Lab 4

Team Name: TornadoSharpShooters

Project Name: Rate My Park

Park XML data set

http://vancouver.ca/your-government/open-data-catalogue.aspx

http://data.vancouver.ca/datacatalogue/parkListing.htm

ftp://webftp.vancouver.ca/opendata/xml/parks_facilities.xml

Notes:
Facebook Web API
http://developers.facebook.com/docs/web/

Git repo:
https://github.com/UBCCPSC310/TornadoSharpShooters

USER STORIES:
- 3 sentence description about the story
- written in customer's language (ie "As a ____, I want to _____, so that I can ___)
- acceptance criteria that defines what is meant for it to be "DONE"

Independent, **N**egotiable, **V**aluable to users or customers, **E**stimable, **S**mall, **T**estable

Import and clean data

1. As a user, I would like to have access to the data stored on data.vancouver.ca through an easy-to-use web application UI, so that I can to retrieve correct and updated data on local parks in Vancouver.

Priority: 1
Story Points: 5 (10 - 12 hours)

Summary:

- The web application should pull all the park data from data.vancouver.ca. A timestamp should show when the park data was last updated. The web application should take care of importing and cleaning data from data.vancouver.ca.

Acceptance Criteria:
- Able to retrieve data from http://data.vancouver.ca/datacatalogue/parkListing.htm
- Data from data.vancouver.ca is correct and up to date
- Data is retrieved periodically
- All needed data is retrieved, and not only a portion of it

Tasks Breakdown:
- Retrieve data from data.vancouver.ca site (**maybe** using HTTP GET request) - (3 hours)
- Parse XML data and print to console - (4 hours)
- Schedule tasks with a cron job to update data regularly - (3 hours)
https://developers.google.com/appengine/docs/java/config/cron)

Storage

2. As a site admin, I should be able to access stored data so that I can moderate and edit content.
Priority: 1
Story Points: 8 (16-20 hours)

Summary:
- The web application should store all data within its own storage once its loaded. Site admins should have access to the server in order to modify all content submitted to the site.
-
https://developers.google.com/appengine/docs/java/datastore/jdo/creatinggettinganddeletingdata

Acceptance Criteria:
- Able to access cached, persisted data, without polling the data.vancouver.ca site.
- Able to remove content that users upload to the site.
- Able to modify user uploaded content as desired.
- Able to make backups of data on site.

Task Breakdown:
- Specify the data schema for our database (2 hours)
- Setup database using Google's App Engine (DataStore) (5 hours)
- Create park objects and import the park data to database using JDO PersistenceManager (hours)
- Write and run test cases to verify functionality of park data in Google's App Engine's

Datastore (modifying content, backup, etc) (2 hours)

Data display
3. As a user, I want to see the most recent, up-to-date park data so that I am able to determine which park I want to visit.

Priority: 1
Story Points: 8 (16-20 hours)

Summary:
        - Users should be able to access the latest data from data.vancouver.ca through the web application's user interface. They should also be able to easily compare parks with one another, to decide which parks to visit.

Acceptance Criteria:

        - Showing tabular data for park, personal reviews / ratings of other logged in users
        - Comparing all attributes of two parks on the same screen

Task Breakdown:
        - Fetch park data from database to display into tabular format (2 hours)
        - Create the UI on web page for listing out the parks, and their individual data (8-10 hours)
        - Create the UI for the compare functionality, to be able to select parks and then "Compare" **Lower priority** (5 hours)
        - Implement a compare method for retrieving data for parks from the backend database, based on user input from the front-end UI **Lower priority* (3 hours)

4. As a user, I should be able to see park locations on a map so that I can locate where are the parks, and get proper directions to a selected park.

Priority: 2
Story Points: 8 (16-20 hours)

Summary:
        - The web application should be able to use google maps in order to locate park locations specific to the user's requirements. Mapping specific addresses or mapping parks around a user specified area should all be handled by the web application.

Acceptance Criteria:

        - Draw path from a given location to a specific park
        - Show all parks on single map, and allow clicking on each for more data
        - The points on each map should give the correct location for each park

5. As a user, I would like to see suggested parks so that I can make easily make decisions as to which park to visit next.

Priority: 3
Story Points: 8 (16-20 hours)

Summary:
      - The web application should be able to generate different parks to give non-logged on users suggestions on which parks to visit (Highest ranked, most popular, etc). For logged on users, it will generate parks depending on their previous rated/visited parks

Acceptance Criteria:
      - Be able to suggest the most popular (visited) parks
      - Be able to suggest the highest ranked parks
      - Suggest a list parks that the user has not yet visited.
      - Or any combination of the above, based on user preference

6. As a user, I should be able to use a search option so that I can find a specific park based on a query or filter.

Priority: 3
Story Points: 5 (10-12 hours)

Summary
      - Users will be able to select fields from in a form in order to filter parks by certain features. They should also be able to specify a query to find parks with that query in their name or attributes.

Acceptance Criteria
      - Able to look up parks by their names, or some attributes
      - The results returned after a search should be correctly filtered
      - If unable to match query with any parks, it should show "Parks not found" to the user

Social network integration

7. As a user, I should be able to use Facebook so that I can Like, Share, and create events for various parks.

Priority: 3
Story Points: 8 (16-20 hours)

Summary:
      - The web application should be able to be linked with Facebook to provide functionality specific to it for social networking purposes, e.g. being able to use the web app to "Like" parks or to share parks with friends/family.

Acceptance Criteria:

> - Should be able to like a park by clicking on a Like button
> - Sharing a park should post the park to the user's Facebook wall
> - Schedule and share events through Facebook to other users

Access

8. As a logged in user, I would like to login using my site account so that I can access my visited and personally preferred parks, reviews, etc, conveniently.

Priority: 2
Story Points: 5 (10-12 hours)

Summary:

> - The web application should allow users to access the web app's functionality and content (such as preferred parks, reviews, etc.) by logging in with an account created through the web app.

Acceptance Criteria:

> - After logging in, should be able to view account specific data such as name, email, parks I've liked, reviewed, etc.

9. As a logged in user, I want to be able to edit personal information saved to my account so that I can ensure my account is up-to-date and secure.

Priority: 1
Story Points: 13 (26 to 30 hours)

Summary:

> - Users will be able to login to the site. Logged in users can add and edit personal information which will be saved by the application.

Acceptance Criteria:

> - Able to create a new account and protect it by specifying a password
> - Able to log into the web application via webapp account after the correct password is entered
> - Able to add, edit and view personal information to the account only when logged in

Task Breakdown:

> - Create User objects as Java classes, and add/update user data on database using the PersistenceManager framework (8 hours)
> - Create UI for login page, and account management (8 hours)

       - Implement password authentication and validation using the jBCrypt framework (6 hours)
       - Create test cases to test user account creation, input and login functionality (2 hours)

10. As a logged in user, I would like to rate and review parks that I have visited so that I can offer recommendations to other users.

Priority: 2
Story Points: 8 (16-20 hours)

Summary:
       - The web application should allow logged in users to input ratings/reviews for parks of their choosing through the web app's user interface. Other users will then be able to see these reviews when they visit the corresponding park page.

Acceptance Criteria:
       - Able to input and submit a rating and review for any park that the web app knows about.
       - Able to browse ratings and reviews posted by others.

EXTRA THINGS TO CONSIDER!

       - allow users to upload photos of parks

<u>SPRINTS</u>

Sprint goals:

For Sprint 1, our goal is to ensure that we have a working demonstration of our web application, i.e. that we can import data from our data source, view and access park data through a basic, functional UI, and implement a working login screen.