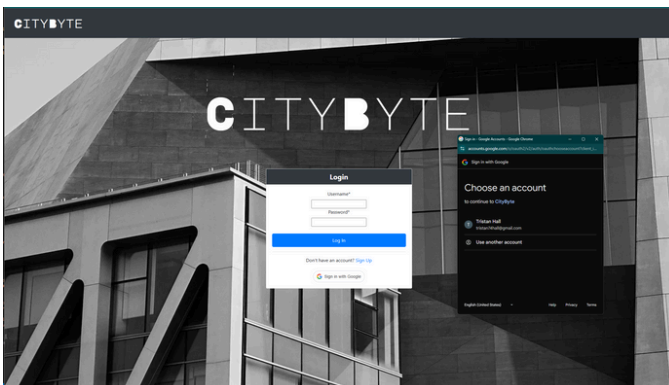# CITYBYTE

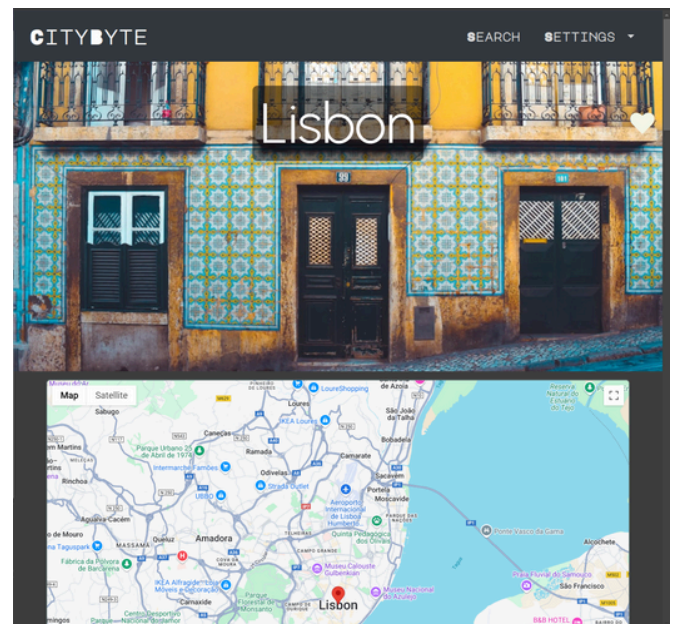**Going on a trip soon? Like to virtually explore? Look to CityByte as your guide for exploration!**

Prior Improvements

- **UI Overhaul:** New shiny, modern, user-friendly interface
- **Maps Display:** Find the locations near you on a map!
- **Google Accounts:** smoother CityByte registration and login via Google authentication API
- **Trip Itinerary:** save cool locations in each city!
- **And so much more:** favorite cities, leave comments, get results fast with the magic of caching, and register accounts without Google.

Proposed Improvements

- **Multi-point Maps:** what's better than one map pin? A lot! Allow users to drop multiple pins to figure out what's close (or not).
- **Better Searching**: It works, but it ain't stellar. Overhaul how user's find and get their results.
- **Organize Your Itinerary**: It isn't always the best thing to

CITYBYTE

SEARCH SETTINGS

Lisbon

CURRENT NO. OF TESTS: 56
+ 4 ON LOCAL SYSTEMS

GitHub Repo: https://github.com/Gunabana/CityByte

Note	Points	Evidence
Workload is spread over the whole team (one team member is often Xtimes more productive than the others…	Brody: 3 Chaitanya: 3 Tristan:2	in GH (commits, merge requests, etc.,)
but nevertheless, here is a track record that everyone is contributing a lot)	Brody: 3 Chaitanya: 3 Tristan:2	in GH (commits, merge requests, etc.,)
Number of commits	Brody: 3 Chaitanya: 3 Tristan:1	in GH (commits or Insights > Contributor)
Number of commits: by different people	Brody: 3 Chaitanya:3 Tristan:1	in GH (commits)
Issues reports: there are **many**	Brody: 3 Chaitanya: 2 Tristan:2	in GH (Issues)
Issues are being closed	Brody: 3 Chaitanya: 2 Tristan:2	in GH (Issues)
Docs: doco generated, format not ugly	Brody: 3 Chaitanya: 3 Tristan:2	in GH
Docs: what: point descriptions of each class/function (in isolation)	Brody: 3 Chaitanya: 3 Tristan:3	Yes
Docs: how: for common use cases X,Y,Z mini-tutorials showing worked examples on how to do X,Y,Z	Brody: 1 Chaitanya: 3 Tristan:2	Yes, in GH
Docs: why: docs tell a story, motivate the whole thing, deliver a punchline that makes you want to rush out and use the thing	Brody: 1 Chaitanya: 2 Tristan:3	I don't know what this rubric item means - Brody
Docs: short video, animated, hosted on your repo. That convinces people why they want to work on your code.	Brody: 1 Chaitanya:1 Tristan:3	Yes, in GH
Use of version control tools	Brody: 3 Chaitanya: 3 Tristan:3	GH itself
Test cases exist	Brody: 3 Chaitanya: 3 Tristan:3	Tests: 51, Code Cov: 89%
Test cases are routinely executed	Brody: 3 Chaitanya: 2 Tristan:3.	GitHub Actions

Issues are discussed before they are closed	Brody: 3 Chaitanya: 3 Tristan:2	Typically done via Discord voice call meeting (if team members are available)
Chat channel: exists	Brody: 3 Chaitanya: 3 Tristan:3	Link
Test cases: a large proportion of the issues related to handling failing cases.	Brody: 3 Chaitanya: 3 Tristan:3	If a test case fails, open an issue and fix it
Evidence that the whole team is using the same tools: everyone can get to all tools and files	Brody: 3 Chaitanya: 3 Tristan:2	No, not really
Evidence that the whole team is using the same tools (e.g. config files in the repo, updated by lots of different people)	Brody: 3 Chaitanya: 3 Tristan:2	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)	Brody: 3 Chaitanya:3 Tristan:2	No, not really
Evidence that the members of the team are working across multiple places in the code base	Brody: 3 Chaitanya: 3 Tristan:2	In GH (see merges requests)
Short release cycles	Brody: 2 Chaitanya: 2 Tristan:2	in GH (see commits)
The file .gitignore lists what files should not be saved to the repo. See [examples]i(https://github.com/github/gitignore)	Brody: 3 Chaitanya: 3 Tristan:3	in GH (.gitignore)
The file INSTALL.md lists how to install the code	Brody: 3 Chaitanya: 3 Tristan:3	Yes, in GH
The file LICENSE.md lists rules of usage for this repo	Brody: 3 Chaitanya: 3 Tristan:3	Yes, in GH
The file CODE-OF-CONDUCT.md lists rules of behavior for this repo; e.g. see example	Brody: 3 Chaitanya: 3 Tristan:3	Yes, in GH
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	Brody: 3 Chaitanya:2 Tristan:3	Yes, in GH
The file README.md contains all the following	Brody: 3 Chaitanya: 3 Tristan:3	in GH
Video	Brody: 3	Yes, in GH

	Chaitanya:2 Tristan:3	
DOI badge: exists	Brody: 3 Chaitanya: 2 Tristan:2	in GH
Badges showing your style checkers	Brody: 3 Chaitanya:2 Tristan:2	in GH
Badges showing your code formatters.	Brody: 2 Chaitanya:2 Tristan:2	Yes, in GH (Flake 8)
Badges showing your syntax checkers.	Brody: 3 Chaitanya:2 Tristan:2	in GH
Badges showing your code coverage tools	Brody: 3 Chaitanya:3 Tristan:2	in GH
Badges showing any other other automated analysis tools	Brody: 3 Chaitanya:3 Tristan:2	in GH
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, Google it
Question 2.2: Is your project/software name free from trademark violations?		Yes, Google it
Question 3.1: Is your software available as a package that can be deployed without building it?		No
Question 3.2: Is your software available for free?		Yes, see our GH
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, in GH
Question 3.4: Is your software hosted in an established, third-party repository like GitHub…		Yes, GH
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, in GH (INSTALL.MD)

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?		Yes, retained but condensed prior teams' work but added more detail of technical documents
Question 4.4: Do you provide a comprehensive guide to all your software's commands, functions and options?		Yes, documentation and guides
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, in GH (all files have revision control)
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, in GH (Releases)
Question 5.1: Does your software describe how a user can get help with using your software?		Yes, via email
Question 5.2: Does your website and documentation describe what support, if any, you provide to users and developers?		Yes, assistance with running system
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, in GH (Issues)
Question 5.6: Is your project's ticketing system publicly visible to your users, so they can view bug reports and feature requests?		Yes, in GH (Issues)
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?		No
Question 9.1: Does your software adhere to appropriate accessibility conventions or standards?		No
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
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
Question 10.5: Do you back-up your repository?		Yes
Question 11.1: Do you provide publicly-available instructions for building your software from the source code?		Yes
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, requirements.txt
Question 11.5: Does your documentation list the version number for all third-party dependencies?		Yes, in requirements.txt
Question 11.6: Does your software list the web address, and licenses for all third-party dependencies and say whether the dependencies are mandatory or optional?		No
Question 11.7: Can you download dependencies using a dependency management tool or package manager?		Yes, pip
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, for main and certain branches
Question 12.4: Are your test results publicly visible?		Yes
Question 12.5: Are all manually-run tests documented?		N/A
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?		Yes
Question 13.5: Do you list your project's publications on your website or link to a resource where these are available?		Yes, if we had any, but we have none so 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?		Yes, if we had any, but we have none so 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?		No, class assignment
Question 14.2: Do you have a contributions policy?		Yes, internal
Question 14.3: Is your contributions' policy publicly available?		No
Question 14.4: Do contributors keep the copyright/IP of their contributions?		No
Question 15.1: Does your website and documentation clearly state the copyright owners of your software and documentation?		No, just documentation

Question 15.2: Does each of your source code files include a copyright statement?		Yes
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 licence header?		No
Question 15.7: Do you have a recommended citation for your software?		No
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)?		No, class will be done in one month
Question 16.2: Does your website or documentation describe how your project is funded, and the period over which funding is guaranteed?		No, I paid for this class
Question 16.3: Do you make timely announcements of the deprecation of components, APIs, etc.?		No, our APIs are used by three people.