

CSCE 606: Shockwave Kickers

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

Iteration 5

- **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: Jacob Kelly
 - Product Manager: Rahul Shah
 - **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:

- Build CI/CD pipeline using github actions
- Checks for creating course or course of a student or creating student
- quiz and qroster deletes should be done
- segregate student table into common table and student course table
- Implement sort (asc, des) by student last name in course profile
- student profile end to end
- improve tags capability
- implement user profiles
- search student by name, email
- Finish upload
- add quiz scores to course profiles

User stories selected in this iteration.

- **Build CI/CD pipeline using github actions**
 - Built CI pipeline to make sure all unit cases are run before merging as well as committing the change to prod
 - Built CD pipeline to push change to heroku
- **Segregate student table into common table and student course table**
 - Segregate student table into common table for student common data and student course table for student course data
- **quiz and roster deletes should be done; notes concatenate**
 - As part of student and course deletes, delete respective info of Roster and quizzes
- **Checks for creating course or course of a student or creating student**
 - Add regex checks on the UI to make sure the fields match expected format. This includes: UIN, email, major, course name, section and so on
- **Implement sort (asc, des) by student last name in course profile**
 - The user can now sort the student list alphabetically or reverse alphabetically for increased ease of use
- **Student profile end to end**
 - We can now view the course history for any student from their student profile, as well as the semester and year taken.
- **Adding quiz scores to the course profiles**
 - The user can now look within course profiles to see how well they are remembering names so that they can further practice courses that have the lowest scores
- **Completing the student upload functionality**
 - When the user provides the application with a zip file containing a csv of student attributes followed by images of each of the students, students will be posted to the database and will appear on the students page and the courses page of the application.
- **Filtering students by search in both the courses and students page**
 - The user is given an increased ability to look for a student profile to check for specific attributes that may be useful for quizzes or for future letters of recommendation.
- **Filtering students by tags**
 - To make it easier to find students with specific notes or attributes, the filter-by-tag feature allows a user to flag a student with custom attributes to add ease-of-use to name memorization or future letters of recommendation.
- **Implementing user profiles by allowing for password resets**

- Names for accounts and password reset functionality was added in case a user forgets their password or if certain classes need to be in the field of view of one user and not another. (Increased scalability)

Stories in Pivotal Tracker:

Current Iteration/Backl... 5 + Add Story ⋮ ✕			
📅	18 of 18 points	8 • 28 Nov - 4 Dec • 👤 100%	
Hide 11 accepted stories			⬆
★	—	add quiz scores to course profiles (JK)	<input type="checkbox"/>
★	=	Finish upload (SS)	<input type="checkbox"/>
★	=	search student by name, email (CS, SS)	<input type="checkbox"/>
★	=	Build CI/CD pipeline using github actions (KV)	<input type="checkbox"/>
★	—	Checks for creating course or course of a student or creating student (KV)	<input type="checkbox"/>
★	—	quiz and qroster deletes should be done; notes concatenate (KV)	<input type="checkbox"/>
★	=	segregate student table into common table and student course table (KV)	<input type="checkbox"/>
★	—	Implement sort (asc, des) by student last name in course profile (RA)	<input type="checkbox"/>
★	=	student profile end to end (RA)	<input type="checkbox"/>
★	=	improve tags capability (CS)	<input type="checkbox"/>
★	=	implement user profiles (JK)	<input type="checkbox"/>

Stories Added/Changed:

- No stories added except one db design change story that needed to be done where we needed to segregate the students table into a common students table and a student course table

Test Coverage Report

Unit Cases Coverage:

We increased coverage from 92.46% to 93.35%

All Files (93.35%)	Controllers (92.91%)	Channels (100.0%)	Models (100.0%)	Malliers (100.0%)	Helpers (83.33%)	Jobs (100.0%)	Libraries (100.0%)	Generated less than a minute ago
All Files (93.35% covered at 4.57 hits/line)								
21 files in total. 677 relevant lines, 632 lines covered and 45 lines missed (93.35%)								
File	% covered ▾	Lines ▾	Relevant Lines ▾	Lines covered ▾	Lines missed ▾	Avg. Hits / Line ▾	Search: <input type="text"/>	
🔍 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	4.75		
🔍 app/lib/StudentCourseEntry.rb	100.00 %	3	2	2	0	1.00		
🔍 app/lib/StudentEntries.rb	100.00 %	9	6	6	0	16.00		
🔍 app/models/course.rb	100.00 %	16	10	10	0	1.80		
🔍 app/models/qroster.rb	100.00 %	2	1	1	0	1.00		
🔍 app/models/quiz.rb	100.00 %	11	8	8	0	6.25		
🔍 app/models/student.rb	100.00 %	21	12	12	0	2.67		
🔍 app/models/student_course.rb	100.00 %	2	1	1	0	1.00		
🔍 app/models/students_tag.rb	100.00 %	2	1	1	0	1.00		
🔍 app/models/tag.rb	100.00 %	2	1	1	0	1.00		
🔍 app/controllers/students_controller.rb	95.65 %	218	138	132	6	5.85		
🔍 app/controllers/upload_controller.rb	94.59 %	84	37	35	2	3.76		
🔍 app/controllers/courses_controller.rb	90.11 %	288	182	164	18	6.59		
🔍 app/controllers/student_courses_controller.rb	90.00 %	29	20	18	2	1.00		
🔍 app/controllers/home_controller.rb	89.33 %	233	150	134	16	3.24		
🔍 app/helpers/courses_helper.rb	66.67 %	5	3	2	1	0.67		