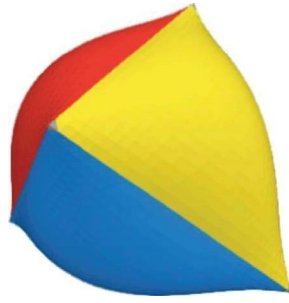


OLEI Survey



Final Report

Team ELRC

Department of Computer Science and Engineering
Texas A&M University

December 2024

Table of Contents

Table of Contents	3
1. Summary	4
2. User Stories	5
2.2 Lo Fi Diagrams	10
3. Updates for legacy code	14
4. Sprint Roles	14
5. Sprint summaries	15
6. Point breakdown	16
7. Customer breakdown	16
8. BDD / TDD process	20
9. Config Management	21
10. Production	23
11. Tools	24
13. Deployment Steps	25
14. Links	26
15. Links to the presentation and the demo	26

1. Summary

This is a legacy project, which aims to bring the **Synergistic Leadership Theory**, a modern take of leadership theory, into a practical and interactive web application. The mission is to develop a platform that not only educates on the theory, but also allows users to access leadership effectiveness through the **Organization and Leadership Effectiveness Inventory** (OLEI). The stakeholders include **Dr. Irby** and **Ms. Minoo Mohammadi**, alongside a team of students committed to transcending traditional leadership paradigms. The platform will feature an animated, interactive tetrahedron model representing the theory's core factors: *Leadership Behaviors, Organizational Structure, External Forces, and Attitudes, values, and beliefs*. This tool will not only facilitate a deeper understanding of leadership dynamics, but also analyze general personalized leadership style analysis.

We have revamped the legacy code by adding new features suggested by the client. One of the core features was implementing an admin dashboard and an admin login feature. The second core feature was providing an option to download the survey responses as a csv file, which the client did not have access to previously. The third core feature was to extend the existing three user roles to seven user roles when the user is signing up to the website. The fourth core feature was ensuring that the user responds to all the questions of the survey before submitting it. This feature was not implemented previously, and this feature is paramount for the tetrahedron to get assigned correct values.

Customer

Dr. Beverly Irby Ms. Minoo Mohammadi

Developers

Vinayaka Hegde Chih-Chuan Hsu

Manoj Peta Kunal S

Manoj Gurram Sai Aakarsh

Sai Nithin

2. User Stories and lo-fi UI mockups

2.1. User stories

Story	Points	Status
Understand the legacy code by reproducing it (Kunal S, Chih-Chuan) [As a] legacy code development team, [So that] we can start developing the leadership style analysis tool efficiently, [I want to] set up our development environment with all necessary tools and dependencies.	1	Done
Responsibilities of Product Owner (Sai Aakarsh) [As a] product owner, [So that] I can setup client meetings and update the weekly progress, [I want to] Ensure I deliver updates to the client, and inform developers about the client feedback every week.	2	Done
Responsibilities of Scrum master (Manoj Peta) [As a] Scrum master, [So that] I have clear visual direction of the features to be implemented [I want to] Split and assign tasks between the developers, and take care of sprint backlog	2	Done
Administrator Login (Vinayaka Hegde, Manoj Peta) [As a] admin user, [So that] I can securely access the system using separate login page with privileged access, [I want to] Login to the system using a dedicated login page, and view all surveys taken by all the users	2	Done
Survey Page (Vinayaka Hegde, Manoj Peta, Sai Aakarsh) [As a] administrator user, [So that] From the admin dashboard once the user clicks a survey result, it must redirect to the survey page [I want to] see the details of each survey submitted by users across all the roles	3	Done

Admin Management Display page (Kunal S) [As a] administrator, [So that] I have the access to user survey data and the admin tasks from a single page, [I want to] access the management dashboard so that I can view and manage user data and administrative tasks from a central location.	3	Done
Setup and deploy on Heroku (Manoj Peta) [As a] developer, [So that] Both the users and admins can access the new application [I want to] Ensure that the new application is accessible remotely and globally	1	Done
Basic Tetrahedron Model (Chih-Chuan) [As a] user, [So that] Once a user opens the survey results, is must generate 2 tetrahedron models with each face corresponding to different parameter, [I want to] be able to see the tetrahedron model based on the results of the survey	3	Done
Test the legacy code by participating in the survey (Kunal S) [As a] developer [So that] all of the developers have a clear understanding of the existing features of the legacy code [I want to] Analyze the features if they meet the client expectations	1	Done
Admin Management Display page (Kunal S) [As a] administrator, [So that] I have the access to user survey data and the admin tasks from a single page, [I want to] access a management dashboard so that I can view and manage user data and administrative tasks from a central location.	3	Done

Solve the bug of delete response after submitting the survey (Chih-Chuan) [As a] developer [So that] I can evaluate the existing cucumber scenarios for correctness, and make sure all tests pass with 100% accuracy [I want to] solve the bug that throws an error after submitting the survey responses	1	Done
Provide data access to the client (Sai Aakarsh) [As a] admin user, [So that] the client has sufficient access to the survey results and the database [I want to] be able to download a csv file which contains the responses of the data submitted by all the users	3	Done
Create the tetrahedron model and display it (Manoj Gurram, Chih-Chuan) [As a] user, [So that] the tetrahedron model is displayed when the user navigates to the survey results page [I want to] Create a model that accounts for all 4 factors provided by the user during the survey	3	Done
Modify the heading to College of Education and Human Development (Sai Nithin) [As a] developer, [So that] end users can understand the name of the survey website correctly [I want to] modify the name from College of Education to “The college of education and human development”	1	Done
Display page for all the surveys in pre-defined for admin. (Sai Aakarsh) [As a] admin, [So that] When I am Admin dashboard page, the survey page must display all responses along with names and roles. [I want to] See a button to redirect to all surveys	2	Done

Admin Login feature - Production version (Kunal S, Manoj Gurram) [As a] admin, [So that] the admin login feature is tested and deployed [I want to] access the page with special privileges for admin	2	Done
Product owner responsibilities (Manoj Peta) [As a] product owner for the current sprint, [So that] I can carry out the product owner responsibilities for the current sprint [I want to] Discuss with the scrum master and break down the tasks and get the feedback from the client	2	Done
Handle empty responses in surveys. (Vinayak Hegde) [As a] developer [So that] No page is left empty, and all the survey pages are filled before hitting the submit button [I want to] ensure that users can click submit only after filling all the pages	3	Done
Integration of Admin login with Admin Dashboard. (Kunal S) [As a] developer, [So that] the access to admin dashboard as per admin access and login is implemented [I want to] integrate the admin login with admin dashboard	2	Done
Update the display and function of tetrahedron (Chih-Chuan) [As a] developer [So that] the tetrahedron displays correct survey responses for own and other surveys [I want to] Update the tetrahedron according to the new requirements of the client discussed in the latest sprint	1	Done

Implement a button to download the complete survey response into excel. (Sai Nithin) [As a] developer [So that] the users can download the complete response of the survey [I want to] Add a download button, on click of which downloads all the responses in the excel file	2	Done
Survey page Mode value for survey responses (Manoj Gurram) [As a] developer [So that] while accessing the survey, the admin must be able to view average of the total surveys against a person [I want to] implement the survey page mode value	3	Done
Survey page mode value fix for tetrahedron (Chih-Chuan) [As a] developer [So that] While accessing the survey, the admin must be able to view the average value of the total survey responses against a person [I want to] display and implement the survey page model value for tetrahedron	2	Done
Write the document to tell professor about the admin assignment (Manoj Gurram, Manoj Peta) [As a] Developer [So that] The client is aware about the admin assignment component [I want to] Write a document to discuss with the client about the future access and management of the database, and help them deploy the current application	2	Done
Admin permission modification (Sai Aakarsh) [As a] user, [So that] If I am not an admin and go directly to the path /admin, I should not be able to perform the admin permission tasks [I want to] ensure that the admin features are only restricted to the admin user	1	Done
Output Excel with two other columns (Sai Aakarsh) [As a] developer [So that] the two columns: Supervisor's responses and Supervisee's responses are viewable [I want to] View the additional two columns	1	Done

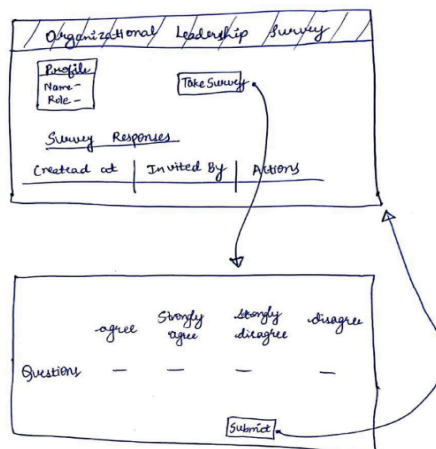
Output Excel with two other columns (Sai Nithin Reddy) [As a] developer [So that] For the welcome page and the about pages, the users can see the page as needed by the client [I want to] modify the welcome and the about pages	2	Done
Wrote the test supervisor's and supervisee's response (Chih-Chuan) [As a] developer [So that] I can evaluate if the new feature of adding these two new fields are working as expected [I want to] rewrite the tests for the supervisors and supervisee's responses	1	Done
Expand principal to 7 evaluation middle positions (Vinayaka Hegde, Kunal S) [As a] developer [So that] All 7 options are displayed while signing up as expected by the client [I want to] expand the existing 3 positions to add 4 more positions	2	Done

2.2 Lo-Fi Diagrams


Following sections indicate the Lo-Fi diagrams and their corresponding screenshots for the four core features of the application:

1. User navigating the survey pages, and filling out the details, and finally submitting the survey by clicking the submit button.

a. lo-fi UI wireframe:



b. Corresponding screenshot:



School of Education & Development
EDUCATION LEADERSHIP RESEARCH CENTER

[Home](#) [About](#) [Logout](#)

ORGANIZATIONAL AND LEADERSHIP EFFECTIVENESS INVENTORY - Principal

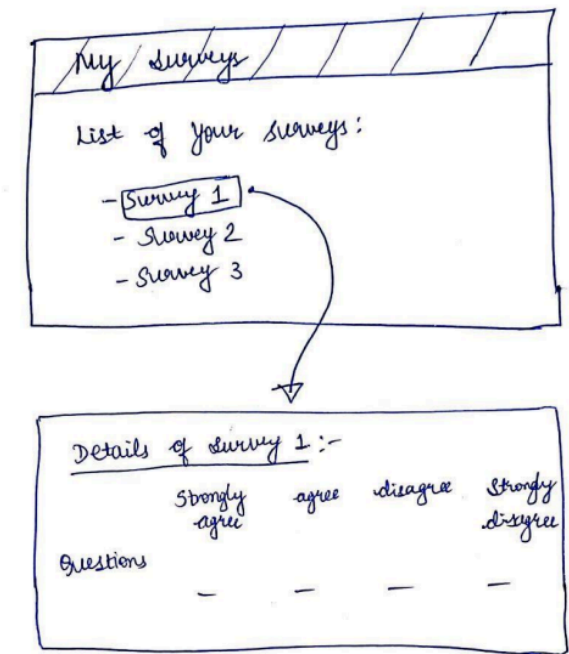
Part 1: Leadership Behavior - Management

To what extent do you agree the following behaviors reflect your personal leadership behaviors?

	Strongly Disagree	Disagree	Agree	Strongly Agree
1. Leads by example	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2. Ability to "juggle" many things in professional and personal life	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
3. Communicator	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
4. Lifelong Learner	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
5. High expectations of self and others	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
6. Strong academic self-concept	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
7. Motivational	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
8. Communicates vision	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
9. "Can do" philosophy (resourceful)	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
10. Persistent	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

2. A user must be able to view all their surveys by going to the home page

a. lo-fi UI wireframe



b. Corresponding screenshot:

ORGANIZATIONAL AND LEADERSHIP EFFECTIVENESS INVENTORY

Welcome vinuvinayaka2011100@gmail.com

Profile

FIRST NAME	Test
LAST NAME	user
CAMPUS	Brazos
DISTRICT	Bryan
ROLE	Principal

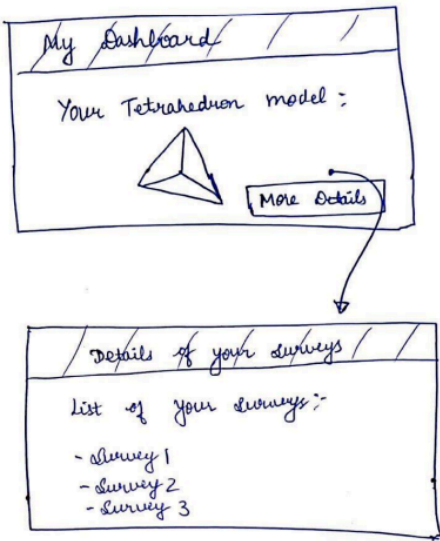
Survey Responses

Created at	Invited by	Invite Token	Actions
2024-11-27 02:01:27 UTC	N/A	N/A	Show

Take Survey

3. The tetrahedron model must be displayed as expected

a. low-fi UI wireframe



b. Corresponding screenshot:

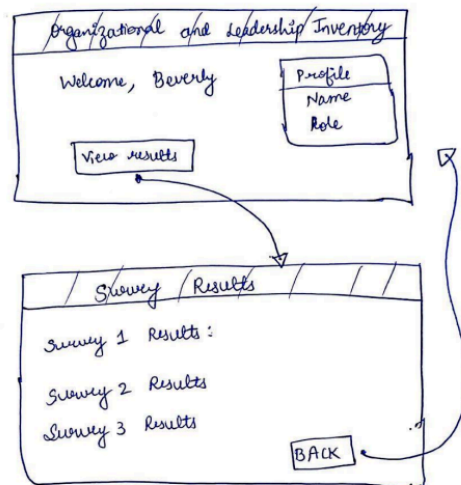
The Tetrahedron Model



The distinct feature of the Synergistic Leadership Theory is its depiction through a tetrahedron model, offering a visual framework to analyze the theory's four foundational elements and their interconnections. The tetrahedron model visually represents the SLT's four factors as vertices, emphasizing their equal importance and interactivity. The model's invariance, meaning it retains its shape regardless of which vertex is the base, symbolizes the non-hierarchical nature of the factors. The six edges represent the non-linear relationships between the factors, inferring dynamic interactions rather than linear cause-effect relationships. This model underscores the complex, interconnected nature of leadership within an organization, highlighting that changes in one factor can significantly influence the others.

4. The admin (Dr Irby) must have access to all the results of the survey, and there must be a provision for the admin to download the survey results in a csv file

a. low-fi UI wireframe:



b. Corresponding screenshot:

Profile

FIRST NAME	Dr Beverly
LAST NAME	Irby
CAMPUS	Brazos
DISTRICT	Bryan
ROLE	Supervisor

Survey Responses

Created at	Invited by	Invite Token	Actions
2024-11-27 02:01:27 UTC	N/A	N/A	Show
2024-11-27 02:06:05 UTC	N/A	N/A	Show
2024-11-27 02:06:10 UTC	N/A	N/A	Show
2024-11-27 02:06:14 UTC	N/A	N/A	Show
2024-11-27 02:06:18 UTC	N/A	N/A	Show

3. Updates for Legacy code

Since this was a legacy project, a large amount of time of the first Sprint was spent in setting up the github repo in the developer's local environment, followed by understanding the legacy codebase, and testing the complete code end to end to ensure that all the features of the legacy code worked as expected.

Understanding the existing code: Understanding the existing codebase was very easy, thanks to the previous team who documented all their efforts meticulously. Since the codebase was very large, we split up the entire codebase amongst all the 5 developers in the first sprint, and every developer explained in detail what they garnered from their part of the codebase.

Refactoring/Modification performed on legacy: The OAuth component was re-written since we did not have access to the OAuth credentials of the previous team, we had to create a new OAuth application, and configure the new application’s credentials on our local Ruby code. The client requested for a bunch of minor changes in the Welcome page, About page, etc. These changes were handled, and the code was appropriately modified. One major refactoring that the legacy code went through was in ensuring that all the survey responses are completely filled before the user submits the survey, and this was not taken care of by the previous team. Apart from this, the client did not require any changes in the existing codebase.

Therefore, all the existing cucumber scenarios and Rspecs were intact since we built on top of that. Therefore, we preserved the testing suite for the existing codebase, and added new cucumber and Rspec tests for the new features implemented by us.

4. Sprint Roles

	Product Owner	Scrum Master
Sprint 1	Vinayaka Hegde	Sai Nithin Reddy

Sprint 2	Manoj Reddy Gurram	Sai Aakarsh Padma
Sprint 3	Manoj Kumar Reddy Peta	Kunal SomendraSingh
Sprint 4	Manoj Reddy Gurram	Chih-Chuan Hsu

5. Sprint Summaries

5.1 : Sprint 1

(3 stories | 5 points)

This sprint was to establish the groundwork for the project. This included setting up the development environment, cloning the github repo, learning about the Synergistic Leadership Theory by meeting Dr. Irby, and understanding about the Organization and Leadership Effectiveness Inventory. Once we understood about the project in depth, we created multiple stories to understand and familiarize ourselves with the code. To achieve this, we first cloned the repo, followed by delving deep into the ruby code, followed by understanding the cucumber test cases and the Rspecs. One major problem we faced initially was that the master key was not made available by the previous team due to security reasons. However, the previous team was kind enough to provide the master key, post which we were able to deploy the code to heroku. Finally, we were up to speed with the code, the data models, and the test cases, post which we started the development.

5.2 : Sprint 2

(10 stories | 20 points)

In this sprint, the client asked us to implement the Admin Login feature, followed by a Survey page. After implementing both of these features, we set up the application and deployed it on heroku. Post which we implemented the admin management display page. After implementing all these features, Chih-Chuan started with implementing a basic tetrahedron model. During this sprint, many developers found multiple bugs in the existing legacy code. For instance, Vinayaka found a bug where an error popped up on click of the delete response page after completing any survey. Finally, the first week of the sprint ended with providing database access to the client, which satisfied the client since they were very eager to check out the database since the first week of the project.

5.3: Sprint 3

(10 stories | 21 points)

This sprint involved lots of coding, and it contributed to a large amount of the project. This is because there were not many roadblocks or dependencies since we had access to the data from the client, and all the requirements were understood by all of the developers. Therefore, this enabled the developers to be productive. Hence, this sprint was a very crucial one in the whole project, since it contributed the most number of points to the final project.

5.4: Sprint 4

(6 stories | 10 points)

The final sprint started off with us drafting a document to inform Dr Irby about the admin assignment. We also walked her through the process of modifying the admin permission. In this sprint, Sai Aakarsh implemented a feature that downloads an excel with multiple columns. In this sprint, Dr Irby asked us to modify the welcome page and the about page to add more details, which we were able to accomplish. Finally, in the last week, when all the other features were completed, Dr. Irby asked us to add 4 additional roles in the dropdown, while signing up, and we implemented this last feature, which wrapped up the project.

6. Point Breakdown

Assignee	Sprint 1	Sprint 2	Sprint 3	Sprint 4	Total
Vinayaka Hegde	1	5	3	1	10
Chih-Chun Hsu	1	3	3	3	10
Kunal S	2	3	5	2	12
Manoj Gurram	1	4	3	2	10
Manoj Peta	1	3	4	2	10
Sai Aakarsh Padma	3	2	2	3	10
Sai Nithin	2	3	2	4	11

7. Customer Breakdown

7.1: Harrington Tower, 2024-10-07 5:30 PM

Summary

- Began meeting with introductions from team and the client, Dr Beverly Irby
- Firstly, Dr Irby introduced us with the origin of the Synergistic Leadership Theory, post which she explained about the Organizational Leadership Effectiveness Inventory
- She introduced the 4 factors that constituted the tetrahedron model - Beliefs and values, Leadership behaviours, External factors, and organizational structure.
- Essentially, a supervisor considers all these elements while evaluating their set of teachers.

- Out of our curiosity, we asked what the other projects that ELRC does, and Dr Irby described completely the various kinds of projects the Educational Leadership Research center is currently working on.
- Then, she explained about the website, that contains the tetrahedron model, the surveys that analyze the leadership styles of a person

Feedback

- The client was very excited interacting with us, and held a positive outlook for the future of our partnership.

7.2: Harrington Tower, 2024-10-20 1:30 PM

Summary

- We met Minoo Mohammadi, a PhD student under the guidance of Dr Irby, who would be the primary contact of our project.
- We explained to her the lo Fi user sketches, the storyboards, and the website's framework. Minoo mentioned that she would be sharing the data of the list of supervisors and the corresponding teachers in a csv format, since this mapping was essential for us to get started with developing the features.
- We primarily discussed the following points throughout the meeting:
- **Database Model:**
 - A person (principal/superintendent, or candidate for the job) takes a survey, this will create a case number.
 - Other evaluators (school board, administrative team, teachers) can use the same case number to fill out surveys evaluating this person.
 - A person can take surveys at different times to reveal the changes in their leadership style.
- **Survey Types:**
 - A person can take a survey reflecting their own leadership style and estimation of their supervisors, organization, and external community.
 - People from the school board (supervisors), the organization, and the external community can take survey reflecting their opinion on the leadership style
- **Survey Analysis:**
 - The survey consists of 5 parts, the first part will be compared with the other four parts to give a matching score on the four aspects of the synergistic leadership theory.
 - Questions are correlated across different parts. For example, questions 1-3 in part 1 and question 1 in part 5 all ask about the openness to changes. If the answers in part 1 and part 5 both tend to agree on the openness to changes, then the matching score for the Values, Attitudes, and Beliefs aspect increases.

- **Visualization:**

- The four aspects of the synergistic leadership theory correspond to the four faces of a tetrahedron. If the matching score in an aspect is high, its corresponding face is flat; otherwise, the face will bulge out (being convexly curved).

Feedback

Dr Irby and Manoo were extremely content with our Lo-Fi User Interface Sketches, Storyboards, and the website's framework. They are willing to share the csv data of the mapping and some additional materials for the website's content.

7.3: Online, 2024-10-21 5:30 PM

Summary

- The client expressed a desire to add an admin login page and an admin dashboard. They also indicated the need of a survey page and a basic tetrahedron model.
- It was also mentioned that they need the database access using which they could view the results of all the data of the surveys.
- The client expressed a desire to query survey information related to their respective teachers or subordinates.
- At the current state, a user could submit the survey without completing all the survey questions. The client desired to change this behavior by enforcing a check that all the questions were answered prior to submitting the survey. This makes sure that incorrect results are not submitted, and the tetrahedron model is unaffected.

Feedback

The client was extremely content about our development progress and the communication experience with us. They understood our queries, and were very helpful to address all our concerns.

7.4: Online, 2024-10-28 6:00 PM

Summary

- The client wanted a display page for all the surveys in pre-defined form for the admin user. Additionally, they had a requirement that all the survey pages must be filled completely before submitting the survey.
- The client also inquired about the plan for the next sprint. The product owner outlined the remaining features to be implemented and any anticipated challenges.
- Set the expectation for the entire project to be wrapped up in the next two sprints,

followed by a round of user testing.

- Discussed around the updated website and some of the challenges encountered in the previous sprint. Also, the developers walked through the changes made to make sure that the application's architecture is clearly understood and maintainable.

Feedback

The client expressed satisfaction with the progress made during sprint 2 and the team's responsiveness to feedback. They reiterated their enthusiasm for the project and willingness to provide any additional materials or input needed to support the next phase of development.

7.5 : Online, 2024-11-03 12:30 PM

Summary

- Discussed realistic expectations and directions for the project deliverable at the end of the semester.
- The developer team discussed more about the features implemented in the previous sprint. The most important one being, the integration of the admin dashboard and the admin login.
- One of the main features was also to update the display and function of the tetrahedron, which was discussed in great detail during the meeting.

Feedback

The clients were satisfied with the progress made in sprint 3, and were understanding of the end-of-semester timings in order to get the project completed on time.

7.6: Online, 2024-11-16 1:30 PM

Summary

- The client expresses a need for a download button to download the survey response into an excel format. The client also mentioned that the admin must be able to view the average value of the total survey responses against a person, so that when comparing the results of self versus that of the others, the user can view the average value displayed on the page.
- The client also expected a feature wherein if the user is not an admin, and goes directly to the path `base_path/admin`, he must be able to see the admin dashboard and perform admin actions.
- In this meeting, we also got an opportunity to discuss and present the documentation that we had been working on in the previous sprint. The documentation involved details on future access and management of the database, and how to help the client deploy the current application.

- The most important feature layed out in the next sprint was to add 5 additional fields in addition with the existing principal and the supervisor. Therefore, a total of 7 fields would be created.

7.7: Online, 2024-11-21 1:30 PM

Summary

- The client expected 2 additional columns for the output excel file - The supervisor's response, and the supervisee's response, which was implemented.
- Also, the client requested for some minor changes in the about and welcome page. In the about page, the client wanted the text "About the SLT and OLEI", which initially consisted of "About the SLT"
- Moreover, in the welcome page, the client required "Welcome to the OLEI survey", which initially had only the text "Welcome".
- One major concern is that the platform is currently hosted on heroku, and the client needs to get access to the platform. Therefore, we would be having a discussion with the IT team of ELRC to hand over the details of the platform and hosting of the code.

Feedback

In the final meeting, the client requested access to the hosting platform, and the developer team agreed to meet the IT team of ELRC, hand over the access and explain to them in detail how to set up and host the platform on heroku.

8. Process

8.1: BDD/TDD Process

Requirement Analysis: We began with gathering and analyzing requirements. This involved close collaboration with stakeholders to ensure that the user stories accurately captured the desired functionalities.

8.1.1 : Test-Driven Development (TDD):

1. Write Tests First: For each functionality, we started by writing RSpec tests before writing any code. These tests were designed to fail initially.

8.1.2 : Behavior-Driven Development (BDD):

2. Feature Files: We used Cucumber to write feature files, which describe the behavior of the application from the user's perspective.
3. Scenarios: Each feature consisted of multiple scenarios, written in plain language that non-technical stakeholders could understand.
4. Step Definitions: These scenarios were then linked to step definitions, which are Ruby

blocks that execute the scenarios.

8.1.3: Integration:

We integrated TDD and BDD by ensuring that:

5. Unit Tests (from TDD) covered the technical aspects.
6. Feature Tests (from BDD) covered the user interactions and workflows.

8.1.4: Continuous Integration (CI):

We used GitHub Actions to run tests on the application. Only once all the tests pass can the code be deployed.

8.2: Benefits

1. **Improved Communication:** BDD helped improve communication between developers, testers, and non-technical stakeholders by using plain language to describe product features.
2. **Higher Quality Code:** TDD led to more robust, error-free code, as it required developers to thoroughly test each piece of functionality before it was integrated.
3. **Early Bug Detection:** Both methodologies helped in detecting issues early in the development cycle, reducing the cost and effort of fixing bugs in later stages.

8.3: Problems

- **Initial Learning Curve:** The team needed time to adapt to the rigor of writing tests before actual code and learning BDD tools and syntax.
- **Time Investment:** Initially, writing tests, especially feature files for BDD, required more time than traditional development methods.
- **Maintenance:** Maintaining a large suite of tests could become cumbersome as the application grows and changes, potentially slowing down the development process.

8.4: Conclusion

Integrating TDD and BDD in a Ruby on Rails project significantly enhanced our development process by fostering better communication and producing higher-quality code. While the initial investment in terms of time and learning was significant, the long-term benefits in terms of reduced bugs and better collaboration were well worth the effort.

9. Config Management

We did not officially have any spikes during any of our sprints. Our high level of communication and familiarity with the framework covered most of the technical issues we encountered.

9.1: Branches & Pull Requests:

We have a total of 26 merged pull requests, each corresponding to a unique feature branch that was created to modify the codebase or add a new feature.

tamu-edu-students / csce606-ELRC-OLEI_Project

Q Type [Z] to search

+ | | | |

code

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

Filters

Q is:pr is:closed

Labels 9

Milestones 0

New pull request

Clear current search query, filters, and sorts

☐

0 Open

✓ 26 Closed

Author

Label

Projects

Milestones

Reviews

Assignee

Sort

☐

updated the views file ✓

#26 by Vinayaka2k was merged 2 days ago

☐

Revert "Revert "Additional position in profile change"" ✗

#25 by kunal-singh-tamu was merged 3 days ago

☐

Minor change to re merge this branch to main.

#24 by kunal-singh-tamu was merged 2 days ago

☐

Revert "Additional position in profile change" ✓

#23 by kunal-singh-tamu was merged 4 days ago

☐

Additional position in profile change ✗

#22 by kunal-singh-tamu was merged 4 days ago

☐

finish test for survey_response and survey_answer ✓

#21 by LeoHsuProgrammingLab was merged 4 days ago

We had only 1 release to ensure that there is a single point of exit to the code.

Final release

Latest

 Vinayaka2k released this now · 45 commits to main since this release  Sprint4  9234062 

What's Changed

- Added a page with path /admin for admin dashboard by [@saiaakarsh-tamu](#) in [#1](#)
- Dev/leo by [@LeoHsuProgrammingLab](#) in [#2](#)
- Dev/leo by [@LeoHsuProgrammingLab](#) in [#3](#)
- Admin management display page by [@kunal-singh-tamu](#) in [#4](#)
- Revert "Admin management display page" by [@LeoHsuProgrammingLab](#) in [#5](#)
- Revert "Revert "Admin management display page"" by [@LeoHsuProgrammingLab](#) in [#7](#)
- Dev/leo by [@LeoHsuProgrammingLab](#) in [#10](#)
- Modified admin_dashboard rspec to restrict user access to admin dashb... by [@kunal-singh-tamu](#) in [#11](#)
- scaling for the distortion of tetrahedron by [@LeoHsuProgrammingLab](#) in [#12](#)
- Added fields first_name, last_name and role for admin dashboard by [@saiaakarsh-tamu](#) in [#14](#)
- Excel Download by [@SaiNithin001](#) in [#15](#)
- Checking whther a user is admin first instead of his user id for surv... by [@saiaakarsh-tamu](#) in [#16](#)
- ensure that all the survey questions are filled by [@Vinayaka2k](#) in [#13](#)

Full Changelog: https://github.com/tamu-edu-students/csce606-ELRC-OLEI_Project/commits/Sprint4

Details on the branches:

We had 16 branches, including the main. For every main feature, we created a series of commits in a new branch, and then squashed all those commits to create a PR, which was merged with the main once the Product Owner reviewed the code. Creating multiple branches ensures that each developer can work on their features in isolation, and then push the code to the remote branch, and raise a PR so that the code is merged with the main branch. This ensured that there would be no merge conflicts. Moreover, we deleted the branches as soon as the PR was merged with the main.

main	yesterday	✓ 1 / 1	Default	...
additional_roles_dropdown	2 days ago	✓ 1 / 1	5 0	#26
revert-23-revert-22-additional-position-in-profile-change	3 days ago	✗ 0 / 1	8 0	#25
additional-position-in-profile-change	3 days ago		11 0	#24
revert-22-additional-position-in-profile-change	4 days ago	✓ 1 / 1	10 0	#23
dev/Leo_survey_response_test	5 days ago	✓ 1 / 1	24 0	#21
dev/Leo	5 days ago		66 3	
ensure_survey_questions_filled	5 days ago	✓ 1 / 1	26 0	#20
sprint3-nithin	5 days ago	✗ 0 / 1	32 0	#17
bug/admin_page	2 weeks ago	✗ 0 / 1	40 0	#18
feature/survey_responses_admin	3 weeks ago		58 0	#14
manoj850	3 weeks ago	✓ 1 / 1	64 0	
admin-dashboard-login-integration	last month	✓ 1 / 1	79 0	#11
manojpeta	last month		115 4	
admin-management-display-page	last month		108 0	#4

10. Production

We deployed our application using a PaaS provider Heroku, and have set up continuous deployment hooks inside our GitHub repository.

The automatic deployment pipeline catches any code that fails any tests and does not deploy the new application code. Once the code passes the test checks, Heroku detects a successful push to the main branch and creates a new deployment.

Heroku instantiates a production postgresql database for our application to use in its production environment.

One of the issues that we encountered initially was with production database configuration. Every time we cloned the Git repository onto a new local machine, the default settings in the Gemfile would attempt to load the PostgreSQL adapter required for the production environment. However, since our local development environment used SQLite (and we didn't want to install or configure PostgreSQL locally), this caused errors when running commands like rails server or rails db:migrate.

To resolve this, we had to run:

bundle config set --local without 'production'

This ensured that the production-specific gems, like the PostgreSQL adapter, were excluded from the local bundle. This step was necessary for local development to work smoothly.

The Heroku environment automatically handled the configuration for the production database, allowing seamless deployment without the need for these adjustments. While this

workaround worked, it served as a reminder of the importance of environment-specific configuration management.

11. Tools

11.1 CI/CD with GitHub Actions

For the Rubocop badge in the CI/CD workflow on GitHub we used the **Dynamic Badges action** plugin. To set this up, a developer must create a GitHub OAuth token with gist permissions and store it as a repository secret under **GIST_SECRET**. Then, navigating to <https://gists.github.com> and make a blank gist named "**badge.json**" and copy the Gist ID into the workflow configuration; the link will look like <https://gist.github.com/username/:gist-id>

11.2 AuthO

We are using **AuthO authentication service** to authenticate users within our application (OAuth OpenID service).

A new Auth0 can be configured by going to the site <https://auth0.com>, and creating a new application, followed by configuring the local Ruby application details on Auth0. Finally, the Auth0 id and the code snippet must be copied from the Auth0 dashboard to our local Ruby folder in order for the Auth0 to start working.

11.3 SimpleCov & CodeClimate

We are using SimpleCov to measure our application's code coverage, which is the percentage of our code that is covered in our automated behavioral and unit level tests. This tool is incredibly useful, as it reveals hidden areas of our code that were untested, and therefore unpredictable by our test-driven development standards.

CodeClimate is where we publish the results of our SimpleCov coverage report, such that interested parties may see the maintainability and relative health of our codebase. The link to our public CodeClimate page is in our GitHub readme.

11.4 Testing Framework

We are primarily using the Cucumber testing framework for our behavioral tests, and using the Rspec testing tool for our unit-level tests.

13. Deployment Steps

1. Create Heroku application

```
heroku create <app_name>
```

2. Add buildpacks

```
heroku buildpacks:add https://github.com/timanovsky/subdir-heroku-buildpack.git --app  
<app-name>
```

```
heroku buildpacks:add heroku/ruby --app <app-name>
```

3. Add config vars

```
heroku config:set PROJECT_PATH=rails_root
```

```
heroku config:set RAILS_MASTER_KEY=<master key here>
```

4. Install [Heroku Postgres](#) and attach to application

a. Push to heroku app

```
git push heroku main
```

b. Generate database

```
heroku run rails db:migrate
```

```
heroku run rails db:seed
```


5. Set up Github Action (Different from Github Integration with Heroku)

1. Set up `secrets.RAILS_MASTER_KEY` in your project Github Repository
2. Set up `secrets.GIST_SECRET` in your project Github Repository by valid token with gist scope
3. Set up `secrets.GIST_ID` (Create your own gist, and set up the gist ID in your project Github Repository)

14. Links

Github: https://github.com/tamu-edu-students/csce606-ELRC-OLEI_Project
Pivotal Tracker: <https://www.pivotaltracker.com/n/projects/2720653>
Slack Workspace: <https://app.slack.com/huddle/T07NQ098G0G/C07NQ0G0TSL>
Live Deployment Link: <https://elrc-6ad76821be30.herokuapp.com/about>

15. Links to Presentation and Demo

Presentation:  FinalPresentation
Demo Video Link: <https://www.youtube.com/watch?v=cNhBqhwOIKI>