

Course Project: Tripster

Your goal in this project is to develop a site to plan trips together and share every aspect of it. Your project will exercise the following aspects: schema design (relational, XML) and view-based access control; cloud hosting; XML interchange; REST and JSON data interchange; SQL-based recommendation; caching in a key-value store; performance evaluation. The project is designed to offer lot of flexibility in the features and to think about various intricacies in implementing them.

For starters, the application should have the following features:

1. Joining as a user
2. Add friends
3. Create a trip
4. Invite people to the trip
5. Sharing photos/albums among the group
6. Notify when your friends are going for trips and request to join
7. Comment and like pictures in the trip
8. Publish photos
9. Privacy settings for albums
10. Rate and recommend locations, friends based on their trips.

You can extend the project and add every aspect from day one of the trip to the last day, in making a schedule, search for accommodation, places to visit, expenses tracker (like split wise), have a check-list before the trip. Conceptually, the site should be designed along the lines of any social networking site, but restricted to trips. It will be your choice as to which details you want to emphasize: Photos, videos, expenses, schedules, hotels etc. However, your platform should support sharing of hyperlinks to any kind of external content (Like pictures of the locations to visit, website links to hotels, Amazon links for items to buy etc.). Even though the application makes much more sense in the mobile version, the intent of the project is to help you understand the importance of good database design in implementing and extending your features.

Phase I: Form a Team (size 3 or 4) and Set up Infrastructure [October 29th]

The initial step is to select three teammates and do the following:

1. Determine whether you wish to standardize on Node.js or whichever technology you want to work on . Your group should all adopt the same platform.
2. One of your group members should set up Subversion/Git for sharing of source code and starter data files. See <http://www.seas.upenn.edu/cets/answers/subversion.html> for subversion and be sure to grant everyone in the group permissions!
3. Upload a PDF document via Canvas, notifying your course Instructors who is in your group and what features you plan to working on (Including extra credit features if you are planning on implementing them).
4. A timeline of how you plan different milestones of your project and division of responsibilities.
- 5.

We will give you four Amazon Web Services tokens, which each grant \$100 in usage credits. You should expect that each credit can only be applied to a single ID, i.e., you likely won't be able to apply all of the credits to the same user account. Each of your team members should register with Amazon at <http://aws.amazon.com/>.

Phase II: Schema Design and AWS Database [November 12th]

In the next phase, you should design a relational schema to capture the various aspects of the application:

1. User profiles, including ID's , encrypted passwords, affiliation, and other information such as the locations visited and a dream list
2. Friends
3. Trips (Add locations and invite people)
4. Content sharing (Photos, Videos, list of items to take)
5. Searching for people/places
6. Create albums within the trip
7. Rating a trip, photos
8. Comments on a trip or photos
9. And anything else you think is of value!

*NOTE: Privacy settings on user profile, trips, content sharing (and any other extra credits you implement) should be provided.

You should use the AWS Getting Started handout to create your own Oracle database on Amazon RDS. For the milestone you should submit a text file with a full JDBC/SQLPLUS connect string, including guest user ID and password and database schema name, to us via Canvas. (From this we should be able to dump your SQL tables.)

Phase III: Data Import / Export [November 19th]

Now that you have a baseline schema to work on, the next part is to integrate content to and from a standard interchange format used by the class. This will allow us to collectively build a collection of users, trips, places, photos and links.

Phase IV: Project Initial Deliverables [December 3rd]

By the time you reach this phase, we expect you to have a running application with the basic features that have been mentioned initially in this documentation. Submit the source code and a brief document of the list of features through Canvas.

Phase V: Project Final Demonstrations [December 15th-18th]

For the course project, you should agree from the beginning on specific project component responsibilities – each person should have aspects that they “own” and are responsible for. “Own” doesn’t necessarily mean they will be doing all of the coding / development, but rather that they are responsible for making sure the feature is complete.

The components can be outlined as (but not restricted to):

1. User Profiles and Login

As with any social Web site, you should have support for the creation of user accounts, including profile information. Things like affiliation, interests, etc. might be useful. Once the user has an account, he or she should be able to log in. This should be done in a secure way. The user’s home screen should show content, with a portion dedicated to his or her own content; and a portion dedicated to other content that might be of interest. There should also be a search capability.

2. News Feed

You should develop metrics that, given a user profile, determines content likely to be of interest (depending on the time of update and interest).

This metric will return a score for each piece of content based on:

1. Your photos/trips
2. Friends photos/trips
3. Comments/ratings on photos
4. Recommend users who have similar interests
5. Recommend trips based on their previous trips
6. Recommend items for trips based on the places visited, or most number of items bought
7. Estimated expense for your current trip (if you have an expense tracker).
- 8.
9. This is not an exhaustive list and can depend on any factors you wish to consider based on the features you are implementing.

You should periodically regenerate the news feed for each user based on the metrics.

3. Caching

We will use a NoSQL database (e.g., MongoDB, Amazon S3) to implement caching. We will be caching multimedia which is most frequently accessed.

Like top news feeds, most visited places, most liked images, popular items or any other data you feel needs immediate access. It is up to you to decide which ones you would like to cache. The basic idea is to improve performance and have control over your top data.

4. Search and REST Search

Users should be able to search for places, people or anything of value from your database (extra-credit for web search)

Extra credit:

1. Use the Yelp API to find hotels and good places at locations.
2. You can trigger Bing Search, see <http://datamarket.azure.com/dataset/bing/search>, to return additional results for place/image searches.
3. Expense tracker like Splitwise
4. Import user information from Facebook.
5. Anything you think is intuitive and adds great value to the application.

PS: Even though Front-end enhancements are appealing for the application, emphasis for extra credits is on back-end.

Final Deliverables:

The entire code has to be zipped and submitted along with the following report.

Experimental Validation and Report

A modern software infrastructure project isn't done until you understand how it performs, and where the bottlenecks are.

Instrument your application to collect timings on various aspects. You should at least be able to determine what the latency in handling each request is, and extra credit will be awarded if you can also see what happens under multiple concurrent requests.

Your final report should include a write-up of:

1. Introduction and project goals
2. Basic architecture (not a dump of the classes)
3. Key technical challenges and how they were overcome
4. Performance evaluation
5. Potential future extensions

All the deliverables have to be submitted on Canvas by December 16th 11:59 pm.