

CompSci 316 - Project Report

Meng Yuan, Shuai Yuan, Chengen Xie, Shangru Huang

Open Project Description

By Fall 2014, there are totally 6,417 undergraduate students at Duke with 10% international students. Besides, there are 8,379 graduate students and 24% of them are foreign students. Therefore, Duke totally has about 2,653 international students. Unlike native students, studying abroad is never an easy thing for international students. All the difficulties start from arrival – how to get to the apartment/campus from airport. Duke Chinese Student and Scholar Association (DCSSA) has a long history for providing free airport ride for incoming Chinese students. But so far all the volunteer-student match is manual work. Every summer, DCSSA needs to organize a special team to handle hundreds of requests from incoming students and assign them to available volunteers, which is inefficient, tedious and error-prone.

Therefore, a fully automated volunteer-student matching system would be necessary and beneficial. Our group is intended to design a web-based airport-pickup application, which can largely simplify the process of volunteers and new students matching. Furthermore, we hope this tool can be further extended to help more students.

Data Sources

After contacting with DCSSA, we can have access to previous new students and volunteers' information (They are stored in some Google Docs), which can be used to populate our database.

Data Samples:

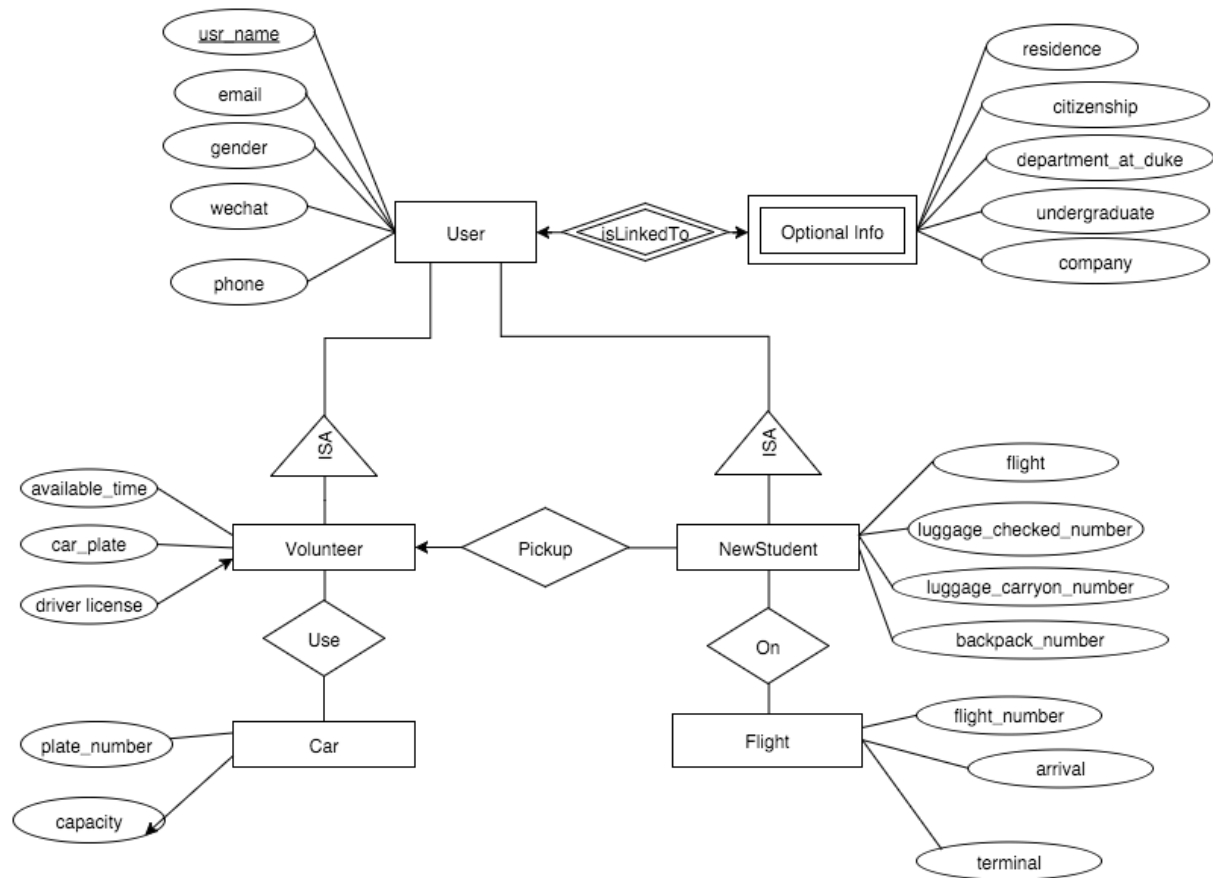
New students: goo.gl/gZhb92

Volunteers: goo.gl/T2ysBp

Implementation Details & Assumptions

- To register with the Web site, a user must provide name, email and intended identification (new_student or volunteer), optional extra information like major, age, gender, hobby, etc.
- Each volunteer is capable of driving different cars indicated by car plate referencing plate number of the car.
- Each volunteer need to provide availabilities (available time period). The system will provide optional time slots, which can reduce the trouble of parsing unpredictable user inputs.
- Each new student should provide arrival time, flight number (referencing flight number in table Flight), number of people, number of total luggages.

E/R Diagram (UPDATED)



System Architecture & Platform

In our project we are planning to use Django as the framework of our web application. Django is a open-souce framework written in Python. Comparing to other existing frameworks, Django is lightweight, fast and scalable. Therefore, it would be our optimal choice for a agile prototyping project approach. One of the distinguishing feature of Django is the model-view-controller architectural pattern. It also come with many bundled applications that we might need in our project including authentication system and GIS applications frameworks (for example, building a GIS visualization measuring the distance of free ride provider and the RDU airport.)

As for the backend database, Django is compatible with multiple existing SQL database including PostgreSQL, MySQL, SQLite and Oracle. Initially we are planning to use SQLite for our database. Other possible choices might include Postgresql and MySQL. By default, the configuration of Django uses SQLite. If we want to use the PostgreSQL, install the appropriate database binding, and change the some keys(including ENGINE - Either and NAME) in the database 'default' item to match the database connection settings.

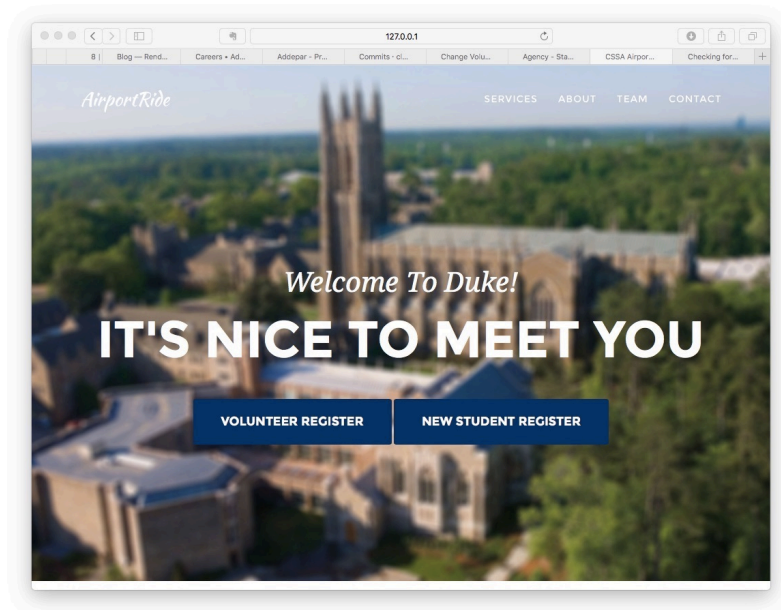
For the User Interface part, once our models are defined, Django can create a professional interface called administrative interface automatically, which is a Web site that allows authenticated users manipulate objects. A beautiful URL scheme is critical to a Web

application. For designing URLs for our web app, we can use the Python module called URLconf, which provides a simple mapping between URL patterns and Python callback functions.

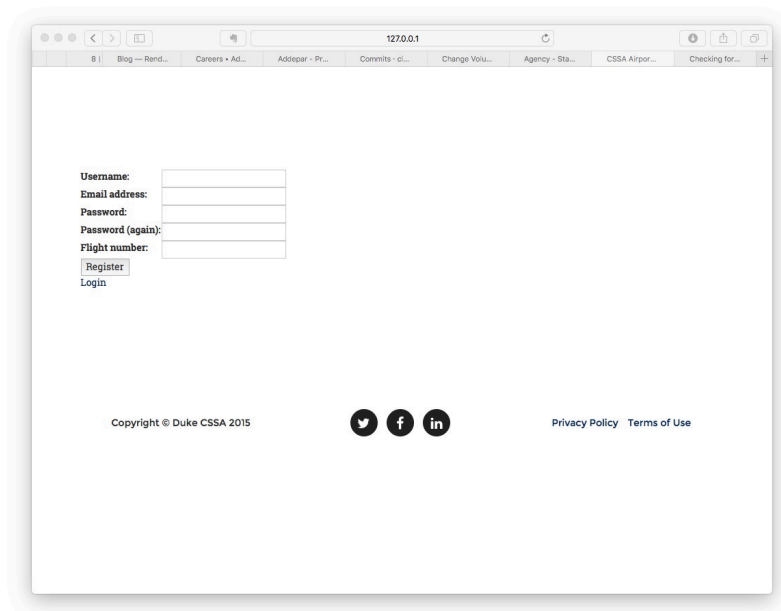
Progress Summary

- Built fundamental knowledge of Django framework and web application design.
- Successfully designed and implemented registration and login processes
- Creatively built the back-end database with SQLite and tested it with sample data.

Initial page:



Login page for new students:



Login page for volunteers:

A screenshot of a web browser showing the login page for volunteers. The browser's address bar displays '127.0.0.1'. The page has a header with navigation links: '8 | Blog -> Rend...', 'Careers + Ad...', 'Adcepap - Pr...', 'Commits - ci...', 'Change Volu...', 'Agency - Sta...', 'CSSA Airpor...', and 'Checking for...'. The main content area contains a login form with the following fields: 'Username:', 'Email address:', 'Password:', 'Password (again):', and 'License number:'. Each field is followed by a text input box. Below the 'License number' field are two buttons: 'Register' and 'Login'. At the bottom of the page, there is a copyright notice 'Copyright © Duke CSSA 2015', three social media icons (Twitter, Facebook, LinkedIn), and links for 'Privacy Policy' and 'Terms of Use'.

Administration page:

A screenshot of the Django administration interface for a project named 'Airportapp'. The browser's address bar shows '127.0.0.1'. The page has a dark blue header with the text 'Django administration' and a welcome message 'Welcome, siechengen. View site / Change password / Log out'. Below the header, there is a breadcrumb trail 'Home > Airportapp'. The main content area is titled 'Airportapp administration' and contains a table with the following rows: 'Cars', 'Flights', 'NewStudents', 'Pickups', 'RideUsers', and 'Volunteers'. Each row has two links: 'Add' (with a green plus icon) and 'Change' (with a pencil icon).

Future Work

- Improve User Interface design.
- Implement the match function for new students and volunteers.
- Add more social network login method, like Duke NetID, Wechat etc.
- Optimize database design or try to improve performance of website

Feedback & Response

TA: For each user, it is very likely that he or she is unwilling to disclose the extra/optional information when registering.

Response/Updates: Update E/R diagram and back-end database by splitting optional registration information from user table.

TA: It is worthwhile to check out the Duke NetID Auth API from Co-lab so users do not have to remember another pair of account password and alleviate the burden of account security. It is available in Mainland China as oppose to Google or Facebook.

Response/Updates: We are currently learning about Duke NetID Auth API and Wechat API and trying to apply them to our user registration.

References

1. https://duke.qualtrics.com/jfe/form/SV_dbtFkV1ohJUe8ex
2. <https://docs.djangoproject.com/>
3. <https://wingz.me/>