



Generative AI Campus Assistant

ELECOMP Capstone Design Project 2026-2027

Sponsoring Company:

Amazon Web Services (AWS)

410 Terry Avenue North, Seattle, WA 98109

Website: <https://aws.amazon.com>

AWS Public Sector: <https://aws.amazon.com/government-education/>

Company Overview:

Amazon Web Services (AWS) is the world's most comprehensive and broadly adopted cloud platform, offering over 200 fully featured services from data centers globally. Millions of customers—including the fastest-growing startups, largest enterprises, and leading government agencies and universities—use AWS to lower costs, become more agile, and innovate faster. AWS provides a deep portfolio of services spanning compute, storage, databases, analytics, machine learning and generative AI, Internet of Things (IoT), robotics, and edge computing.

AWS is deeply committed to education and workforce development. Through this ELECOMP Capstone sponsorship, AWS aims to give URI's senior Electrical and Computer Engineering students hands-on experience building real-world systems powered by generative AI—mirroring the architectures that power modern intelligent products.



Technical Director(s):

Fang Cong

Sr. Solutions Architect, AWS WWPS SLG/EDU East

fcong@amazon.com

Brandon Artz

System Dev Engineer, RDPI Hardware Solutions, Amazon

branartz@amazon.com

Consulting Technical Director(s):

Hui Lin

Associate Professor

Electrical and Computer Engineering Department, University of Rhode Island

huilin@uri.edu

Project Motivation:

Generative AI is reshaping how software is built and how people interact with information, yet most engineering graduates leave school having only used AI tools—not having built systems on top of them. This project is designed to close that gap. It gives URI's ELECOMP seniors direct, hands-on experience architecting a production-grade generative-AI application on a leading cloud platform, developing precisely the skills employers are urgently hiring for.

The project is deliberately designed to cover the full span of applied GenAI:

- **Real-world relevance and workforce readiness.** Retrieval-Augmented Generation (RAG) is one of the most in-demand architectural patterns in the industry today. Students will graduate fluent in the exact architecture powering modern enterprise AI products—from document ingestion and semantic search to conversational AI with guardrails and citations.
- **Full-stack, interdisciplinary engineering.** This project exercises the complete cloud-native software stack: serverless compute (Lambda, API Gateway), managed AI services



(Amazon Bedrock, OpenSearch), authentication (Cognito), frontend deployment (Amplify), and responsible-AI practices (guardrails, grounded citations). Students build production-grade software end to end.

- **Lasting institutional value.** The Campus Assistant is an immediately useful tool the College of Engineering can continue to operate after the capstone year. Over two semesters, the team builds the core system first and then iteratively improves it—integrating additional data sources, expanding to other campus systems, and refining the experience based on real usage data.
- **A reusable foundation.** The resulting RAG-on-Bedrock architecture is directly applicable to any institution or organization that needs a grounded, cited AI assistant over its own documents. Published as a blog and open-sourced, it can be adopted by other universities and teams—amplifying the project's impact well beyond URI.

Publication & open-source opportunity: Students will have the opportunity to co-author a technical blog post with their AWS mentors documenting the project and its results, with the goal of publishing it on the AWS website (e.g., the AWS Public Sector or Architecture blog). The resulting solutions can also be open-sourced so that other universities and organizations can adopt and benefit from the students' work—amplifying the impact of the capstone beyond URI.

Anticipated Best Outcome:

A production-quality conversational assistant for URI's College of Engineering that answers student and faculty questions about courses, program requirements, lab schedules, department policies, and campus events—grounding every response in authoritative university documents using Retrieval Augmented Generation (RAG) and citing its sources. The best outcome is a deployed, authenticated web application that achieves at least 80% answer accuracy on a curated evaluation set, responds in under three seconds, and is ready for the College of Engineering to operate.



Project Details:

The team will design and build an AI-powered conversational assistant for URI's College of Engineering. Students and faculty frequently need answers scattered across dozens of university documents—course catalogs, handbooks, lab safety policies, advising guides, and event calendars. This assistant serves as a single, intelligent point of access to that institutional knowledge, while avoiding the hallucination problems of ungrounded chatbots by grounding responses in source documents and citing them.

Project A — Generative AI Campus Assistant (Agentic RAG)

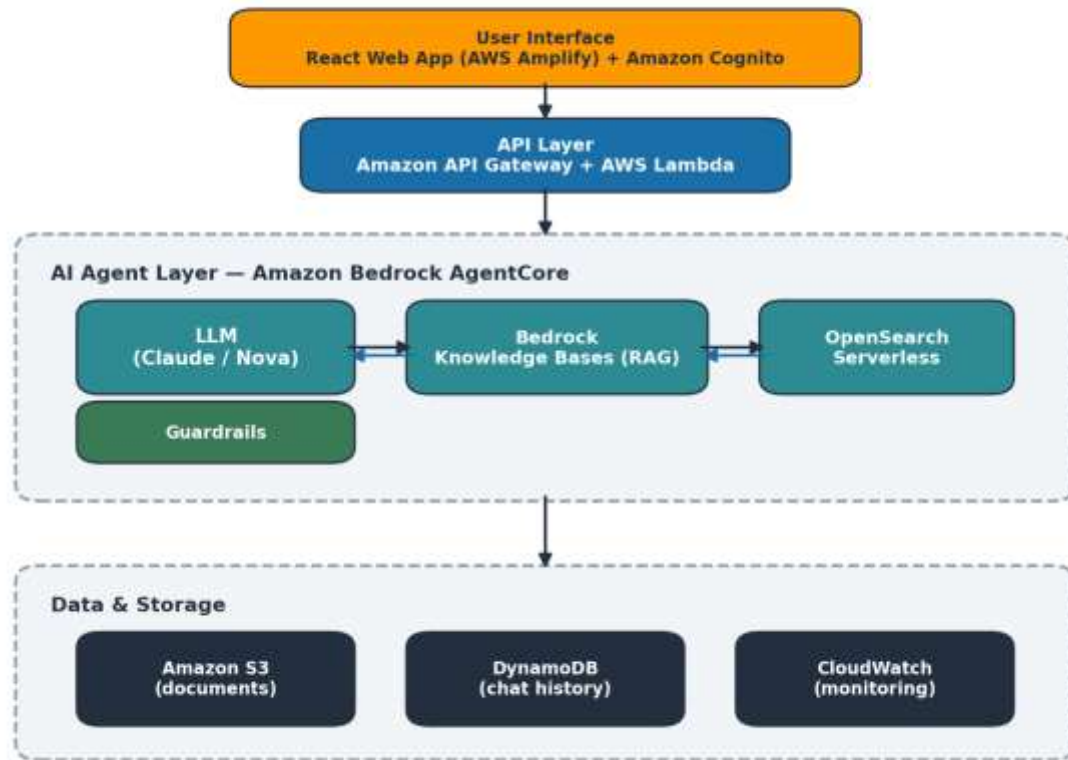


Figure A-1: Generative AI Campus Assistant System Architecture.

The project is organized around the following outcomes:

ABO A-1: Document Ingestion and Knowledge Base- Collect and curate URI College of Engineering documents (course catalogs, handbooks, lab schedules, policies) - Chunk, embed, and index documents into **Amazon Bedrock Knowledge Bases** with a vector store (**Amazon OpenSearch Serverless**) - Store source documents in **Amazon S3**; implement an automated refresh pipeline so new/updated documents are re-indexed on a schedule - Target: at least 50 documents indexed with a validated chunking strategy

ABO A-2: Conversational AI Agent with Grounded, Cited Responses- Build an agentic assistant using **Amazon Bedrock AgentCore** with retrieval actions over the knowledge base - Include a source citation in every response, linking back to the originating document - Apply **Amazon Bedrock Guardrails** to block off-topic or inappropriate queries and reduce hallucination -



Support multi-turn conversation with context retention; target at least 80% accuracy on a 100-question evaluation set

ABO A-3: Web Interface and Deployment- Build a web-based chat interface (**React**) hosted on **AWS Amplify**; authenticate users via **Amazon Cognito** (URI email login) - Store conversation history in **Amazon DynamoDB**; expose backend logic via **AWS Lambda** and **Amazon API Gateway**- Provide a feedback mechanism (thumbs up/down); target sub-3-second response time and mobile-responsive design

ABO A-4: Evaluation Framework and Continuous Improvement- Build an automated evaluation harness that scores answer accuracy and citation correctness against a curated question set. Establish a continuous-improvement loop feeding evaluation results back into prompt and retrieval tuning

ABO A-5 (Stretch): Voice Interface- Add speech-to-text input and text-to-speech output (**Amazon Transcribe** + **Amazon Polly**) for accessibility—reusing the voice pipeline developed in the companion AWS project, the Voice-Controlled Autonomous Robot

Composition of Team:

This project is delivered by a team of **2 Computer Engineers**.

Skills Required:

Students entering this project should have, or be prepared to develop, the following skills. All 3 Technical Directors will provide guidance and enablement on the cloud- and AI-specific areas.

- **Programming:** Proficiency in Python and JavaScript/TypeScript; familiarity with Git and collaborative development
- **Cloud & Generative AI:** Interest in (and willingness to learn) Amazon Bedrock, RAG, prompt engineering, and serverless architecture (Lambda, API Gateway, DynamoDB)
- **Web development:** React front-end development; REST API integration; basic authentication concepts

AI Usage:

The AI Use Policy for this project is defined in Appendix A at the end of this proposal. Technical Directors: complete the checklist in Appendix A to indicate which AI uses are permitted on this



project. Any item left unchecked is not allowed.

Anticipated Best Outcome's Impact on Company's Business, and Economic Impact

The anticipated best outcomes create economic impact along several dimensions:

- **Talent pipeline.** The project produces graduates fluent in Amazon Bedrock, RAG, serverless, and AI—skills in acute demand across AWS and its customers and partners. This strengthens the hiring pipeline for AWS and the broader cloud economy.
- **Reference architectures and reusable assets.** The Campus Assistant (RAG on Bedrock) is a reusable reference implementation. Published as a blog and/or open-sourced, it lowers the cost for other institutions and customers to adopt the same pattern, accelerating AWS service adoption.
- **Brand and community impact.** A co-authored AWS blog post and a demonstrable Spring Summit showcase raise AWS's visibility in higher education and provide proof points reusable across the education field. The open-source solution extends the economic benefit to the wider community.

Broader Implications of the Best Outcome on the Company's Industry:

Beyond AWS, the outcomes have implications for the cloud, AI, and education-technology industries:

- **Advancing responsible, grounded AI in higher education.** The Campus Assistant models a responsible pattern—retrieval-grounded answers with citations and guardrails—that directly addresses higher education's concerns about hallucination and trust in AI. A published, open-source blueprint helps set a practical standard other institutions can follow.
- **Democratizing applied generative AI.** By open-sourcing the solutions and documenting them in a public AWS blog, the work lowers the barrier for smaller universities and organizations—those without large AI teams—to deploy their own grounded assistants and LLM-driven automation, broadening participation in the AI economy.
- **Workforce transformation.** Producing engineers who can *build* on generative AI (not merely use it) helps the industry close a critical skills gap, informing how engineering curricula and industry-academia partnerships evolve to prepare the next generation of AI-fluent engineers.



Appendix A: AI Use Policy – Generative AI Campus Assistant

Allowed

Students may use AI tools for

- ☒ brainstorming support
- ☒ concept development
- ☒ literature and publicly available documentation review
- ☒ software development
- ☒ debugging
- ☒ data analysis
- ☒ test development
- ☒ non-proprietary paragraph editing
- ☒ non-proprietary single-slide formatting
- ☒ preparation of project deliverables

Not Allowed

- Note taking or transcription of meetings
- Using AI in violation of university policies, applicable laws, or course-specific rules identified by the capstone instruction team and project-specific rules identified by the technical directors.
- Entering proprietary, confidential, personally identifiable, security-sensitive, or otherwise restricted information into an AI system without authorization.
- Submitting AI-assisted work without understanding, reviewing, verifying, and appropriately disclosing its use.
- Submitting entire documents or presentations to an AI system for editing or formatting.
- Any unchecked items from the allowed list above.

Requirements

- If any use is allowed, AI tools may only be used to enhance, but not replace, students' technical reasoning, engineering judgment, and understanding. Students must be able to explain, justify, and defend all submitted designs, code, analyses, test results, documentation, and conclusions.
- During the first two weeks of the project, the student team shall work with the Technical Directors to: identify any proprietary or confidential information that may be involved in the project; determine what information may or may not be shared with AI systems; identify the AI tools, models, accounts, and configurations approved for project use; and document any project-specific restrictions or approval requirements.



- Students shall transparently document the uses of AI using the URI CYPHER AI Usage Marking guidelines (<https://web.uri.edu/cypher/ai-marking/>). A workshop is planned for September to familiarize students with these guidelines.
- Students retain full professional responsibility for all design decisions, analyses, implementations, testing, documentation, presentations, and final deliverables, regardless of whether AI tools contributed to the work.