CO550
Web Applications
Coursework 1

Design a Simple Web Database Application

James Harris
21606555

Alex Paulo da Costa
21611431

1st November 2017

Computing & Web Development
Buckinghamshire New University

Contents

Project Outlines

1.1 Enterprise Model

A clerk in head office (or operational manager) sets the routes and times they want for the bus lines.

Drivers are assigned to the routes, reducing the total time of the route to their contracted hours: the closest it gets to 0 the better(hours left to allocate).

Customers look at the timetable for the line, without knowing who the driver is, and choose the bus they want to take,

They board the bus, pay the driver, or show them a previously bought and still valid ticket, and take a seat.

1.2 Business Goals

The goal of the business is to provide customers with reliable, fast and relevant public transportation services, all whilst making the running costs (infrastructures, staff, etc.) as low as possible to make as much profit as possible.

In order to make the service fast and relevant, the company tracks what stops seem to be the most used by customers, either through driver feedback or through the use of the website, and sees what sections and portions of the main lines are used the most at certain times of the day, adapting the routes and services available at certain times of the day according to the user needs (examples of adaptation would be school holiday timetables and term timetables, as well as not running the whole line at certain times of the day).

To make sure that the service is reliable, it constantly verifies if the rotas are properly allocated (no drivers assigned to two different routes at the same time) and checks against traffic patterns and provisions the time that it takes to go between two stops, reflecting that difference in the timetable.

To avoid costs, the company makes sure that all the drivers have their hours allocated properly (not making less or more hours than what they are contracted for), as well as some other systems that are used to make the service fast and reliable (an example would be the adaptation of the lines and routes to the time of the day, by not having too many buses running at a time where they aren't needed).

To provide a better service to their customers, the company now wants to provide the customers with an online service where they can see the updated timetables for their favourite services at all times, provide staff with a service that makes it easier for them to update the timetables and make rotas, where all the information is centralized and where the computer systems eliminate the tedious work of consistency check.

1.3 Functional Requirements

- Enable users to login and register

    - Normal (unprivileged users) can register anytime
    - Privileged users must be added by an admin

- Enable users and visitors to see bus timetables

- Enable registered users to save their favourite routes

    - Requires personal dashboard

- Allow admins and authorised staff to change timetables and rotas

  - Create bus lines and routes
  - Allocate staff to lines
  - Make sure the staff are working the hours they were contracted for
  - Make sure no driver is assigned to multiple routes at the same time

- Allow staff to see their rota

  - Staff assigned to particular routes
  - Staff dashboard allows to print weekly rota as a pdf

- The users should be able to search routes by terms, e.g route code, town names or bus stop names

1.4   Outline of the Project

Diagrams & Schematics

2.1 Entity Relationship Diagram
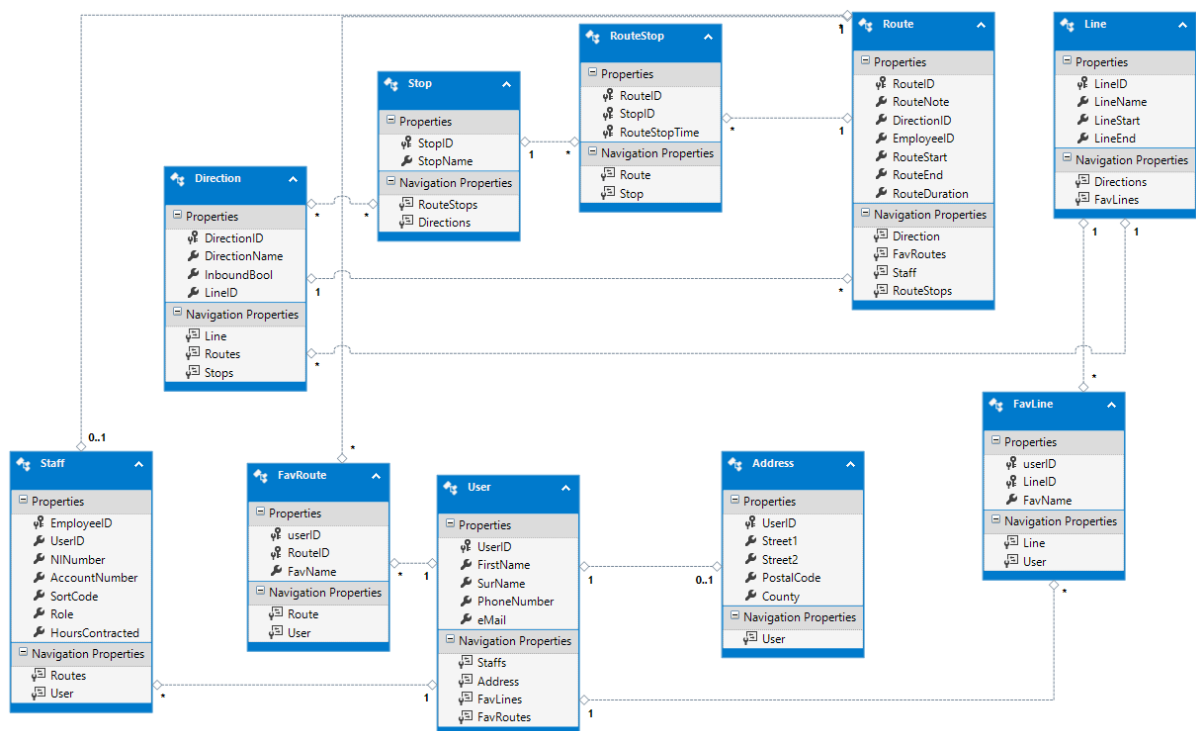
The Entity Relationship Diagram shows our revised database model after multiple revisions.

The $User$ entity has a 1 to 0..1 relation with the $Address$ table, where the address information was filled in during registration (or later in account preferences) or not.

Each $User$ entity has a one to many relationship with $FavouriteRoute$ and $FavouriteLine$, which are association entities to $Route$ and $Line$ respectively. These link multiple routes or lines to a single user.

A $Staff$ entity has a many to one relationship with the $User$ entity, as the $Staff$ entity extends the basic information stored by the $User$ entity. However a $User$ may not have a relation with the $Staff$ entity if they are just a customer.

$Lines$ are the main entity for the bus timetabling system, where a single line will have multiple $Directions$ (i.e. A to B and B to A).

Each $Direction$ has many $Routes$, as each time slot a line operates for is stored as its own route. For example, on $LineA$, busses leaving at 0900 and 0930 are two separate routes assigned to the same $Direction$ and $Line$ with different drivers.

A $Direction$ also has many $Stops$, which contain the stop's name and coordinates. However these stops may not be serviced by all of the busses on the same line (e.g. limited services on weekends or holidays). For this, there is another association entity called $RouteStop$ which maps stops serviced by the $Line$ and stops the $Route$ will actually be stopping at.

2.2 UML Class Diagrams
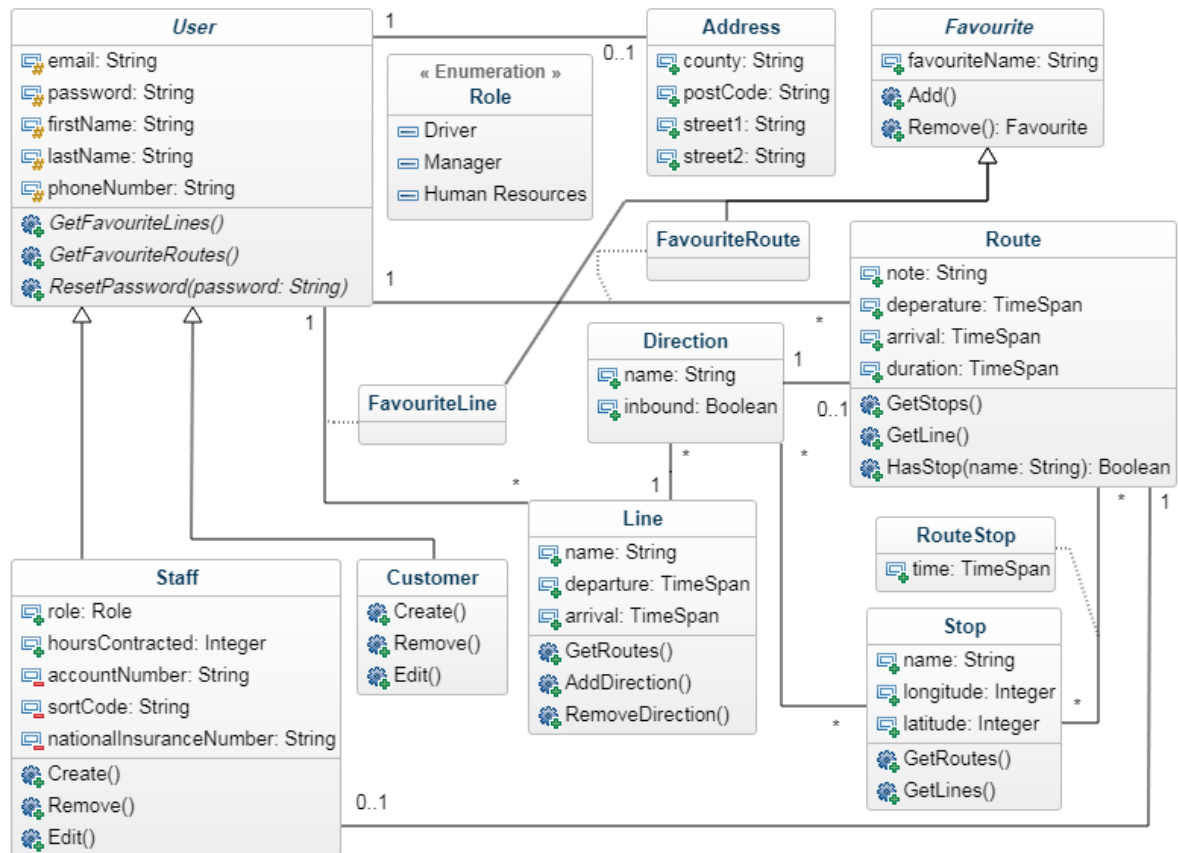
2.3 Interface Prototypes

2.3.1 Homepage
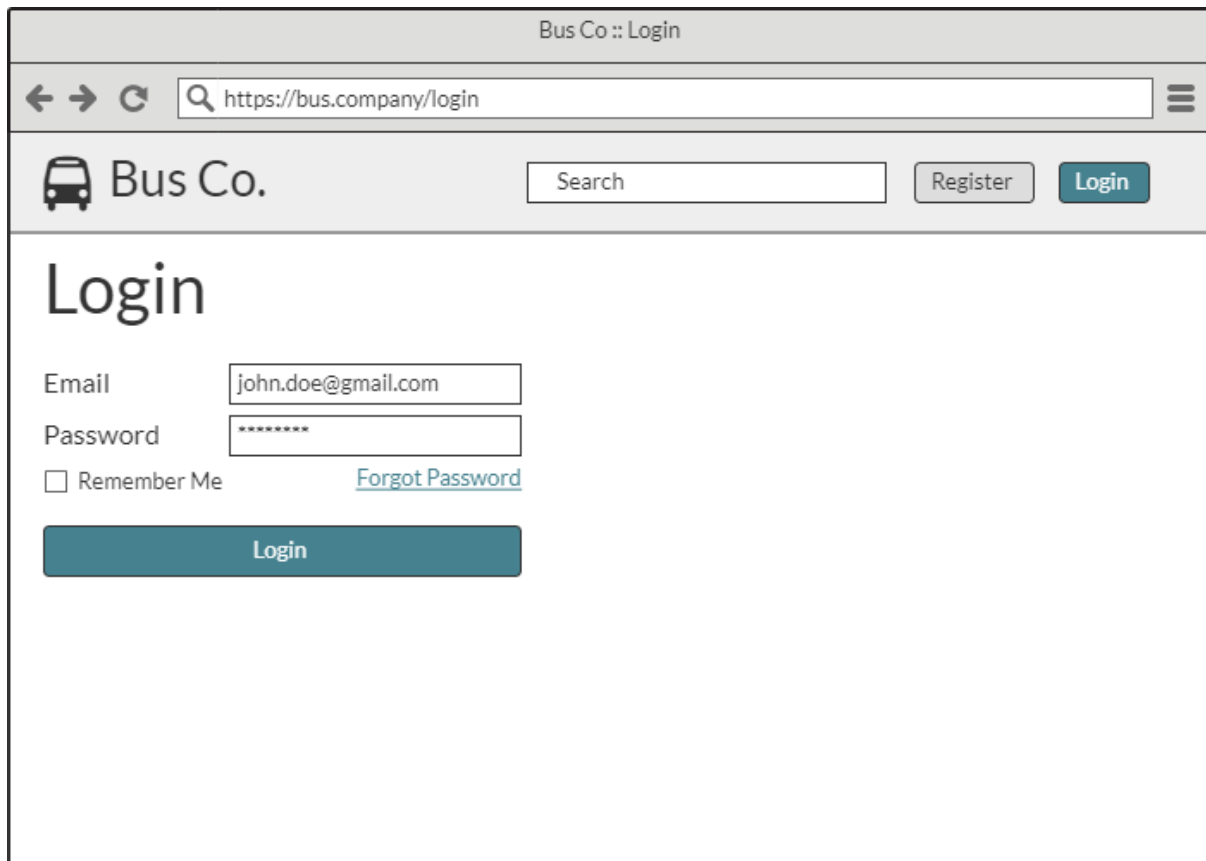
The homepage is where customers will be able to find routes from a specific town or bus stop to another location. In the header, users can also directly search a route or line to go straight to its route details page.

Below this is a table that lists all the active lines operated by the bus company. It's broken down by region to allow users to find their desired routes quickly.

For users that are signed in and have added routes to their favourites, these routes will appear at the top of the route table in the relevant region.
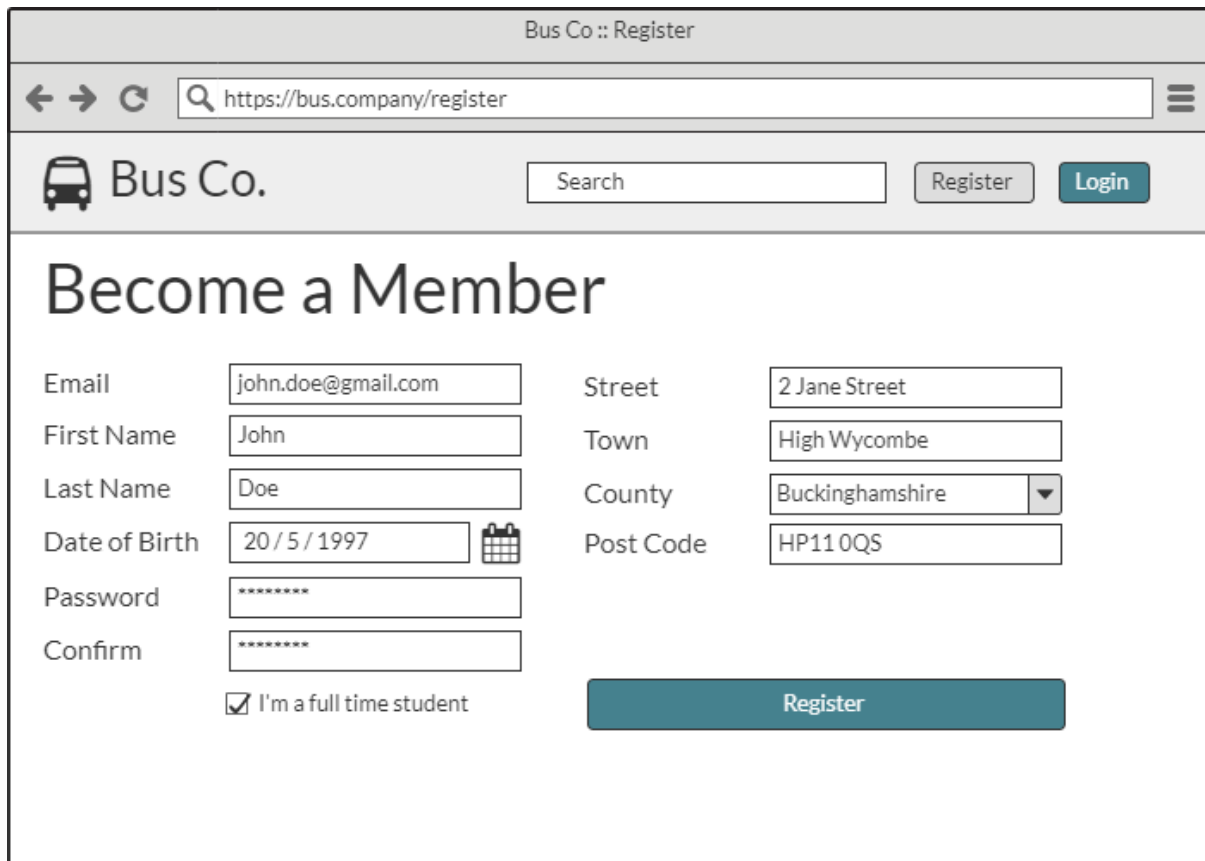
2.3.2 Login

Staff and users that have previously registered on the service can log in here with their email and password.

Additionally, cookies or localstorage will be used to remember their login session on repeat visits if they tick the $Remember Me$ checkbox. Clicking $Forgot Password$ will send an email, which will allow the user to enter a new password.

2.3.3 Register

Registering will add a new user in the database. When registering, users can enter their details and address. Entering an address will allow the region they reside in to be the default tab on the homepage.

Users that are in full-time education can tick a checkbox which will show student ticket prices for routes instead of the normal price. If the user's age is 60 or above (from the DoB field), the price will be omitted entirely.

Drivers/staff cannot register themselves and have to be added by an existing staff member with the correct *role*.

2.3.4 Preferences

In the account preferences, users can modify all of their data. If they set a new password they must enter their current password for the change to take effect.

The favourites tab shows a list of all routes the user has previously favourited, where they can go to the route details or remove the route. Removing a route will display a confirmation prompt to avoid accidental removals.

2.3.5   Route Details

The route details page shows all the busses running the route with the estimated time to arrive at each stop along the route.

In the top left section, if a user is signed in there will be a *star* icon which will add or remove the route from the users favourites. This icon won't be shown if they are not signed in.

Below the route name is the town the route starts and ends in. To the right of this is an icon to invert the direction, which will allow the user to see the return times. At the bottom of this section, there is a series of buttons to switch between showing route times for the current day, the next day and any other day. Clicking *Other* will open a calendar to pick a specific day to show.

On the right is a route map which will show a plotted route with all the stops using the Google Maps API.

2.3.6 Admin Panel

The admin panel allows staff with the required *Role*(s) to navigate to the necessary page to edit data in the database directly using the ASP.NET MVC edit views.

2.3.7 Driver Dashboard

The Driver Dashboard is what staff with the *driver Role* will see instead of the admin panel. Here they can see the hours they are assigned for the current week, along with a weekly timetable.

The timetable shows the routes they are driving on and at which times. On desktops, this may show more details such as the bus bay or town the assigned route is starting and ending in alongside the route name. Clicking on the route bar will open a modal containing all the required information and allow the driver to view the relevant route details page.

The driver can also access previous and future weeks, as well as print a printer-friendly version of the timetable.

2.3.8 Line Management

The line management page in the admin panel lists all the lines operated by the bus company. Key details are shown in this list and can be deleted or edited by staff members. New lines can be added by the button in the top right.

2.3.9 Route Management

Route management can be accessed from the $AdminPanel$ or through the $LineManagement$ page.

The page shows all the routes assigned to a specific line with their start times and other important information. It defaults to showing all the route data for the current week which can be navigated on a week-by-week basis.

New routes can be added to a line here, as well as be removed or modified.

2.3.10 Route Editing

The route edit page is accessed from clicking the *Edit* button on a route in the *RouteManagement* page.

The driver assigned to the particular route is displayed along with a list of all the stops assigned to the route. These can be added, deleted or modified.