Northeastern University- Red Eye Registration Automation application

Authors

Priyanka Shenoy

Email : shenoy.p@husky.neu.edu

NUID : 001680531

Sameer Motwani

Email : motwani.s@husky.neu.edu
NUID : 001791310

Sravani Beeram

Email : beeram.s@husky.neu.edu

NUID : 001705306

Anvita Surapaneni

Email : surapaneni.a@husky.neu.edu

NUID : 001626386

Abstract

The purpose of the application is to create a centralized shuttle service system which will provide ease of access to the students and organizers of the service.

Introduction

- We are building a portal which students will be able to login with their myNEU credentials and book the time they wish to avail the Red Eye service.
- The student while booking provides the destination he wishes to go to.
- He / She can cancel the booking up to 15 minutes before his / her travel time.
- There will be an additional wait-list of 10 students who can go in the shuttle if there is any cancellation.
- The driver gets a list of all students who are going to travel 10 minutes before the schedule time. This way he does not need to check student ID, ask for destinations or inform his superior about the details of the travel since all this will be managed by the system.

- This system ensures students and drivers both don't have to go through the trouble which they have to now.

Requirements

The actors that make use of the application are the NUPD (Northeastern University Police Department), the student of University and Driver of the Red Eye. The following are the use cases:

- *Register for a ride at a given time*
  - ➢ Student registers himself/herself on a ride at a particular time (one hour before ride).
  - ➢ System books seat for a student.
- *Request waitlist slot*
  - ➢ Student register himself/herself for a waitlist position (max 10)
  - ➢ If number of waitlist positions are less than 10, student is registered on waitlist by the system.
- *Cancel ride*
  - ➢ Student can cancel his/her ride or waitlist 15 minutes before schedule time.
  - ➢ System updates status for other students on waitlist.
- *Waitlist cleared confirmation*
  - ➢ System allots the next student on waitlist a seat in case of cancellation.
  - ➢ Student can take ride at the time slot.
- *Monitoring student information*
  - ➢ Students register themselves for a ride at a time.
  - ➢ Details of all students and their destinations collected by system.

- ➢ These details are provided to NUPD for maintaining security by system
- ➢ NUPD maintains log of students for all time slots.

- *Monitoring Driver Information*
  - ➢ Driver provides his availability during the week.
  - ➢ NUPD allots drivers at different time slots for ride.

- *Monitor details about vehicle*
  - ➢ NUPD maintains log of vehicle registration number and time slots at which the rides are scheduled to happen in a particular van. This is done to ensure security.

- *Notify Start of Ride*
  - ➢ Driver notifies NUPD about start of ride.
  - ➢ NUPD maintains log of the same

- *Notify End of Ride*
  - ➢ Driver notifies NUPD about end of ride.
  - ➢ NUPD maintains log of the same.

- *Notify about absence of student*
  - ➢ Driver notifies NUPD about absence of a student.
  - ➢ NUPD maintains log of absentees.

- *Notify on the spot addition to the ride*
  - ➢ Informs NUPD about new student added on the spot in case of vacant seats if any.
  - ➢ NUPD maintains log for security.

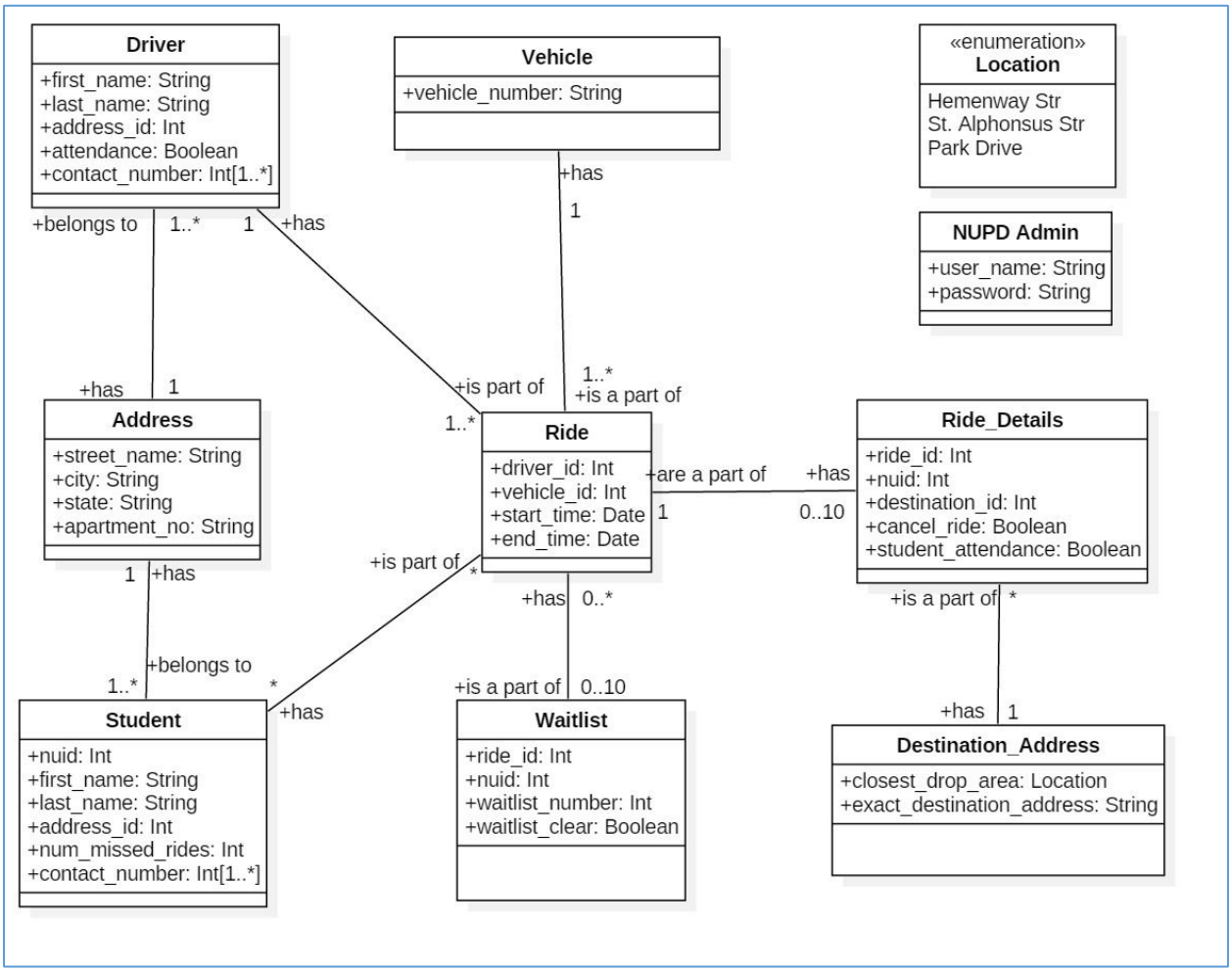
- *Update driver/vehicle information*

  - Adds new recruited driver to system.
  - Adds new vehicle information to system.

- *Penalizing students for absence during ride*

  - If student fails to turn up more than 5 time for rides he registers, he/she is penalized.
  - Student needs to visit NUPD office with valid reasons for absence to revoke penalty.
  - NUPD may revoke penalty if absence is justified.

Design


Driver
+first_name: String
+last_name: String
+address_id: Int
+attendance: Boolean
+contact_number: Int[1..*]
Vehicle
+vehicle_number: String
«enumeration»
Location
Hemenway Str
St. Alphonsus Str
Park Drive
NUPD Admin
+user_name: String
+password: String
Address
+street_name: String
+city: String
+state: String
+apartment_no: String
Ride
+driver_id: Int
+vehicle_id: Int
+start_time: Date
+end_time: Date
Ride_Details
+ride_id: Int
+nuid: Int
+destination_id: Int
+cancel_ride: Boolean
+student_attendance: Boolean
Student
+nuid: Int
+first_name: String
+last_name: String
+address_id: Int
+num_missed_rides: Int
+contact_number: Int[1..*]
Waitlist
+ride_id: Int
+nuid: Int
+waitlist_number: Int
+waitlist_clear: Boolean
Destination_Address
+closest_drop_area: Location
+exact_destination_address: String
+belongs to
1..*
1
+has
+has
1
+has
1
+has
1
+is part of
1..*
1..*
+is a part of
+are a part of
1
+has
0..10
+is part of
*
1
+has
+belongs to
1..*
*
+has
+has
0..*
+is part of
0..10
+is part of
*
+has
1

Implementation

Technologies used

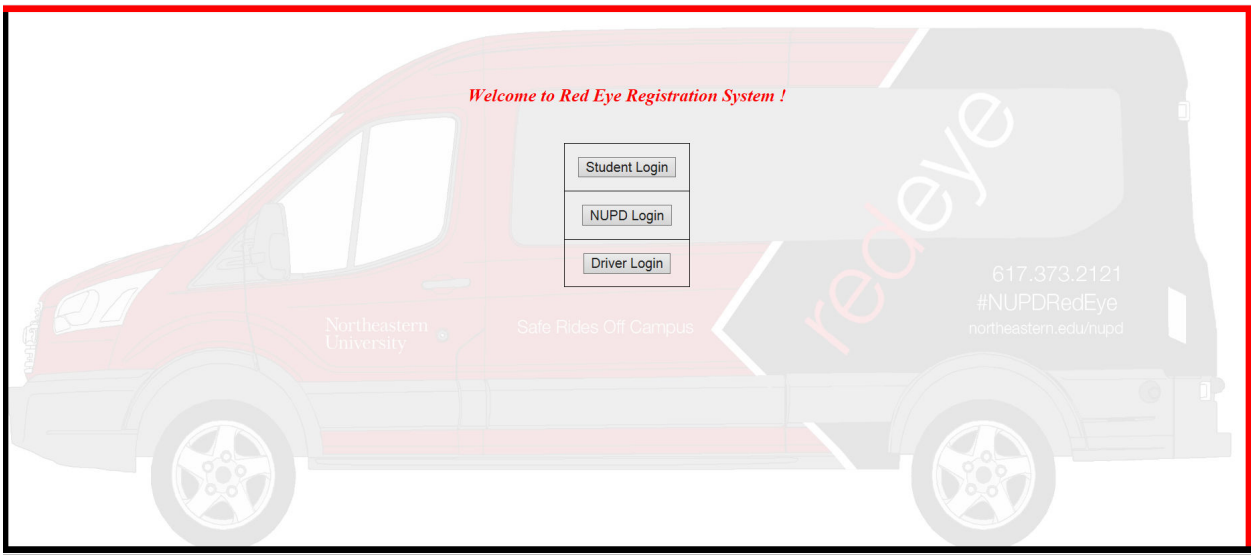
SERVLETS
Front End: HTML, CSS, JSP
Back End: MySQL DB

Architecture

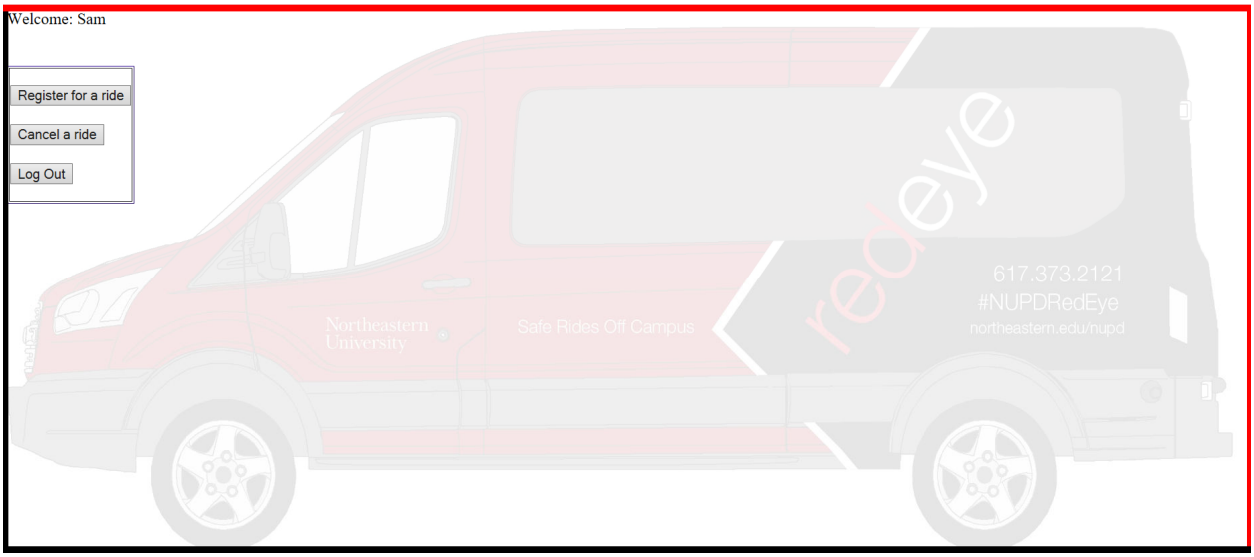
The online systems uses a MVC architecture
The client views and request information via JSP pages
Data is modelled as an object using JAVA classes.

UI Pages Screenshot

Login Page

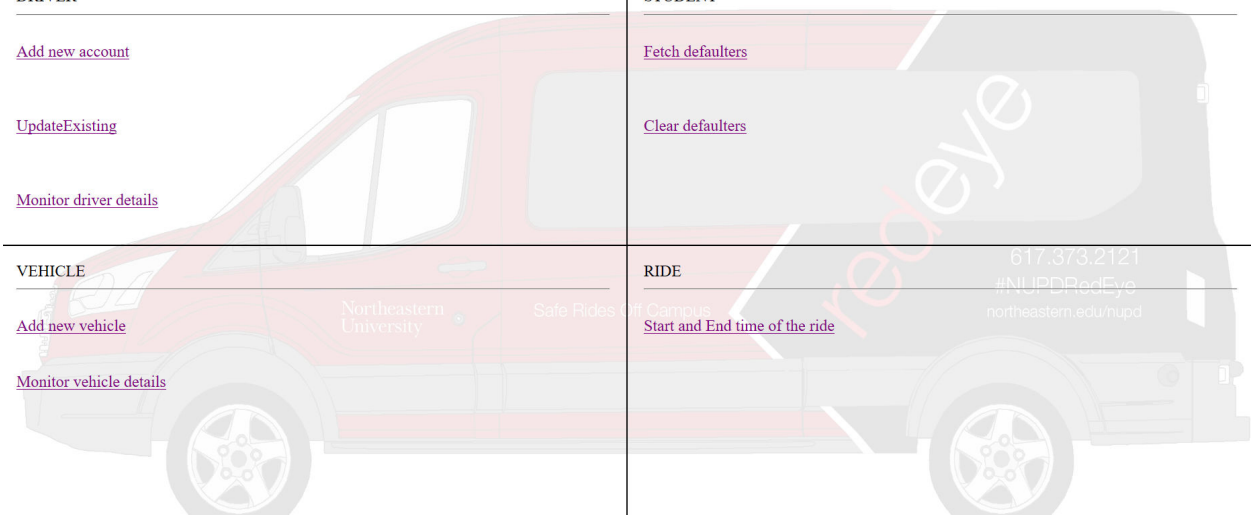


Functionality for student module

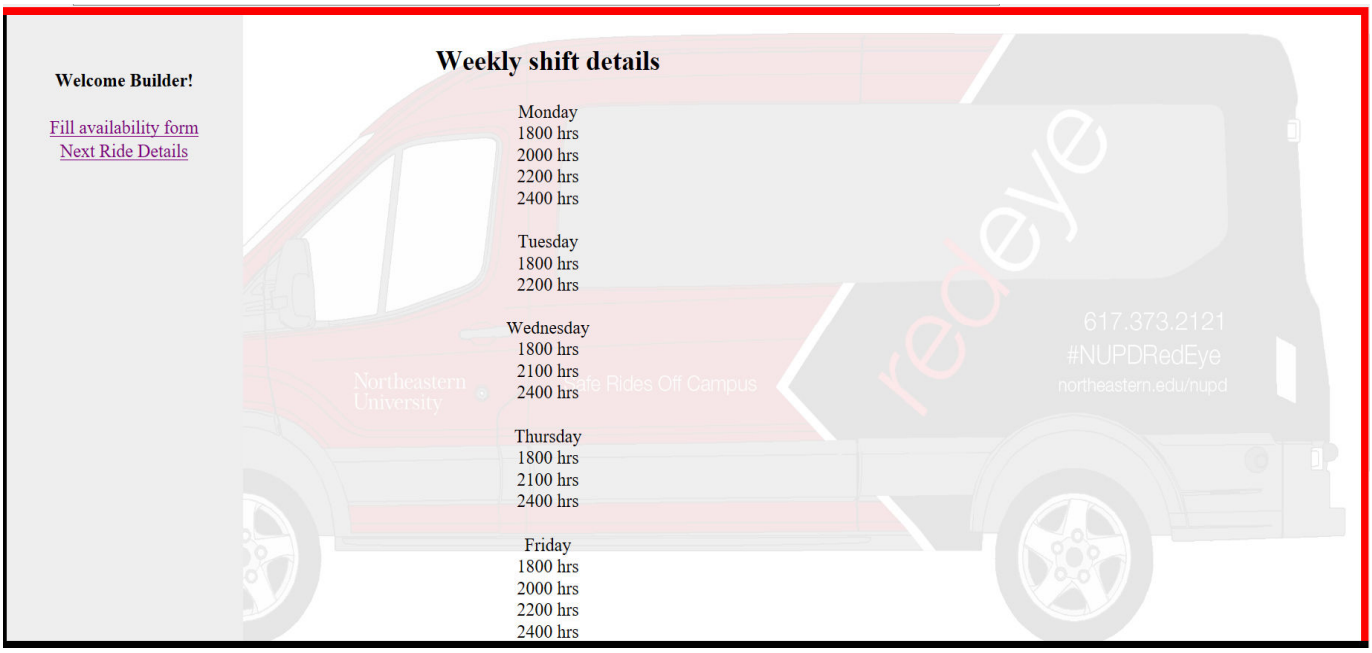


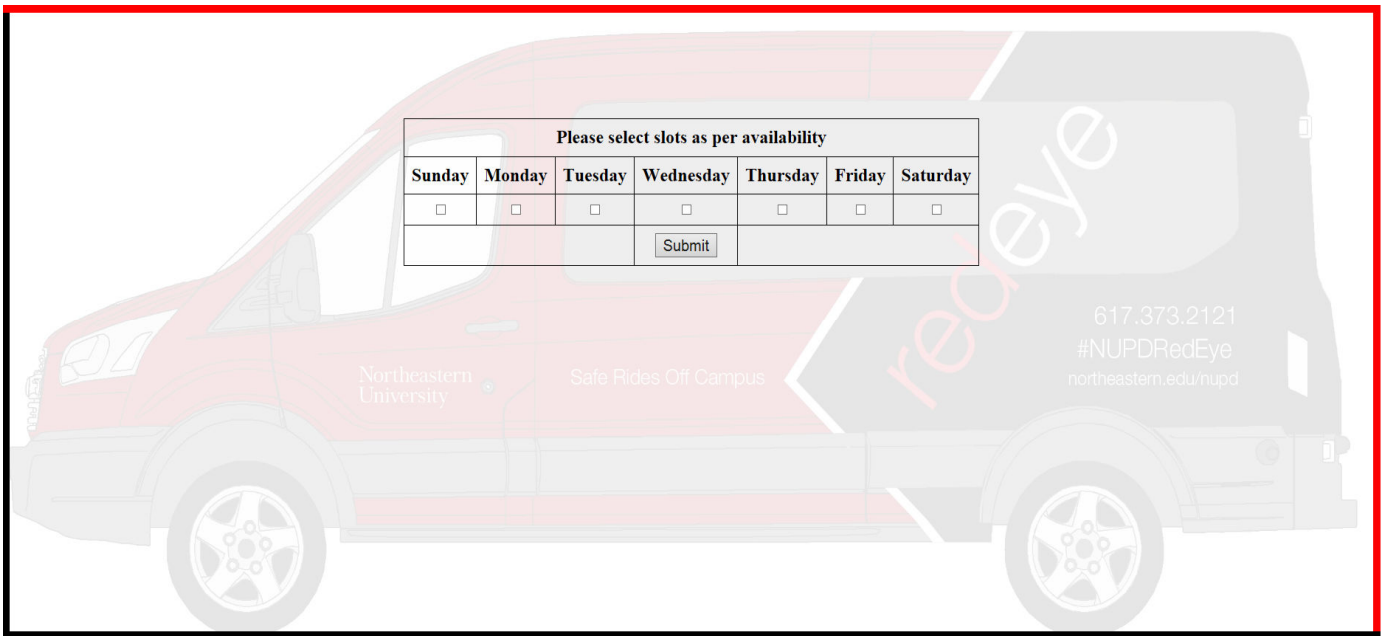
Functionality with NUPD module

DRIVER	STUDENT
Add new account	Fetch defaulters
UpdateExisting	Clear defaulters
Monitor driver details	
VEHICLE	**RIDE**
Add new vehicle	Start and End time of the ride
Monitor vehicle details	

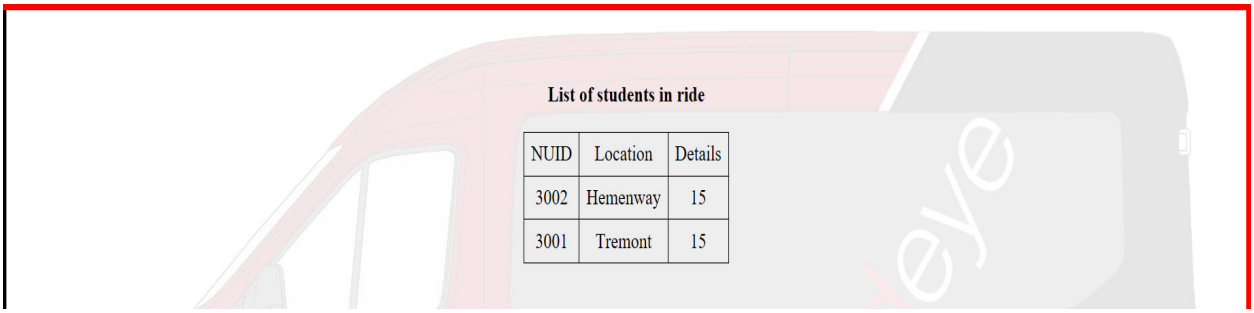


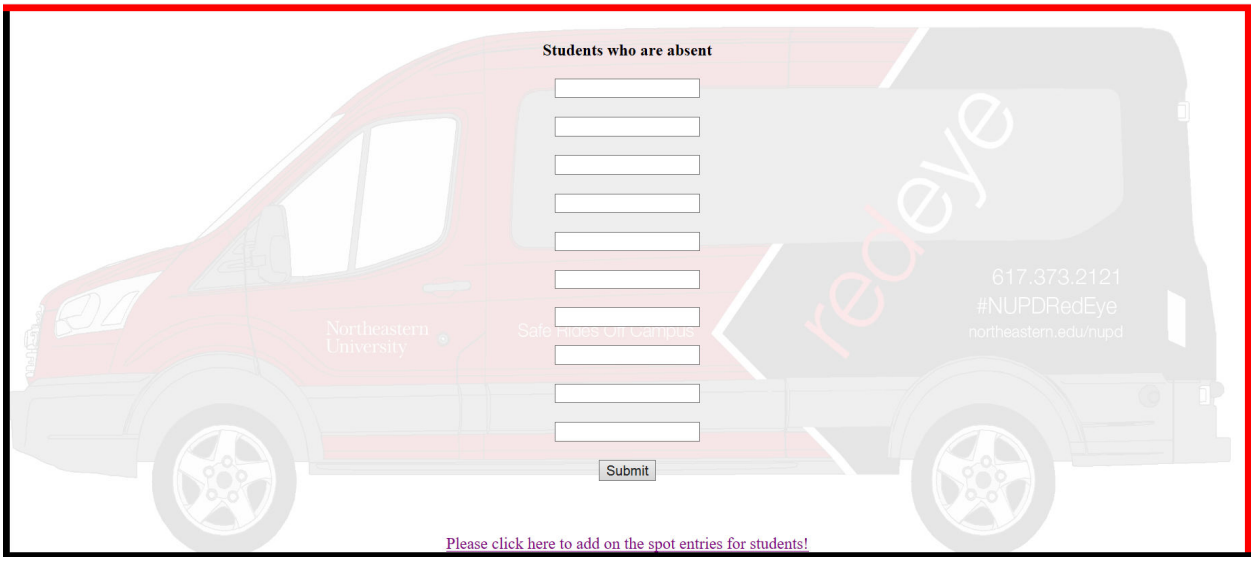
Functionality with driver module

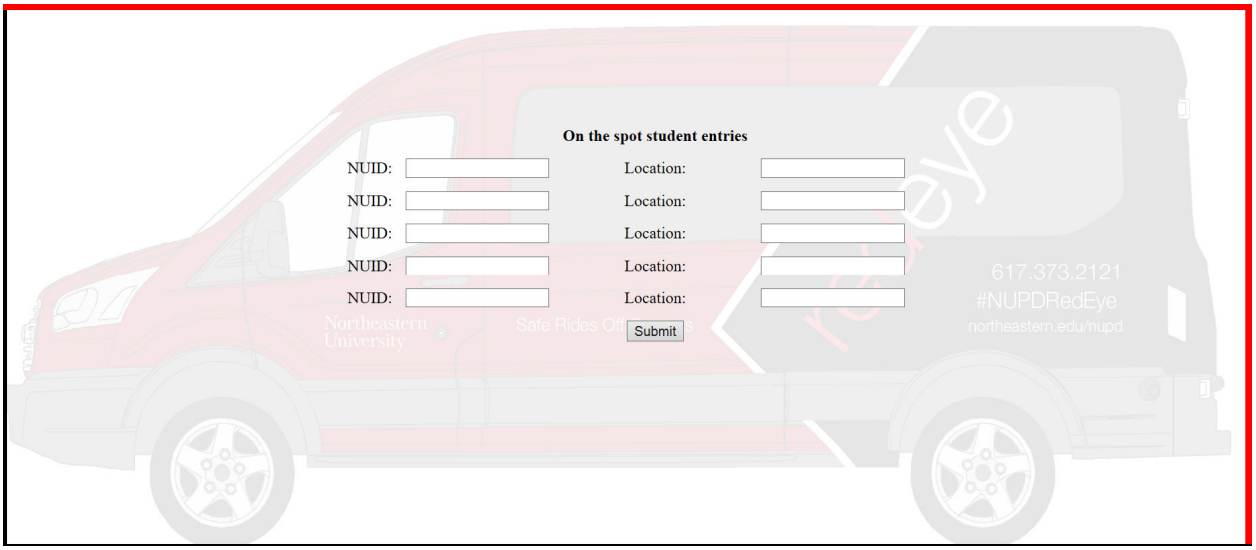




Functionality with driver module







Conclusion

The Northeastern University- Red Eye Registration Automatic application was developed to facilitate the students with an easier way to access the university Red Eye service. Access to the application is via a secure website. The application automates the registration, cancelation, waitlist of ride for the student and also the reporting of student attendance by the Driver. The application also allows the driver to select their available slots and view the students on his ride. He can also add a new student on the spot. The NUPD has the facility to monitor all these interactions and also waiver a student's penalty.

References

- Most of the programming in java was done by referring "**Database Design and Implementation by Edward Sciore".**

- For troubleshooting issues with the program we referred online forums like –
  - ➢ http://docs.oracle.com/javaee/6/tutorial/doc/bnafd.html
  - ➢ https://docs.oracle.com/javaee/5/tutorial/doc/bnagx.html
  - ➢ http://tomcat.apache.org/tomcat-7.0-doc/index.html
  - ➢ http://www.w3schools.com/html/default.asp
  - ➢ http://www.coderanch.com/forums/f-50/JSP
  - ➢ http://www.tutorialspoint.com/jsp/
  - ➢ http://stackoverflow.com/

- We have uploaded the source code, report and presentation for our project at the following locations-

  - ➢ Source Code - https://github.ccs.neu.edu/shenoyp/DBMS_Project.git
  - ➢ Report - https://drive.google.com/open?id=0By6fMKfCLKH9OG4wY1hVYWlTV2M
  - ➢ Presentation - https://docs.google.com/presentation/d/1NcxCHAYpI0Tr3CsQOjd4Bg5nCo2FcQbKTMJzl_UeFX0/edit?usp=sharing