



## UEF25-014 Engineering Navigator: AI-Enhanced Engineering Opportunities Search Bot Final Report

### Project Scope

DiscoverE received a grant of \$150,000 to support the development of an AI-enhanced search tool that finds and shares engineering-related opportunities for students from PreK through 12th grade. Collaborating with an advisory board of education and industry stakeholders (ASCE, ASME, ASCE, and IEEE, among others), DiscoverE and our AI partner, Creative 2, spent 2025 designing, testing, and iterating to prove the hypothesis that an AI bot can improve access to engineering education experiences for early-grade learners, their parents, and educators. As of November 24, 2025, the Engineering Navigator is fully operational and prepared to begin its second year of development, with a soft launch scheduled for January.

The Engineering Navigator uses generative AI to search the internet, extract program details, and organize them into a centralized, user-friendly platform. With filters for location, topic, age range, and date, the dashboard is modeled on familiar tools like Airbnb and Zillow. It is accessible to all users—regardless of technical fluency—and provides a one-stop resource for discovering engineering-related learning experiences.

### Completed Deliverables

Over the past year, DiscoverE successfully completed Phase 1 (Scraper Bot) and Phase 2 (Learning Bot) of the Engineering Navigator project, marking a full proof-of-concept and technical validation of the core AI engine.

Through a carefully sequenced series of sprints, DiscoverE and Creative 2 have developed a scalable technical foundation, established hosting and security parameters, built administrative and public-facing interfaces, trained the bot to scrape and parse event data, and enhanced the system to crawl the broader web. The system is now operational, tested, and refined, positioning DiscoverE to continue iterative improvements, partner onboarding, and data validation in the upcoming year.

### Phase 1: Scraper Bot Development

Phase 1 centered on building the foundational technical architecture and AI ingestion pipeline that powers the Engineering Navigator. This phase was focused on building a system capable of accurately identifying, extracting, validating, and structuring STEM event data from diverse partner websites while minimizing noise, processing inefficiencies, and server costs.

At the core of Phase 1 is a multi-stage intelligent parsing pipeline that combines traditional web scraping with generative AI interpretation. When a URL is submitted, the system evaluates whether the page contains direct event data, links to event listings (such as calendars or multi-event collections), structured data feeds (API, RSS, XML), or recognizable CMS frameworks; from there, recognized site patterns are flagged and routed through the most effective extraction pathway automatically.

## **Phase 2: Learning Bot Development**

Phase 2 transformed the Scraper Bot into an autonomous Learning Bot capable of independently discovering, evaluating, and classifying STEM opportunities at scale across the broader web. Phase 2 introduced system intelligence, scalability, and pattern recognition that allowed the platform to move toward self-directed discovery and continuous learning. The focus of this phase was to move beyond partner-provided websites and enable the system to: identify new STEM-related sources across the internet; interpret increasingly complex and inconsistent site structures; and scale parsing operations while maintaining relevance, accuracy, and cost control.

The Learning Bot operates by generating search queries, identifying potential opportunity sources, evaluating relevance signals, and feeding validated URLs back into the parsing pipeline for extraction, normalization, and classification. This phase also formalized the system's ability to more autonomously classify ambiguous results, suppress noise, and intelligently determine whether content qualifies as an "engineering opportunity" as defined by DiscoverE.

## **Advisory Committee Engagement**

Over the course of multiple meetings, the advisory committee members—including ASEE, ASME, ASCE, IEEE, Campbell University, Museum of Science–Boston, Minnesota State Engineering Center of Excellence, MathWorks, and the STEM Education Coalition—shaped: data field definition; treatment of ambiguous terminology; search and filtering rules; expectations for the Navigator's output.

## **Lessons Learned**

- Normalizing fragmented opportunity data into a single structured database unlocks downstream capabilities – impact measurement, long-term program planning, missing areas of need, etc., potentially positioning the Engineer Navigator not only as a search and delivery platform, but as a critical component for national STEM access strategy.
- Event definition consistency is harder than topical event detection; the most recurring accuracy problem was not finding related STEM/engineering opportunities, but aligning inconsistent real-world content to a single, enforceable definition of what qualifies as an "engineering opportunity" for this specific audience.
- STEM opportunity pages rarely follow predictable structural standards, requiring dynamically adaptive extraction logic instead of fixed rule-based scraping. Partner and university websites varied much more dramatically than originally anticipated. A large majority of sites were invalid (syntactically) or not following common web patterns or standards.
- Relative website content was not nearly as helpful as anticipated in determining on-page event topic confidence scores – showing a low percentage increase in accuracy.
- Date structures were inconsistent, requiring normalization and iterative training for semantics and time-specificity
- Lower-quality sources/sites skewed training sessions significantly.
- Keyword filtering fails rapidly at large scale, semantic interpretation and contextual reasoning are required to prevent massive false-positive contamination.
- Prompt drift is real and measurable when including as much content as is required to parse an entire web page of data – chunking and re-stitching were required.
- Dual-branch model testing revealed that extraction methodology and previously learned context matters more than model specifics and direct page content.