**Read-a-Thon Project Report**

**1. Introduction**

The purpose of the Read-a-Thon website is to support a fundraising event (the Read-a-Thon) commonly used by schools and supporting booster clubs/organizations. My school district’s booster club for the highly capable program runs an annual read-a-thon, which they currently run with offline tools (paper & pen). This project could provide them or similar organizations with an online platform to: 1) further encourage student’s participation; 2) make it easier for students (and their parents/teachers) to track their reading and pledge progress, and; 3) provide another method for supporters to make pledges and donations.

In terms of personal goals, this project would show that I can work with a team to complete a website from start to finish.

**2. Design and Implementation**

**System description:**

One page design with:

*Home*: description of read-a-thon, useful tools such as reading list and book selector (using API)

*Contact:* link to contact information

*Register/Login*: modal to allow for registration and login. As time allows, we will create individual registration/logins by student/parent, and teacher.

*Donate*: modal or other component to collect donation information and redirect to payment processor

**Development in Stages Plan** (from previous report)**:**

1. Static Site (links work, informational, no logins or donate button)

*For Student*

1. Students can register, log in, fill in information & submit. Students may/may not receive feedback after submit.
2. Student functionality increased
   1. Students receive feedback alert after they submit.
   2. They can see what information they have submitted, visuals to encourage them (such as badges, alerts like “You’ve read 10 hours, congratulations!)

*For Donor*

1. Donor button activated – donor can fill out form, submit, and be redirected to paysite to complete transaction

*Reporting/Mongo*

1. Reports can be generated from student’s submissions, which can be sent to class teacher

*Additional Functionality (If time allows)*

* Students can set goals (I want to read X# books/X# hours/raise $X.)
* Main page: Leaderboards on main page of top achieving class(es); Student visuals showing their progress

**Converting from design plan:** The idea for the design came from a pre-existing offline project, as well as research into similar websites. From here I created a wireframe for the index page, with the plan for other pages to follow the same structure.

**Modifications made to plan**: After reviewing initial work with teammate (Josh K) we decided to move forward with a template using the assets already created in the first version of the website. This should give additional time later to work on other aspects of the project, such as login set up for multiple users (student/parent/teacher) and donation/payment set up. We also decided to move forward with a more modern, one-page design, using modals and similar components for login/registration.

**Modules/Libraries:**

* Bookshare API (reference below): we can setup book search by reading level
* **Josh, can you add libraries/modules you intend to use?**

**Website Screenshots:**

**Screenshots of website: Josh, can you add?**

**3. Conclusions**

*Results*: So far, we have a working, static website that will provide students and parents with information about the read-a-thon.

­*Features:*

Platform for 1) students to input their reading and pledges and see their progress, and 2) parents/teachers to monitor students’ activities

*Data captures*:

* Visitor contact information (from contact form)
* Registration and login data
* (Registered users) reading/pledge input captured
* (Donors) donation and pledge input captured

*External links*:

* Social media
* External purchases (Amazon)

*Child Safety*: We are still discussing how best to set up the website with the main user (children under 14) in mind. The user’s online safety and privacy is very important. At minimum, no full names would be used on the site, and donors would be asked to provide the name of the child they wished to donate to. We may also set it up so that an administrator (such as a teacher) registers all users to simplify the user’s experience.

*Choices*: In retrospect, I would have started out with a clearer vision of what my minimum viable project is. I think this would have helped a lot in helping me focus on what I think are the project’s most important features. I also would have started out using a template; as design is not my forte, this would have helped me to save time on initial build and give me more headspace to focus on content and planning for further functionality.

**4. References**

Similar Sites:

<https://www.read-a-thon.com/>

<https://risefundraiser.com/campaign/readathon2018?design-mode=TILE>

<https://p.pledgestar.com/>

<https://prathamusa.org/>

API: <http://developer.bookshare.org/docs/read/api_overview/Request_and_Result_Formats#GradeSearch>

Payment processing: <https://stripe.com/>

Donor information:

<https://www.donorschoose.org/>

<https://org.amazon.com/> - we will be including book titles, and we can link those titles to amazon, using a non-profits amazon smiles account

Child internet safety:

Guidelines: <https://www.ftc.gov/enforcement/rules/rulemaking-regulatory-reform-proceedings/childrens-online-privacy-protection-rule>

Example form: <https://www.read-a-thon.com/copa-compliant.php>