Name	Sameer Pandey
Current Job Role	Salesforce Developer
Degree	Bachelor of Technology
Experience	4 Years
Current Employer	Cloud Analogy CRM Specialist Ltd, Noida (July 2019 - Now)
Contact	+919793235939, sameerpandey129@gmail.com
Certifications	**Platform Developer I** **Platform Developer II**
Salesforce Clouds	**Sales, Service, Community**
Programming Languages :	Apex, Javascript, Salesforce lightning Design system, Lightning Component, Lightning Web Component
Web Technologies :	HTML, Javascript, CSS,VisualForce,AURA Framework
Development Tool :	MS- Office,VS code,Eclipse, Intellij Idea, Data Loader
Salesforce Expertise	Apex Script, Triggers, Lightning Aura Components, VisualForce, LWC.
Salesforce Components :	Process Builder, Apex classes, Triggers, Workflows, Dashboards & Reports, Validations rules, Formulas
Salesforce Admin:	Process Builder, Workflows, Approval Custom Reports, Flows, Sharing Setting, Approval Process, Roles and Hierarchy.
Integration	**Quickbooks, Orion, Zoom, QR-Tiger**
Salesforce AppExchangeProducts	Zoom Meeting, Ring Central, HubSpot Connector, Drag N Drop, DLRS, Docusign, FormStack, GetResponse, Intelliverse, PandaDoc, Sharepoint, MailChimp,Conga Composer,Adobe Sign
Database :	SOQL,SOSL
PROJECT-1	**LWC Development : Genpact** 1. During my 9-month tenure as a contractor at Genpact, I held the role of an LWC Developer, primarily responsible for implementing LWC solutions across various projects. One of the notable projects involved the development of a dynamic custom lookup field LWC component.

	This component served as a powerful tool for enhanced data management and interaction within the Salesforce platform. Furthermore, I contributed to the integration efforts between Outlook and Salesforce, enabling seamless synchronization of meetings and events from Outlook to Salesforce. 2. In addition to these significant contributions, I actively participated in the development of multiple LWC components throughout my time at Genpact. These components were designed and implemented to address specific project requirements, offering improved functionality and a seamless user experience. 3. Overall, my role as an LWC Developer at Genpact allowed me to contribute to diverse projects and leverage my expertise in developing innovative solutions using Lightning Web Components. Use : **LWC** component, Apex script, Triggers, Flows, Aura component, Batch class, Integration Role: Salesforce LWC Developer
PROJECT-2	**Zuora-CPQ Billing Project:** 1. In the given project, our primary objective was to automate the billing process using Zuora and CPQ. To achieve this, we employed a combination of custom code-based automations, such as triggers, along with standard tools like Flow. This ensured efficient and streamlined billing operations. 2. Furthermore, we implemented custom Lightning pages that played a crucial role in facilitating the business process of purchasing and managing subscriptions. These pages were developed using Lightning Web Components (LWC) and Visualforce (VF) page frameworks, enabling us to create interactive and user-friendly interfaces. By incorporating an assisted flow, we aimed to simplify the subscription management process for users. 3. Overall, our approach involved a blend of code-based automations, standard tools, and custom UI development to automate billing using Zuora and CPQ effectively. This comprehensive solution addressed the specific requirements of the project and aimed to enhance operational efficiency and user experience. Use : **LWC** component, Apex script, Triggers, Flows, Aura component, Zuora Role: Salesforce Developer
PROJECT-3	**<u>Handwrytten Notes</u>**

	1. During this project, my focus was on developing multiple Visualforce pages to facilitate the process of sending handwritten notes to relevant contacts, accounts, leads, or opportunity owners. This feature allowed users to select from a diverse range of note templates available on a website through the integration of an API. The notes were accessible at a minimal cost, ensuring cost-effective communication. 2. To implement this functionality successfully, I leveraged a combination of Visualforce, Apex classes, JavaScript, and CSS. Visualforce pages provided the foundation for creating dynamic and interactive user interfaces, while Apex classes handled the backend logic and data manipulation. JavaScript was utilized to enhance the user experience by implementing dynamic functionalities and form validation. Additionally, CSS was employed to style the Visualforce pages, ensuring a visually appealing and cohesive design. 3. By utilizing these technologies in synergy, I was able to deliver a solution that enabled users to effortlessly select and send handwritten notes using predefined templates or by composing their own personalized messages. This implementation significantly enhanced the communication process and provided users with a seamless and efficient means of expressing themselves. Use : Visual Force, Apex script, Triggers, Manage Package, Change set Role: Salesforce Developer
PROJECT-4	**FSL - Salesforce** 1. In this project, I played a key role in developing and setting up the Field Service Lightning (FSL) functionality from the ground up. This involved creating various components and triggers to support both the user interface (UI) and backend operations. I successfully developed LWC components and Aura components, which were integrated into the Opportunity Record Detail page to showcase related child object records effectively. 2. Furthermore, I contributed to the project by designing and implementing Lightning Flows, which streamlined the workflow processes. I also established validation rules to ensure data integrity and enforce business logic. As part of the project, I crafted HTML email templates that were utilized for sending email notifications, enhancing communication and keeping stakeholders informed. 3. Overall, my involvement encompassed a wide range of tasks, including UI development, backend triggers, component creation, Lightning Flow implementation, validation rules setup, and email template creation. This comprehensive approach aimed to deliver a robust and efficient

	solution for the project, empowering users and optimizing business processes. Use : Validation rule, Email template creation, Apex script, Triggers, Flows, Aura component Role : Salesforce Developer
PROJECT-5	**<u>Community Project</u>** In this project, I created multiple communities in Salesforce and implemented functionality to display and save data according to the particular community user. Specifically, 1. Created a Lightning component to save and display data based on the user's community. 2. Created a Lightning component to add new cases and display data from the article base. Use : Apex script, Triggers, Flows, Aura component Role : Salesforce Developer
PROJECT-6	**<u>Integration of QRTiger to SF for QR Generation</u>** 1. During this project, I successfully accomplished several tasks that contributed to its overall success. Firstly, I developed a visually appealing LWC component to enhance the user interface of QR. This component played a crucial role in providing an intuitive and interactive experience for users. 2. Additionally, I created Apex classes that effectively handled data and seamlessly facilitated REST API requests. These classes were responsible for sending REST API requests and processing the received data, ensuring smooth integration with external systems. 3. To ensure secure and reliable authentication, I implemented the use of Named Credentials and Auth Provider. Named Credentials allowed for the secure storage and management of authentication information, such as usernames and passwords or OAuth tokens, for external services. The Auth Provider was set up to authenticate and authorize external services or identity providers, adding an extra layer of security to the system. 4. Furthermore, I successfully completed the project's deployments using change sets. Change sets are a deployment tool in Salesforce that allowed for bundling and migrating metadata components, such as Apex classes, triggers, and LWC components, between different Salesforce environments.

	5. Throughout the project, I utilized various technologies and features such as Apex, Trigger, HTTPMock, Named Credentials, Auth Provider, LWC, and Quick Action Button. This diverse set of tools and technologies enabled me to effectively address the project requirements and deliver a robust solution. Use : **LWC** component, Apex script, Triggers, Validation Rule, Aura component Role: Salesforce Developer
PROJECT-7	**<u>Zoom-Salesforce Integration</u>** In this project, one of the key tasks was to integrate Zoom with Salesforce, enabling the synchronization of Zoom call logs into Salesforce. To achieve this integration, I employed the following steps: 1. Zoom Authentication: I utilized named credentials and an auth provider in Salesforce to authenticate with the Zoom API. This authentication was implemented using the OAuth 2.0 authentication method, ensuring a secure and authorized connection between the two platforms. 2. Custom Object: To store the Zoom call log data within Salesforce, I created a custom object. This object was specifically designed to capture and store the relevant information from the Zoom call logs, allowing for seamless data synchronization and easy access within the Salesforce environment. Use : **LWC** component, Apex script, Triggers, Named Credentials, Auth Provider, Validation Rules, Custom Objects and fields Role: Salesforce Developer.
PROJECT-8	**<u>Data Migration: Ant Migration tool</u>** 1. In this project, a crucial task involved migrating data from one Salesforce org to another Salesforce org. To accomplish this, we utilized the Ant migration tool, a powerful tool provided by Salesforce for automating deployments and data migration. 2. Using the Ant migration tool, we successfully migrated various components, including objects, fields, record types, page layouts, Aura component bundles, Apex classes, and flows. This comprehensive approach ensured a smooth transition of all essential elements from the source org to the target org. 3. The migration process involved defining the necessary metadata components in the deployment package and leveraging the Ant migration tool's capabilities to deploy them efficiently. The tool allowed

	us to specify the specific components to migrate, validate the deployment, and perform the actual migration with ease. Role: Developer & Metadata Import/Export
PROJECT-9	**<u>Custom lightning component development</u>** As the lead developer in this long-term project, our primary objective is to develop Lightning components and pages that replicate the functionality of Case Peer in Salesforce. To accomplish this, we have employed a comprehensive range of tools and technologies. 1. Firstly, we have created custom objects and fields to store and manage the necessary data for the project. These customizations allow us to tailor the solution to meet specific requirements. 2. Additionally, we have developed various Lightning components that are responsible for displaying data related to each object. These components provide an intuitive and user-friendly interface for users to interact with the data effectively. 3. To ensure seamless integration and business logic enforcement, we have utilized Apex classes, Apex Triggers, and Data Model creation. These components handle backend logic and data manipulation, ensuring the smooth operation of the application. 4. Furthermore, we have implemented various features such as validation rules, workflows, process builders, and quick actions to streamline and automate processes within the system. 5. To enhance user experience and visual appeal, we have leveraged JavaScript and CSS to create dynamic functionalities and design attractive interfaces. 6. Additionally, we have made use of flows and custom metadata to simplify complex processes and store configurable settings. Use : **LWC** component, Apex script, Triggers, Flows, Aura component Role: Lead Salesforce Developer.
PROJECT-10	**VisualForce form page development:** 1. In this project, I have to develop several visual force forms filtered on the basis of member number. There were almost 850 member numbers. 2. Also implemented the inline fields validations and integrated a payment gateway after submitting the forms.

	3. Made LWC components and called from VF pages to show in classic view of salesforce. 4. Also made some visual flow in this project to delete the invalid or expired records. Use: Visual Force pages, Apex script, Test classes, Validation rules, Flow Role: Developer
PROJECT-11	**Salesforce-Field service lightning (FSL) - SST** In this Field Service Lightning (FSL) project, I have undertaken various tasks related to configuration and development to ensure effective implementation and customization. 1. Configuration Setup: I have worked on setting up the configuration for FSL, which includes defining field service settings, service territories, work types, and other relevant configurations to align the system with specific business requirements. 2. Apex Triggers: I have developed Apex triggers to handle specific requirements and automate processes within the FSL environment. These triggers enable customization and provide the ability to execute specific actions based on predefined conditions. 3. Work Order Setup: I have implemented the setup and customization of work orders, ensuring that their status is updated dynamically based on specific criteria defined on work order line items. This allows for efficient management and tracking of work orders throughout their lifecycle. 4. Lightning Components: I have developed Lightning components to enhance the user interface and provide customized functionalities within the FSL application. These components enable a more intuitive and tailored user experience, improving productivity and efficiency. 5. Lightning Flows: I have worked on Lightning flows to automate and streamline complex business processes within the FSL environment. By designing and implementing flows, I have created visual representations of the steps involved in the processes, guiding users through various stages and automating repetitive tasks. By performing these tasks, including FSL configuration setup, Apex trigger development, work order customization, lightning component development, and lightning flow implementation, I have contributed to the overall success of the FSL project. The combination of these efforts ensures a comprehensive and tailored solution that optimizes the field service management processes and improves user productivity.

	Use: **LWC** component, Apex script, Triggers, Aura component, batch class, Flow, Configurations Role: Developer
	<u>Public Site Design and Implement-Salesforce</u> In this project, my focus was on developing and customizing pages for the Salesforce public site. To achieve this, I undertook the following tasks: 1. Visualforce Pages: I implemented multiple Visualforce pages to create the desired user interface and functionality for the public site. These pages are tailored to meet specific requirements and provide a seamless browsing experience. 2. Apex Classes: I developed Apex classes to handle the backend logic and data manipulation required for the public site pages. These classes enable the execution of custom business processes and ensure data integrity and security. 3. Aura/LWC Components: I utilized both Aura and LWC components to enhance the functionality and interactivity of the public site pages. These components enable a rich user experience and allow for the seamless integration of custom features and data display. 4. Triggers: I implemented triggers to automate specific actions or enforce business rules based on certain events or conditions. These triggers ensure data consistency and enable real-time updates when interacting with the public site. 5. Static Resources: I leveraged static resources to include JavaScript bundles required for specific functionalities on the public site. This approach ensures optimized performance and provides a scalable solution for managing client-side resources. 6. Permissions and Data Access: I worked on configuring permissions and managing data access for the public site. This involved setting up site content and defining appropriate access controls to ensure the proper visibility and security of data. By combining Visualforce pages, Apex classes, Aura/LWC components, triggers, static resources, and permissions management, I successfully implemented and customized pages for the Salesforce public site. These efforts resulted in a user-friendly and feature-rich public site that meets the project requirements and enhances the overall user experience.
PROJECT-12	Use: **LWC** component, Apex script, Triggers, Flows, VF, Aura

	Role: Developer