CSCE 606: Shockwave Kickers

Team members: Colton Simpson, Jacob Kelly, Kunal Vudathu, Rahul Shah, Stuart Nelson

Iteration 1

- **Team roles**: The team should elect a scrum master and product owner.
  - We have an agreed upon rotation that will shift with each iteration
  - Scrum Master: Rahul Shah
  - Product Manager: Kunal Vudathu
- **Customer meeting date/time/place**: The team product owner should contact the customer and set up a regular weekly meeting.
  - Every Friday at 2pm in PETR
- **SCRUM meeting date/time/place**:
  - Every Tuesday and Thursday at 10am
- Github: https://github.com/Lightfoot-Heavy-Machinery/shockwave_kickers
- Pivotal Tracker: https://www.pivotaltracker.com/n/projects/2595510
- Slack Workspace name:
  - Workspace: tamu.slack.com
  - Channel Name: csce606_shockwave_kickers
- Heroku Deployment: https://shockwave-kickers.herokuapp.com/courses

Work Accomplished:

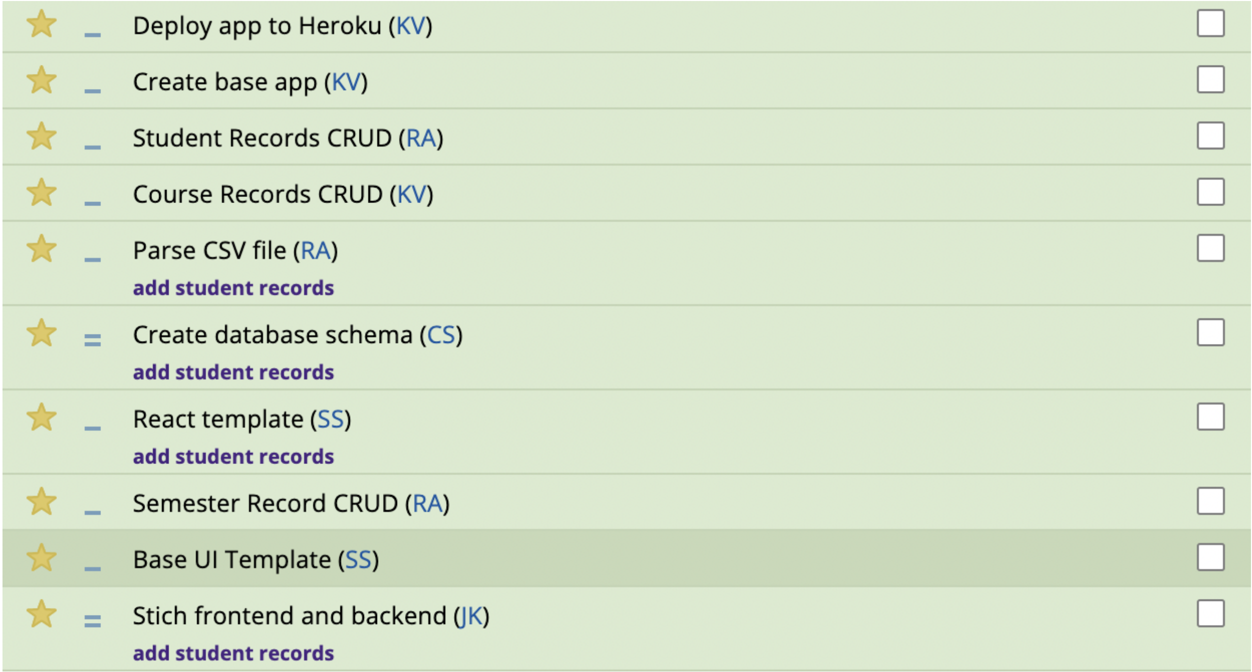
- Got base app working with frontend being react framework and backend being rails.
- We implemented CRUDS to add semester, course and student.
- Got Base UI template with upload page working
- Deployed base app to Heroku (all the changes are reelected there)
- Finalized DB schema

User stories selected in this iteration.

- **Deploy app to Heroku**
  - Deploy backend with simple static web pages to Heroku
- **Create base app**
  - Create the Ruby on Rails template
  - Create and attach a database to the backend (Postgres for dev and prod)
- **Student records CRUD**
  - Create CRUD routes for the student records controller

- **Course records CRUD**
    - Create CRUD routes for the course records controller
- **Parse CSV file**
    - Read in a CSV and split data into a JSON
- **Create database schema**
    - Develop a an outline of the database tables and necessary columns
    - Identify primary keys
- **React template**
    - Create a base react app template
- **Semester records CRUD**
    - Create CRUD routes for the semester records controller
- **Base UI template**
    - Within the react app, add pages for the UI
    - Based upon our Figma wireframe
- **Stitch frontend and backend**
    - Be able to pass file uploaded in react to backend parser
    - Begin connecting react views to backend controllers/models
- **CI/CD Pipeline (Not accomplished)**
    - Integrate CI/CD pipeline into Github and Heroku
    - Pushed to a later iteration

Stories in Pivotal Tracker:



★ Deploy app to Heroku (KV) □

★ Create base app (KV) □

★ Student Records CRUD (RA) □

★ Course Records CRUD (KV) □

★ Parse CSV file (RA) □
add student records

★ Create database schema (CS) □
add student records

★ React template (SS) □
add student records

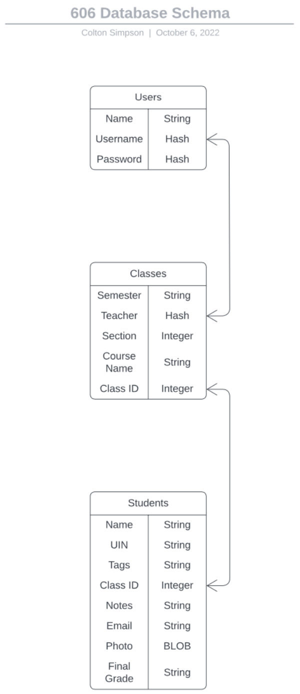
★ Semester Record CRUD (RA) □

★ Base UI Template (SS) □

★ Stich frontend and backend (JK) □
add student records

Design Diagrams for this iteration:

1) DB schema diagram:


606 Database Schema

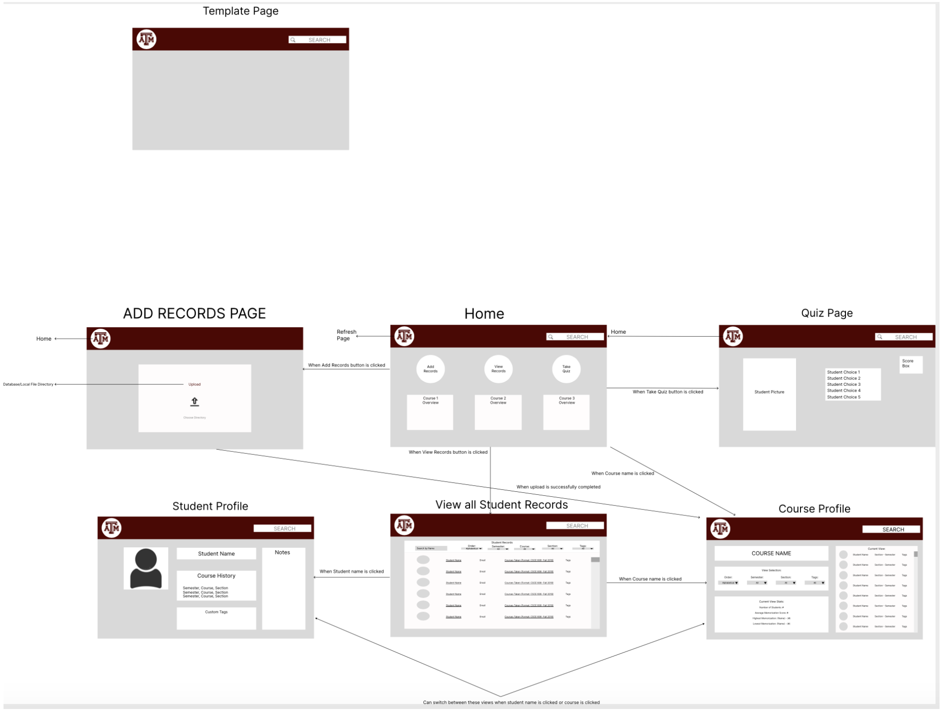
Colton Simpson | October 6, 2022

Users	
Name	String
Username	Hash
Password	Hash

Classes	
Semester	String
Teacher	Hash
Section	Integer
Course Name	String
Class ID	Integer

Students	
Name	String
UIN	String
Tags	String
Class ID	Integer
Notes	String
Email	String
Photo	BLOB
Final Grade	String

2) Features:



Stories planned but not implemented for this iteration:

- Build CI/CD pipeline using github actions
    - We planned to implement CI/CD using github actions. Basically the plan was to run tests and deploy to heroku after every code merge.

Stories Added/Changed:

- For Iteration 2, we created stories to perform db migrations on the finalized schema.
- Created stories for end to end development work of the major features that the client requested.

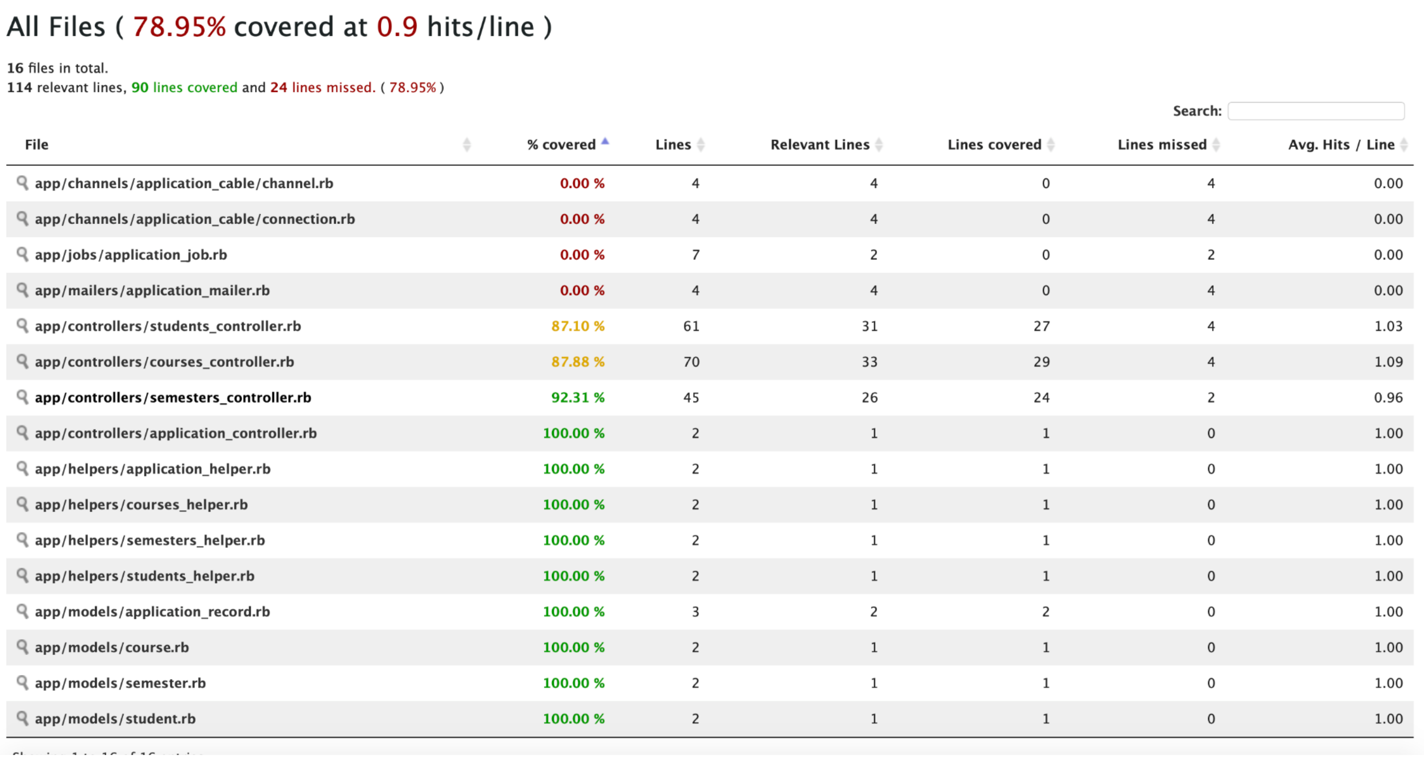
Iteration 2 work:

- Based on the finalized DB schema, we will need to perform migrations on the CRUD stories as the DB scheme is slightly different from production.
- Get the upload feature working end to end, with it updating correctly on the new DB schema
- Basic functionality for Quiz feature
- Get CI/CD pipeline working

Test Coverage Report

Unit Cases Coverage:

Achieved 78.95% coverage from 0% in this iteration:



All Files (78.95% covered at 0.9 hits/line)

16 files in total.
114 relevant lines, 90 lines covered and 24 lines missed. (78.95%)

File	% covered	Lines	Relevant Lines	Lines covered	Lines missed	Avg. Hits / Line
app/channels/application_cable/channel.rb	0.00 %	4	4	0	4	0.00
app/channels/application_cable/connection.rb	0.00 %	4	4	0	4	0.00
app/jobs/application_job.rb	0.00 %	7	2	0	2	0.00
app/mailers/application_mailer.rb	0.00 %	4	4	0	4	0.00
app/controllers/students_controller.rb	87.10 %	61	31	27	4	1.03
app/controllers/courses_controller.rb	87.88 %	70	33	29	4	1.09
app/controllers/semesters_controller.rb	92.31 %	45	26	24	2	0.96
app/controllers/application_controller.rb	100.00 %	2	1	1	0	1.00
app/helpers/application_helper.rb	100.00 %	2	1	1	0	1.00
app/helpers/courses_helper.rb	100.00 %	2	1	1	0	1.00
app/helpers/semesters_helper.rb	100.00 %	2	1	1	0	1.00
app/helpers/students_helper.rb	100.00 %	2	1	1	0	1.00
app/models/application_record.rb	100.00 %	3	2	2	0	1.00
app/models/course.rb	100.00 %	2	1	1	0	1.00
app/models/semester.rb	100.00 %	2	1	1	0	1.00
app/models/student.rb	100.00 %	2	1	1	0	1.00

Showing 1 to 16 of 16 entries