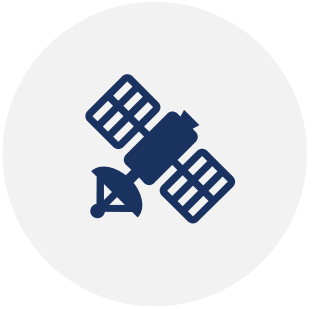
COLD CALLER

NICHOLAS BONAT, ZACH DOMKE, QI HAN, VU VO, JERRY XIE

BASIC FUNCTIONALITY

IMPORT



EXPORT

COLD CALL

LOGS

DEMO TIME

TECHNOLOGIES USED

- Python | Tkinter
- py2app

SYSTEM REQUIREMENTS

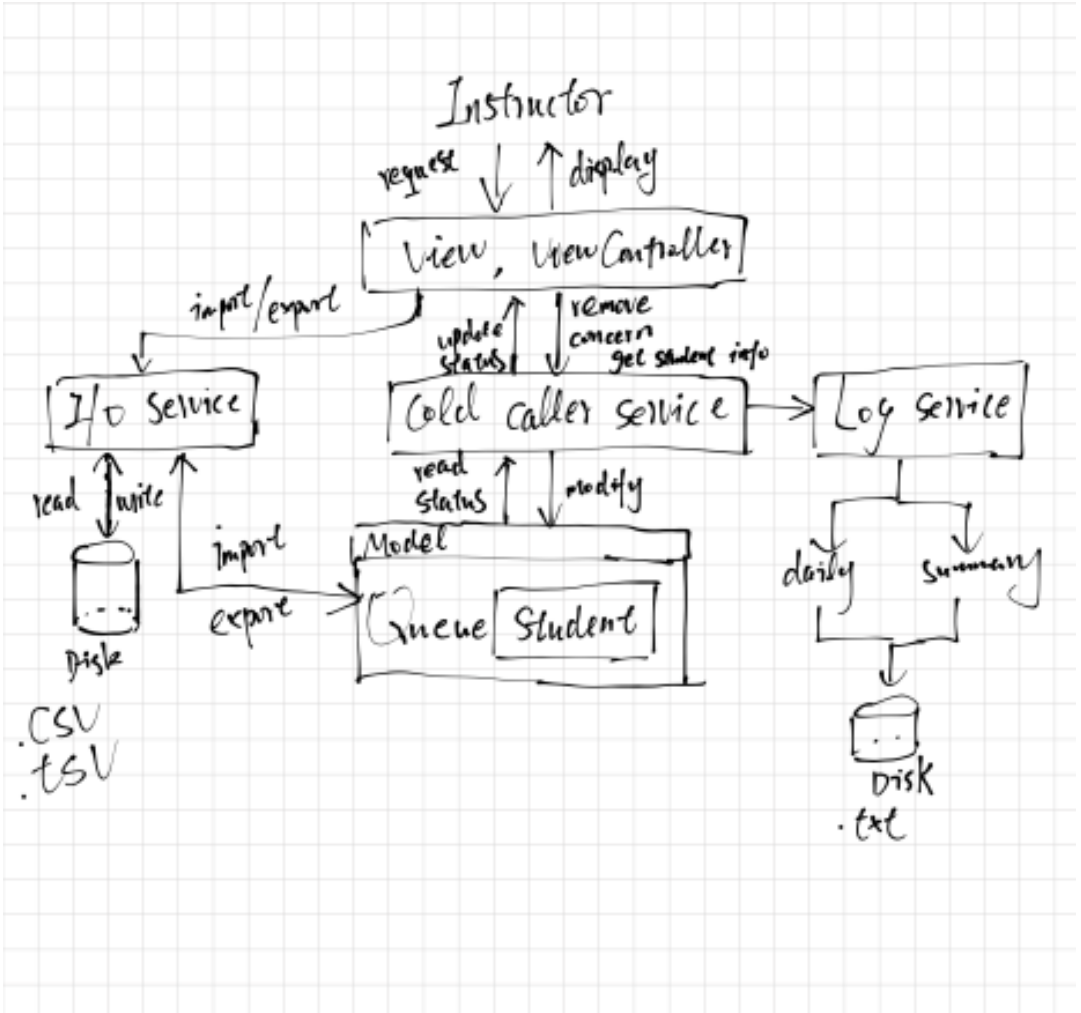
- Dissected the given SRS and chose what exact requirements we wanted to implement
- We learned that ambiguous system requirements gives us more room to be creative, but limits our knowledge of what the client exactly wants

DESIGN REQUIREMENTS

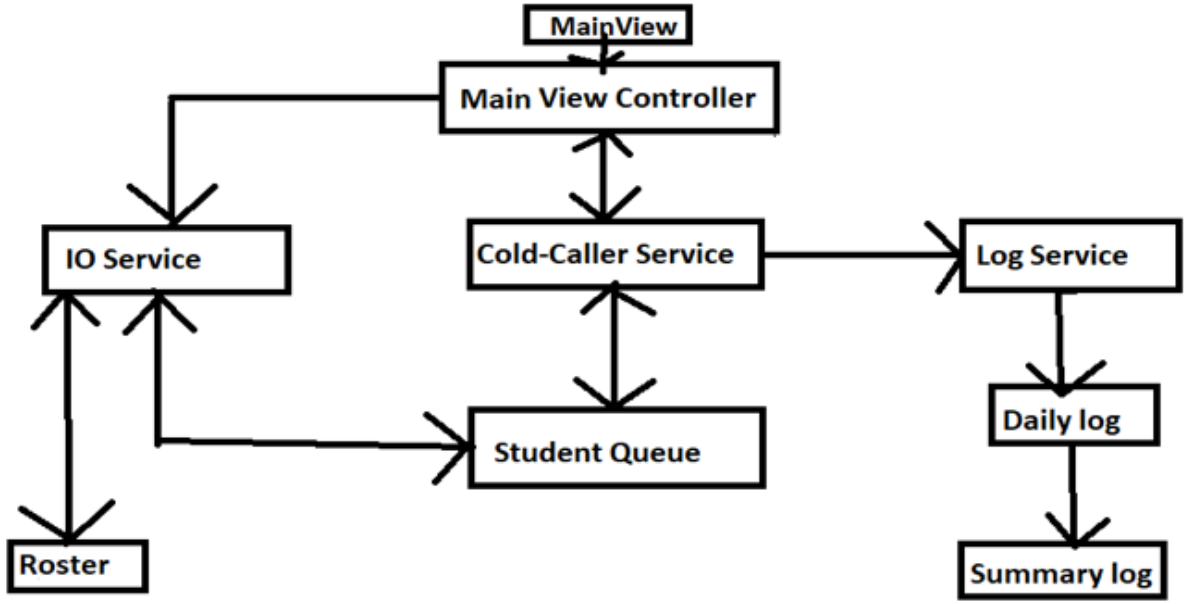
- Applied design patters
- Split up the tasks(importing/exporting/logging/reinsertion)

INITIAL DESIGN

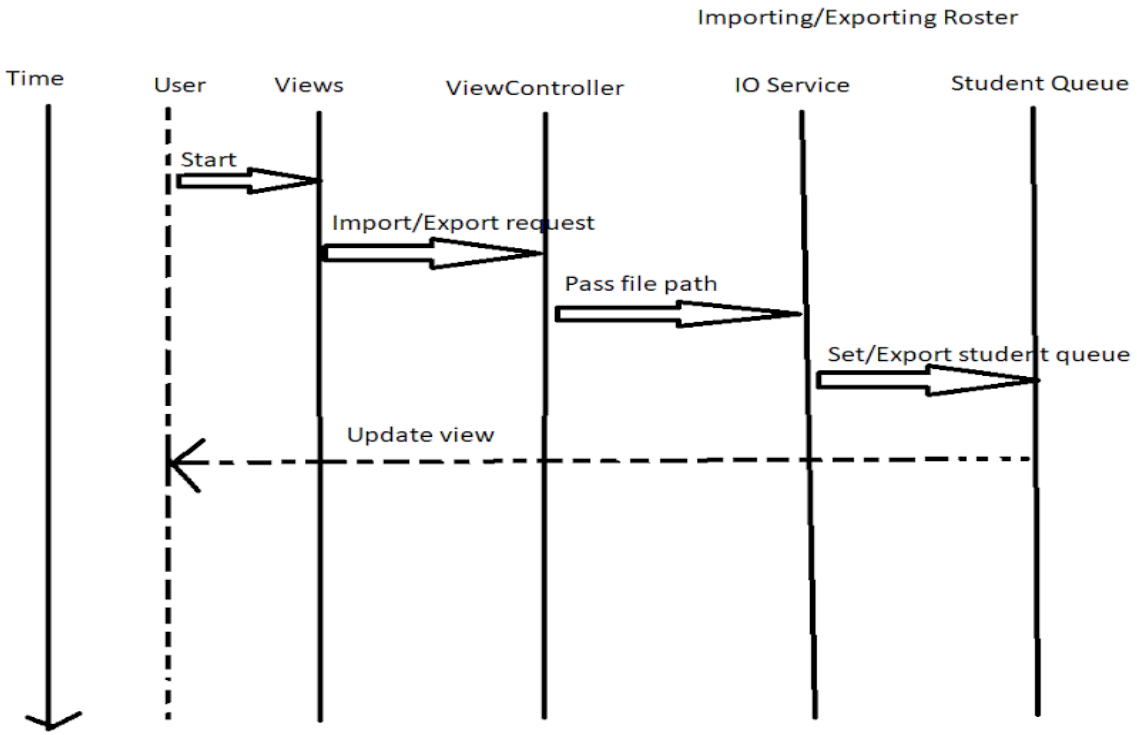
- BAD
- Mix of static and dynamic



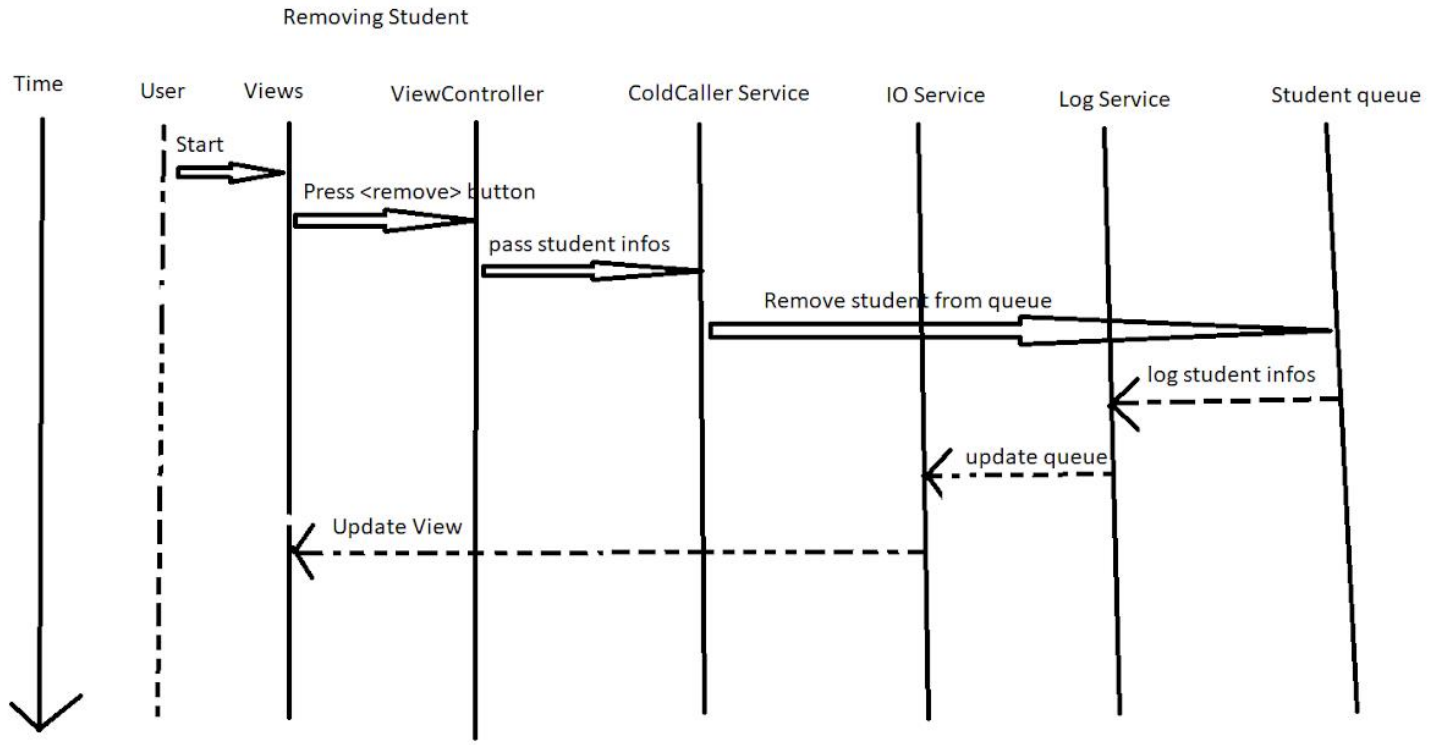
STATIC DIAGRAM



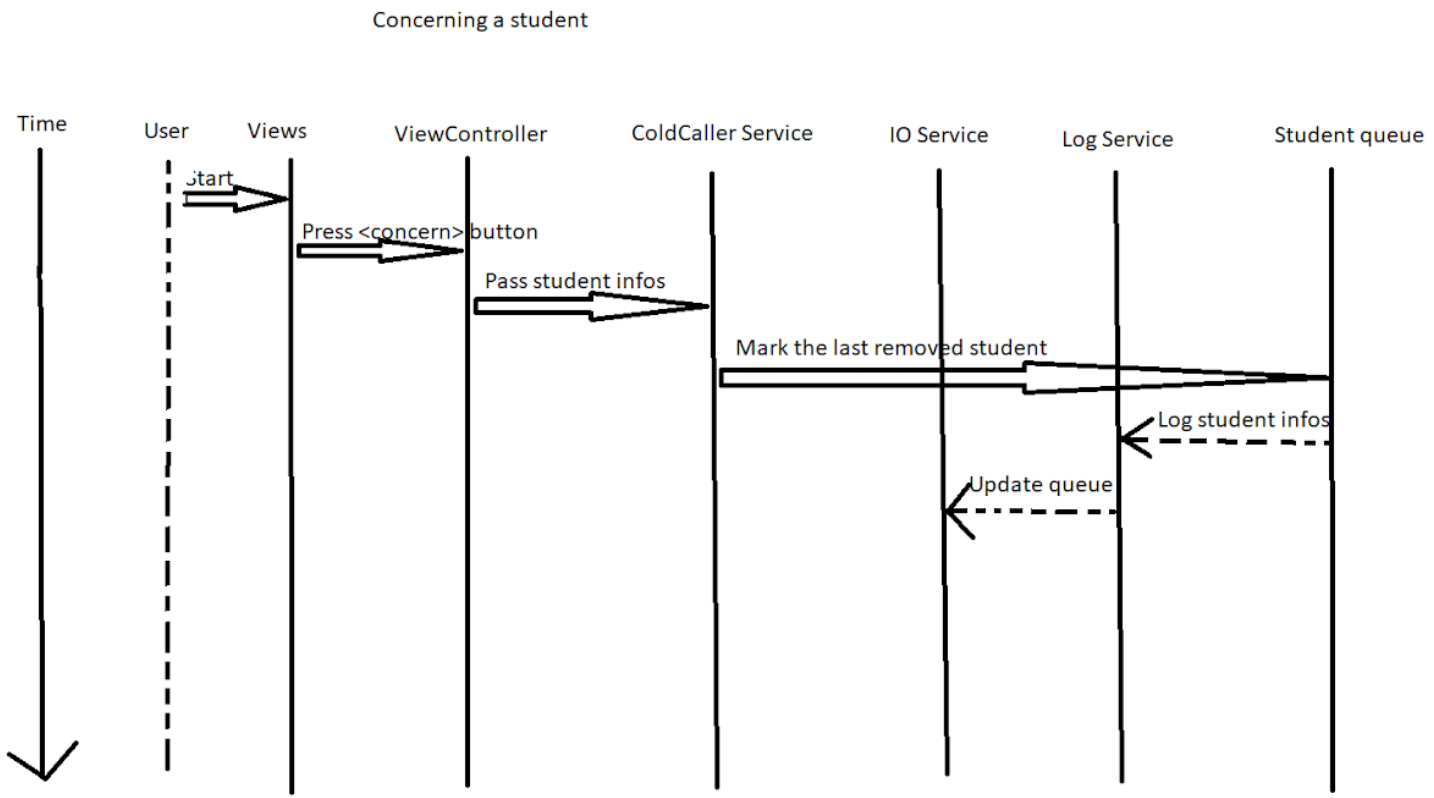
SEQUENCE DIAGRAM



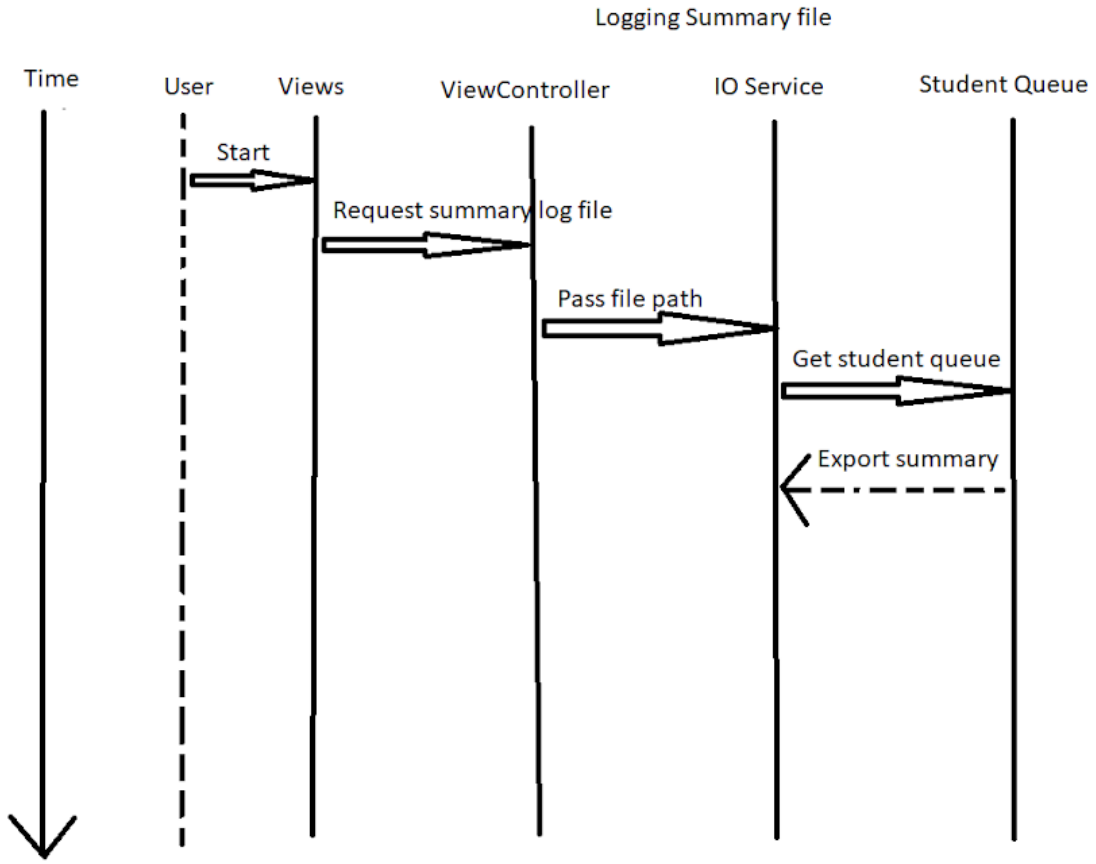
SEQUENCE DIAGRAM



SEQUENCE DIAGRAM



SEQUENCE DIAGRAM



INTEGRATION AND TESTING

- Incremental development
- Testing individual functions – then modules – then system as a whole
- What we learned – software testing takes more time than coding the actual function

MANAGEMENT / EXPERIENCE

- Management

- Assign a project manager
- Assign a developer to a module
- Meet together twice a week and four times in week 4

- Experience

- Underestimated workload (especially documentation)
- Various members were good at specific things(i.e. GUI interface, IO, etc.)

TAKEAWAYS

- Stable SRS/SDS is a must in order to create the intended program
- Knowing member's skills is important for splitting up work
- Teamwork makes the dream work