All Aboard Attendance
July 24, 2019

- **Matt Jacobs**
mtjacobs@ucsc.edu
- **Trevor Weger**
teweger@ucsc.edu
- **Suleyman Saib**
sysaib@ucsc.edu
- **Donghyun Kim**
dkim198@ucsc.edu
- **Wesley Smith**
wetsmith@ucsc.edu

Traditional attendance taking methods

- Roll call – Too slow
- Sign-in sheets – Easily tricked
- ID scanners – Prohibitively expensive

What we wanted to achieve

- Simple process that can be done from a cell phone or laptop
- Faster than traditional attendance taking
- More difficult to trick
- Requires no additional equipment or hardware

Challenges we faced

- Entire team new to web development
- The framework and almost every package we used had a learn curve
- Learning new management style
- Time constraints: Both in general length of projects and scheduling

What we achieved

It’s fast

It’s simple to use

No extra equipment

Something even cooler…

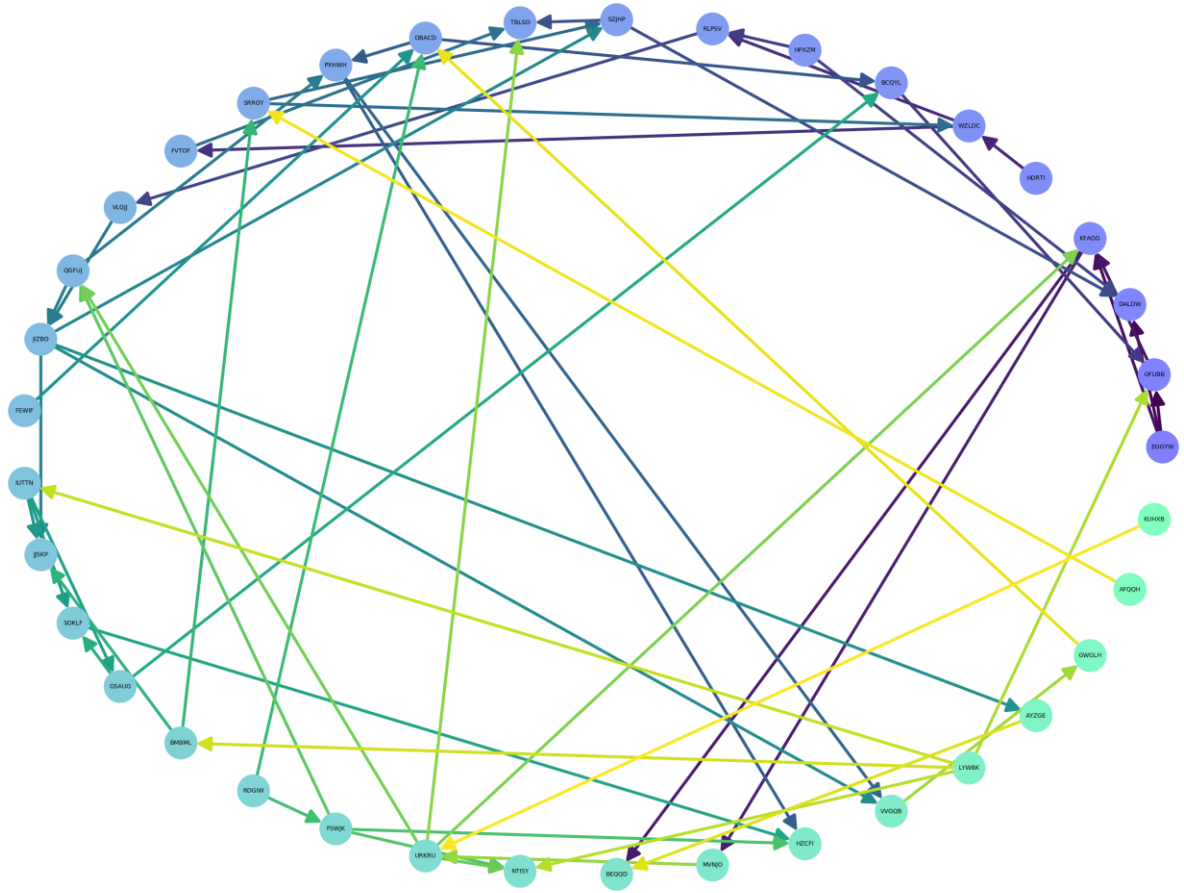
Something cooler

Each student receives a unique code for a lecture

To be marked present, students exchange their code with other nearby students

We keep track of these proximity-based code exchanges

Data!


TBLSO
SZJHP
RLPSV
HFXZH
OBACD
PKHWH
BCQYL
SKROY
WZLDC
FVTOF
HDRTI
VLOJJ
KFAQG
QOFUJ
DALDW
JIZBO
OFUBB
FEWIF
EOGYW
IJTTN
KUHXB
JISKP
AFQOH
SOKLF
GWGLH
OSAUG
AYZGE
BMBML
LYWBK
RDGIW
VVGOB
FSWJK
HZCFI
LPUKRJ
NTISY
MVNJO
BEQQD

Why is this data useful?

See which students regularly attend together

Correlate with MOSS reports for plagiarism detection

Make final/midterm seating charts that break up students who regularly sit near one another

Generate new data for research

Frameworks and Technologies

Python, MySQLite

Django Web Framework

QR Code, NetworkX

Frameworks and Technologies

Python, MySQLite

- **Python:** High level object oriented language
- **MySQLite:** C-language library common in applications. It creates a fast and efficient database

Frameworks and Technologies

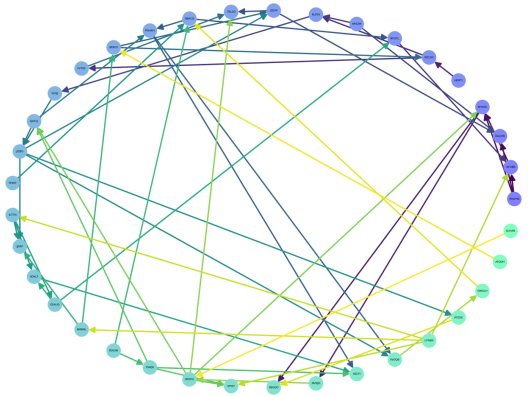
Django Web Framework

- Python based
- Simple organization and built in functions
- Efficient and fast
- Models: Interfaces with MySQLite

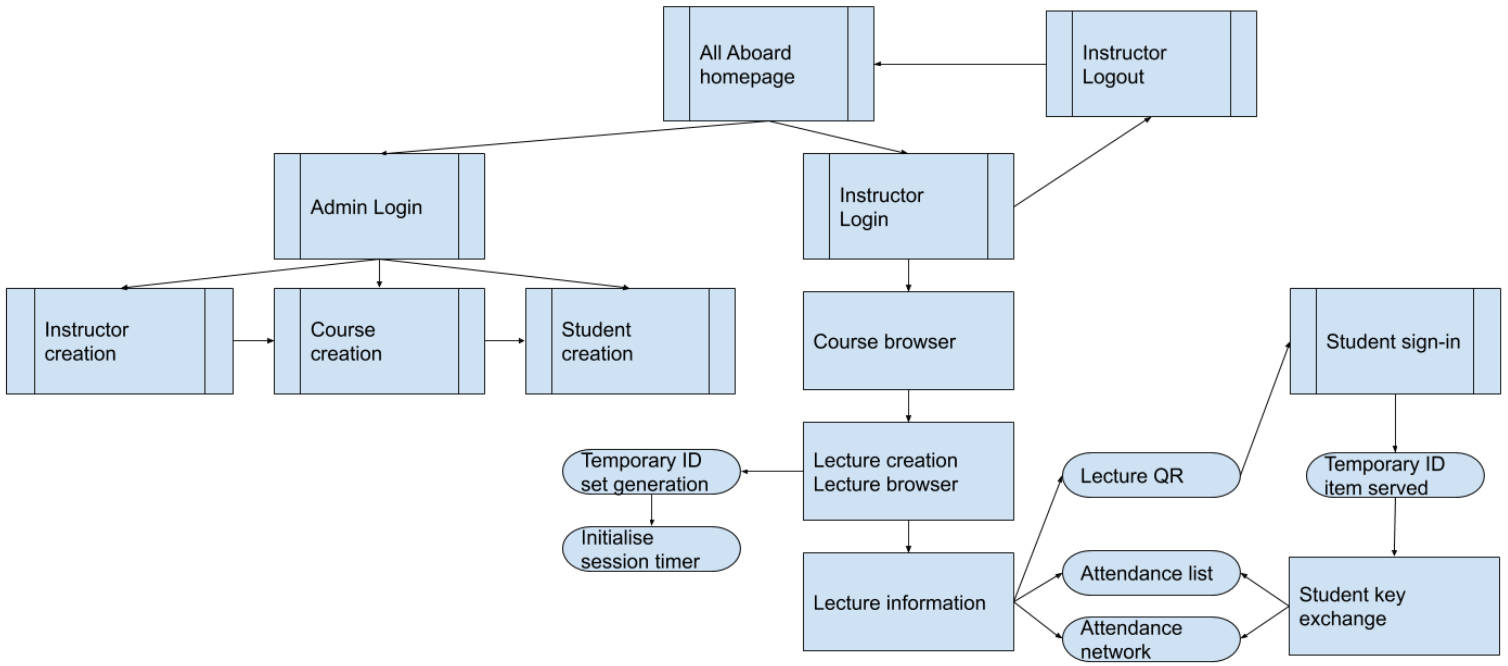
Frameworks and Technologies

QR Code, NetworkX

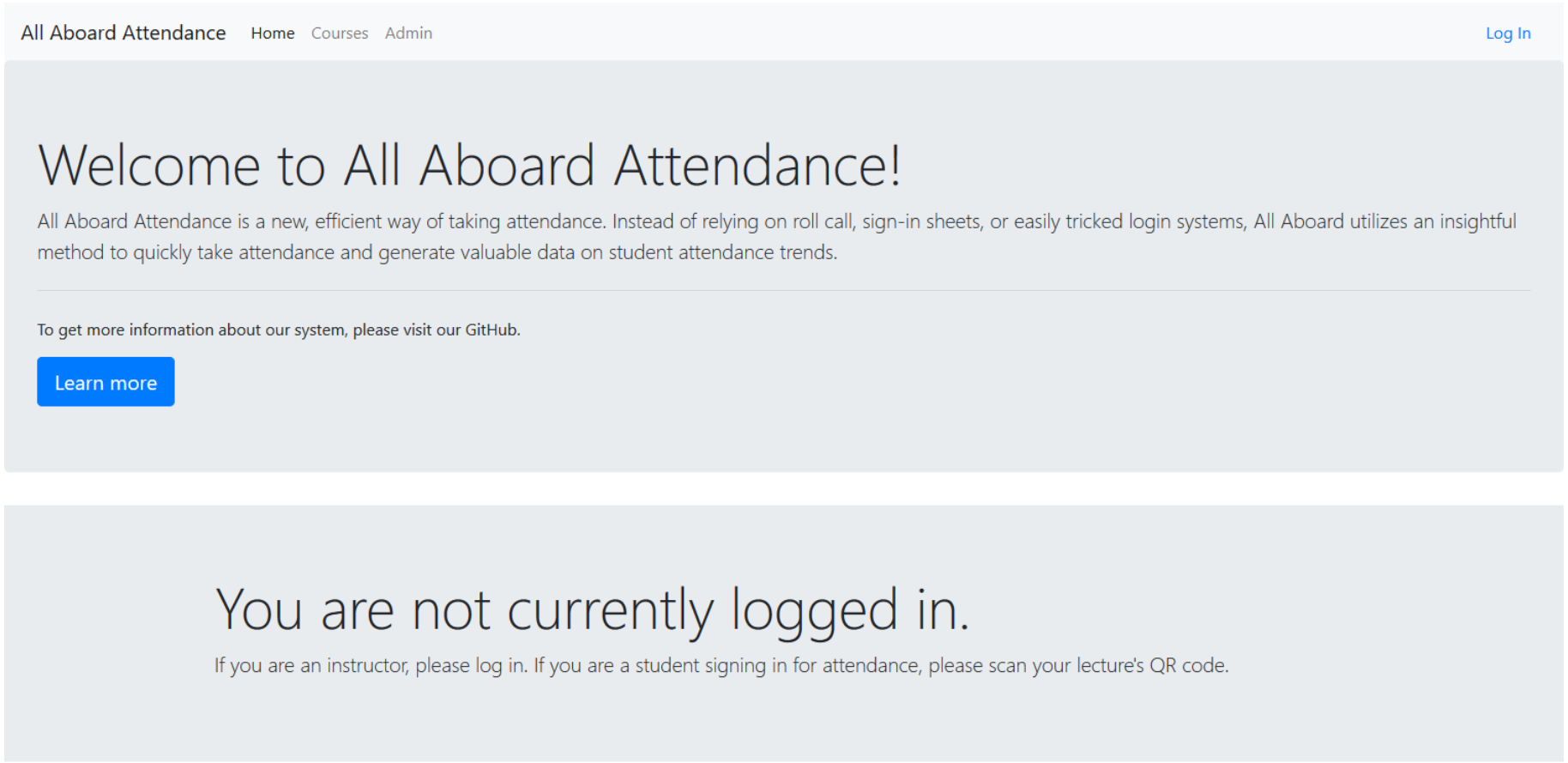
- Python libraries
- You'll see where specifically in the next slide



System Diagram/Overview

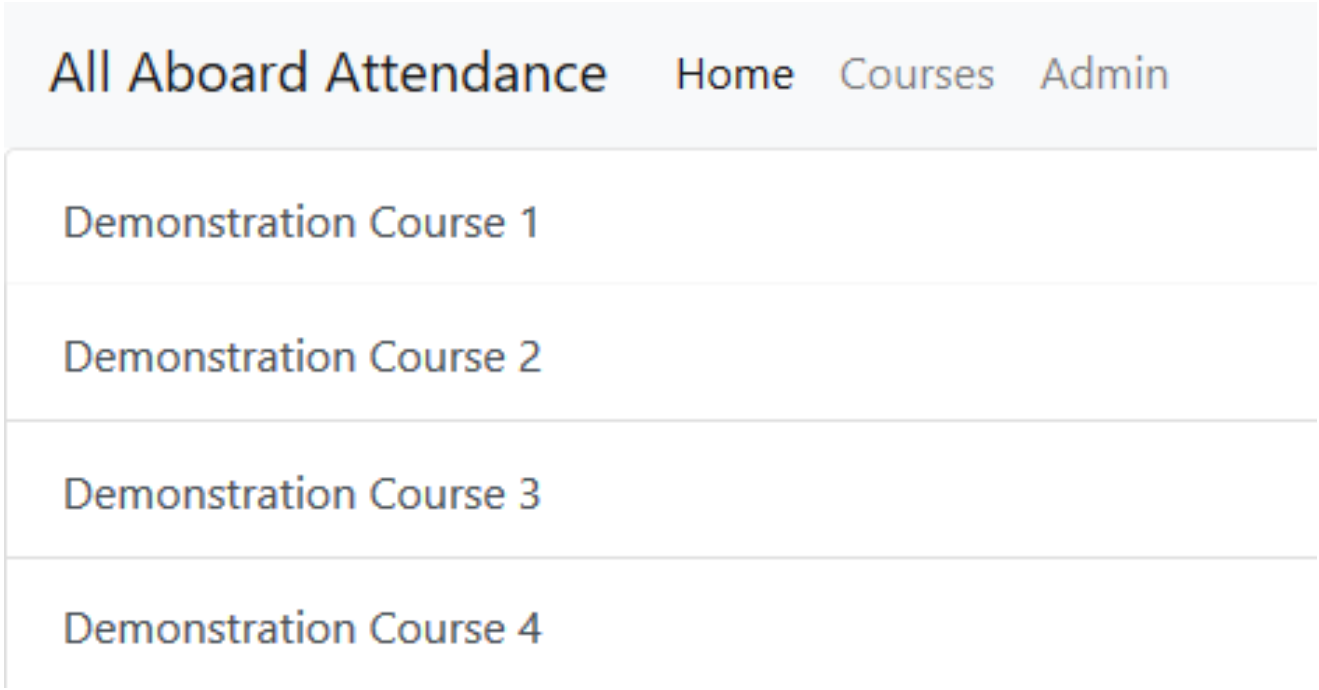


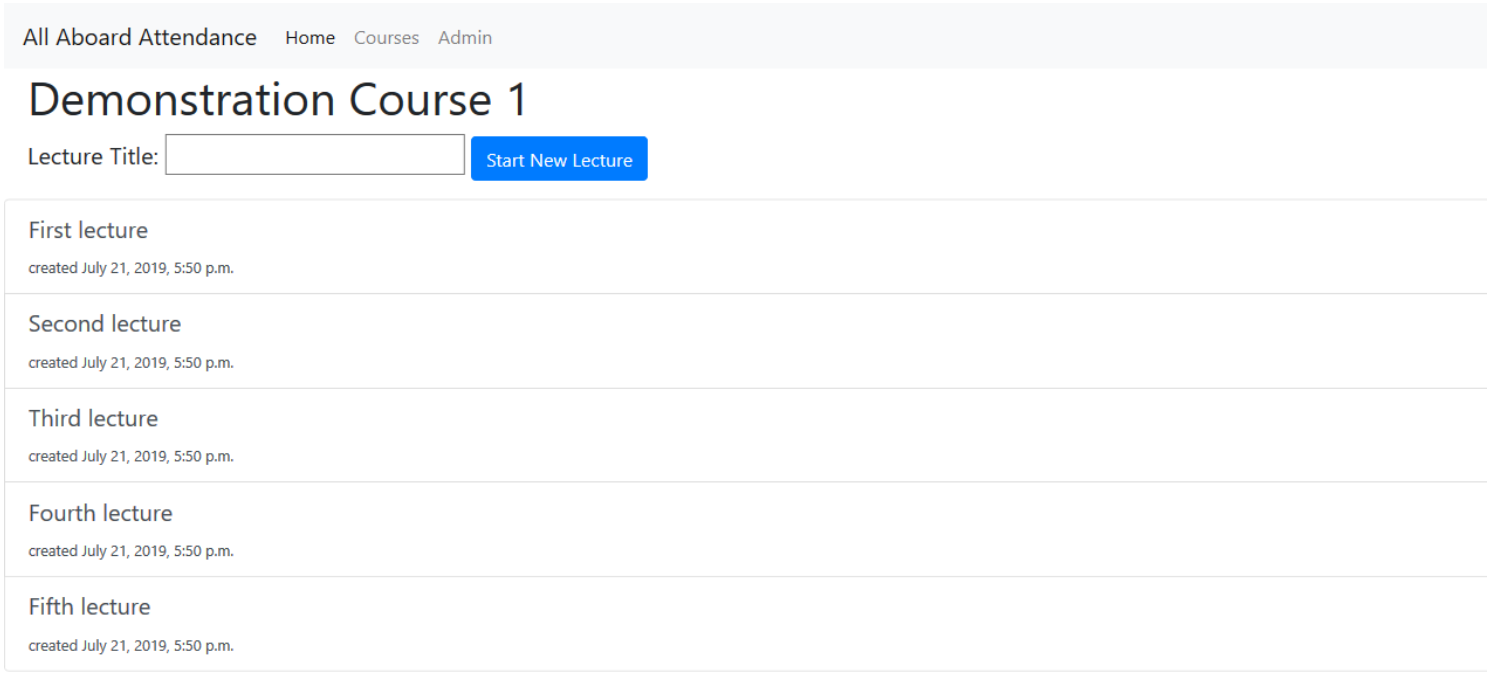
Home page


All Aboard Attendance
Home
Courses
Admin
Log In
Welcome to All Aboard Attendance!
All Aboard Attendance is a new, efficient way of taking attendance. Instead of relying on roll call, sign-in sheets, or easily tricked login systems, All Aboard utilizes an insightful method to quickly take attendance and generate valuable data on student attendance trends.
To get more information about our system, please visit our GitHub.
Learn more
You are not currently logged in.
If you are an instructor, please log in. If you are a student signing in for attendance, please scan your lecture's QR code.

Course index

- Inaccessible to students
- Only shows the courses associated with a specific instructor login
- Allows instructors to navigate the courses they are teaching
- Each course is a Django model in a sqlite3 database



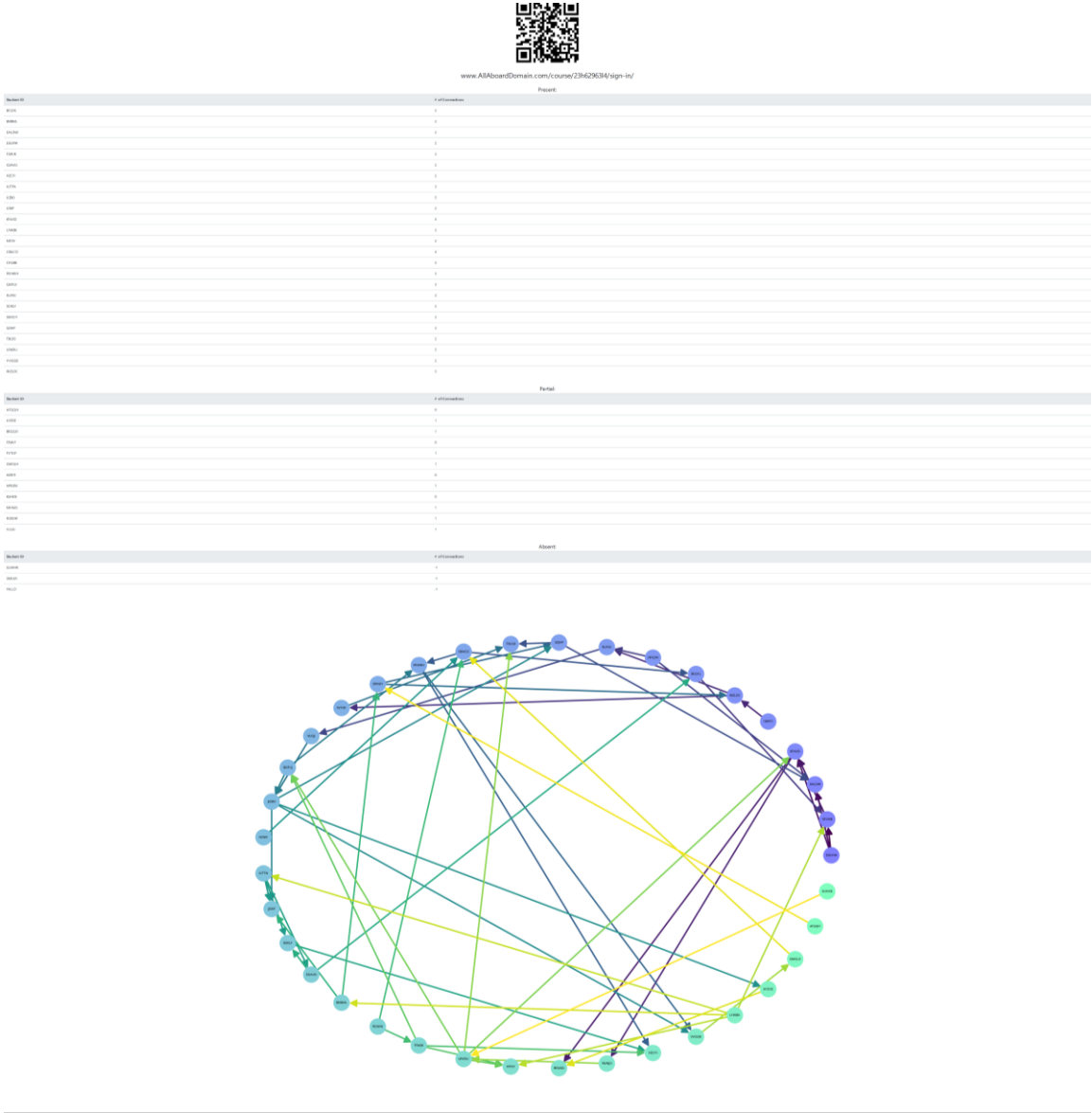


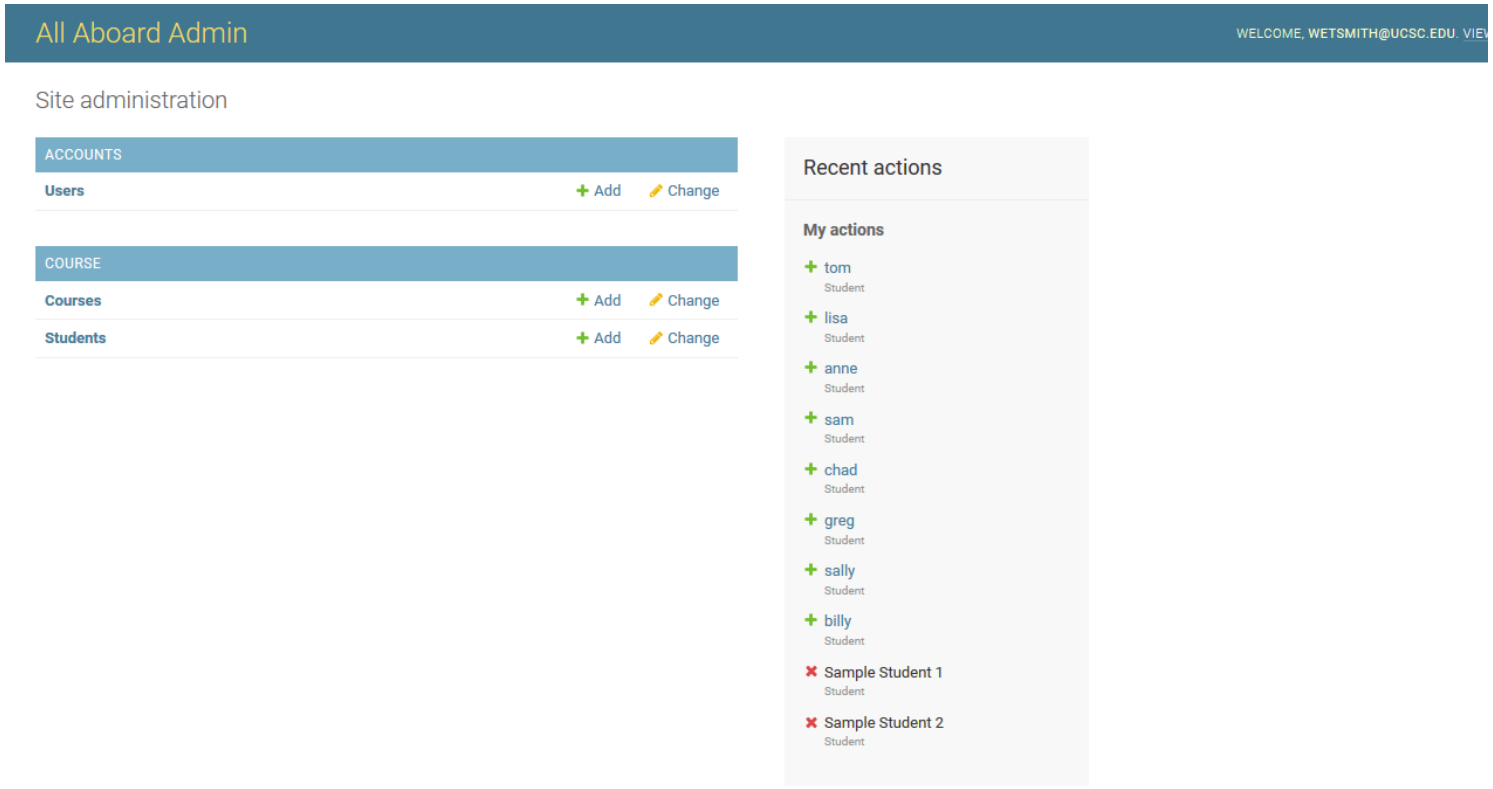
Lecture index

- Lectures contain the attendance information for a course
- From this page, instructors can navigate to previously held lectures or create new ones
- Like courses, lectures are Django models in our database

Lecture attendance page

- Attendance information is rendered into a user-friendly format
- A QR code containing the student sign-in URL is presented for instructor use
- Sign-ins are time locked for 15 minutes after lecture creation
- The connection counter tracks how many attendants a specific student has exchanged ID with
- A NetworkX graph is rendered on page refresh that matches the current lecture attendance.





Admin page

- Courses can be created and assigned students
- A course represents a class offered for a term, e.g., CMPS 101 Fall 2019
- Instructor accounts can be made and assigned to courses
- An instructor account is generally held by the professor for the course

Management Techniques

- Everything Scrum (Google Sheets, Discord, GitHub)
- Required tutorials and educational material
- Style and comment standards
- Routine meetings, as close to daily as possible

Minimum Viable Product

As a **student**, I want my attendance to be accurate so I receive fair credit

- Currently student login is easily tricked, resulting in false positives for attendance.
- We have limited false negatives by providing detailed warnings and clear instructions on student attendance pages.
- Future releases should tether institution-wide logins with attendance logins. For instance, a Canvas login could be used.

As an **instructor,** I want attendance to be taken quickly so I can maximize lecture time

- The closer students sit together, the faster the process.
- Exchanging ID with two adjacent students takes less than thirty seconds. In optimal seating cases, attendance can take less than a minute!
- In larger (100+ student) lectures with sub-optimally seated students, we anticipate a worst-case of five minutes for accurate attendance.

As a **researcher,** I want data on student seating trends so I can create new insights

- All necessary data is stored and can be queried from the database.
- Student-to-student connection direction is also tracked, with the edge pointing to the student who enters another student's ID.
- Only one network visualization is built in to this release.
- In future releases, a large variety of visualizations and analytics should be implemented.

Likes / Dislikes

WHAT WE LIKED

Learning how to track activity and productivity for group members

Working on our own project without an assignment description

Wading into web development and database management

WHAT WE DIDN'T LIKE

Scheduling in person meetings was difficult

Being new to Django and Scrum caused a lot of errors and delays

What we learned

- Accountability is huge in project management
- We improved our Scrum planning skills significantly
- Remote conferencing isn’t ideal. In-person was better
- Stay Mindful of Global Architecture (i.e. limit dependencies)
- We learned a lot about Django and web development in general
- Wish we had more time. Doesn’t everyone?

Q&A