CT Explored Scavenger Hunt App

Sprint Three Report

Mad Hatterz

3/18/2024

Team members: [redacted]

Functionality

During this sprint, the team was able to successfully implement the leaderboard functionality. We pull the leaderboard information from the database and into the app asynchronously, and then parse and render the information. We also update the leaderboard web snippet and rehost it on a set schedule via a CRON job to reflect those changes. In addition, the team faced a major roadblock due to not having access to the Apple Developer ID account. All remaining stories planned for this sprint were only partially completed due to lack of access to the Apple Developer ID.

We planned for 21 points. Out of those we accomplished about 13 points. We completed the leaderboard story, which was worth 5 points. In addition, we completed part of the two Database stories. There were two new tables created, one to handle the information that would need to be sent to the leaderboard and one to take care of user information and handling of the Instagram tokens. We were able to create those and begin writing the code that will handle the connection and updating of the Instagram tokens, making them refresh over a 60-day period instead of 24 hours. (These two stories were worth 10 points total, half of each done, would be about 5 points). We were also able to get a little work done on the Instagram story, with the posting and getting a hashed value (worth 5 points, accomplished about 2 of the 5).

Demo of the App:

[see attached video]

Individual Contributions

During this sprint, [redacted] worked on the leaderboard functionality, creating a web snippet and hosted it via firebase to show everyone's score (in the future, this can be embedded in CTExplored's website) and created a GitHub action that retrieved the users' Instagram usernames and scores once an hour to reflect changes in Supabase, all while reducing API calls to the DB. Along with this, [redacted] also wrote a script within the mobile app to retrieve the Leaderboard info (Instagram username and score) from the userInfo table in Supabase (New table Created by [redacted] and [redacted] as a public table so the leaderboard can pull the necessary information). After retrieving and parsing the data, [redacted] rendered the data to the front-end (leaderboard screen) from highest to lowest score. [redacted] helped him a bit by creating a toggle that would be able to sort the points status HIGH to LOW or LOW to HIGH, and [redacted] refactored [redacted]'s work a bit. [redacted] and [redacted] worked on the Supabase Database side of the project, implementing a new table and editing the private table to deal with the authenticated users. They also used SQL to create the parts of the table that would handle the Instagram tokens and refresh them during a set time frame. The code for that app side has been written as well by them but is yet to be tested due to lack of Apple ID (as mentioned in the beginning, under functionality).

[redacted] and [redacted] worked on the Instagram story, beginning to implement the authentication and linkage between Instagram and the Scavenger Hunt App. Since there was no access to the Apple Developer ID, they worked with [redacted]'s Instagram account & a generic ctexploreddev account for testing of the flow. They were able to create a caption a user could copy paste and a link into the Instagram app. From there once the user created a post a listener would capture their return to the app and trigger a request to the Instagram Basic display API, from there we are able to decide how many points the user should receive based on the # of posts with the appropriate @ and hashtag.

Customer's Feedback

When meeting with [redacted], we informed her that we implemented the change to the app so that the user would not need to log in if they did not want to. They would only need to log into Instagram if they wanted to post. We also made the access to Instagram a lot more intuitive, so that if the user clicks on the Instagram logo, it'll have them log in. In addition, we had asked her to work on getting the team access to Apple ID Developer account, so that we can create better progress going forward. She informed us that she would get that done and needed specific paperwork from her organization. At the time of writing this report, we still had not gotten access to the Apple ID but there has been some progress made towards getting it.

What Worked Well

The aspect of the sprint that worked well for us was that we worked to our maximum potential, doing as much as we could despite our blocker with [redacted]. While many of the user stories were only partially completed, they were completed as much as they could've without accounting for the blocker. We have a workflow and knowledge of the approaches we need to apply in our Sprint 4 work already, so when [redacted] comes back to us with Apple Developer credentials, we will be able to quickly resolve the blocker. Also, we were communicative with [redacted] with the Apple Developer credentials, reminding her urgently, yet professionally, that obtaining them is key to moving forward with our work.

Problems Faced

One of the biggest problems we faced was the main road blocker of not being able to get the Apple Developer ID. The only thing not relying on that was the leaderboard, which was able to get done successfully. All other functionality for this sprint, such as creating the Instagram authentication and working with the SupaBase Database, relied on our team having access to the Apple Developer ID. Despite the team not having this access, we were able to work around it and create meaningful progress towards the sprint goals.

Lessons Learned

One of the lessons we learned was flexibility when it comes to issues popping up. We faced yet another major roadblock during this sprint, and it was important for us to stay flexible and do as much work as we could, despite the issue. We worked well when it came to figuring out how we can work around the roadblock and what we could do to continue to make progress during this sprint.

Changes to be Made

Based on our experience, we will more conservatively plan for points and user story functionality in Sprint 4. This will allow us time to catch up with our work as impacted by our blocker (obtaining the Apple Developer credentials), and to apply [redacted]'s desired functionality in a timely, productive manner. Scaling back our work will allow us to have clearer roles and maximize our communication and productivity with each other.

Sprint Four Plans

- As a developer, I would like to extend the user DB schema to create a system to store and renew the Instagram access tokens and link to the database. (5 points)

- As a developer, I would like to add Apple ID sign in to the user authentication system to allow for an additional account sign-in option for application users. (1 Point)
- As a developer, I will need to extend authentication flow so a user can link their Instagram account to their app/sign in account and store the necessary token for future requests. (5 points)
- As a user I would like a second walking tour with 5 Points of Interest (POI's) so that the application's tours can be more thoroughly demonstrated. (2 points)

Going into Sprint Four, we plan to accomplish 13 points. The first story combines the two previous stories dealing with the database and will now be combined into one, to finish up the remaining 5 points of the original 10 points. We will continue to work on adding an Instagram authentication/sign once we have access to the Apple ID and storing the token, so the user won't have to re-authenticate every time. Lastly, we plan to add another tour to the app with new POIs.

Challenges Anticipated

Some challenges we might face might be with the Instagram authentication. Once we get the Apple ID (which might take some time), we'll have to be diligent with our time and setting up the authentication as soon as possible. This will need to happen first so that once it's done, we can then finish setting up the backend with Supabase's database and making sure it updates the tokens properly. The Database backend might also provide some challenges, but we hope not.

There is still work to be done with the callback url from the Instagram Oauth request for linking the account. The major issue we have encountered over the past two sprints is Facebooks requirement for a "https://" url for the callback, we have discovered that we need to implement IOS Universal Links and Android App links, these require respective developer accounts & some sort of beta app in the app stores for a unique SHA key for the app to link it to a specific https domain. We plan on configuring this with some throwaway domain/endpoint for the time being as we foresee that communicating with Kathy's team to gain this sort of access to their domain may be a challenge but our research in upcoming sprints may resolve our concerns.