Design Document

Group 1: Bryon Burleigh, Freelin Hummel, Kabir Kang

Components

External:

- Advising System (external, we will use template emails of the same format for testing)
- Email

Internal:

- Procmail Filter
- Appointment Database
- Command line interface

Tasks

Procmail Filter

- Make simple test filter
- Make function for sending email
- Write a parsing function
- Write a function for adding data to the database

Appointment Database

- Create tables
- Add test data
- Write queries

Command line interface

- Learn curses
- Make simple interface
- Have interface use data from the database
- Add email sending function
- Add ability to cancel appointments

Use Cases

Filter updates database and attaches meeting request to email

Actor: Student / Automated advising emails
Preconditions: Email is the correct format for an advising meeting request or cancellation
Postconditions: The email has a meeting request or cancellation attached, appointment database is updated.
Flow of events:

1. Student signs up for an appointment using the advising web form.
2. The form sends an automated email to the adviser.
3. The filter intercepts the email before it reaches their inbox, and does the following:
    a. Adds or removes the appointment information to the appointment database
    b. Attaches a meeting request or cancellation to the email
    c. Continues sending the email

View appointments with client

Actor: Adviser
Preconditions: Adviser has the client
Postconditions: The client displays the advisers current appointments.
Flow of events:

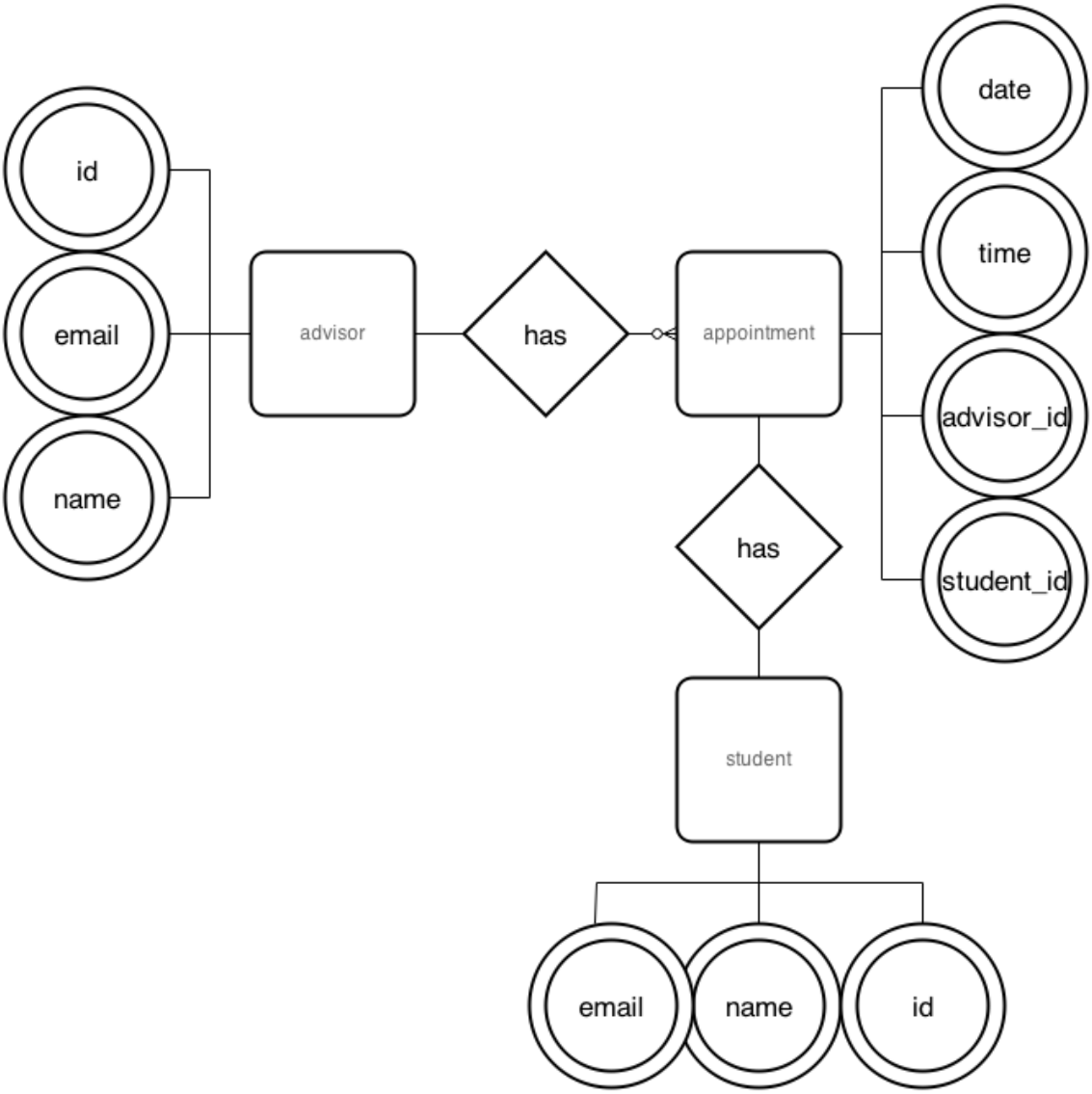
1. The adviser enters their email
2. The client retrieves all the appointments belonging to the email from the appointment database
3. The client lists all these appointments

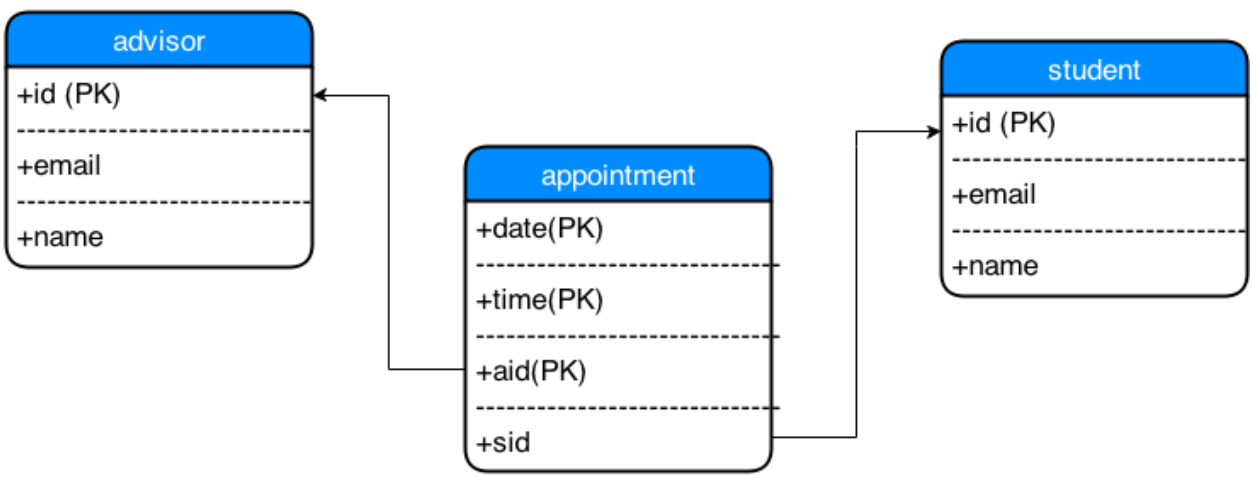
Delete appointments with client

Actor: Adviser
Preconditions: Appointments are displayed accurately, and an appointment is selected by the user.
Postconditions: A cancellation email is sent to the advisor which removes the appointment from the appointment database.
Flow of events:

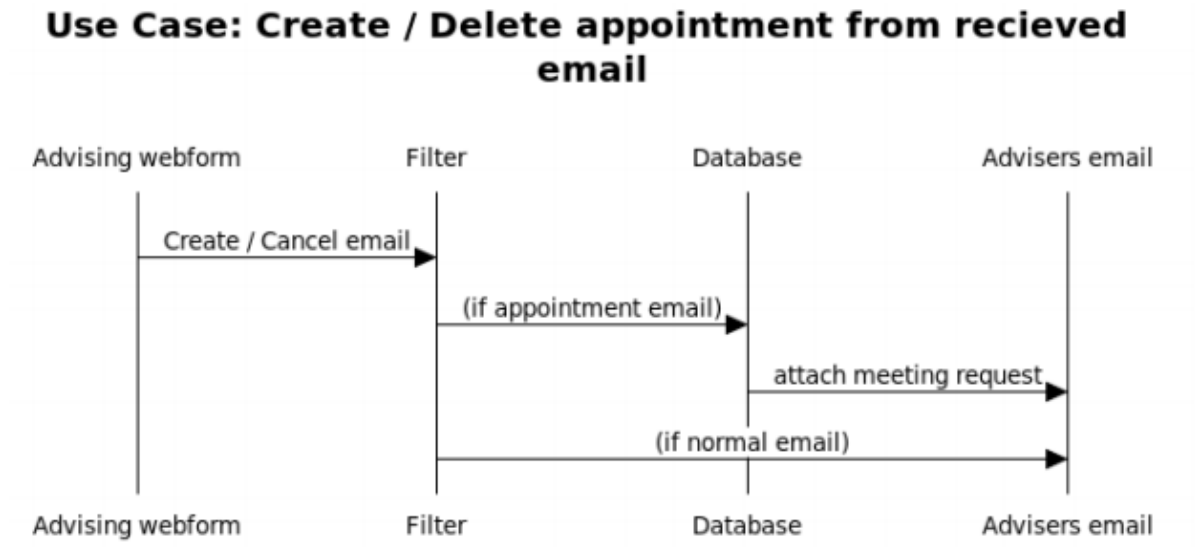
1. The advisor selects to remove an appointment
2. The client sends a cancellation email
    a. The filter removes the appointment from the database
3. The client lets the user know the appointment is being removed

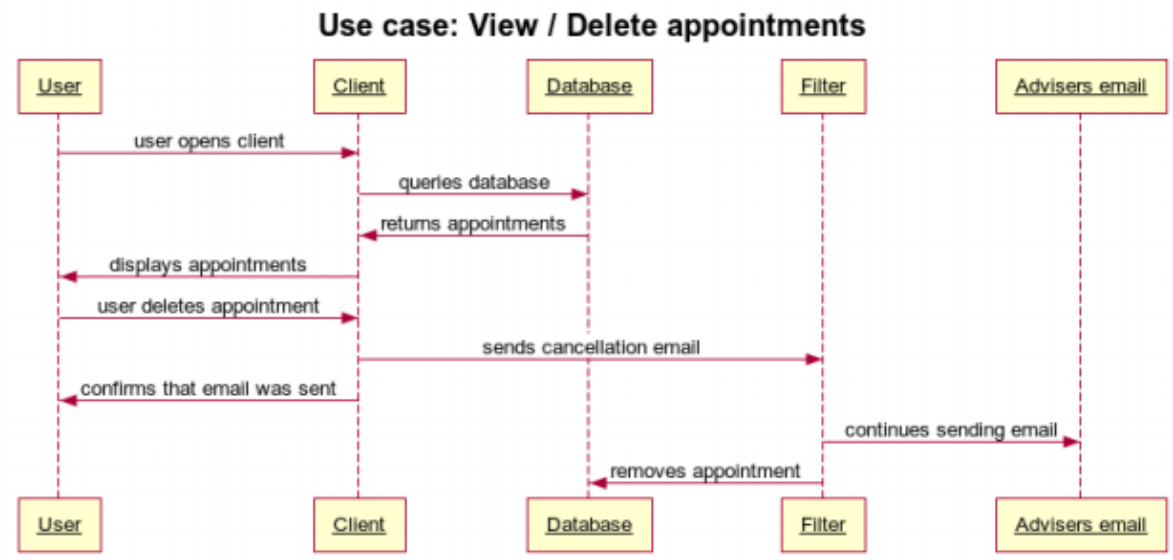
Database ER Diagram & Schema





Message Sequence Diagrams





Project Timeline

Week 4:

- Start working on procmail filter
  - Get email filter working (does not need to do anything at the moment, but should be in place)
  - Get email sending function working
    - Should both be able to send email normally, or with a calendar attachment
  - Start on parsing function
- Test
- Start looking at using curses

Week 5:

- Make simple curses interface
  - Does not use database - placeholder values
- Write queries for database that the client will use
- Test

Week 6: Demo

- Continue testing
- More work on curses interface
- Make demo

Week 7

- Finish work on parsing function
- Finish filter
  - At this point the filter should receive, parse, and send the correct emails.
- If time, more work on client
- Test

Week 8

- Work on client
  - Connect client to show appointments from database
  - Allow sending of cancellation emails
- Test

Week 9

- More testing

Week 10

- More testing
- Work on final report

Week 11

- Work on final report