Team: ANTE

Elias Tracy - Product Owner | Tuong Tran - SCRUM Master
Andrew Imwalle - Developer | Nick McDonough - Developer

Pivotal Tracker: https://www.pivotaltracker.com/n/projects/2598148
GitHub:https://github.com/tamu-edu-students/fighting-aggies-platform
Slack: https://app.slack.com/client/T04GX9UU3V3/C04H08B4N0J
Heroku: https://fighting-aggies-platform.herokuapp.com/

Coach login:

Username:faaplicationuser@gmail.com
Password: password2023

Admin login:

Username: faaplicationmanager@gmail.com
Password: password2023

Data Manager login:
Username: faapdatamanager@gmail.com
Password: faapdata

Customer Meeting Date:	1/30/2023
Time:	3:45pm
Location:	Zoom

Meeting Notes:
- Notes
  - When clicking player name
    - See all the routes they run
    - See total success rates
    - When clicking route
      - It'll show all of his attempts at that route
  - Should never see practice data
    - Shouldn't see the 1's and 0's
    - Want only summary data unless specifically asked for
  - Pick a player -> see their routes
    - Choose a route
      - See all instances of that route
    - Choose a practice
      - See all routes from that practice

-
  -
    -
      - Ability to see time range of practices
        - Only want to see recent data
    - Main stuff shown
      - Show summary data
    - Shouldn't have a set page
    - Increasing specificity should be applied as you add more choices
    - If they don't run the route, it shouldn't be 0 percent
      - It should be N/A or ----
    - If i click on a percentage
      - Go to that players instance of that route
    - Header where you can change the practice set
    - Click a player
      - Goes to player stats
    - Click on route
      - Goes to all players who run that route
    - Choose an intersection
      - Should show all instances of that route
    - Data Manager
      - In charge of video
    - UI Mockups on the data manager site?
      - Haven't made yet, make before showing client site
      - Should look like dropbox or google drive
        - A file explorer
      - Coaches can watch the videos, no edit or upload
    - When a coach drills down on plays and routes
      - Coach can click instance, play video with timestamp or clip of that player in that play doing that route
      - FUTURE: add note section for coaches to add notes to specific instances of routes
    - Main functionality is report cards
      - If coach has specific notes about player doing this route during this practice
        - Player should be able to see that comment in the report card
    - Advice for next client meeting
      - Have a team meeting to collect everything we've done
      - Have one together pitch, don't say i did this, he did this
      - Use "we did this"
      - Bring UI lo fi mockups
      - Get feedback from client before developing things
  - Prof advice
    - Make several templates to see what we need in the back end
    - Find common similarities to see what we need on the front end
    - Will take several semesters
   - "Show me these players on these routes"

- ■ Generates table of players, routes, accuracy
- ○ Make tableau?
- ○ Choose data -> choose data format -> make report
- ○ Big focus
  - ■ Make it easy to pass on

User Stories

User stories successfully implemented this iteration:

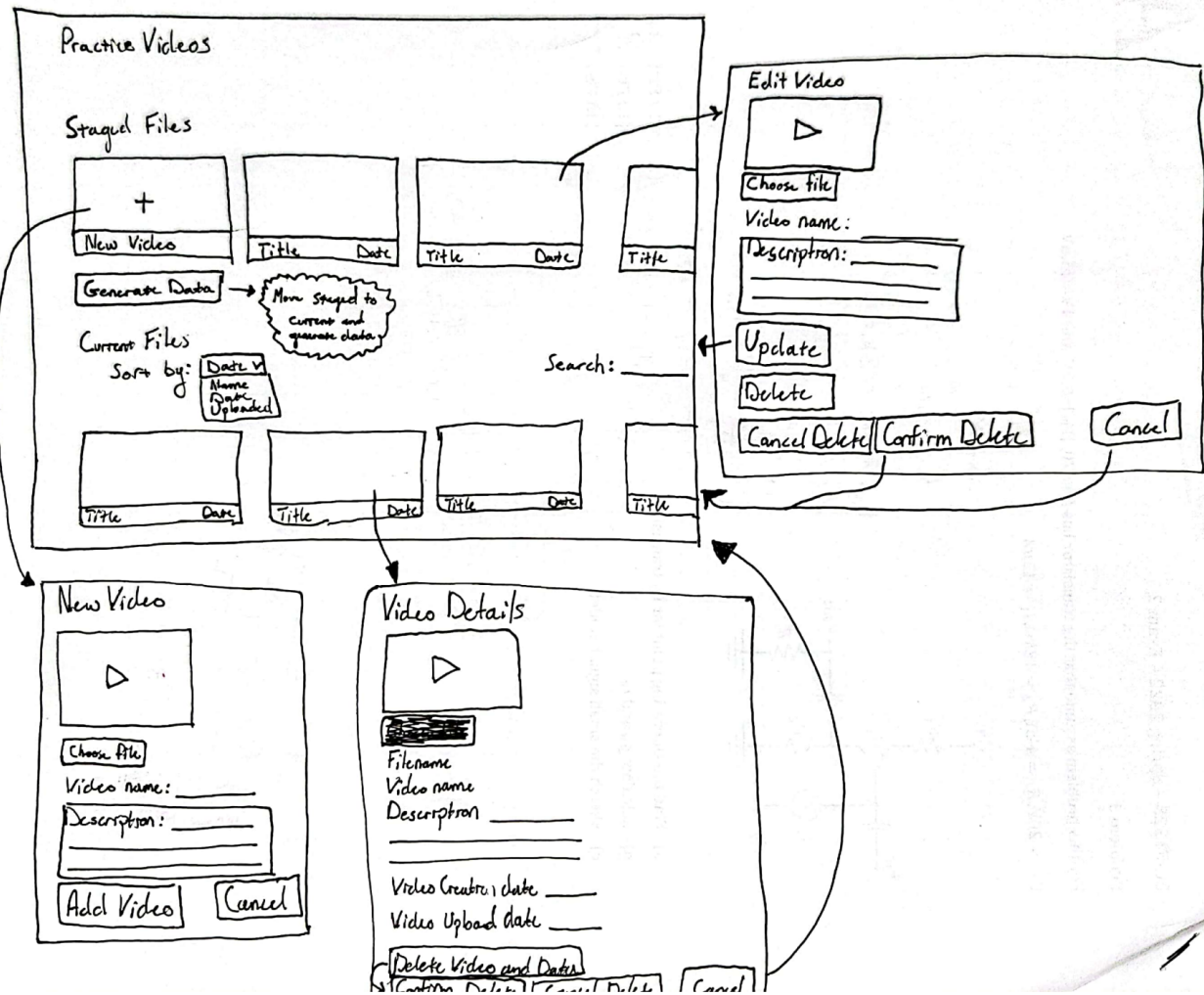| User Story	Points	Completed By
Feature: View Analytics Dashboard	1	Nick
Feature: Edit submitted videos	1	Eli
Feature: Delete submitted videos	1	Eli
Feature: CI/CD Pipeline	1	Eli
Refactor: Dashboard	2	Nick
Feature: View Previously Submitted Video Repository	1	Andrew
Feature: Add Videos to Database	3	Andrew
Feature: Admin Table can be sorted by role	1	Tuong
Feature: Admin Table can be sorted alphabetically	1	Tuong
Feature: Log in/out of Submitted Videos	1	Andrew/Eli
Refactor: Login page styling	1	Eli

Design Diagram
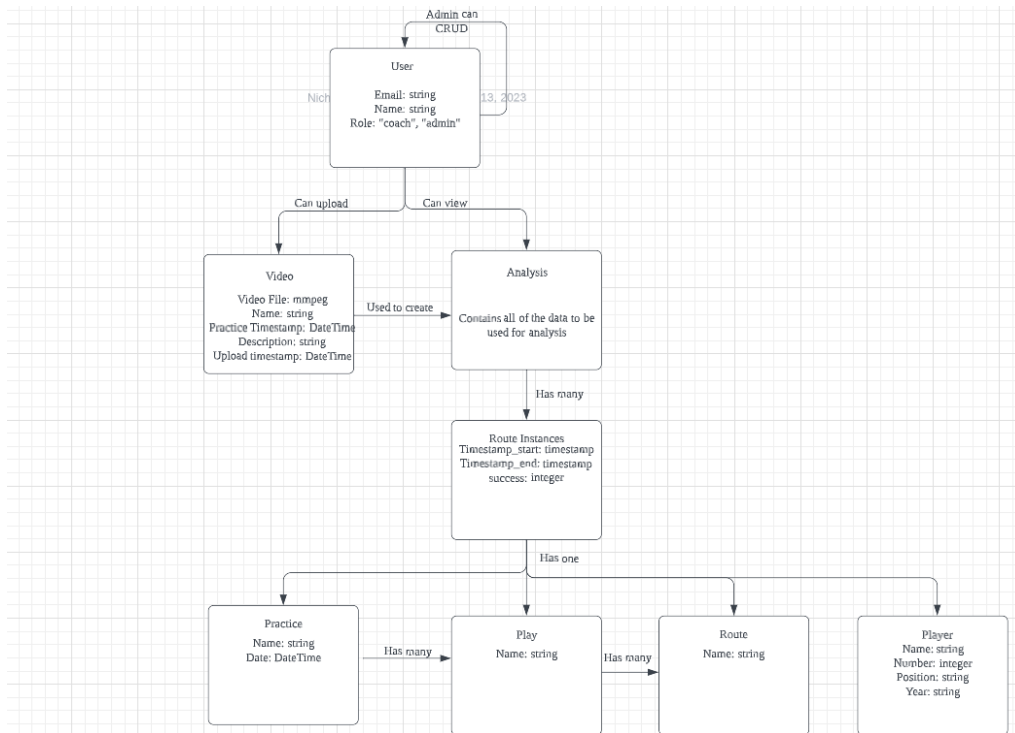
Lo-fi UIs

User Stories related to this User Interface
Feature: Log in/out of Submitted Videos
Feature: Edit submitted videos
Feature: Delete submitted videos
Feature: View Previously Submitted Video Repository
Feature: Add Videos to Database

Data Manager

Practice Videos

Staged Files

+			
New Video	Title Date	Title Date	Title

[Generate Data] → (Move staged to current and generate data)

Current Files
Sort by: [Date ∨ / Name / Date Uploaded]

Search: _____

Title Date	Title Date	Title Date	Title

Edit Video

▷

[Choose file]
Video name :
Description: _____

[Update]

[Delete]

[Cancel Delete] [Confirm Delete]   [Cancel]

New Video

▷

[Choose file]
Video name: _____
Description : _____

[Add Video]   [Cancel]

Video Details

▷

Filename
Video name
Description _____
_____
_____

Video Creation date _____
Video Upload date _____

[Delete Video and Data]
[Confirm Delete] [Cancel Delete] [Cancel]

User Stories related to this User Interface
Feature: View Analytics Dashboard
Refactor: Dashboard

Coach Dashboard

[Log Out]

Player Performance

>Template1
Template2
Template3

Included Practices
    57 Practices Selected

Generate Reports

Player Performance Per Route

	Route1	Route2	Route3	Route4
		%	%	%
Name1	%		%	%
		%		
Name2	%			
		%	%	%
Name3	%			
			%	%
Name4	%			

→Redirect to login Page and end session

If user is logged in and attempts to route to either admin page or login page, return to dashboard
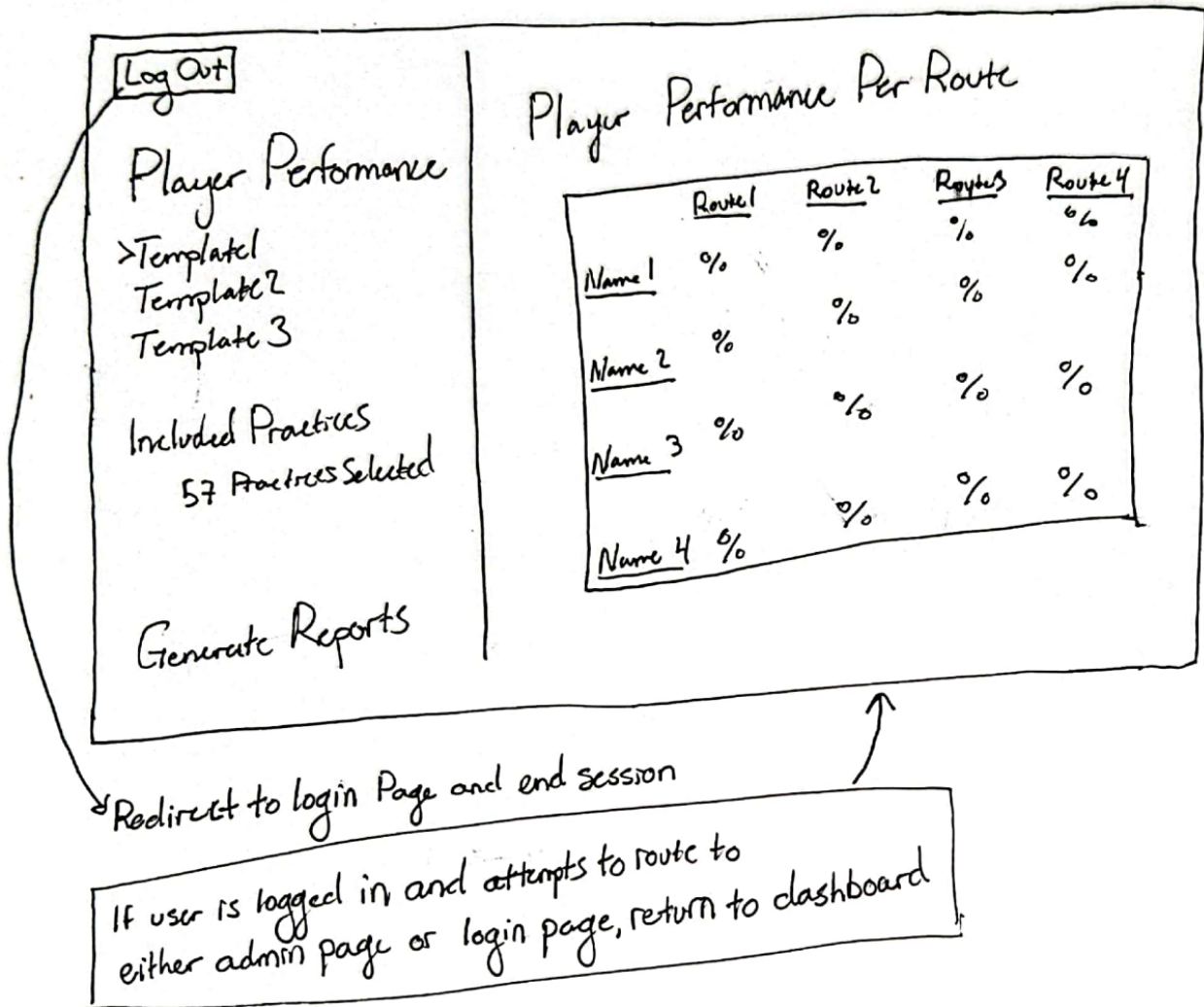
Coach Dashboard   when user selects name 1 from Template 1:

Log Out

Player Performance
>Template 1
Template 2
Template 3

Included Practices
   57 Practices Selected

Generate Reports

Name 1

Route Name	Play Name	Timestamp	Success
		00:00:00	1
		00:00:10	0
		00:00:20	1
		00:00:30	1

If the user is on template 1, and they select a player's name, they will view data relevant to that specific player during the included practices.

Coach Dashboard   When user selects route 1 from Name 1

Log Out

Player Performance
>Template 1
Template 2
Template 3

Included Practices
   57 Practices Selected

Generate Reports

Name 1 > Route 1

Play Name	Timestamp	Success
		1
		0
		1
		0

If the user is on the player's data and they select a specific route, they will view data relevant to that specific player during the included practices, on the specific route.

Updated User Stories

N/A

Incomplete User Stories

N/A

Legacy Report:

During this iteration, the group has had the opportunity to replace the legacy dashboard with our own dashboard. This is the final part of the product that was previously written in react. We have replaced this with ruby on rails which has streamlined the development process and will make it easier for the future developers to begin work. By removing the react code, the application.js file has been reduced to 1/12th the size that it was at the end of the previous iteration. This makes testing and further development much simpler as rails uses a MVC pattern which makes the product easier to scale. Additionally, Ruby on Rails provides better performance compared to React when it comes to server-side rendering. With Rails, the entire HTML page can be generated on the server, reducing the amount of JavaScript that needs to be loaded in the browser and improving the initial load time for users. There is likely some dormant code that remains, but through deleting the excess react, the testing coverage has significantly increased which results in a more stable product.

Code Evaluation:

Cucumber:
- 38 scenarios (1 undefined, 6 pending, 31 passed)
- 209 steps (9 skipped, 2 undefined, 6 pending, 192 passed)
- 211 / 237 LOC (89.03%) covered.

Rspec:
- 649 / 676 LOC (96.01%) covered.
- 57 examples, 0 failures

SimpleCov:

All Files (90.49% covered at 24.58 hits/line)

47 files in total.
862 relevant lines, 780 lines covered and 82 lines missed. (90.49%)

Search:

File	% covered	Lines	Relevant Lines	Lines covered	Lines missed	Avg. Hits / Line
app/helpers/admin_helper.rb	33.33 %	10	6	2	4	0.33
app/controllers/users_controller.rb	36.59 %	82	41	15	26	0.44
app/controllers/practice_videos_controller.rb	38.89 %	75	36	14	22	0.47
app/controllers/application_controller.rb	65.22 %	41	23	15	8	2.22
app/models/practice_video.rb	69.23 %	29	13	9	4	0.69
test/models/user_test.rb	84.62 %	41	26	22	4	0.85
test/controllers/practice_videos_controller_test.rb	92.59 %	52	27	25	2	1.15
test/controllers/users_controller_test.rb	92.59 %	50	27	25	2	1.15
app/controllers/api/v1/analyses_controller.rb	94.59 %	87	37	35	2	3.05
test/controllers/api/v1/analyses_controller_test.rb	96.64 %	194	119	115	4	1.10
app/controllers/api/v1/videos_controller.rb	96.97 %	73	33	32	1	2.85
test/controllers/api/v1/visions_controller_test.rb	97.33 %	132	75	73	2	1.16
test/controllers/api/v1/videos_controller_test.rb	98.46 %	113	65	64	1	2.38
app/controllers/api/v1/visions_controller.rb	100.00 %	59	26	26	0	2.73
app/helpers/api/v1/analyses_helper.rb	100.00 %	8	3	3	0	1.00
app/helpers/api/v1/visions_helper.rb	100.00 %	8	3	3	0	1.00
app/helpers/application_helper.rb	100.00 %	4	1	1	0	1.00
app/helpers/login_helper.rb	100.00 %	4	1	1	0	1.00
app/helpers/pages_helper.rb	100.00 %	4	1	1	0	1.00
app/helpers/players_helper.rb	100.00 %	2	1	1	0	1.00
app/helpers/practice_videos_helper.rb	100.00 %	2	1	1	0	1.00

Rubycritic:

RUBYCRITIC

Overview

| Overview |
| Code |
| Smells |
| Coverage |

82.53 /100

■ A ■ B ■ C ■ D ■ F

Highcharts.com

Churn vs Complexity

Highcharts.com

Summary

A	70 files	596 churns	94 smells
B	6 files	46 churns	30 smells
C	9 files	108 churns	19 smells
D	13 files	79 churns	84 smells
F	3 files	19 churns	30 smells