First I just want to list how to start the server, since I reference that a couple times.

1) Have all code present on computer
2) Go to the directory with manage.py
3) Run 'python manage.py makemigrations'
4) Run 'python manage.py migrate'
5) Run 'python manage.py runserver'
6) On a separate command line, go to the directory with manage.py
7) Run 'python manage.py createsuperuser' and enter in relevant information in the fields
8) Go to http://127.0.0.1:8000
9) Click "Login" and enter the account information for the superuser you created
10) Click on "Users" and then "+" to create a new staff account with login information
11) Click "Save"
12) Click on "Courses" and then "+" to create a new course, make sure to list the new staff user as the teacher
13) Click "Save"
14) Click on "Students" and then "+" to create new students
15) Enter all needed information
16) Click "Save" if you're finished, otherwise click "Save and add another" and go back to step 15

Sprint 1:

URL Generator: Tested by generating a number of unique urls that would be valid when creating new sessions. Very simple.

How to test:

1) Run through the startup I detailed above.
2) Log out, then log back in as the staff member you created in step [10] of the startup phase
3) Go to "Courses," then click on your example course from step [12] of the startup phase
4) Click "Start new lecture"
5) Random URL will be used to create QR code, as well as displayed as a link below the code.

Sprints 2 & 3:

NetworkX Graph Generation and Visualization: Start a lecture, generate sample information, and generate graph. A little more involved than the first.

1) Run through the startup I detailed above.
2) Log out, then log back in as the staff member you created in step [10] of the startup phase
3) Go to "Courses," then click on your example course from step [12] of the startup phase
4) Click "Start new lecture"

5) Random URL will be used to create QR code, as well as displayed as a link below the code.
6) Sign in as some of the students you created in step [15] of the startup phase. This is where it gets a little tricky. The link itself is not yet accurate because due to a limited timeframe we do not have any server hosting the site AllAboardAttendance.com. Luckily, the QR code will lead you to the correct link, and you can also go to http://127.0.0.1:8000 + whatever the link says after /course/ (Example: http://127.0.0.1:8000/course/12345abcdef/sign-in/)
7) Open this page on several tabs of your browser.
8) On each tab, enter a separate student id from the ones you created in step [15] of the startup phase and click “sign in”
9) Mix and match by entering the generated student codes into the fields of other tabs. This is how the students are marked as present and added to the graph as nodes. Remember to click the button so that the codes are properly entered. Sometimes you will get an error because a connection has already been made. This is fine and supposed to happen since we don’t allow for double connections.
10) After you feel you’ve entered a modest amount of this information, go back to the lecture page you started the session from. There is a button at the bottom of the page which you can click to generate your graph. Enjoy!

ALTERNATIVE TESTING METHOD

10) Open a new command line, go to the directory with the file “manage.py”
11) Run the command ‘python manage.py shell’
12) Copy and paste this script into the shell:

```
from django.shortcuts import get_object_or_404, render
from django.http import HttpResponseRedirect
from django.urls import reverse
from django.views import generic
from django.utils import timezone
import string, random
import networkx as nx
import matplotlib.pyplot as plt
from PIL import Image
from io import BytesIO
from django.core.files.base import ContentFile
import qrcode
from .models import Lecture, Attendant, DirEdge
from course.models import Course
from lecture.views import generateGraph
course = Course.objects.all()[0]
L = course.lecture_set.all()[0]
from lecture.views import makeEdgesList
from lecture.views import str2
generate_graph(L)
```

13) Go the directiory: /AllAboardAttendance/AllAboardAttendance/media/graphs
14) The generated image will be in that directory. This is more of a “unit test” thanks to the script and requires minimal setup to be run again and again, since you only need to restart the shell.