<u>**Retrospective Report**</u>

<u>**CSCE 120 Autograder Frontend**</u>

Scrum Master: Md Hasan Al Banna
Product Owner: Max Smith

<u>**Links**</u>
- [GitHub](#)
- [Slack Invite](#)
- [GitHub Projects](#)
- [Heroku Deployment](#)

<u>**Dates of the Sprint**</u>
Planning: 20 September to 27 September
Development: 28 September to 7 October

<u>**Information about team and member contributions**</u>

Members	<u>**Assigned Tasks**</u>	<u>**Points**</u>	<u>**Percentage of Development**</u>
Mainak Sarkar	Create a Login System for the App, Create a New Assignment	4+3 = 7	18.42%
Riddhi Ghate	Create a Login System for the App, Create an Access Control Management System	4+3 = 7	18.42%
Max Smith (Product Owner)	Create a New Assignment, Regenerate .tests file on test creation/update/deletion	3+2 = 5	13.15%
Md Hasan Al Banna	Add a Test to an Assignment, Review Tests in an Assignment, Update test for an assignment, Export assignment as .zip	1+1+1+2 = 5	13.15%
Qinyao Hou	Add a Test to an Assignment, Review Tests in an Assignment, Update test for an assignment, Export assignment as .zip	1+1+1+2 = 5	13.15%

| Ryan Gonzalez | Regenerate .tests file on test creation/update/deletion, Update remote assignment repository | 2+2 = 4 | 10.52% |
| Saksham Mehta | Create an Access Control Management System, Update remote assignment repository | 3+2 = 5 | 13.15% |

Sprint Goal:
As evident from the listed user stories in this sprint plan, all of which we aim to complete in the first sprint, our current objective is to develop the foundational aspects of the autograder application.

Broadly speaking, our tasks can be proliferated as follows:
- Creating the Login System for our App.
- Creating the Access Control Management System that allows the admin(s) of the organization to add new users and manage their level of access.
- Creating the home page of the App, that includes the link to the form for creating new assignments.
- Developing the CRUD operations for tests within individual assignments, and updating the "code.tests" specification file accordingly, depending on the operation performed.
- Enabling the functionality that allows the assignment to be exported for upload to the autograder in Gradescope.

The above tasks have been accrued on the basis of our initial interactions with the client, and we feel that the completion of the above tasks, along with continual guidance and feedback from the client will pave the way for us to dive deep and refine the app's features even further, thereby leading to the final product meeting the expectations of the stakeholders.

Sprint Achievements and Summary of Stories implemented in this Sprint:

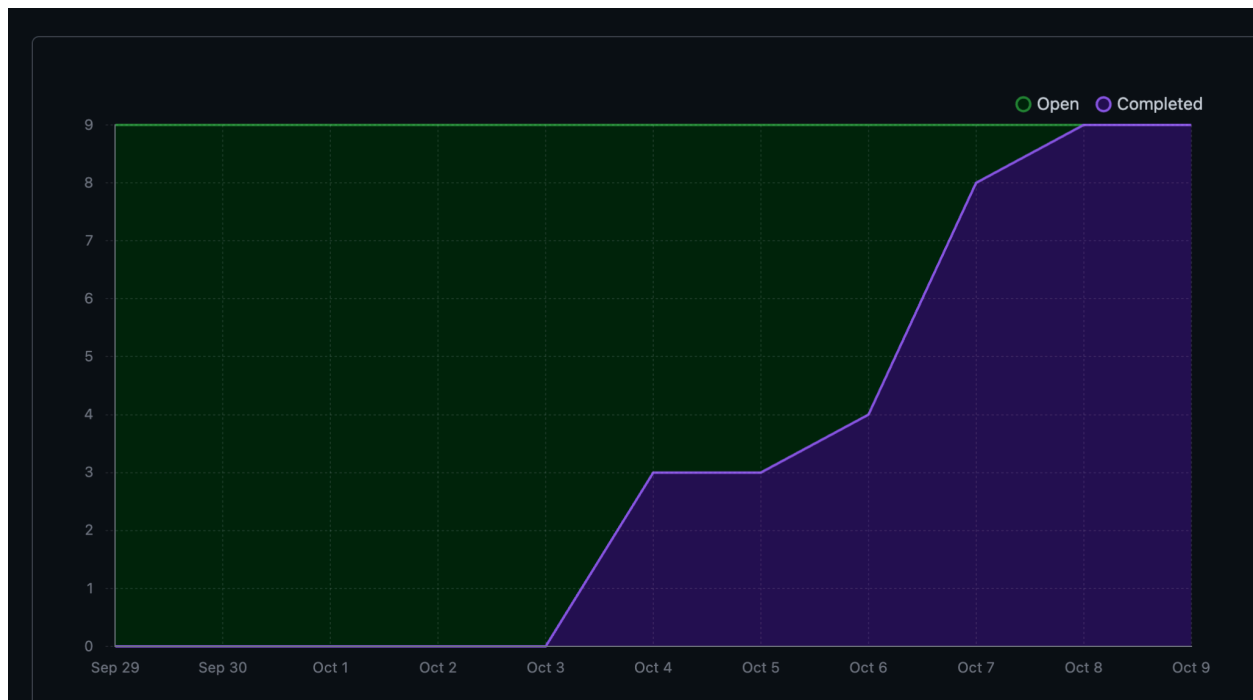| Story Index	Task	Completion
1	Create a Login System for the App	Completed
2	Create an Access Control Management System	Completed (To be continued in the next sprint)

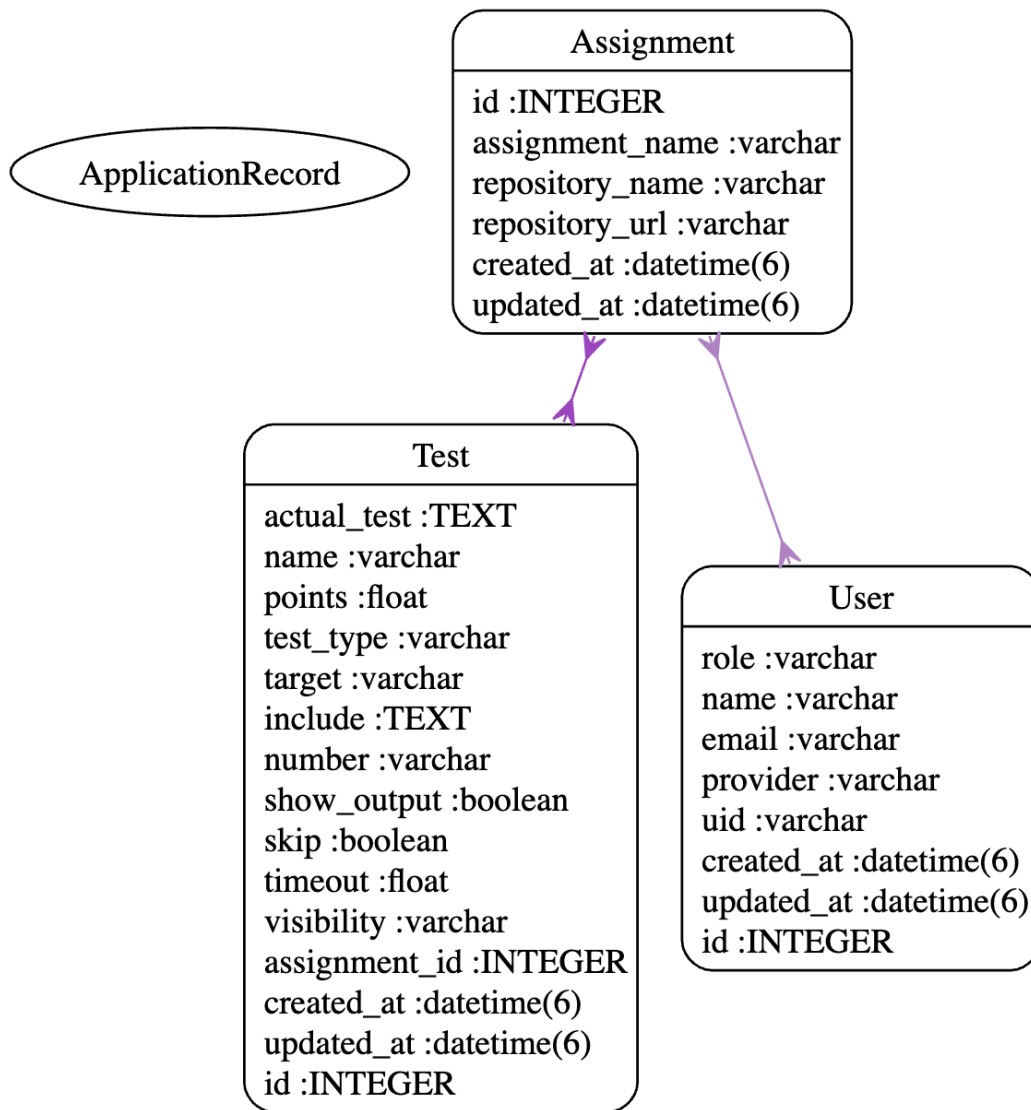| 3	Create a New Assignment	Completed
4	Add a Test to an Assignment	Completed
5	Review Tests in an Assignment	Completed
6	Update test for an assignment	Completed
7	Regenerate .tests file on test creation/update/deletion	Bugs to be fixed in the next sprint
8	Update remote assignment repository	Completed
9	Export assignment as .zip	Completed

Sprint backlog items and status:

	Task	Explanation
	Create a Login System for the App	
	Create an Access Control Management System	
	Create a New Assignment	
	Add a Test to an Assignment	
	Review Tests in an Assignment	
	Update test for an assignment	
	Regenerate .tests file on test creation/update/deletion	
	Update remote assignment repository	
	Export assignment as .zip	
+	Deletion of Tests in an Assignment	Completed along with other CRUD operations
+ / *	Addition of user to GitHub organization	Added after sprint

	through UI	planning, then removed due to time constraints
+ / *	Removal of user from GitHub organization through UI	Added after sprint planning, then removed due to time constraints

Burndown Chart:

Design diagrams:

ApplicationRecord

Assignment

id :INTEGER
assignment_name :varchar
repository_name :varchar
repository_url :varchar
created_at :datetime(6)
updated_at :datetime(6)

Test

actual_test :TEXT
name :varchar
points :float
test_type :varchar
target :varchar
include :TEXT
number :varchar
show_output :boolean
skip :boolean
timeout :float
visibility :varchar
assignment_id :INTEGER
created_at :datetime(6)
updated_at :datetime(6)
id :INTEGER

User

role :varchar
name :varchar
email :varchar
provider :varchar
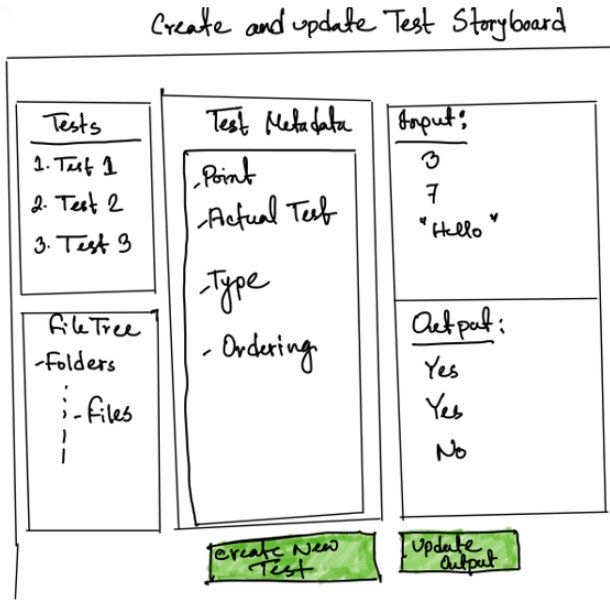uid :varchar
created_at :datetime(6)
updated_at :datetime(6)
id :INTEGER

Documentation of changes

We have updated our working agreement. We had some design change for one of the stories which is given below:

Updated View:

Evaluations of code and test quality: CodeClimate gives us an 'A' with 5 code smells. There was no violation on rubocop.
Coverage: 90.88%

Customer Meeting
Meeting: 07 October 2024, Monday from 02:30 pm,on Zoom Platform

- Misc. feedback
    - Client wants addition of a navigation bar
    - Client wants more descriptive button names
    - Client wants more consistent styling
- Course overview page feedback
    - Client wants search bar/sorted tabular entries for course assignments
    - Client wants removal of redundant data (Assignment Name)
    - Client wants ability to regenerate assignment deploy keys (and possibly clear existing)
- Test creation feedback
    - Client wants ability to create groupings of tests
    - Client wants more organized view/formatting for creating/modifying tests
    - Client wants insertion of whitespace in between code.tests entries
- Access control feedback
    - Client wants "matrix" of checkboxes to select specific permissions
    - Client wants both an assignment view (in which all users' access can be modified for a specific assignment), as well as the existing user view (which allows an administrator to modify all of a user's permissions for all assignments)

bdd & tdd
Both tdd and bdd were followed for this sprint using cucumber scenarios and rspec tests.

Summary:
Positives from this sprint:
1. All the cucumber scenarios were written beforehand.
2. There was a developed prototype.
3. Everybody had a positive frame of mind to help each other.
4. Designed with future design in mind.

Improvements to be done:
1. Better communication and contribution in the meetings.
2. Plan based on the capability of the team mates.
3. Adherence to Git best practices.
4. More effective testing.
5. Better conflict resolution.

6. More accountability.
7. Stories should be created keeping interdependencies in mind.