.com/Chloe-Ku/To-Done/compare/main...CSC510-Team-57:To-Done:main
but nevertheless, here is a track record that everyone is contributing a lot)	3	https://github.com/Chloe-Ku/To-Done/compare/main...CSC510-Team-57:To-Done:main Charlie-Eastin has a lot of commits at one point but that is just because I was trying to fix the yml tests so they were a lot of little commits
Number of commits	3	https://github.com/Chloe-Ku/To-Done/compare/main...CSC510-Team-57:To-Done:main
Number of commits: by different people	3	https://github.com/Chloe-Ku/To-Done/compare/main...CSC510-Team-57:To-Done:main
Issues reports: there are **many**	2.5	https://github.com/CSC510-Team-57/To-Done/issues
Issues are being closed	3	https://github.com/CSC510-Team-57/To-Done/issues
Docs: doco generated, format not ugly	3	In GH
Docs: what: point descriptions of each class/function (in isolation)	3	In GH
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-Team-57/To-Done/blob/main/README.md
Docs: why: docs tell a story, motivate the whole thing, deliver a punchline that	3	https://github.com/CSC510-Team-57/To-Done/blob/main/README.md

makes you want to rush out and use the thing		
Docs: short video, animated, hosted on your repo. That convinces people why they want to work on your code.	3	https://github.com/CSC510-Team-57/To-Done/blob/main/README.md
Use of version control tools	3	We used Git
Test cases exist	3	https://github.com/CSC510-Team-57/To-Done/tree/main/todo/tests
Test cases are routinely executed	3	https://github.com/CSC510-Team-57/To-Done/actions/workflows/main.yml
Issues are discussed before they are closed	3	https://github.com/CSC510-Team-57/To-Done/issues/17
Chat channel: exists	3	 Did you delete todo.sqlite 3? did that work for you? Chester Yesterday at 8:56 PM yes vscode bugs out a bit if you're using the built in source control, but you can just to "git add ." to force it Charlie Eastin Yesterday at 8:57 PM Okay, ready to merge Chester Yesterday at 8:59 PM merged 👍 1 Charlie Eastin Yesterday at 8:59 PM I think the last thing that needs to be developed is the default_templates, and I think dark mode. I believe Aastha is working on both of those Chester Yesterday at 8:59 PM ok Charlie Eastin Yesterday at 9:00 PM Now that we are merged we can start doing tests, we need to do 20 each Chester Yesterday at 9:00 PM ok I'll do ones for notifications first since I know the most about it 👍 1 @Charlie Eastin I think the last thing that needs to be developed is the default_templates, and I think dark mode. I believe Aastha is working on both of those Aastha Yesterday at 9:09 PM yes i am gonna push them soon
Test cases: a large proportion of the issues related to handling failing cases.	2	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	3	In GH

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	3	In GH
Short release cycles	2.5	https://github.com/Chloe-Ku/To-Done/compare/main...CSC510-Team-57:To-Done:main
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-Team-57/To-Done/blob/main/.gitignore
The file INSTALL.md lists how to install the code	3	https://github.com/CSC510-Team-57/To-Done/blob/main/INSTALL.md
The file LICENSE.md lists rules of usage for this repo	3	https://github.com/CSC510-Team-57/To-Done/blob/main/LICENSE
The file CODE-OF-CONDUCT.md lists rules of behavior for this repo; e.g. see example	3	https://github.com/CSC510-Team-57/To-Done/blob/main/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	3	https://github.com/CSC510-Team-57/To-Done/blob/main/CONTRIBUTING.md
The file README.md contains all the following	3	https://github.com/CSC510-Team-57/To-Done/blob/main/README.md
Video	3	https://github.com/CSC510-Team-57/To-Done/blob/main/img/DEMO.mp4
DOI badge: exists. To get a Digitial Object Indentifier, regiser the project at Zenodo. DOI badges look like this:	3	https://doi.org/10.5281/zenodo.14016079

Badges showing your style checkers	3	https://github.com/CSC510-Team-57/To-Done/actions/workflows/pylint.yml
Badges showing your code formatters.	2	https://github.com/CSC510-Team-57/To-Done/actions
Badges showing your syntax checkers.	3	https://github.com/CSC510-Team-57/To-Done/actions/workflows/pylint.yml
Badges showing your code coverage tools	3	https://coveralls.io/github/CSC510-Team-57/To-Done?branch=main
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?	Yes	We have example cases
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	Yes	There is no need to build

deployed without building it?		
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), BitBucket (https://bitbucket.org),LaunchPad (https://launchpad.net) orSourceForge (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	Yes	

provide a comprehensive guide to all your software’s commands, functions and options?		
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
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
Question 5.3: Does your project have an e-mail	No	

address or forum that is solely for supporting users?		
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	Main 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	No	

code?		
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	No	

third-party publications that refer to your software on your website or link to a resource where these are available?		
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	Yes	

documentation?		
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	