

# BOOKING PORTAL

Booking Platform for Businesses and Customers



| System Delivery Document |

*Luke Fetin*

*USYD | ELEC3609 MAJOR PROJECT*

## **Application**

### **Domain**

Bokingportal.com  
Provided by AWS: Route 53

### **Website Accounts**

Businesses Usernames:  
Sydney\_University-Tennis  
SydneyCBD\_SportCentre-LockerHire  
Nationwide\_Parking

Customer Usernames:  
uni\_student  
sydney\_worker  
lfet7715

ALL HAVE PASSWORD '123lukey'

Sign in to View Bookings, Search for Bookings,  
Create Booking Tables, Delete Bookings etc. . . .

## **Configuration**

### **Ec2 Instance**

Name: Booking Portal  
Instance id: i-038ae5dc25b5ec09c  
Public DNS: ec2-18-216-60-222.useast-2.compute.amazonaws.com

### **Access**

#### **AWS Read Only User**

Username: 1  
Password: password

Restricted access with Read Only

#### **SSH Access**

Marker SSH key is place in authorized\_key

To ssh in:

```
ssh -i "markerkey.pem" ubuntu@ec2-18-216-60-222.us-east-2.compute.amazonaws.com
```

#### **Database Access**

Database used: sqlite3

From /home/ubuntu/A2/BookingPortal

Python manage.py shell

## Virtual environment

Venv – app specific modules installed on virtual environemnt.

To activate virtual environment:

```
source /home/ubuntu/env/bin/activate
```

## Python Modules

Django

Crispy forms

Django rest framework

Pillow

## Server Stack (Process Control, WSGI Server Implementation and Proxy)

Supervisor - Process Control and Fault Tolerance

Gunicorn - WSGI Server Implementation

Nginx - Reverse Proxy

Gunicorn configuration found in:

/etc/supervisor/conf.d/gunicorn.conf

Log Files:

Stderr\_log - /var/log/gunicorn/gunicorn.err.log

Stdout\_log - /var/log/gunicorn/gunicorn.out.log

Nginx configuration:

/etc/nginx

Nginx configuration for BookingPortal:

/etc/nginx/sites-available/django.conf

/etc/nginx/sites-enable/django.conf

## Website Security

### SSL

All traffic is transferred with SSL. SSL Certificate provided by <https://certbot.eff.org/lets-encrypt/ubuntubionic-nginx>

HTTPS traffic is redirected to HTTPS.

All traffic restricted to HTTPS.

### Content Protection

Session cookies, CSRF and XSS Filtering protection all provided by Django. See set settings in settings.py.

### AWS Security

Security Group: launch-wizard-2

HTTP	TCP	80	0.0.0.0/0
HTTP	TCP	80	::/0
Custom TCP Rule	TCP	8000	0.0.0.0/0
SSH	TCP	22	0.0.0.0/0
Custom TCP Rule	TCP	9641	0.0.0.0/0
Custom TCP Rule	TCP	9641	::/0
HTTPS	TCP	443	0.0.0.0/0
HTTPS	TCP	443	::/0

## **Unit Test**

Location:

/home/ubuntu/A2/BookingPortal/users/tests.py

Run:

Python manage.py test users

From - /home/ubuntu/A2/BookingPortal

## **Test Cases**

### **Bookings**

View

Make

Delete

Search

### **User**

Create

Update Account

### **Asset**

View

Create

Delete

Edit