

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

1	Project Outlines	2
1.1	Enterprise Model	2
1.2	Business Goals	2
1.3	Functional Requirements	2
1.4	Outline of the Project	3
2	Diagrams & Schematics	4
2.1	Entity Relationship Diagram	4
2.2	UML Class Diagrams	5
2.3	Interface Prototypes	6
2.3.1	Homepage	6
2.3.2	Login	7
2.3.3	Register	8
2.3.4	Preferences	9
2.3.5	Route Details	10
2.3.6	Admin Panel	11
2.3.7	Driver Dashboard	12
2.3.8	Line Management	13
2.3.9	Route Management	14
2.3.10	Route Editing	15

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

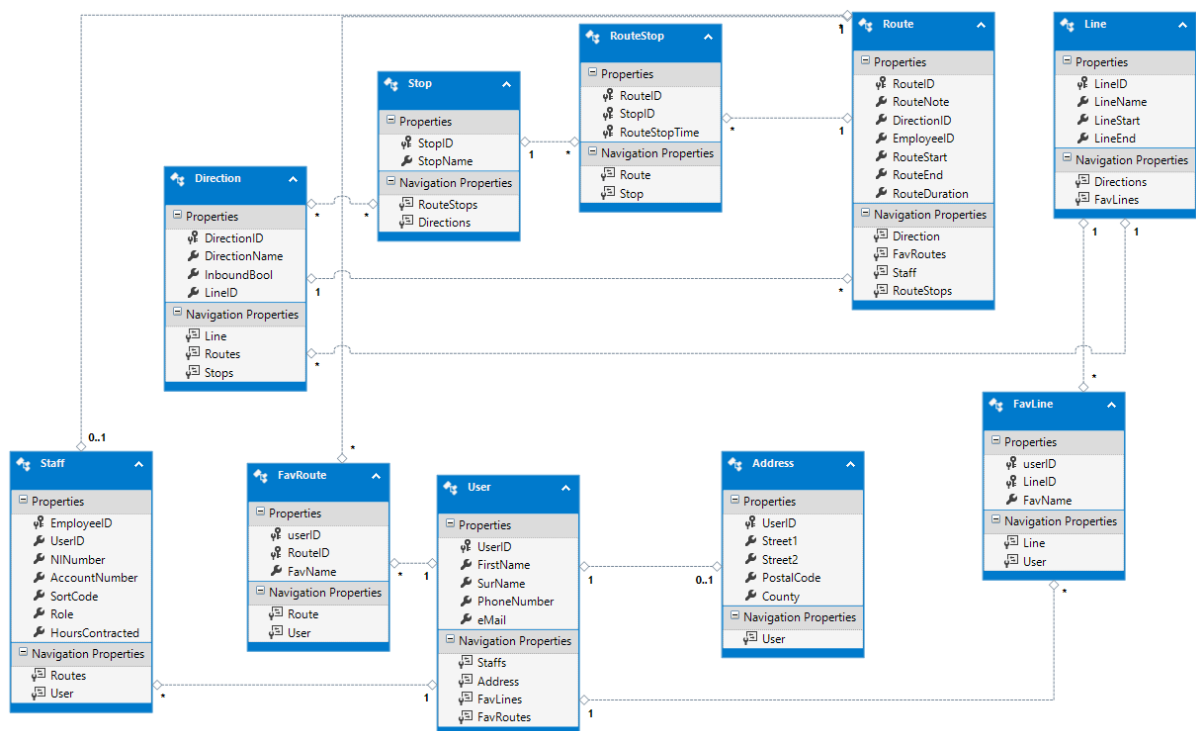
- Enable users to login and register
 - Normal (unprivileged users) can register anytime
 - Privileged users must be added by an admin
- Enable users and visitors to see bus timetables
- Enable registered users to save their favourite routes
 - Requires personal dashboard

- 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.

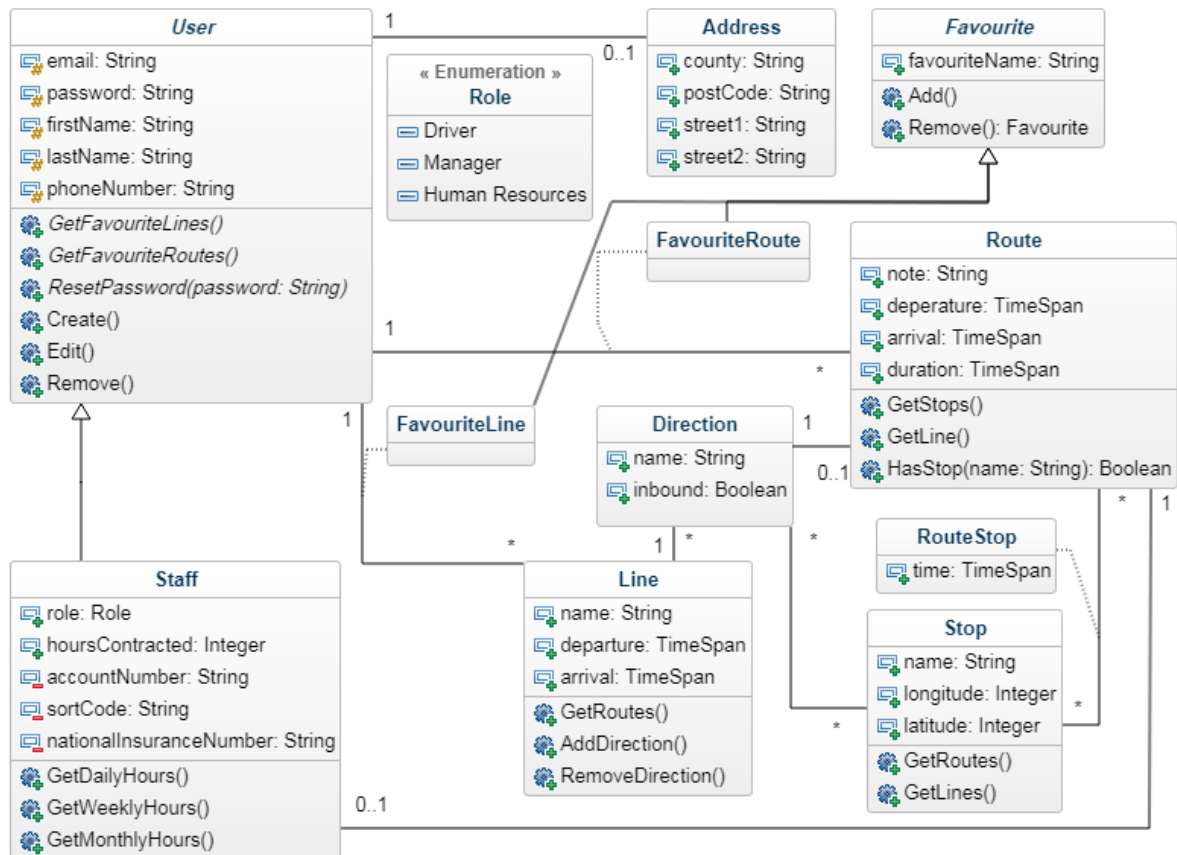
A *Staff* entity has a many to one relationship with the *User* entity, as the *Staff* entity extends the basic information stored by the *User* entity. However a *User* may not have a relation with the *Staff* entity if they are just a customer.

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



Functionally equivalent to the *ER* diagram, the *User* class is an abstract class, which the *Staff* class is generalised from.

As *FavouriteLine* and *FavouriteRoute* share the same attributes and methods, these classes are also a generalisation of an abstract *Favourite* class. As these two and *RouteStop* are just associating data from two tables, these are displayed as association classes in the UML class diagram instead of regular classes.

The *Staff* class has *GetXHours()* methods which calculate the hours the staff member will be working within the duration (*day*, *week* or *month*) from the *hoursContracted* for non-driver staff or route durations the driver is assigned to. This will be used as validation when assigning a driver to new routes, as a driver legally cannot drive for more than 9 hours a day, plus the required breaks (with 2 days a week where they can drive 10 hours). If the route will cause the driver to exceed the limit, the creation/edit view should display an unobtrusive error and prevent confirming the change or addition.

The *Route* class has a *GetStops()* method which will be used by the *RouteDetails* and *RouteEditing* views to list all stops associated with the selected route. *HasStop()* will be used by the homepage search, so only routes that service both selected stops will be shown in the results.

2.3 Interface Prototypes

2.3.1 Homepage

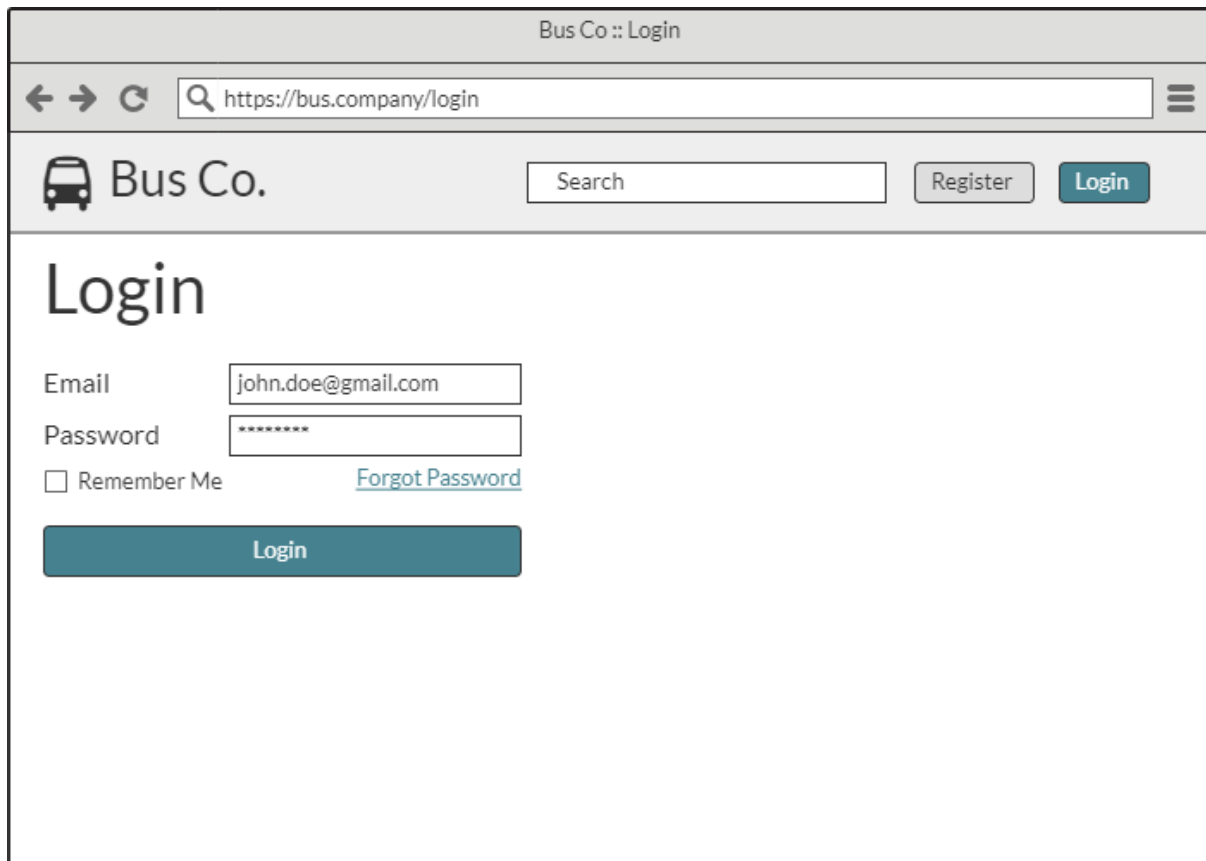
The wireframe shows a web browser window titled "Bus Co :: Timetable Service". The address bar contains "https://bus.company". The header features the "Bus Co." logo, a search bar, and "Register" and "Login" buttons. Below the header, there are two dropdown menus for "From" and "Destination", followed by a "Find Bus" button. To the right of these is a large rectangular area labeled "background image". Below the search section, the heading "Routes" is displayed. Under "Routes", there is a vertical list of buttons: "Buckinghamshire", "Sussex", "Cornwall", and "Other Regions". To the right of this list is a large empty rectangular box for displaying route information.

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

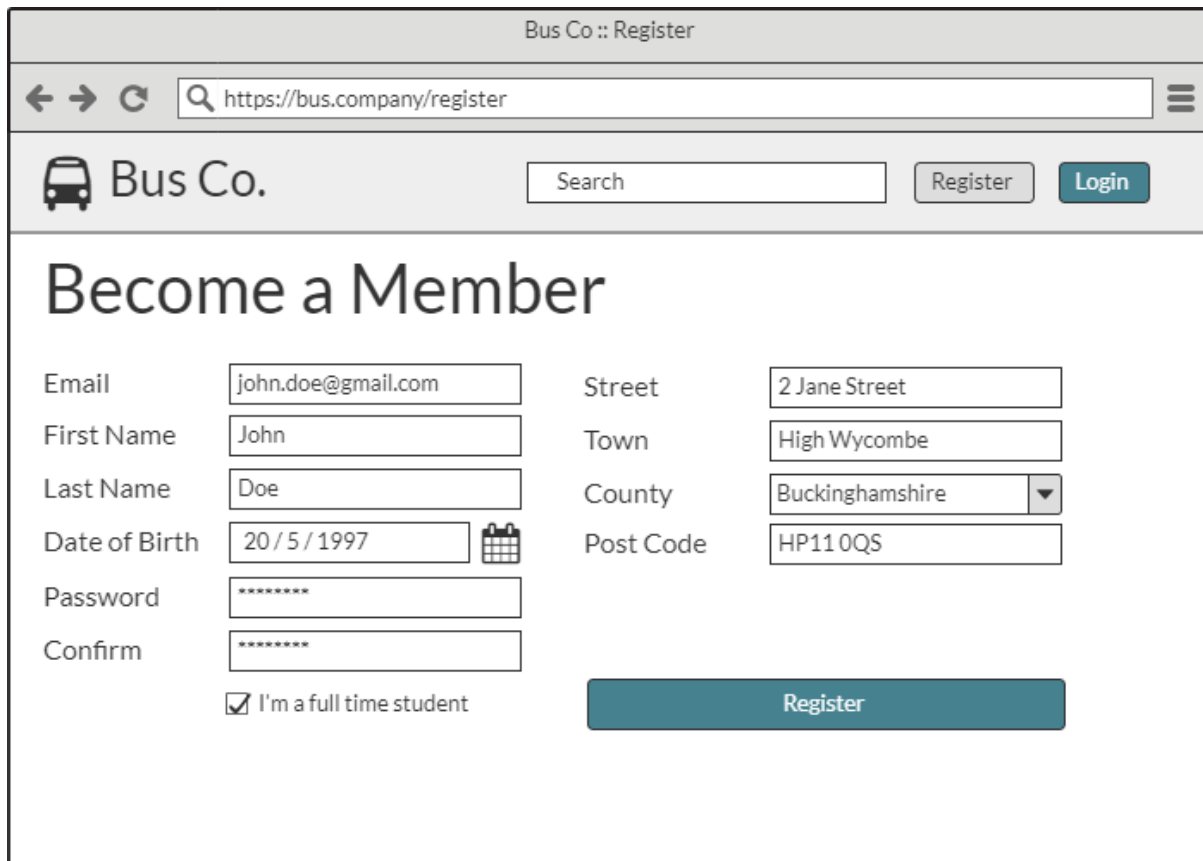


The screenshot shows a web browser window titled "Bus Co :: Login". The address bar displays "https://bus.company/login". The page header includes the "Bus Co." logo, a search bar, and "Register" and "Login" buttons. The main content area is titled "Login" and contains a form with fields for "Email" (containing "john.doe@gmail.com") and "Password" (containing "*****"). Below the password field is a "Remember Me" checkbox and a "Forgot Password" link. A large "Login" button is positioned at the bottom of the form.

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



The screenshot shows a web browser window titled "Bus Co :: Register". The address bar shows "https://bus.company/register". The page header includes the "Bus Co." logo, a search bar, and "Register" and "Login" buttons. The main heading is "Become a Member". The registration form consists of two columns of input fields. The left column contains fields for Email (john.doe@gmail.com), First Name (John), Last Name (Doe), Date of Birth (20/5/1997 with a calendar icon), Password (masked with asterisks), and Confirm (masked with asterisks). The right column contains fields for Street (2 Jane Street), Town (High Wycombe), County (Buckinghamshire with a dropdown arrow), and Post Code (HP11 0QS). Below the form is a checkbox labeled "I'm a full time student" which is checked, and a large "Register" button.

Email	<input type="text" value="john.doe@gmail.com"/>	Street	<input type="text" value="2 Jane Street"/>
First Name	<input type="text" value="John"/>	Town	<input type="text" value="High Wycombe"/>
Last Name	<input type="text" value="Doe"/>	County	<input type="text" value="Buckinghamshire"/>
Date of Birth	<input type="text" value="20/5/1997"/>	Post Code	<input type="text" value="HP11 0QS"/>
Password	<input type="password" value="*****"/>		
Confirm	<input type="password" value="*****"/>		
<input checked="" type="checkbox"/> I'm a full time student		<input type="button" value="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

Bus Co :: Preferences

← → ↻

https://bus.company/preferences

⋮

 Bus Co.

Search

John Doe ▼

My Favourites

Preferences

Sign Out

Preferences

Account

Favourites

Email

john.doe@gmail.com

Street

2 Jane Street

First Name

John

Town

High Wycombe

Last Name

Doe

County

Buckinghamshire ▼

New Password

Post Code

HP11 0QS

Confirm

Current Password


Save

Bus Co :: Preferences

← → ↻

https://bus.company/preferences

⋮

 Bus Co.

Search

John Doe ▼

Preferences

Account

Favourites

Route Name	▲ Start	◆ End	◆ Remove
A300	High Wycombe	Aylesbury	yes
A300	Aylesbury	High Wycombe	yes

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

Bus Co. :: Route A300

https://bus.company/route/a300

Bus Co. Search Register Login

A300
High Wycombe to Aylesbury

★ ↺

Route Map

	Today	Tomorrow	Other
Stop 1	1150 1250 1350 1450 1600 1700 1800 1900 2000 2100 2200 2330		
Stop 2	1155 1255 1355	1910	
Stop 3	1213	1407 1505	

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 screenshot shows a web browser window titled "Bus Co. :: Admin". The address bar displays "https://bus.company/admin". The page header features the "Bus Co." logo on the left and a "User" dropdown menu on the right. The main content area contains three side-by-side panels, each with a title, a list of actions, and a "Go" button:

- Staff**: See Stats, Add Modify. Go
- Routes**: Add, Delete, Modify, Assign Drivers. Go
- Lines**: Add, Delete, Modify. Go

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 screenshot shows a web browser window titled "Bus Co :: Dashboard". The address bar displays "https://bus.company/dashboard". The page header includes the "Bus Co." logo and a "User" dropdown menu. The main content area greets the user with "Hi User," and "You're working 30 hours this week". It also shows the current week "W/C: 30/10/2017" and three buttons: "Previous Week", "Print Timetable", and "Next Week".

The timetable is presented as a grid with days of the week on the vertical axis and time slots from 0900 to 2200 on the horizontal axis. Route assignments are shown in grey boxes:

	0900	1000	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000	2100	2200
Monday	A300				A300				A300					
Tuesday			A500						A300					
Wednesday	A300				A300				A300					
Thursday	A300				A300				A300					
Friday														
Saturday														
Sunday														

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

Bus Co. :: Line Management

https://bus.company/admin/lines

Bus Co. User

Lines [Add](#)

Line	Start	End	Actions
A500	xxxxxxx	xxxxxxx	Delete Edit
A550	xxxxxxx	xxxxxxx	Delete Edit
A300	xxxxxxx	xxxxxxx	Delete Edit

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

Bus Co :: A300 Route Management

← → ↺

https://bus.company/admin/lines/a300

☰

 Bus Co.

User ▾

Routes - Line A300

[Previous](#)

W/C: 30/10/2017

[Next](#)

Monday

Tuesday

Wednesday

Thursday

Friday

Saturday

Sunday

[Add](#)

Time	Start	End	Stops	Driver	Actions
0600	xxxxxxx	xxxxxx	14	xxxxxx	Delete Edit
0620	xxxxxxx	xxxxxx	18	xxxxxx	Delete Edit
0640	xxxxxxx	xxxxxx	14	xxxxxx	Delete Edit

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

Bus Co. :: Edit A300 Route

← → ↺

https://bus.company/admin/lines/a300/0600

☰

 Bus Co.

User ▾

0600 Route - A300

[Change](#) 30/10/2017

Driver: Bob Malik [Change](#) [Add Stop](#)

#	Stop	Actions
1	Bus Stop Number 1	Delete Edit
2	Bus Stop Number 3	Delete Edit
3	Bus Stop Number 4	Delete Edit

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.