“Free Time” Website Project Proposal

Brandon Edgren (bse3)
Yuchen Mou (ycmou)
Connor Moseley (mosel36)

This project is a website for people that have some free time and are looking for suggestions on how to spend it. We expect the main users of this website to be between the ages of 13 and 30, but other than that it could be just about anyone. We have found one site that offers suggestions of things to do (http://www.wtfshouldidotoday.com/). This site’s method however, is to just provide a randomly selected action that is sometimes a link. It also does not help you to do many of the things it suggests and allows “anonymous suggestions”, some of which can be offensive or inane.

The website we are proposing focuses mainly on how long certain activities are expected to take. It also allows the user to narrow down the possible topics to help increase the chances of finding something they are interested in. When a result is displayed, it will show a description and possibly one or more useful links to more information. If the activity is on the internet then there will be a link to that activity. If the activity is not on the internet then it will be described on the results page or the links will be to descriptions/instructions elsewhere. There will also be the option to show another activity if the current result does not appeal to the user.

In its simplest form, this website could use as few as five pages. The home page will allow the user to set the options and request an activity. The result would then be displayed on a separate page. There will also be a button on the home page for users to propose new activities to be added. This will take the user to a page with a form to input details such as name, description, images, and links. The proposed activity would then have to be reviewed by website administrators before it is added to the collection of activities. There would also be a backend login page for site admins, which would lead to a page displaying all pending activity submissions. The admin could then either approve or delete submissions, which would then be updated in the database.

This project could be taken further by adding user profiles, which opens the door to many more possibilities and and complexities. For example, the site could keep track of each user’s history in order to avoid repeated suggestions. It could also provide more customized results based on previously completed activities or personal info such as age, interests, etc. We could also allow users to rate the activities and show a list of most popular or most recently completed activities. Of course, social media can be incorporated for users to share a task that they have completed. Because this website would work just fine without user profiles, it might be wise to first get it in a completely working state without them. Alternatively, if there are a sufficient number of people working on this, it may be possible to work on user profiles simultaneously, but care must be

taken to ensure that at the very least the website works without the profiles by the end.

The main components that interact with each other are the web page, the database, and the system for proposing/reviewing/submitting new activities. The web page should be as simple and clean as possible, both in appearance and function. The database would keep track of all activities (name, description, category, etc.) and user profiles. Activities could either use a flag to differentiate between approved and pending, or be separated into two database tables.

There are a few good options for web development that might be considered as they all should provide the needed functionality. Ruby on Rails would be a good option and would be a useful tool to have knowledge of if you don't already. However it has been known to be tricky to setup. A second option is Python with the Django web framework which should be easier to setup and maybe to use as well. A third option would be to just use HTML, Javascript (or coffeescript), PHP, and some form of SQL (such as SQLite, MySQL, or PostgreSQL).

The main challenge will be getting up to speed with whatever tools are used, though it shouldn't be hard to find some online tutorials or instructions to help get started. Taking the project in small steps should help as well, which was part of reason for starting with a very simple set of features and expanding afterward.