Embrace change: The future of software engineering is GenAI

Brillio's GenAI solutions solves long development cycles with robust responsiveness using automated code generation, bug fixes, and testing to streamline architecture development.

Traditional Software Development **Lifecycle (SDLC)**

The traditional SDLC has long been the backbone of software engineering, providing a robust framework for testing and deploying software applications. However, as technology evolves and the demand for rapid, efficient, and adaptive software solutions increases, the limitations of conventional SDLC methodologies are becoming more apparent. Challenges like prolonged development times, rigid phase transitions, and difficulties accommodating changing requirements often hinder the agility and responsiveness required for today's forward-thinking digital enterprises.

GenAI has emerged as a massive boon in software engineering. Organizations can automate and enhance various aspects of the SDLC, from requirement analysis to code generation and testing. This integration accelerates development processes and introduces adaptability and intelligence previously unattainable with traditional methods. In the subsequent sections, we will explore how Brillio's GenAI solutions pave the way for a more efficient, flexible, and intelligent software development process.

Here's a comparison table of routine SDLC activities and the challenges they pose:

RFP and Proposals • Conduct spend analysis • Define project scope • Prepare documentation • Establish SLAs, negotiate and create contracts.	**Challenges** • Determining ROI • High cost and effort involved in RFP preparation • Communication gaps between stakeholders • Difficulty in providing accurate success metrics and milestones
Requirement Analysis • Understand project requirements • Create scope documents • Communicate requirements to relevant teams • Manage scope changes throughout the project	**Challenges** • Difficult articulating requirements from end users • Time-consuming process to design and validate wireframes and user experience • Undocumented processes and requirements
Solution Architecture • Design high-level architecture diagrams based on requirements • Map technical dependencies between modules • Conduct discussions and validations with stakeholders	**Challenges** • Long and complex process to validate architecture with multiple stakeholders • Difficulty in mapping requirements to architecture designs • Challenges in deploying cloud architecture (both technically and in terms of effort)
Development • Break architecture design into individual modules • Code and integrate functional modules • Ensure security and fix loopholes in the code • Incorporate ongoing changes as needed	**Challenges** • Complex process to deploy the CI/CD pipeline • Time-consuming re-engineering of legacy code • Tedious and repetitive effort for security checks in source code

Testing

- Develop unit test cases for each module
- Build functional test cases for each component
- Create test cases for each user use-case

Challenges

- Low-code coverage in both unit and functional test cases
- Difficulty in translating use-cases into valid tests
- Lack of detailed contextual data for effective test validations

What must organizations consider **before adopting GenAI?**

GenAI's foray into the SDLC offers significant value, but successful implementation requires a thoughtful, strategic approach. Here are seven critical areas to maximize impact and mitigate risks:

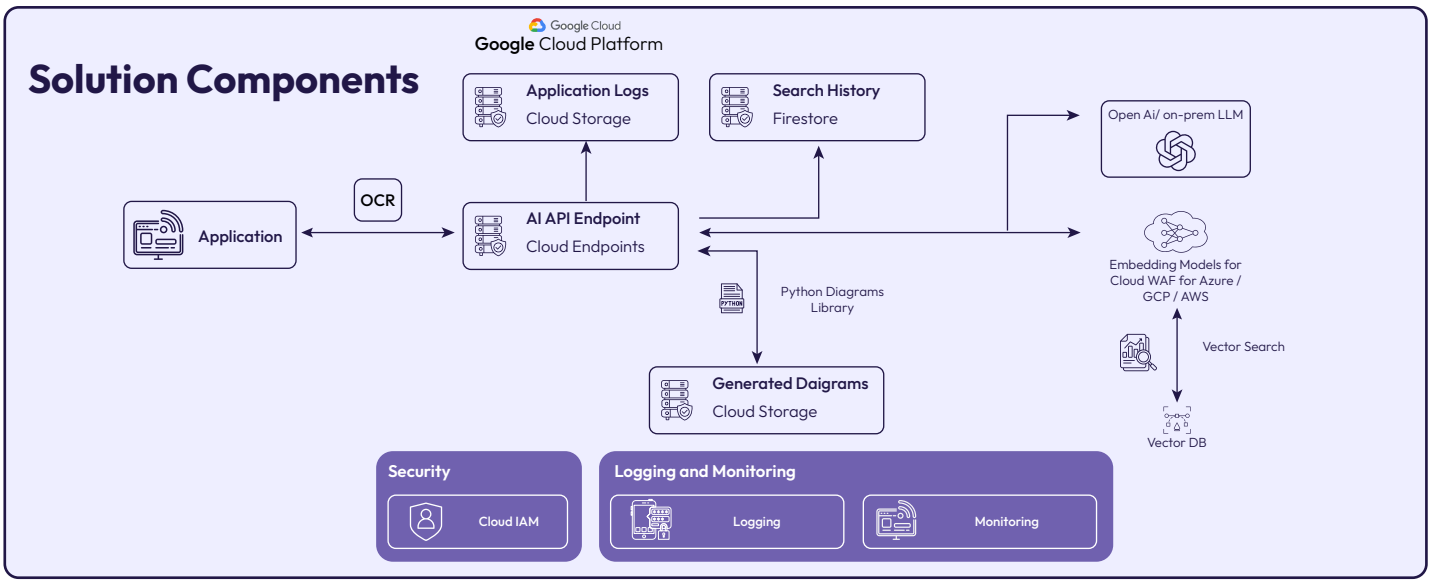
1. **Intellectual property protection:** To protect IP and data, consider deploying GenAI solutions on-premises and audit third-party AI tools that leverage open-source libraries or infrastructure.
2. **Data governance:** Ensure transparency in how first and third-party data is collected and used for training AI models (including source code and other proprietary information).
3. **Accuracy and bias decisions:** Strike the right balance between precision and recall when training GenAI models with diverse datasets through continuous feedback and model tuning.
4. **Economics:** Measure GenAI's effectiveness using engineering productivity metrics after deployment to get a clear picture of ROI and operational benefits.
5. **Change management:** Democratize knowledge to empower leaders and SMEs with robust change enablement through the transition.
6. **Organizational readiness:** Devise a structured roadmap, including governance frameworks, technical readiness, and cultural alignment when introducing AI solutions.
7. **Regulatory compliance:** Instill guardrails for AI models to ensure fairness, strict model governance, and strong ethical considerations.

Brillio's GenAI solutions for **software engineering**

Accelerate software engineering initiatives by integrating GenAI into the DevSecOps pipeline with our five offerings.

- **Architecture Validation and Recommendation**
- **Architecture Deployment**
- **Application Modernization**
- **DevSecOps Pipeline Generation**
- **Testing (Unit and functional testing and test data generation)**

Architecture Validation and Recommendation



Streamline the architecture validation process by automating the assessment, recommending optimizations, and enhancing collaboration between architects and engineering teams.

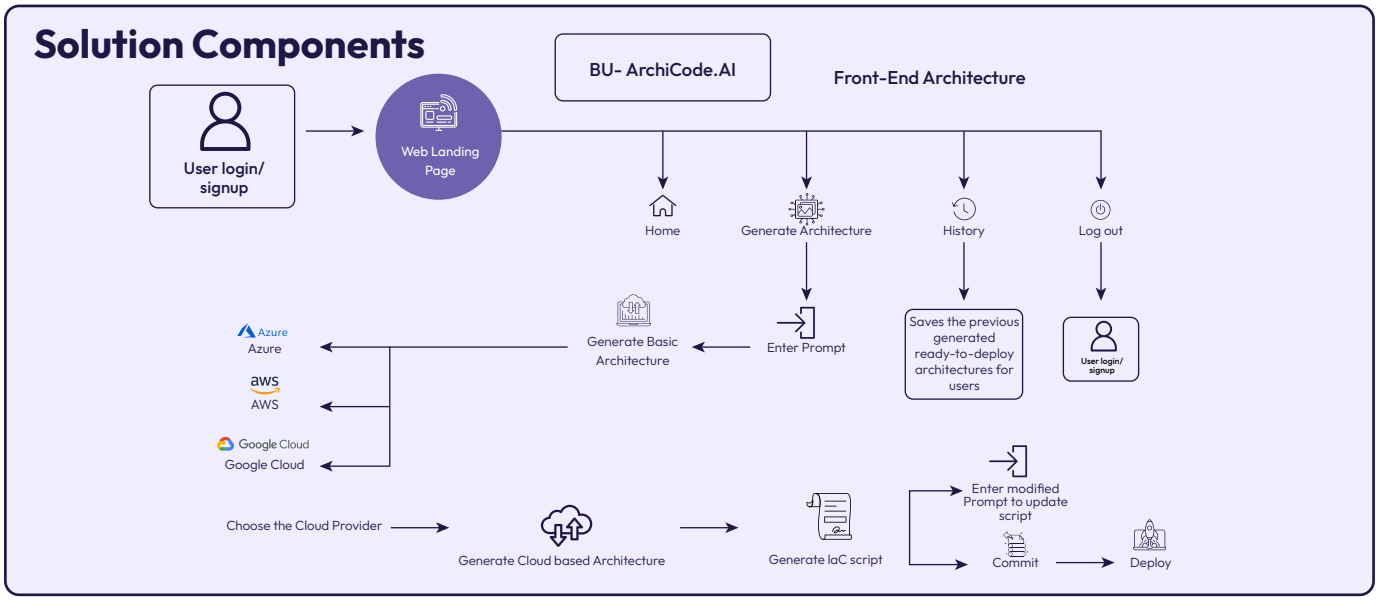
Workflow:

- **OCR for text extraction:** Extracts text from architecture diagrams to identify listed services.
- **Execution of cloud functions:** Uses serverless cloud functions to parse architecture details.
- **Accessing the Well-architected Framework (WAF) model:** Calls a pre-trained WAF model hosted on the cloud to evaluate architecture logic.
- **Suggestions retrieval:** AI model generates recommendations for improvements based on services and architecture.
- **Forwarding to OpenAI:** Suggestions are enhanced using OpenAI for further validation and refinement.
- **Validation and enhancement:** Suggestions are evaluated using Azure OpenAI, incorporating feedback from SMEs.
- **Enhanced diagram generation:** Updated architecture is produced in diagrammatic form using Python libraries.
- **Storage on Google Cloud:** Enhanced diagrams are stored with a retrievable URL for future references and collaboration.

Benefits:

- **Automated assessment:** Reduced manual review efforts and accelerates architecture evaluations.
- **Efficiency:** Scales architecture assessment by processing diagrams rapidly and intelligently.
- **Compliance:** Ensures consistent adherence to WAF best practices, minimizing design flaws.
- **Collaboration:** Enables cross-functional teams to review and refine designs more effectively using a shared AI-driven platform.

Architecture Deployment



Streamline cloud architecture creation and deployment using prompt-based automation and AI-powered tooling to significantly reduce design and rollout time.

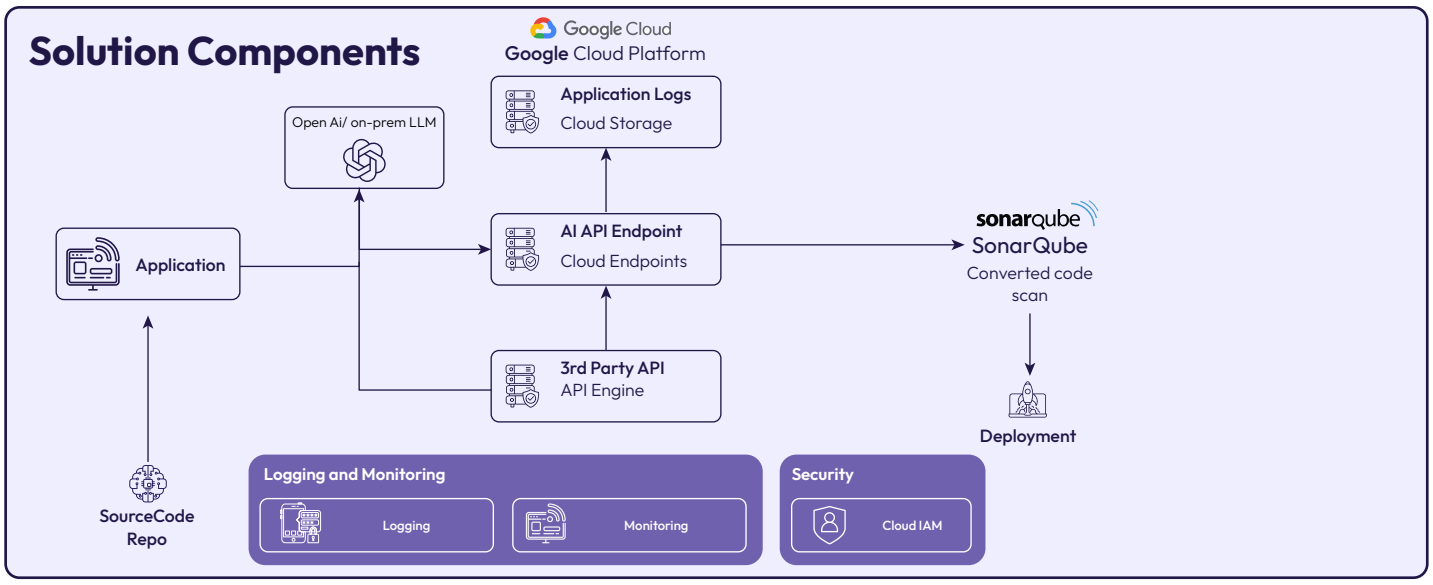
Workflow:

- **Prompt input:** Users enter a simple text prompt outlining the required architecture scope.
- **Azure OpenAI conversion:** Prompt is passed to Azure OpenAI, which generates a text-based architecture diagram.
- **Basic architecture creation:** A base architecture diagram is rendered from the text and passed to Python and Mermaid scripts.
- **CSP selection:** Users select the CSP best suited to their requirements.
- **Cloud-based architecture output:** CSP-specific architecture diagram is generated and refined by Azure OpenAI and Python scripts.
- **IaC script generation:** IaC scripts are generated for the final architecture.
- **Validate and update:** Users can review and update architectural elements as per generated script.
- **Commit and deploy:** Final architecture is deployed on the selected cloud platform.

Benefits:

- **Speed:** Rapid validation and deployment of architectures, significantly reducing time-to-market.
- **Productivity:** Engineers can create and iterate on architectures quickly using a prompt-based interface.
- **Accuracy:** Azure OpenAI accurately interprets scope and transforms it into functional cloud architecture.
- **Automation:** Fully automates cloud resource allocation, scripting, and deployment, reducing manual overhead.

Application Modernization



Automate and accelerate legacy application transformation, ensuring consistent, accurate, and production-ready code conversion using LLMs and custom workflows.

Workflow:

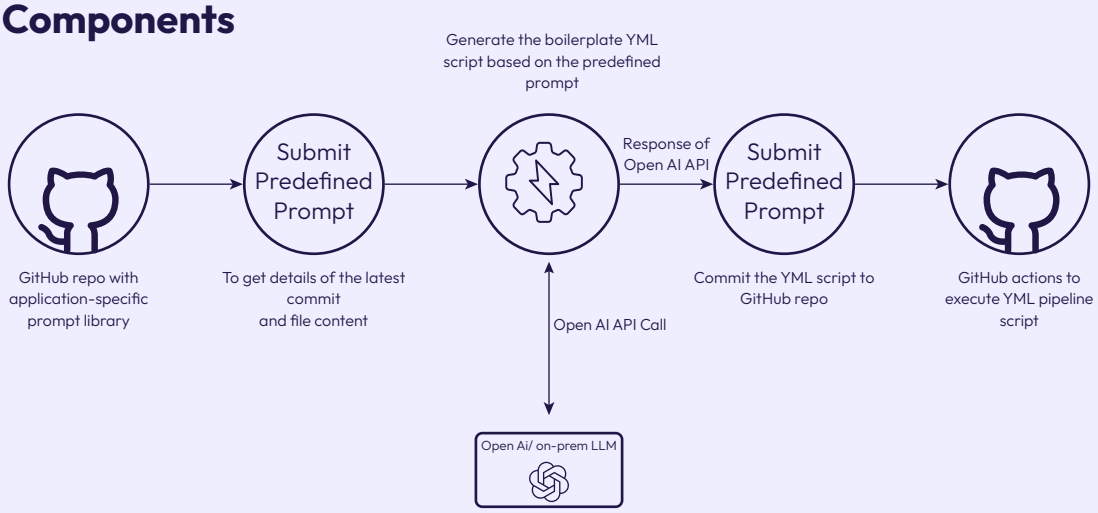
- **Source setup:** Users define the source and destination repositories for one-time code conversion setup.
- **Code identification:** Selected files are identified for conversion using our custom conversion engine.
- **Conversion process:** Multiple conversion path is selected, and the code is sent to Azure OpenAI or an on-prem LLM for processing.
- **Validation via SonarQube:** Post-response, the converted code can be validated using SonarQube for code quality and security.
- **Push to GitHub:** After validation, the application is production-ready and pushed to the target GitHub repository.

Benefits:

- **Consistency:** Maintains functionality and behavior across codebases for seamless transitions.
- **Speed:** Automates repetitive tasks and accelerates feature releases, reducing time to market.
- **Productivity:** Frees developers from manual refactoring, letting them focus on strategic development.
- **Accuracy:** Advanced ML and GenAI techniques ensure reliable and efficient code conversion.

DevSecOps Pipeline Generation

Solution Components



Automate the creation of secure CI/CD pipelines by generating YAML scripts through intelligent prompts, enabling rapid, secure, and customizable DevSecOps implementation.

Workflow:

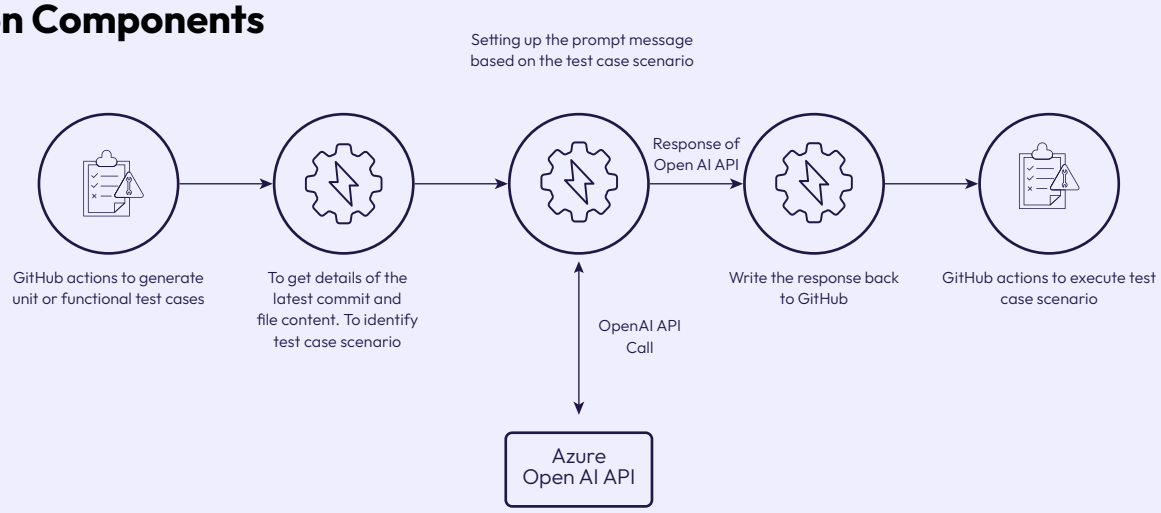
- **Prompt definition:** User defines the pipeline generation scope using application-specific prompts.
- **Submit to OpenAI:** Prompts are passed to OpenAI, which returns a YAML template with the necessary DevSecOps configurations.
- **OpenAI output:** Customized YAML is generating with GenAI, based on the defined pipeline structure.
- **Pipeline creation:** YAML is pushed to GitHub, where the actual CI/CD pipeline is set up.
- **Script optimization:** LLM refines the generated scripts with resuable components from GitHub across code repositories.

Benefits:

- **Speed to market:** Significantly accelerates pipeline setup, increasing overall delivery velocity.
- **Rapid development:** Reduces time spent to build, test, and deploy pipeline configuration.
- **Customization:** Offers high flexibility with predefined and user-specific prompts for tailored pipelines.
- **Integration:** Seamless integration with GitHub, GitLab, Azure, DevOps, and AWS for automated deployments.

Testing-as-a-Service

Solution Components



Automate the generation, execution, and validation of unit and functional test cases with Azure OpenAI to boost productivity, accuracy, and test coverage in CI/CD pipelines.

Workflow:

- **Prompt setup:** User provides prompts for unit or functional test case generation tailored to the application stack.
- **Trigger points:** Code changes are detected using GitHub integrations to identify when tests are needed.
- **OpenAI call:** Customized prompts are sent to Azure OpenAI, which generates the test cases.
- **Test case generation:** Generated cases are written back to the repository automatically.
- **Validation and execution:** Test cases are run via GitHub actions, validated and committed to the codebase.
- **Reporting:** Reports are generated for each test stage and published for review.

Benefits:

- **Productivity:** Saves developer time and effort with automatic test case generation.
- **Quality:** Improves quality and coverage by generating realistic and diverse test data.
- **Accuracy:** Predicts potential defects or failures based on past data and patterns.
- **Integration:** Integrates test case generation in CI/CD pipeline.

Securing the future of software **delivery with GenAI**

GenAI is transforming what is possible in software engineering by accelerating delivery, expanding test coverage, and reducing costs throughout the SDLC. However, fully harnessing this potential requires more than simply having the right tools; it demands a strategic approach that balances innovation with governance, risk mitigation, and measurable outcomes. Organizations that excel in this area will address today's engineering challenges and set the standard for the future of software delivery. Are you interested in discovering how to leverage GenAI for your engineering organization? Connect with us to learn more.

About Brillio

Brillio is one of the fastest growing digital technology service providers and the partner of choice for many Fortune 1000 companies seeking to turn disruptions into competitive advantages through innovative digital adoption. We help clients harness the transformative potential of the four superpowers of technology: cloud computing, Internet of Things (IoT), artificial intelligence (AI) and mobility. Born digital in 2014, we apply our expertise in customer experience solutions, data analytics and AI, digital infrastructure and security, and platform and product engineering to help clients quickly innovate for growth, create digital products, build service platforms, and drive smarter, data-driven performance. With 14 locations across the US, the UK, Romania, Canada, Mexico, and India, our growing global workforce of 6,000 Brillians blends the latest technology and design thinking with digital fluency to solve complex business problems and drive competitive differentiation for our clients. Brillio was certified by Great Place to Work® in 2021, 2022, 2023, and 2024.

https://www.brillio.com/
Contact Us: info@brillio.com