CSCE 606: Shockwave Kickers

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

Iteration 4

- **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: Colton Simpson
  - Product Manager: Jacob Kelly
  - **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:

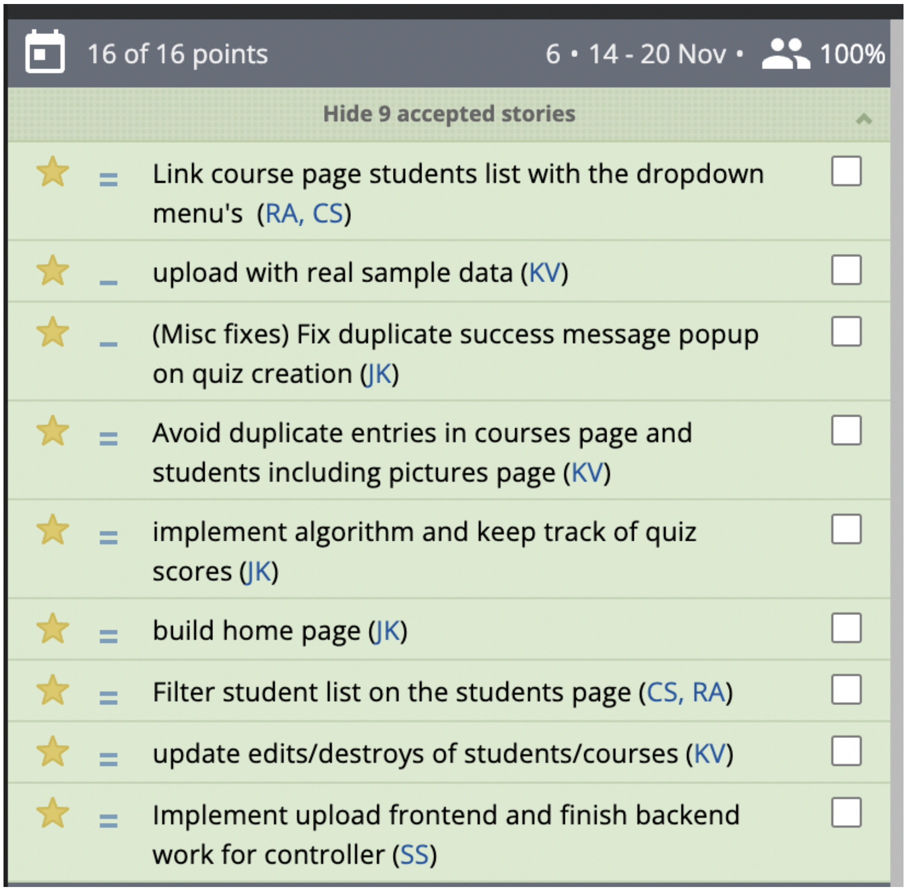
- Filter list of students on Course Profile Page based on semester, section or tags
- Filter list of students on Students Page based on semester, course name or tags
- Added submit button for dropdown menu filters
- Implemented frontend for file uploads
- Implemented home page
- Implement quiz algorithms
- Implemented course history page
- Implemented course edit page where you can change section, course name and semester
- Edited student edit page where the professor can now view student course history as well as add grade of the student for a particular course.
- Added destroy course of a student functionality where you can delete course attached to a student
- Removed duplicate course (based on same course name) and student entries (based on UIN) from UI. Concatenated them into a single table entry on the UI.

User stories selected in this iteration.

- **Upload with real sample data**
    - Get an actual sample CSV to test our parsing
- **(Misc fixes) Fix duplicate success message popup on quiz creation**
    - When a quiz was created/saved, the success/fail message would appear twice
    - To improve the UI/UX, this needed to be changed
- **Filter student list on the students page**
    - A user should be able to filter the list of students by selecting semester, course, and tag filters through the dropdown menus
    - Should work with consolidated student entries (tags, semesters, course, etc.)
- **Link course page students list with the dropdown menus**
    - A user should be able to filter the list of students by selecting semester, section, and tag filters through the dropdown menus
- **Avoid duplicate entries in courses page and students including pictures page**
    - Due to db design, the UI was showing duplicate entries for courses and students, the story was concatenate course entries with same into one entry
- **Implement algorithm and keep track of quiz scores**
    - When a quiz is taken, the user should be able to see their score for the quiz
    - A user should be able to view past quiz scores
    - A user should be able to see how many they've gotten correct in a row on a quiz, and what their longest streak in that quiz has been
    - Users should be able to see how many times they incorrectly guessed a student's name before it was guessed correctly
- **Build home page**
    - A dashboard to provide an overview of this user
    - It will show counts of semesters, classes, and years taught
    - It should show average quiz and streak scores for the most recent semester and overall
    - It should display the most relevant students regarding memorization
    - It should display some of the most recent courses
- **Update edits/destroys of students/courses**
    - The story includes add edit and destroy pages for Courses
        - When deleting a course, all students under the course will be deleted

        - When editing a course, course info across all students in the course will update
    - Added a page for viewing course history
    - For student pages, the story includes:
        - Updating student’s grade of that class
        - Updating info across all students
    - Added destroy course of a student functionality where you can delete course attached to a student
- **Implement upload frontend and finish backend work for controller**
    - The upload page UI is finished, and shows the upload button, as well as the upload history.
    - Users can delete previous CSV uploads
    - Backend CSV parsing is done

Stories in Pivotal Tracker:



16 of 16 points — 6 • 14 - 20 Nov • 100%

Hide 9 accepted stories

★ = Link course page students list with the dropdown menu's (RA, CS) □

★ – upload with real sample data (KV) □

★ – (Misc fixes) Fix duplicate success message popup on quiz creation (JK) □

★ = Avoid duplicate entries in courses page and students including pictures page (KV) □

★ = implement algorithm and keep track of quiz scores (JK) □

★ = build home page (JK) □

★ = Filter student list on the students page (CS, RA) □

★ = update edits/destroys of students/courses (KV) □

★ = Implement upload frontend and finish backend work for controller (SS) □

Stories Added/Changed:

- Pushed CI/CD and sample CSV use to Iteration 5
- Based on professor Lightfoot's request, want to add search by email functionality (Iteration 5)

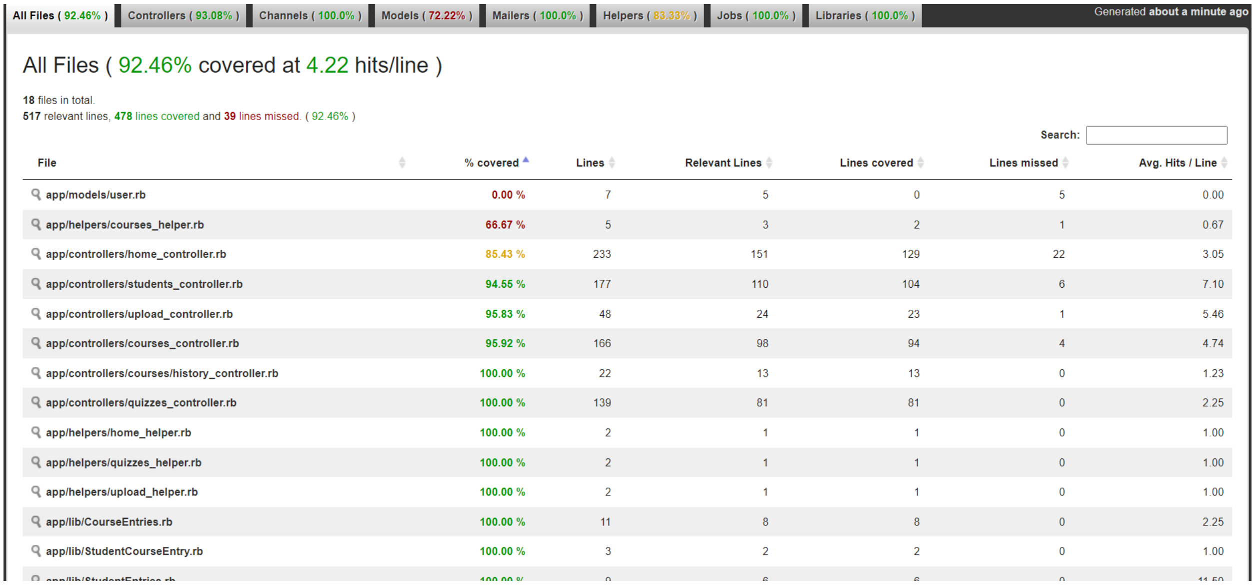
Iteration 5 work:

- Make a CI/CD Pipeline
- Add a search bar for the Students page so that you can find students by email and name
- Search bar for the Courses page so that you can find courses by name
- Add password reset functionality
- Add a profile page for the current signed in user
- Work on student profile page
- Get an actual CSV of sample data and test on it

Test Coverage Report

Unit Cases Coverage:

We increased coverage from 91.74% to 92.46%



All Files (92.46% covered at 4.22 hits/line)

18 files in total.
517 relevant lines, 478 lines covered and 39 lines missed. (92.46%)

File	% covered	Lines	Relevant Lines	Lines covered	Lines missed	Avg. Hits / Line
app/models/user.rb	0.00 %	7	5	0	5	0.00
app/helpers/courses_helper.rb	66.67 %	5	3	2	1	0.67
app/controllers/home_controller.rb	85.43 %	233	151	129	22	3.05
app/controllers/students_controller.rb	94.55 %	177	110	104	6	7.10
app/controllers/upload_controller.rb	95.83 %	48	24	23	1	5.46
app/controllers/courses_controller.rb	95.92 %	166	98	94	4	4.74
app/controllers/courses/history_controller.rb	100.00 %	22	13	13	0	1.23
app/controllers/quizzes_controller.rb	100.00 %	139	81	81	0	2.25
app/helpers/home_helper.rb	100.00 %	2	1	1	0	1.00
app/helpers/quizzes_helper.rb	100.00 %	2	1	1	0	1.00
app/helpers/upload_helper.rb	100.00 %	2	1	1	0	1.00
app/lib/CourseEntries.rb	100.00 %	11	8	8	0	2.25
app/lib/StudentCourseEntry.rb	100.00 %	3	2	2	0	1.00
app/lib/StudentEntries.rb	100.00 %	9	6	6	0	11.50