WPI

Aiding in the Migration of the E-TRIALS Platform

Kristine Guan

Sharon Wu

In collaboration with:

May Dong (MQP)

Michael Oliveira (MQP)

Submitted to:

Dr. Neil Heffernan

Abstract

E-TRIALS was launched by the ASSISTments Foundation in 2022 to facilitate educational studies for researchers investigating different teaching methods and supports. With the evolution of the released product, a need for a content migration has been identified to provide more interactive features. This MQP adapted to the Agile environment established by ASSISTments full-time engineers to assist in the changes necessary for the migration. During the duration of the project, we made improvements to existing code, while also aiding with migration.

Acknowledgment

We would like to acknowledge the ASSISTments community and those who supported us throughout our project. In particular, we would like to say thank you to Ryan Emberling for providing great direction and unrelenting support throughout the entirety of our project. Thank you to May Dong and Michael Oliveria for dedicating time and energy to passing down their knowledge and being optimistic while troubleshooting. Thank you to Dr. Neil Heffernan for the pleasure of working alongside the E-TRIALS team. Thank you to Hannah Pandolph for the welcoming and friendly environments during meetings. Last but not least, thank you to Angela Kao for handling administrative matters with ease and always with a smile.

Table of Contents

1. Introduction

The prevalence of online learning has grown incredibly since the Covid-19 pandemic beginning in 2020. Teachers had quite the disruption in their learning plans and implementations and had to aggressively pivot to online platforms (Ali & Nath, 2023). Now in 2024, some schools and institutions have returned to in-person, though many have adjusted to hybrid environments which integrate online learning into their curriculum. As the use of online learning persists and evolves, it is important for teachers to ensure they are providing a positive and successful learning experience to students within an online learning model. Given unknowns of what is optimal for students and teachers in such a setting, research is vital to identify what is best for all users of online learning.

Created by Neil and Christina Heffernan, the ASSISTments Foundation is an online learning resource that serves not only as a learning platform, but a research endeavor that addresses questions of what works well for students online. ASSISTments is "dedicated to improv[e] student learning through responsive online technology," and this mission is supported by their team of learning scientists, educators, and software engineers (About the ASSISTments Foundation). Learning resources are free and available to all teachers and students, and researchers are able to utilize available education content to conduct studies.

ASSISTments strives for continuous growth with their product, and has gone through phases of improvements and iterations, as software naturally does. A large push for a new iteration of the software was to address issues such as outdated design and unintuitive features for users. Developed at Worcester Polytechnic Institute (WPI), the Foundation works in collaboration with the institute to further the growth of ASSISTments. These projects largely involve Major Qualifying Projects (MQP) with computer science undergraduate students, where students apply their programming knowledge to help address design and feature improvement.

One MQP in 2020 coined the term E-TRIALS to represent the new iteration of ASSISTments, and basic designs and prototypes were born (Krichevsky & Spinelli, 2020). Continuing their progress, another MQP group aimed to improve and build on the original designs to reach their goal of preparing a minimum viable product (McCarthy, 2021). The official launch of E-TRIALS in February 2022 was made possible with the help of yet another MQP group which worked towards final preparations (Philippo & Spofford, 2022). These former

MQP groups illustrate the journey E-TRIALS has taken the past few years, and it is now active and readily available for researchers. The focus for E-TRIALS has since shifted to hone in on refinements and improvements, and MQP groups are continuing to aid E-TRIALS through this leg of its growth.

2. Background

2.1 ASSISTments

ASSISTments is a free online educational platform that provides practice and assessment tools for K-12 math teachers, allowing them to select over 200,000 problems from certified textbooks or Open Educational Resource (OER) materials. It integrates with platforms like Google Classroom and Canvas, allowing teachers to easily assign problem sets to their students. Upon completion of problem sets by students, teachers receive in-depth reports on both individual and class performance. These insights are helpful in customizing teaching strategies to meet students' needs. ASSISTments offers many student supports such as immediate feedback after the completion of a problem, hints, and tailored guidance for common wrong answers. These supports enable students to deepen their understanding of the subject matter and provide a guide for navigating similar challenges in the future. The platform is also pioneering in educational research through E-TRIALS, simplifying the execution of randomized controlled trials (RCTs) with the use of real student data gathered from ASSISTments.

2.2 E-TRIALS

E-TRIALS (Ed-Tech Research Infrastructure to Advance Learning Science) is a learning research platform integrated with ASSISTments. It allows researchers to conduct large-scale education-based experiments in an efficient and cost-effective way. This built-in research tool enables researchers to use ASSISTments' existing student user base to quickly find participants for studies aimed at uncovering the most effective teaching methods and educational content. By simplifying the logistics of participant recruitment, E-TRIALS allows researchers to put more focus on the quality and content of their studies, including the design and deployment of RCTs. The goal is to enhance learning outcomes by identifying the optimal conditions for student learning, making it a valuable asset for education researchers and ultimately benefiting educators and students alike.

On E-TRIALS, researchers can choose from four different types of studies to deploy. The "Support Comparison" study allows for the creation of two or more conditions with student supports, which are in the form of hints or direct explanations of the answer for a given problem

set. These supports are then randomly assigned to students and are shown when the student asks for help on the given problem set. This setup enables a comparative analysis of the effectiveness of each support based on the students' performance. E-TRIALS also offers "Single Support" studies which focus on testing a single type of support compared to the platform's default support. "Problem Varied" studies vary the problem sets given to the students instead of the supports. Lastly, "Common Wrong Answers" studies are aimed at providing specific feedback for common wrong answers provided by students to address common misconceptions and knowledge gaps.

Upon the conclusion of a study, researchers receive a report detailing student performance across each of the given conditions that were implemented. These reports are helpful in understanding the impact of the varied supports or problem sets tested in the study. These diverse study designs offer researchers multiple approaches to investigate the most effective educational supports, thereby contributing to improved learning strategies and outcomes.

2.3 The History of E-TRIALS

2.3.1 Assistments 1.0 Builder

The origination of E-TRIALS traces back to the functionality of the ASSISTments 1.0 Builder, which is a tool designed for teachers to create problems and problem sets for student assignments. The Builder also served as a tool for researchers that enabled them to create studies that would later evolve into the capabilities of the current E-TRIALS platform. The inherent complexity of the builder, designed to accommodate both content creation and research study construction, presented significant usability challenges for both teachers and researchers. Teachers found the research-oriented features unnecessary for their purposes, while researchers struggled with the study's tree structure visualization, which did not align with the linear and flowchart-like designs they were accustomed to. Figure 1 demonstrates this complexity, depicting two different studies integrated within the tree-like framework of the 1.0 Builder. Researchers unfamiliar with such tree structures often required direct assistance from the ASSISTments team, leading to a time-consuming process (Krichevsky & Spinelli, 2020).

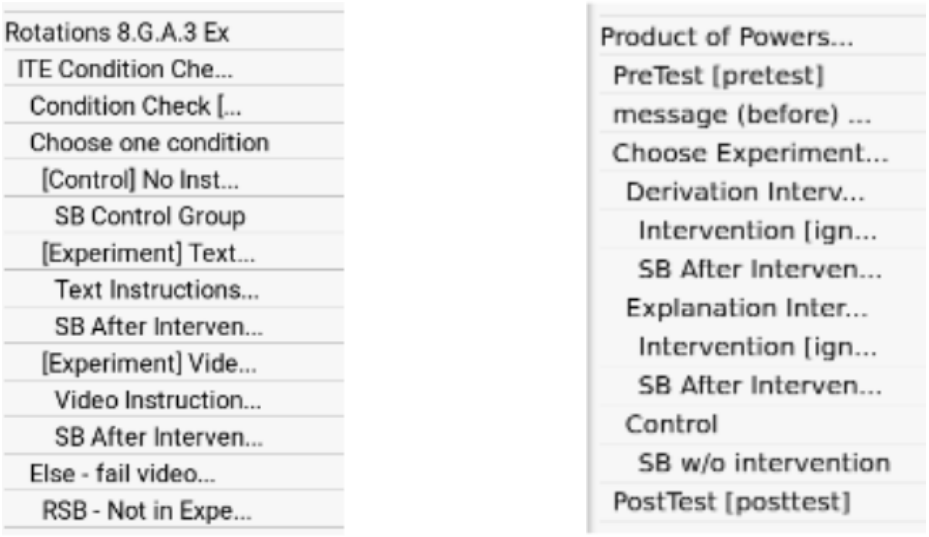


Figure 1. Two studies created in Assistments 1.0 Builder

2.3.2 E-TRIALS Prototyping

The drive to resolve usability issues and streamline the study creation process led to the development of the E-TRIALS platform. Initiated through an Interactive Qualifying Project (IQP) by students Nicholas Krichevsky and Kamryn Spinelli in 2020, the project aimed at redesigning the process of creating studies within ASSISTments. Through user research and prototype development, the students sought to simplify documentation, visualization, and content selection aspects of study building. Their efforts, which involved creating wireframe models to demonstrate new study construction flows and conducting user testing with ASSISTments team members, highlighted the need to limit granular control in the study building process (Krichevsky & Spinelli, 2020). This feedback was instrumental in shaping the final E-TRIALS prototype (shown in Figure 2), which utilized the Vue.js JavaScript and Vuetify Material Design frameworks for its user interface.

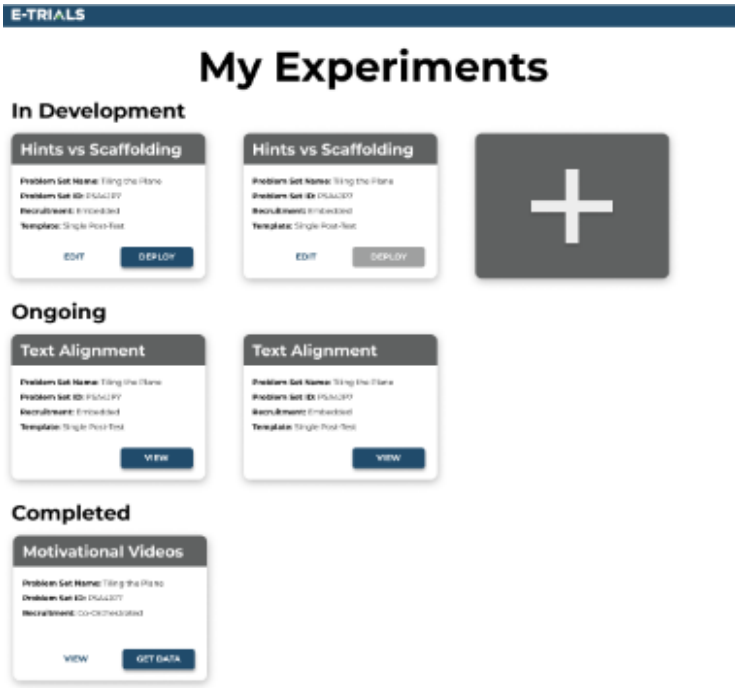


Figure 2. Homepage of the E-TRIALS interface prototype

Despite the initial project focusing more on interface design rather than full functionality, it laid the groundwork for what E-TRIALS would become: a more accessible and efficient platform for conducting educational research, addressing the original builder's limitations and setting a new direction for research study creation within the ASSISTments ecosystem.

2.3.3 Full-Stack Implementation and Development

The development of E-TRIALS saw further progression in 2021 through the efforts of Timothy McCarthy's IQP. The project aimed at refining the user interface created in the previous IQP, and establishing the necessary backend infrastructure. The integration of the frontend with a newly developed backend, with assistance from the E-TRIALS development team, was pivotal in ensuring that data could be persistently stored and retrieved from an external server, rather than being lost once the browsing session ended. This backend integration also set the stage for more robust testing of the E-TRIALS application, including comprehensive end-to-end tests that simulated user interaction and helped to identify and correct potential bugs. Moreover, it allowed for real user testing in a dynamic environment that mirrored actual use cases more closely.

Along with these technical enhancements, the project focused on simplifying the study building process to make it more intuitive for researchers. This involved creating a modular

structure where parts of the study could be toggled, facilitating a more manageable approach to building studies while still providing guidance through the process. The translation of study data from the frontend to the backend was mapped to database tables, as conceptualized in Figure 3, reflecting the relational structure of the E-TRIALS tables.

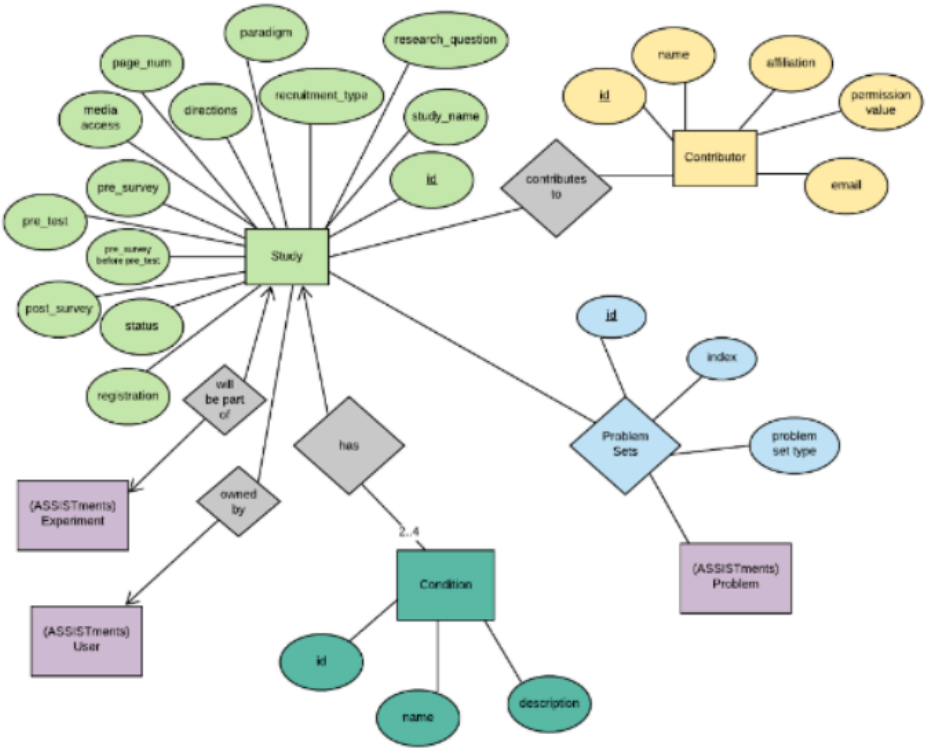


Figure 3. Entity Relationship Diagram of E-TRIALS tables

Another innovation during this phase was the reformation of the content selection process. The initial design's sequence was streamlined to prevent researchers from getting sidetracked by interjecting steps, allowing for a smoother transition from content selection to condition association. This improved continuity and helped researchers to better comprehend the problem sets within their studies. Enhanced functionality was also introduced to the content selection interface, with the addition of filters and metadata to inform decision-making about content inclusion, as illustrated in Figure 4 (McCarthy, 2021).

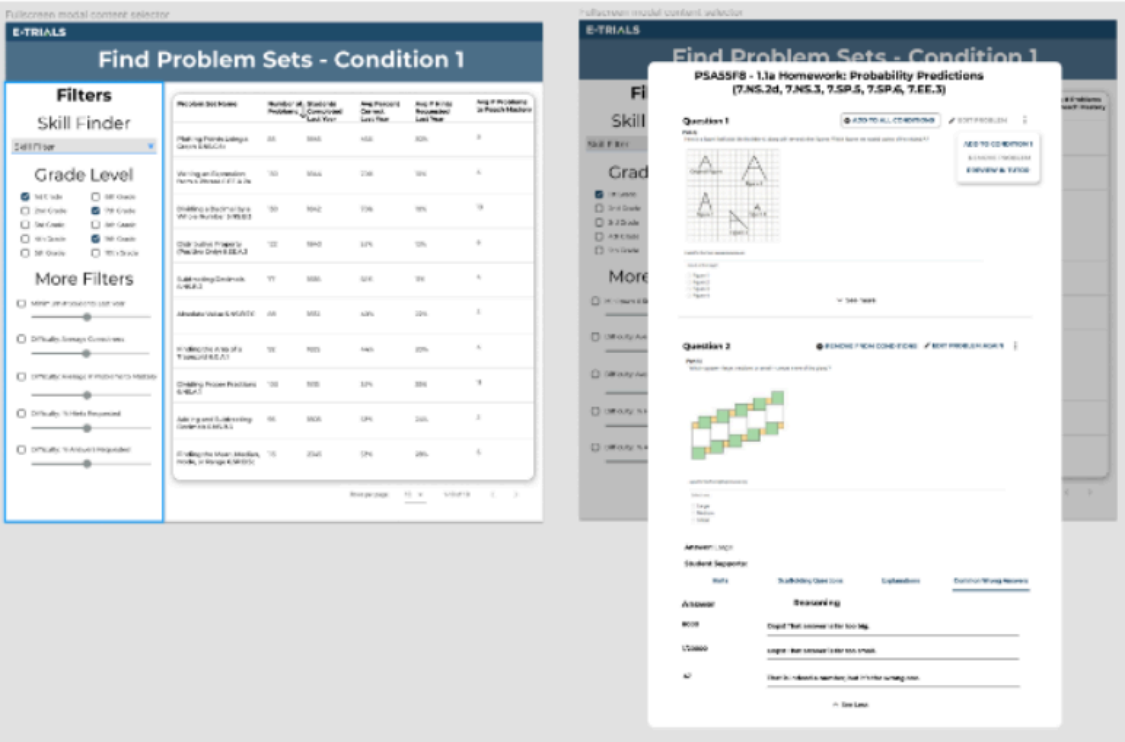


Figure 4. Content selection modals in Figma

Despite these advancements under McCarthy's IQP, there remained unfinished areas in E-TRIALS' development, including gaps in functionality, security, and infrastructure. Consequently, additional work was needed before E-TRIALS could be rolled out as a minimal viable product for the educational research community.

2.3.4 The Release of E-TRIALS as a Minimum Viable Product

Development resumed with an MQP by Edward Philippo and Matthew Spofford in 2022. Their objective was to advance the E-TRIALS platform towards becoming a minimum viable product (MVP). Collaborating with the E-TRIALS development team, their efforts focused on enhancing the platform's functionality, security, and infrastructure. This project was pivotal for E-TRIALS, with the launch aligning with the project's conclusion in February 2022 (Philippo & Spofford, 2022).

A critical aspect of their work involved addressing functionality issues that had surfaced during initial user testing phases. A significant portion of these improvements was linked to integrating E-TRIALS with the Open Science Frame (OSF), a platform designed to facilitate research sharing within the scientific community. By using OSF's API, Philippo and Spofford

tackled challenges related to the seamless loading of study data and ensuring that updates made within E-TRIALS were accurately mirrored on the OSF platform.

Moreover, they contributed to the redesign of the platform's user interface elements, such as the problem set stepper. This redesign aimed to provide researchers with a clearer understanding of their progress within a study, indicating both the sections they were currently editing and those they had completed. Such improvements were not only functional but also enhanced the overall user experience, making the research process more intuitive and efficient.

Security was another major area of focus. The team identified a significant vulnerability: the ease with which any user, provided they had the study URL, could access and modify study details. To mitigate this risk, backend modifications were implemented to ensure that only study owners and administrators could view or edit study information. This was complemented by the introduction of new role-based permissions, further refining the platform's security framework by describing what actions users and admins could perform at various stages of study development.

By March 2022, the contributions of Philippo and Spofford, in collaboration with the E-TRIALS development team, were instrumental in launching the platform. Their work facilitated the transition of E-TRIALS into a functional MVP, ready for use by researchers conducting educational studies through ASSISTments.

2.3.5 Additional Features and Modifications

In 2022, an MQP by Patrick Spillane focused on enhancing the user experience on the E-TRIALS platform, specifically on revamping the deploy page to streamline the process for researchers conducting educational studies. Recognizing the confusion caused by the deploy page's redundancy and lack of informative feedback on the state of a study, Spillane's project transformed it into a single, scrollable page. This new design allows researchers to easily view all supports written for the study in one consolidated space, significantly improving usability and efficiency. Additionally, Spillane introduced updates to how correct answers for problems are displayed to researchers, ensuring alignment with the student's view and thereby enhancing clarity and preventing misconceptions (Spillane, 2022). Through these improvements, Spillane's project contributed to making the E-TRIALS study builder more intuitive and valuable for researchers, ensuring a consistent and well-organized interface.

In 2022, Jin Ryoul Kim's IQP focused on implementing an onboarding process for new researchers and addressing the accumulated technical debt due to rapid development phases. This introduced an onboarding prompt after account creation, enabling the platform to collect detailed user information, such as teaching grades and curriculum preferences. To tackle the platform's technical debt, Kim streamlined the coding and styling processes by integrating SASS and Vuetify for a unified styling framework, enhancing maintainability and ensuring a consistent user interface. Furthermore, the shift towards using enum objects for better code clarity and migrating to TypeScript helped minimize type-related errors, improving overall code readability and structure (Kim, 2022).

May Dong's IQP completed in 2023 focused on introducing new features and quality-of-life improvements to E-TRIALS. She modified the TinyMCE text editor used within E-TRIALS, which originally threw errors for image uploads exceeding 500kb. Dong's solution involved applying compression to larger images, ensuring they could be uploaded without issues. Additionally, the project introduced backend endpoints for retrieving and editing Project Page IDs, enabling researchers to easily manage and display this information within their studies. This facilitated a smoother integration with the Open Science Framework (OSF), making it simpler for researchers to link their E-TRIALS studies with OSF project pages. Furthermore, Dong initiated the implementation of a new study type named "Common Wrong Answer," aimed at providing tailored feedback to students based on frequent incorrect answers. This feature was designed to allow researchers to explore more effective ways of supporting students who encounter common misconceptions in problem-solving (Dong, 2023).

Emily Gorelik's MQP in 2023 aimed at enhancing the design consistency and usability of the E-TRIALS frontend, alongside addressing various bugs identified post-release. Gorelik's project targeted the uniformity of interactable UI elements, which previously lacked consistency in size, shape, and color. She also developed a UI element enabling direct editing of OSF information within E-TRIALS, eliminating the need to navigate to the OSF study page. This effort was part of a broader initiative to refine E-TRIALS following its MVP launch, focusing on usability and visual components that were initially overlooked due to prioritizing functionality (Gorelik, 2023). Gorelik's work also included making the Buildplace Stepper persistently visible on scrolling, correcting improperly nested buttons that affected interaction cues, and ensuring that edit functionalities are appropriately toggled based on the study's status. Additionally,

improvements were made to enhance user interaction, such as displaying study names in confirmation pop-ups to prevent publishing errors and making OSF Reference Information and the number of participants in a study readily accessible and editable (Gorelik, 2023).

In their 2023 MQP, May Dong and Michael Oliveira focused on enhancing the E-TRIALS platform by finalizing the “Common Wrong Answers” study type, facilitating content migration for new answer types, and conducting comprehensive maintenance including UI/UX improvements. They developed frontend UI and logic to enable researchers to create support for common wrong answers, enhancing educational feedback by directly addressing students' misconceptions. They also tackled the migration of database schemas to accommodate new answer types like drag and drop, updating IDs to CERIs and removing outdated columns, which involved creating PostgreSQL scripts and updating both backend and frontend to reflect these changes. Maintenance efforts addressed bugs, such as the visibility of action icons in various study phases, and reduced technical debt by enhancing the platform's mock backend capabilities and simplifying content addition processes. Additionally, they focused on UI/UX consistency and responsiveness, updating UI images across study content pages for uniformity and introducing skeleton loaders for improved user feedback, significantly improving the platform's usability and visual appeal (Dong & Oliveira, 2023).

2.4 Migration

Content migration within E-TRIALS is a critical process as the platform's extensive database continues to expand, accommodating new studies and diversifying data types to support educational research. This migration is not just about scaling up; it's an enhancement aimed at significantly improving the application's functionality and robustness. With the platform's ambition to broaden the range of user interactions, content migration targets the introduction of more dynamic question formats like drag and drop and dropdown menus. These formats are crucial for accurately reflecting the complexity and varied nature of math questions, moving beyond the limitations of single input boxes to more interactive and engaging question types. The migration to SDK 3.0 is a pivotal step in this migration, necessitating a comprehensive refactor of E-TRIALS' architecture to ensure compatibility with more interactive features. Understanding the existing infrastructure and the necessary changes is crucial for progressing

towards a more interactive and user-friendly environment, making content migration a cornerstone of E-TRIALS' ongoing development.

3. Methodology

3.1 Software Development Process

The E-TRIALS project follows the Agile Scrum Methodology, a common software programming practice designed to ensure efficient and streamlined progress. “[S]crum describes a set of meetings, tools, and roles that work in concert to help teams structure and manage their work.” MQP groups working on E-TRIALS projects work alongside full-time ASSISTments employees and adopt this methodology as well (Atlassian).

Recurring meetings are conducted to maintain the flow and cycle of a Scrum Agile environment. These meetings, held on Zoom, include stand-ups, story refinement, and story planning, which allocate time for team members to share their progress in their work, for a leader to clarify on any tasks that may need more detail, and for user stories to be built on.

The E-TRIALS team operated in two-week sprints, which are defined periods of time dedicated to working on assigned tasks known as tickets. Tickets represent the tasks assigned to team members for completion during a sprint, and their progress can be communicated through the mentioned meetings throughout the two-week periods.

Tickets were assigned to the MQP student group by Ryan Emberling, the Director of Engineering at The ASSISTments Foundation. Characteristics of the Scrum Agile methodology were reflected in standups in Slack between Ryan and the MQP group, in addition to weekly check-in meetings. Besides standups, Slack served as a platform used for messaging and holding impromptu meetings.

Organization of tickets was managed by Jira, a tool that offered a framework of timelines and Kanban boards for team members to easily keep track of their tickets. Additionally, each ticket was linked to an associated branch managed by GitHub, providing a seamless integration with Jira through automatic linkage between tickets and branches.

3.2 Tech Stack

Similar to many applications, E-TRIALS consisted of a frontend and backend to behave and function properly. The frontend is responsible for what the user sees and interacts with, and is made with design and accessibility aspects in mind. The backend is responsible for managing

and storing all the data that the application receives and creates. The database is the part of the backend in particular that sends data to the backend to be transformed into a usable form.

The frontend is programmed in JavaScript and uses the Vue.js framework. Building on standard HTML, CSS, and JavaScript, Vue "provides a declarative and component-based programming model that helps you efficiently develop user interfaces" (Vue.js). Vue comes with its own state management pattern and library, Veux store, where information that is available globally to all components is stored. Used in conjunction with the frontend, Vue,js devtools allows for debugging of the frontend in the browser. The tool is a Chrome extension that can break down components and assets to effectively identify necessary information during the debugging process. Vite is utilized for building frontend projects, and is "a build tool that aims to provide a faster and leaner development experience for modern web projects" (Vite).

The backend is programmed in Java, and uses the Spring framework, which "provides a comprehensive programming and configuration model for modern Java-based enterprise applications - on any kind of deployment platform" (Spring). Though the backend for E-TRIALS has historically run on Apache Tomcat servers, the use of Docker has been an appealing alternative approach. "Docker helps developers build, share, run, and verify applications anywhere — without tedious environment configuration or management" (Docker). Because of the efficient and consistent nature of Docker, it was implemented and utilized during the last term of the project, and has proven itself to be a more seamless and intuitive experience for developing. With running the backend using Docker, more time was allotted for focusing on programming, rather than setting up the programming environment.

When it comes to hosting the frontend and backend, it can be done both locally and on Amazon Web Services (AWS). For deployment purposes, the application is hosted on AWS.

Version control is managed through Github, where repositories of code exist in the means of branches. The production and dev branches follow convention and are default in each E-TRIALS repository. The production branch holds code that is currently in use, and the code for the application that users interact with. The dev branch is for code that is not quite ready for production, as it may require testing or is anticipated to have on-going changes. To reflect and support the content mutation that is currently undergoing, many repositories also have a migration branch, which holds code that reflect changes identified as a part of the migration.

3.3 Technical Approach

In addressing the requirements of our tasks, our technical strategy was divided into different components, each fulfilling a significant role in the development of E-TRIALS.

3.3.1 UtilTools

UtilTools is a repository tool for generating Java files that mirror the database structure, creating direct interaction between the application's backend and its database. This automation creates three file types needed to make this connection:

1. Domain File: Represents a database table row as a Java object, enabling object-oriented database interactions.
2. DAO (Data Access Object) File: Defines the interface for methods to interact with database tables.
3. JdbcDao File: Implements the DAO file with SQL queries and mappings, bridging domain objects and table rows.

3.3.2 Backend

ETrialsService, the backend component of E-TRIALS, handles the core logic, database communication, and API services. It is responsible for processing data, executing logic, and serving as the backbone for the platform's functionality. This backend infrastructure uses the Java files generated by UtilTools to ensure seamless database integration and operations.

3.3.3 PostgreSQL

PostgreSQL scripts play a role in managing database schemas and facilitating data migration. These scripts allow for data manipulation and structure alteration, ensuring the database's evolution aligns with the application's growing needs. This component is vital for maintaining data integrity and consistency across the platform's development lifecycle.

3.3.4 Frontend

ETrials, the frontend component of the project, focuses on user interaction, presenting a visually engaging and intuitive interface. It utilizes modern web development frameworks to create responsive and dynamic user experiences, making educational research data accessible and manageable. The frontend communicates with ETrialsService to fetch and display data, ensuring users have a seamless interaction with the platform.

3.3.5 Debugging with Vue DevTools

Vue DevTools is a tool for inspecting and debugging the frontend, allowing developers to view and manipulate application state, observe component structures, and troubleshoot issues. This tool enhances the development efficiency and aids in ensuring the reliability and performance of the E-TRIALS frontend.

3.3.6 Full-Stack Development

Implementing changes to the E-TRIALS platform often requires a full-stack approach, especially for content migration that impacts both the frontend and backend. This process sees the Vue frontend initiating API calls to the Java Spring backend, which in turn queries the PostgreSQL server. Such a workflow is contingent on the database being accurately structured, with all necessary columns and tables in place, to ensure successful builds. Backend files, often autogenerated from .tdf files in UtilTools, are aligned with the database structure to maintain consistency and functionality. This interconnected system requires a deep understanding of how each layer of the stack influences the others, highlighting the importance of a cohesive full-stack perspective. Successfully navigating this complexity was particularly important for tasks like migrating to SDK 3.0, where changes had to be integrated across all levels of the E-TRIALS infrastructure.

4. Results

4.1 Migration

Content migration stands as a critical task currently in progress within E-TRIALS. Numerous changes to the database need to be reflected across the full stack. In the context of API consumption, IDs are changed to string-based CERIs (Content External Reference Identifier) to enhance the usability and readability of the system. This adjustment allows internal methods to operate with CERIs, which are more recognizable and user-friendly for both teachers and students than numerical IDs. By using strings that are easier to identify, the system simplifies the translation process and ensures a more intuitive interaction at the API level. Although switching from integers to strings incurs a performance tradeoff—since comparing integers is more efficient than comparing strings—the impact is generally minor relative to the overall benefits.

We have actively contributed to this migration effort, specifically changing the problem set IDs to CERIs in three key tables: wip_problem_study and problem_set_usage within ETrialsService, and problem_experiment in CAS. To make this change across the full stack, our initial step involved updating the .tdf files in UtilTools for each affected table, transitioning problem_set_id or problem_id from an integer to problem_set_ceri or problem_ceri as a string. Following this, UtilTools was used to generate the three files—domain, dao, and jdbcDao—needed to bridge the database with the backend, based on the revised .tdf files. These files were then integrated into the backend, which also needed updates to both the variable names and types within table objects to align with the updated database schema. Additionally, certain methods were modified to adapt to the data type transition.

For the problem_experiment table, this shift from problem set IDs to CERIs had an impact on several methods, particularly publishExperiment() and registerAssignment() within ExperimentManagerPrivate, both of which interact with the problem_experiments relation table. This required updates not only to the methods themselves but also to their declarations and the code that calls these methods, transitioning them to utilize CERIs instead of database IDs. The final step in this migration process involved updating the frontend to accommodate new or

revised APIs from the backend, alongside variable name changes, ensuring a seamless transition and maintaining the integrity of the E-TRIALS platform's functionality.

Another task in our migration efforts involved updating the admin view study table on the E-TRIALS website. Initially, this table was populated using the StudySummary object; however, we transitioned to using the new StudyDTO data structure. The reason behind this switch was that StudyDTO offers a more comprehensive dataset than StudySummary, effectively reducing the need for additional API calls, such as tracking metrics like participant count. When we took on this task, the relevant database tables had already been updated to accommodate this new approach. Yet, these updates had not been reflected in both the backend and the frontend, leading to a situation where the admin view study table failed to populate, a problem that is shown in Figure 5.

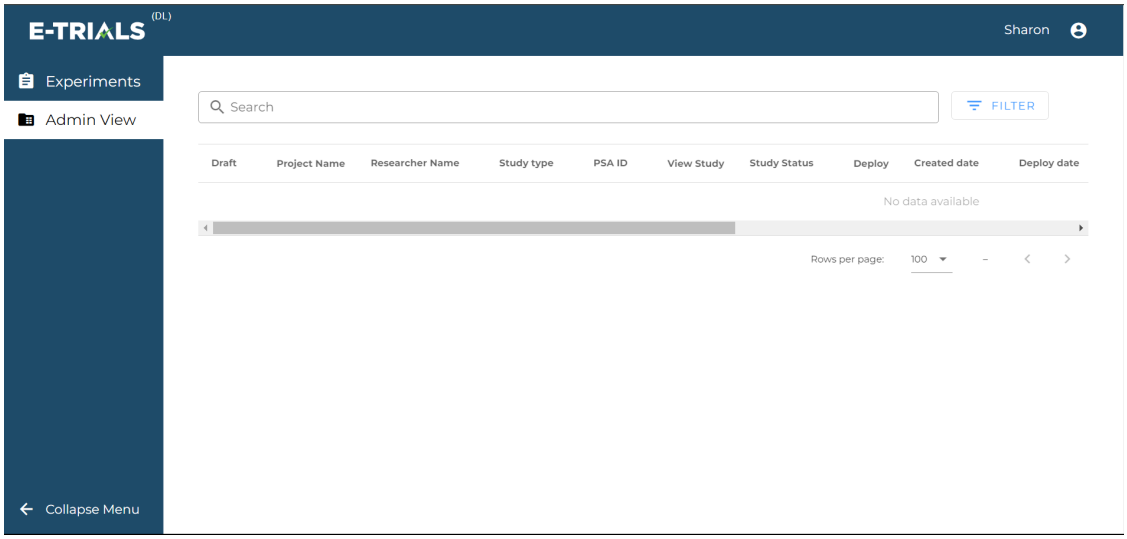
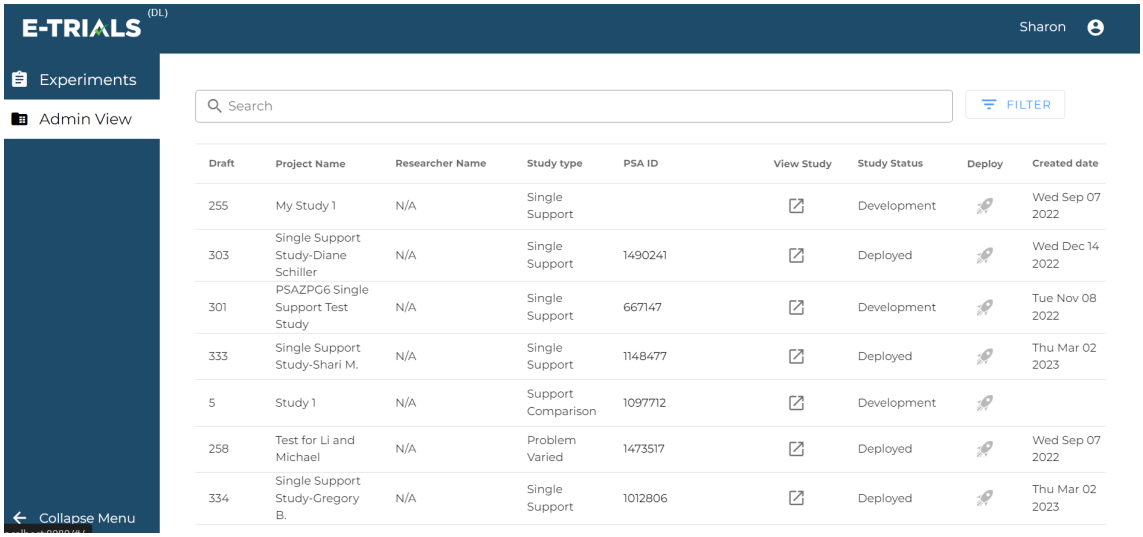


Figure 5. The admin view table before frontend and backend migration changes.

To ensure the admin view table was populated with the updated StudyDTO object, we needed a series of backend and frontend modifications. On the backend, we updated the loadUserStudies method in the StudyController to return StudyDTO instead of StudySummary. Additionally, we refactored the numParticipants attribute to ensure that all studies within the study list utilized the StudyDTO object.

Reflecting these backend changes on the frontend was also needed. We updated the studyList store along with its consumer pages, such as the Home page, and the numParticipants attribute to adopt the StudyDTO object, moving away from the now-deprecated StudySummary

object. This update across the stack was needed in resolving the issue with the admin view study table, which, after the implementation of these changes, was successfully populated, as illustrated in Figure 6. This successful update not only improved the admin interface's functionality but also underlined the importance of maintaining consistency and synchronization between the backend and frontend components of the E-TRIALS platform.



Draft	Project Name	Researcher Name	Study type	PSA ID	View Study	Study Status	Deploy	Created date
255	My Study 1	N/A	Single Support			Development		Wed Sep 07 2022
303	Single Support Study-Diane Schiller	N/A	Single Support	1490241		Deployed		Wed Dec 14 2022
301	PSAZPG6 Single Support Test Study	N/A	Single Support	667147		Development		Tue Nov 08 2022
333	Single Support Study-Shari M.	N/A	Single Support	1148477		Deployed		Thu Mar 02 2023
5	Study 1	N/A	Support Comparison	1097712		Development		
258	Test for Li and Michael	N/A	Problem Varied	1473517		Deployed		Wed Sep 07 2022
334	Single Support Study-Gregory B.	N/A	Single Support	1012806		Deployed		Thu Mar 02 2023

Figure 6. The admin view table after frontend and backend migration changes.

4.2 Maintenance

Maintenance is a key aspect of keeping E-TRIALS in good condition in order to continue growing the product. Contributions were made towards code maintenance in this project with the use of Mirage, “a JavaScript library that lets frontend developers mock out backend APIs” (Mirage JS). Mirage serves as a helpful tool when developers want to avoid working with the backend, either hosted locally or on AWS, when it is not completely functional. Providing recreations of dynamic scenarios, Mirage mocks the typical behavior of a real server to allow programmers to work on frontend issues independent from a real backend. The mock backend is especially relevant and helpful during the content migration of E-TRIALS, where the code base is adjusted in real-time to accommodate changes.

The use of Mirage was beneficial when it came to approaching Ticket ET-303 described in Appendix A, which targeted a code maintenance issue where studies on the admin table were not able to be searched by the problem set ID. With migration changes affecting the backend,

Mirage became a key aspect of the approach to completing the ticket. Mock seeds and endpoints were created to simulate the admin table on the mock backend. Once the admin data table was properly simulated, the symptoms of the bug were easier to see, and Vue Devtools was used to identify the root of the problem. The ticket was successfully completed, and the fix allowed for admins to have more ease when searching for studies (shown in Figure 7), improving and maintaining the user experience.

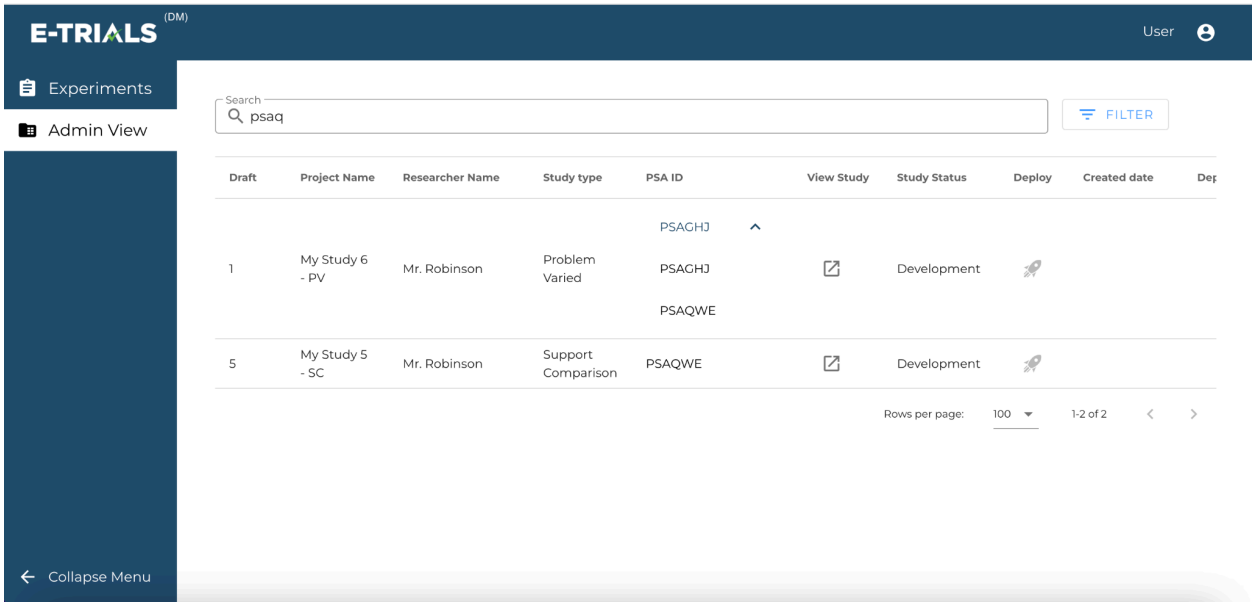


Figure 7. Search by Problem Set ID Implemented with mock data from Mirage

Mirage was also helpful during our work on Ticket ET-304, which required the integration of a new endpoint to fetch a list of skills for the Skill Finder feature on the Browse Problem Sets Table page. The Skill Finder, a tool designed to enable filtering of problem sets by skill, was impacted by backend disruptions caused by ongoing content migration efforts. To navigate this challenge, we opted to use Mirage to create mock data representing the skills. This approach allowed us to simulate the expected backend response, filling the gap left by the temporarily unavailable backend services. By implementing this mock data and utilizing the mock backend setup, we successfully populated the Skill Finder with the necessary skills, restoring the filter functionality. The effectiveness of this solution was demonstrated in the operation of the Skill Finder, as highlighted in Figure 8, showcasing how Mirage can serve as a critical tool in maintaining frontend functionality amidst backend development and migration processes.

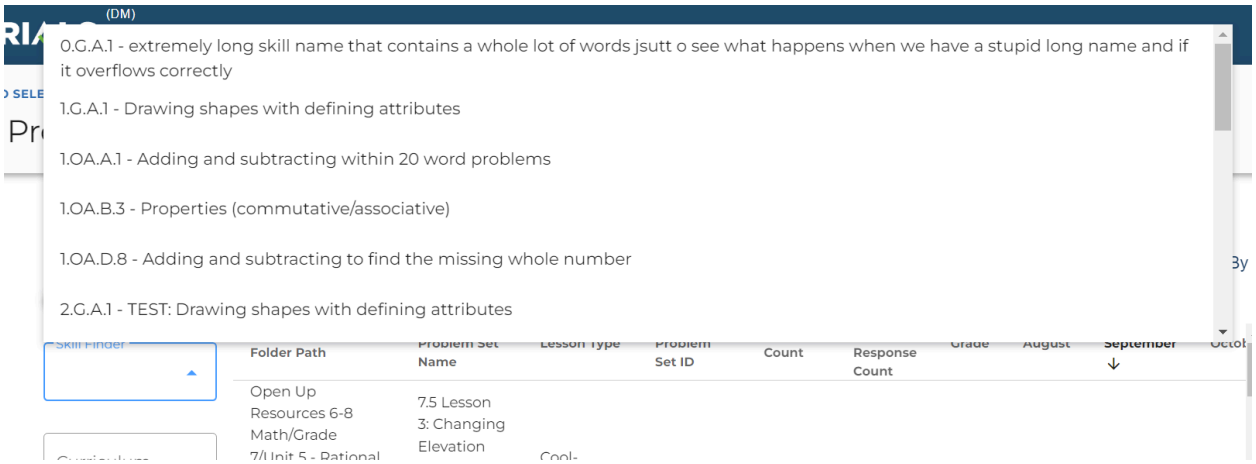


Figure 8. Skill Finder populated with mock data from Mirage

5. Future Work

A large scope of the work towards E-TRIALS is maintenance, which is routine and natural for a released product. As bugs are being discovered and properly identified in Jira tickets, they need to be tasked and worked on routinely by engineers. New bugs will continue to be identified and will consistently need to be dealt with.

Another identified area of growth would be surrounding the testing infrastructure. Tests are put in place to ensure the persistence of existing functionality of E-TRIALS, and currently there are limited automated tests. Future work could focus on developing more automated tests to increase code coverage. More end to end tests could be made as well, to confirm proper integration between the frontend and backend.

In a more urgent manner, E-TRIALS will continue to go through content migration, where changes are being made in the methods various types of data are being stored and processed. Updates will need to continue to be made to ensure the platform remains compatible with the migration changes. The future of student projects will likely aim towards assisting with this migration.

6. References

About the ASSISTments Foundation. ASSISTments. (n.d.). https://new.assistments.org/about

Ali, R., & Nath, S. (2023, June 20). *Pre-service teachers' experiences of remote online learning: reimagining teacher education post-pandemic*. Taylor & Francis Online.

Atlassian. (n.d.). What is agile? https://www.atlassian.com/agile

Develop. (n.d.). https://www.etrialstestbed.org/run-a-study/develop/

Accelerated Container Application Development. Docker. (2024, January 23). https://www.docker.com/

Dong, M. (2023). Implementing New Features and Quality-of-Life Additions Within E-TRIALS. : Worcester Polytechnic Institute.

Dong, M., & Oliveira, M. (1213). Developing Features and Maintenance Improvements for the E-TRIALS Platform. : Worcester Polytechnic Institute.

Gorelik, E. (2023). Developing Quality of Life Features for the E-TRIALS Platform. : Worcester Polytechnic Institute.

Introduction. • Mirage JS. (n.d.-a). https://miragejs.com/docs/getting-started/introduction/

Kim, J. (2022). Improving ASSISTments E-TRIALS Service. : Worcester Polytechnic Institute.

Krichevsky, N., & Spinelli, K. (2020). E-TRIALS: Developing a Web Application For Educational Research. : Worcester Polytechnic Institute.

McCarthy, T. (2021). Continuing the Development of E-TRIALS. : Worcester Polytechnic Institute.

Philippo, E., & Spofford, M. (2022). Preparing the E-TRIALS Minimum Viable Product for Release: Improving Platform Usability, Security, & Infrastructure. : Worcester Polytechnic Institute.

Spillane, P. (2022). Improving the E-TRIALS User Experience. : Worcester Polytechnic Institute.

Spring framework 6.1.4. Spring Framework. (n.d.). https://spring.io/projects/spring-framework

Vite. (n.d.). https://vitejs.dev/guide/

Vue.js. Introduction | Vue.js. (n.d.-a). https://vuejs.org/guide/introduction.html

7. Appendix

Appendix A: Jira Tickets Completed

Ticket Title	**Description**
[ET-303] Add problem set ID to search filter	Investigated existing shape of data for problem set information, and worked with lead dev to find optimal data structure for problem set ID. Adjusted code to work with new data structure, and identified bug to successfully allow search by problem set ID on the admin table. Contributed to mock backend by creating mock seeds and endpoints to simulate the admin table.
[ET-304] Frontend: New end point for getting Skills	Changing skills endpoint, renaming for migration, creating mirage endpoint and skills data
[ET-308]v1 Removed Autogenerated OSF Project ID	Removed Autogenerated OSF Project ID
[ET-308]v2 Backend: Update Problem Set Usage Table	Replace problem_set_id to problem_set_ceri. Get the list of the problem sets from the new CAS content database.
[ET-318] Backend: Update WIP_Problem_Study	Updated the etrials_wip_problem_set_study to replace problem_id with problem_ceri and use the post-migration CERIs
[ET-319] UtilTools: Update WIP_Problem_Study	Updated the etrials_wip_problem_set_study to replace problem_id with problem_ceri
[ET-338] Backend: Update problem_experiment	Updated problem_experiment to replace problem_set_id with problem_set_ceri. Fixed errors that came with implementing new data structure

[ET-371] UtilTools: Update Problem Set Usage Table	Use UtilTools to generate new files to replace pre existing files in EtrialsService that needed different attribute(?) name
[ET-372] Update `StudyController.loadUserStudies()` to use new `StudyAdminDTO` return value	Backend implementation of StudyController.loadUserStudies returns the newer StudyDTO data structure instead of the old StudySummary. StudyDTO has more data than StudySummary and reduces the need for additional API calls to track things like participant count
[ET-373] Update frontend studyList store to use StudyAdminDTO instead of StudySummary	Implemented StudyDTO in the front-end to be used in place of StudySummary. This change allows for the api calls to become more streamlined and less redundant.