**A blue and red background with white text

Description automatically generated**

**Generative AI**

*The information contained in this document is not to be used for any purpose other than the purposes for which this document is furnished by GENPACT, nor is this document (in whole or in part) to be reproduced or furnished to third parties or made public without the prior express written permission of GENPACT.*

[1. Introduction 3](#_Toc156913578)

[2. Technologies Adoption 3](#_Toc156913582)

[3. Solution Overview 3](#_Toc156913583)

[4. Business Use Case 5](#_Toc156913584)

[4.1 Software Development with GenAI: An SDLC Automation Solution 5](#_Toc156913589)

[4.1.1 Overview 5](#_Toc156913590)

[4.1.2 Key SDLC Challenges 5](#_Toc156913592)

[4.1.3 Implementation of Automated Generation of User Stories, Code, and Test Cases 5](#_Toc156913593)

[4.1.4 User Experience 6](#_Toc156913594)

[4.1.5 Key Benefits 7](#_Toc156913595)

# Introduction

# In today's fast-paced and ever-evolving technological landscape, the need for efficient and streamlined processes has become more crucial than ever. As organizations strive to stay ahead of the competition, they are constantly seeking ways to optimize their software development life cycle (SDLC) and increase productivity. With GenAI, businesses can now achieve faster and more accurate software development, reducing costs and increasing efficiency.

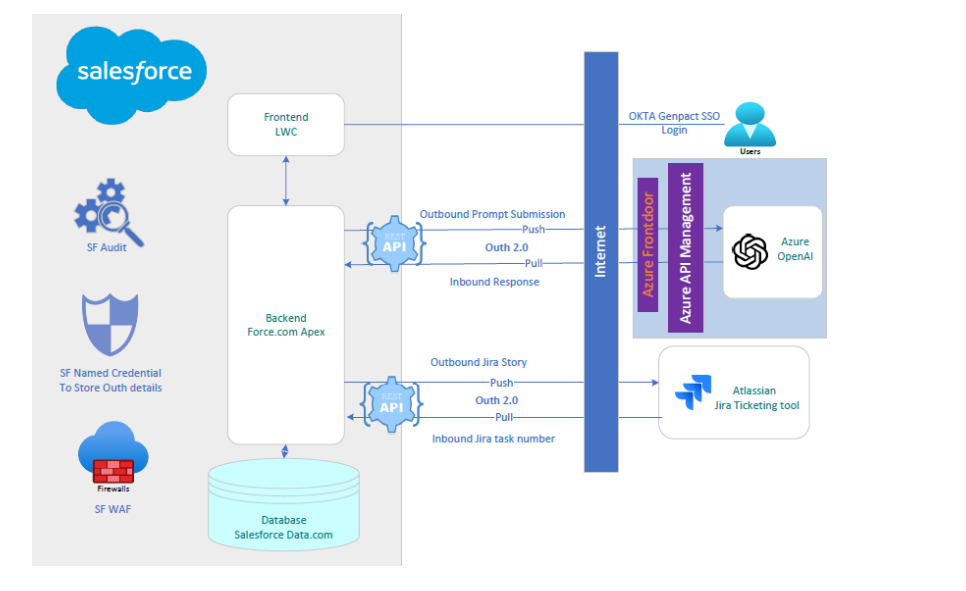
# In this white paper, we will explore the use case of GenAI in SDLC automation which is an innovative solution designed to automate the software development lifecycle (SDLC) process on the Salesforce platform. The GenAI powered app will help developers to automatically create and manage Jira stories, generate code, and test cases within the platform.

# We will delve into the business case for implementing GenAI, its benefits, key business challenges, benefits, and potential impact on the bottom line.

# Technologies Adoption

|  |  |
| --- | --- |
| User interface | Salesforce LWC (Lightning web component) |
| Generative AI tools/platform | AZURE Open AI |
| Deployment Server | AZURE Cloud |
| Backend API | Salesforce (API Version 59.0), Atlassian (version: 1001.0.0-SNAPSHOT) |
| AZURE Open AI /AI Stack | Azure Open AI (Turbo 3.5) |
|  | DB Service: Salesforce Object data model for application and chat data |

# Solution Overview



The above is the architecture outline.

**Solution Overview**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

The Generative AI App has been designed with a modular approach to provide maximum flexibility and scalability to developers. The app is built on the Salesforce platform using Apex programming and Lightning web component UI framework, leveraging the latest AI technology Microsoft AZURE OPEN AI to provide advanced automation capabilities. Each module works independently of the others, allowing developers to choose the modules that best fit their needs.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* **Generate JIRA Stories**: This module allows users to create and manage their JIRA stories within the Salesforce platform. The functionality automates the process of generating JIRA stories based on the user's input.
* **Generate Code**: This module generates code from natural language descriptions or requirements. The user can choose the programming language they want to generate code.
* **Generate Test Cases**: This functionality allows users to automatically generate test cases for their code using GenAI. The module analyses the code to determine the different test cases required for the software development task.

The Salesforce LWC web app provides the user experience, presentation, context, and user interaction.

**SDLC Automation Solution GenAI Implementation details**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* User can input details about the software development project, including its scope, goals, deliverables, and Jira stories in terms of business requirements.
* The GenAI-powered application creates a backlog of software development tasks along with their dependencies, generates code and test cases for software development and testing.
* GenAI created JIRA stories are made available to view and edit. Users can download in csv or upload to a JIRA project.
* User can generate code from GenAI based on the input programming language chosen by the user from Language Dropdown.
* Users can generate test cases from GenAI which are made available to view and edit. Users can download test cases in CSV/JSON/XML formats.

# Business Use Case



# Software Development with GenAI: An SDLC Automation Solution

## Overview

## Software Development Life Cycle (SDLC) is a critical process in the creation of high-quality software, encompassing various stages from conception to deployment. However, traditional SDLC approaches often face several challenges that hinder the efficiency and effectiveness of the development process.

## Key SDLC Challenges

* + - **Time-Consuming Processes:** Manual execution of tasks, such as generating user stories, creating BDD feature files, generating test data, and writing code, contributes to time delays in project timelines.
    - **Lack of Consistency:** Manual processes are prone to human error, leading to inconsistencies in documentation, test data, and code, which can result in quality issues and project setbacks.
    - **Scalability Concerns:** As projects grow in complexity, managing the increasing volume of tasks and documentation becomes challenging, often leading to scalability issues**.**

## Implementation of Automated Generation of User Stories, Code, and Test Cases

* + - **JIRA User Story Automation:** Users can now automatically generate JIRA stories directly within the Salesforce platform using GenAI, breaking away from traditional manual user story creation.
    - **Code Generation Automation:** Users can now automatically generate code using Salesforce GenAI powered app allowing the flexibility to choose their preferred programming language for code generation.
    - **Test Case automation:** Enhanced testing efficiency through GenAI driven test case generation providing the functionality to automatically generate test cases by analyzing the code.

## User Experience

* + - **User Friendly Interface:** The seamless integration of JIRA story creation within Salesforce provides a user-friendly interface, eliminating the need for developers to switch between platforms. This cohesive experience contributes to a smoother and more intuitive workflow.

A screenshot of a computer

Description automatically generated

* + - **Programming Language Flexibility:** Users have the flexibility to choose their preferred programming language for code generation. This customization caters to the diverse needs of development teams and ensures a more personalized and user-centric coding experience.

A screenshot of a computer

Description automatically generated

* + - **Intelligent Test Case Automation:** Users now benefit from the reduction of manual efforts in creating test cases, allowing them to focus on more strategic aspects of software development. The automated test case generation enhances accuracy and thoroughness in the testing phase.

A screenshot of a computer

Description automatically generated

## Key Benefits

* + - **Increased Efficiency in Software Development Processes:** GenAI automates time-consuming and repetitive tasks in the software development life cycle, such as user story generation, test data generation, and code generation. By streamlining and expediting various stages of development, GenAI contributes to faster time-to-market for software products.
    - **Optimized Resource Utilization:** Gen AI powered SDLC automation reduces the reliance on manual inputs, minimizing the chances of human error in tasks such as creating user stories, code writing, and documentation so that human resources can be redirected to more strategic and creative aspects of software development.
    - **Improved Accuracy and Consistency in Code Generation:** GenAI enforces coding standards and best practices consistently across the entire codebase. This standardization ensures that the generated code is of high quality, adheres to organizational coding conventions, and is easier to maintain.
    - **Enhanced Collaboration Between Development and Testing Teams:** Automation of tasks like test data generation ensures that testing teams have access to relevant and diverse datasets, allowing them to work concurrently with development teams. This parallel workflow increases the speed of development and testing cycles, resulting in a more agile and collaborative development process.