

Georgia Institute of Technology
College of Computing
School of Computing Instruction
Course: CS 2340 – Objects and Design
Instructor: Pedro Guillermo Feijóo-García, Ph.D.
Project design: Aidan Stickán (Head TA)

Team Project – *Spotify Wrapped*

Objective

The goal of this project is to design and develop a Spotify listening statistics summary website similar to [Spotify Wrapped](#) using Django and the Spotify API.

Description

This is designed to be a fun project that you can have a lot of personal control over. While there are base user stories that everyone must complete that form the core of what the website is, the rest of the stories can be chosen based on the team's personal vision, capacity, experience, and interests.

Each team has to finish a total of **21 story points** from the customizable user stories, with each customizable user story having a set amount of story points based on difficulty. This means that choosing a more difficult user story will not penalize you, as you will be able to do less stories overall! Exceeding the minimum amount of story points **may** make your client TA's grading slightly more lenient.

Grading

The project will be graded based on two core factors: product and process.

- **Product** (35% of your final course grade): Your team will need to **satisfy their client (your mentor TA)**. User stories will be graded as whole-or-nothing: partial functionality will not earn you points!
- **Process** (5% of your final course grade): Your team will need to follow **Scrum** and its different elements. Communication, cooperation, planning, feedback, and balance are essential. **This part of the assignment is graded separately from your first project and counts for half of your first evaluation grade.** Issues that will sanction your team are:
 - Lack of effective communication with the client.
 - Lack of effective communication among teammates.
 - Lack of planning and follow-ups (daily standups).
 - Unbalanced distribution of work.
 - Lack of cooperation and conflict resolution.

At the end of the project, based on these criteria, your “Product” grade will be calculated as follows:

1. **Outstanding** (100% of the grade): *if all of the following are true:*
 - a. Your website is working flawlessly
 - b. Your website satisfies all base user stories
 - c. Your website meets or exceeds the point requirements for customizable user stories
 - d. Your website has an extremely well-designed GUI
2. **Good** (85% of the grade) *if any of the following are true:*
 - a. Your website has minor bugs.
 - b. Your website is 1-2 points off the points requirement for customizable user stories
 - c. Your website has minor flaws in its GUI design.
3. **Okay** (70% of the grade) *if any of the following are true:*
 - a. Your website has some bugs.
 - b. Your website doesn't fully satisfy any of the base user stories.
 - c. Your website is 3-4 points off the points requirement for customizable user stories.
 - d. Your website has noticeable flaws in its GUI design.
4. **Flawed** (50% of the grade) *if any of the following are true:*
 - a. Your website has major bugs and functionality issues.
 - b. Your website doesn't fully satisfy 2 of the base user stories.
 - c. Your website is 4-7 points off the points requirement for customizable user stories.
 - d. Your website has major flaws in its GUI design.
5. **Rejected** (0% of the grade) *if any of the following are true:*
 - a. Your website's functionality is severely impacted by bugs.
 - b. Your website doesn't fully satisfy 3+ of the base user stories.
 - c. Your website is 8+ points off the points requirement for customizable user stories.
 - d. Your website's GUI design severely hinders its usability and function.

Disclaimer: This is a *very simple* rubric. The cornerstone is that you and your team *communicate* with your mentor TA. Be **proactive** when reaching out and showing your progress on the project!

Individual penalties

Not actively participating in the project **can and will lead to penalties applied to your individual grade.**

Some things that may lead to you being penalized (this is **not** an exhaustive list):

- Skipping team meetings or standups
- Skipping class (without a valid excuse)
- Not contributing an adequate amount to your project
- Mistreating your team members or mentor TA

We operate on a three strikes system, with the penalties for each strike being outlined below. We reserve the right to apply multiple strikes at once.

Strike 1: You will receive a warning to improve your behavior and/or performance

Strike 2: You will receive a 30 point penalty on your project grade

Strike 3: You will get a 0 on the project and potentially be removed from your team

Strikes are decided on by the instructional team, and **are not able to be appealed.**

Repeatedly reaching out to the head TA, section lead, or professor to contest a strike will result in further punishment.

Outside Resources

We encourage you to use ChatGPT and other generative AI tools to help you develop. We also encourage you to use the internet and resources like StackOverflow to help you debug issues and find helpful code snippets.

We are okay with the use of some outside libraries, but **you must ask your mentor TA** before adding them to your project. Failure to do so may result in a penalty. Since we want you to gain experience requesting and processing data from an API, **we will not be allowing any libraries that wrap the Spotify API (e.g., Spotipy).**

Copying work from other CS 2340 groups is plagiarism and will result in **both groups** receiving a 0 on their project and being reported to the Office of Student Integrity.

Getting Help

This is a challenging project! **Please do not be afraid to reach out to your mentor TA if you need assistance.** Their primary job this semester is to help you.

Submission requirements

This project also involves extending your project 1 team website (**this is not your Django web application**) to additionally showcase your project 2 submission. This is a chance for your team to highlight your hard work, creativity, and technical achievements!

Objectives:

You must add the following information to your team website:

1. Team Presentation: Introduce your team, detailing each member's name, role, and contributions.
2. Product description: Elaborate on the web app your team designed and developed and describe how the different screens and features respond to the proposed user needs you selected.
3. Process description: Explain how your team followed Scrum and the design decisions behind your solution: principles, patterns, etc. Use diagrams as tools to communicate your rationale.
4. Video Demonstration: Include a demo video showcasing your web app and its features.

Submission Guidelines:

- All the information from project 1 should still be present
- The team website should be aesthetically pleasing, easy to navigate, and well-organized.
- Ensure the content is clear, concise, and free of grammatical errors.
- The video should be high-quality, with clear audio and visuals.

Deliverables

1. An URL to your team website.
2. A link to the GitHub repo for your project

Project Disclaimers:

- Payment disclaimer: User stories with this disclaimer may require the use of paid services/APIs. We are unable to reimburse you for any associated costs.
- Credential disclaimer: Please do not share any of your personal credentials, even with your teammates. We are unable to reimburse you for any damages that result from you sharing your usernames/passwords/API keys.

Base User Stories:

All projects are expected to satisfy the requirements in these four stories.

Base User Story #1: As a user, I want to be able to view a presentation of the different aspects of my personal Spotify music listening tastes displayed in colorful and fun ways.

Acceptance criteria:

- Able to parse through a Spotify account's data to generate a detailed and creative summary of the user's music listening habits and tastes
- Summary must consist of at least 8 distinct "slides" (transition slides count and are encouraged)

Base User Story #2: As a user, I would like to be able to create and login to an account that saves my previously generated Spotify wraps.

Acceptance criteria:

- Able to create and log into an account associated with your Spotify info
- Able to log out of your account
- Account information is persistent (exists after exiting the website)
- Provide a screen where past Spotify wraps (the entire wrap, not just a summary) can be accessed, viewed, and deleted
- Provide a screen where you can delete your account

Base User Story #3: As a user, I would like the UI to be aesthetically-pleasing and responsive.

Acceptance criteria:

- UI matches your client TA's personal design tastes and expectations
- UI is not hard-coded for a specific screen resolution, will display correctly on different laptop/monitor sizes

Base User Story #4: As a developer, I would like to avoid security leaks by not hosting secrets like API keys on GitHub.

Acceptance criteria:

- No secrets should be committed to GitHub
- We recommend storing all secrets in a file that has been added to your .gitignore to prevent accidental commits

Customizable User Stories (choose any combination, minimum 21 points):

All user stories have a set number of base points you can earn by satisfying the requirements outlined in the user story. Many of the user stories have modifiers (indicated by a +), which are additional point incentives for enhancing the functionality of that user story. You must finish the base user story to earn points from the modifier.

*More difficult user stories will award more points. Earning story points is **all or nothing**, you will not earn any points if you only partially complete a user story.*

1. As a user, I would like to be able to invite a friend to join a Duo-Wrapped that displays and compares both of our tastes in a creative combined way.
 - a. **8 points**
 - b. **+2 points** for saving the Duo-Wrapped
2. As a user, I would like for a LLM API to dynamically describe how someone who listens to my kind of music tends to act/think/dress during my Spotify Wrapped.
 - a. **3 points**
 - b. **+1 point** if you use the LLM API to compare your taste with a friend's during a Duo-Wrapped
 - c. **See the payment disclaimer!**
3. As a user, I would like to be able to access the website on my phone with a well-designed and mobile-friendly UI.
 - a. **8 points**
4. As a user, I would like to be able to generate and save special and distinct holiday versions of my Spotify Wrapped on at least 2 specific days of the year (Halloween, Christmas, etc.)
 - a. **3 points**
5. As a user, I would like to be able to hear clips from some of my top songs play during my Spotify Wrapped
 - a. **3 points**
 - b. **+1 point** for integrating music playback into Duo-Wrapped
6. As a user, I would like to create Spotify Wraps over short, medium, and long terms of my listening history (each of these options must be distinguishable when saved)
 - a. **2 points**

7. As a user, I would like fun games embedded into my Spotify Wrapped that quiz me on my own taste
 - a. Base points scale based on game complexity and creativity:
 - i. Simple/uncreative games = **2 points**
 - E.g. guess your most listened to musician
 - ii. Moderately complex/creative games = **5 points**
 - E.g. guess one of your top songs based on an audio snippet
 - iii. Complex/highly creative games = **8 points**
 - Ask your mentor TA for more information
 - b. **+3 points** for integrating new games (completely distinct from the base Spotify Wrapped ones) into either Duo-Wraps or the special holiday wraps
8. As a user, I would like to view a page where I can post my Spotify Wrapped and see all Spotify Wraps that other users made public
 - a. **5 points**
 - b. **+2 points** for adding the ability to like Wraps and filter by posts you've liked
 - c. **+2 points** for being able to "follow" other users and filter by only their posts
9. As a developer, I would like a GitHub Actions CI/CD pipeline to verify that my changes build and are formatted correctly (according to pylint or pep8) and to prevent PRs with bad changes from being merged into main
 - a. **2 points**
 - b. **+5 points** for integrating code coverage libraries into the CI pipeline that prevent the merging of PRs with less than 80% code coverage
 - i. Tip: We highly recommend starting this user story early and iteratively adding new unit tests with each new PR
 - ii. Note: your codebase must meet or exceed this coverage goal during demo time to qualify. Your unit tests must also be non-trivial, as we will not accept purposeless tests.
10. As a user, I would like to be able to toggle a dark-mode that applies to all UI on the website.
 - a. **2 points**
 - b. **+1 points** for adding the ability to swap to another visually distinct theme of your choice
11. As a user, I would like to be able to click a button to easily share my Wrapped on social media like Instagram, LinkedIn, and X (Twitter)
 - a. **2 points**

12. As a user, I would like to have the option to toggle between English and at least 2 additional languages for the UI
 - a. **3 points**
13. As a developer, I would like to run my Django website on a hosting service (like Heroku) so anyone can access my Spotify Wrapped application
 - a. **5 points**
 - b. **See the payment disclaimer!**
14. As a user, I would like to see transition animations between each Spotify Wrapped page and hover effects on some of the pages.
 - a. **2 points**
15. As a user, I would like to be able to access a “contact the developers” page where I can read more about the development team and easily send them feedback.
 - a. **1 point**
16. As a developer, I would like all functions to have proper documentation so I can easily understand them at a glance (use docstrings for Python functions and JSDoc for JavaScript functions)
 - a. **1 point**
17. As a front-end developer, I would like to have a wireframe-flow diagram on Figma or Balsamiq for all planned screens that clearly guide me in implementing the website’s UI.
 - a. **1 point**
 - b. Note 1: to earn the point for this story, you must show your finalized UI flow diagram and Figma wireframes to your TA by the end of sprint 1
 - c. Note 2: This is what we refer to as top-down software development.