Part 1: System Overview

1. **System Description:**
   System: Car Rental System (Auto Vista Car Rentals)

2. **System Features:**
   Purpose:
   The system is made to make running a vehicle rental business easier by giving clients a smooth experience and handling administrative tasks for the company.

   Function:

   - Create an Account: Customers are able to register ang manage their persona information.
   - Place a Rental: It allows customer to select and rent vehicles.
   - Cancel a Rental: Gives option for the customer to cancel their reservations if needed.
   - View Rental History: The system provides a thorough record of their past rentals.
   - Monitor Sales: Allows admins to track revenue and rental transactions.
   - Manage Customer Rentals: Provides tools for admins to oversee and assist with customer rentals.
   - Manage Employees: Enables admins to add, remove, and update employee records.
   - Vehicle Inventory: Allows admins to add, remove, and update available car stocks.

   Users:

   - Customer: People looking to rent a vehicle.
   - Admins: Business administrators responsible for managing customer rentals, inventory, and employee records.

Part 2: SDLC Model Selection

1. **SDLC models:**

Model	Advantages	Disadvantages	Best Use Cases
Waterfall	Structured, easy to manage, clear phases	Rigid, late testing, poor for changing needs	Small projects with clear, stable requirements
Agile	Flexible, fast delivery, client-centric	Requires strong client involvement, scope creep risk	Dynamic projects with evolving requirements
Spiral	Risk management focus, adaptable	Complex, costly, requires skilled management	High-risk, large-scale projects
Iterative	Allows early feedback, manageable risk	Requires good initial requirements, costlier	Projects needing early releases, evolving specs

2. **Chosen SDLC Model:**

   Agile Model

   For the SCDL model, we opt Agile approach to ensure flexibility and adaptability throughout the development process. In simple terms, Agile methodology breaks down the project into smaller, manageable chunks called sprints. Each sprint typically lasts for a few weeks and focuses on completing a specific set of tasks or features.

Part 3: Phases of the Chosen SDLC Model

Phase 1: Planning

- **Purpose of the Phase:**
  - The main goal of this phase is to define the project objectives, identify project stakeholders, and outline initial requirements. This phase is crucial in setting the direction and foundation for the project.
- **Planned Activities:**
  - Initial project scoping and risk assessment.
  - 一 Resource and timeline estimation.
  - 二 Required gathering through meetings and interviews.
  - 三 Setting up a backlog of features and tasks to be addressed in upcoming sprints.
- **Expected Outcome:**
  - A thorough understanding of the project's goals and limitations, together with a roadmap and backlog of features and activities that have been prioritized.

Phase 2: Design

- **Purpose of the Phase:**
  - In order to direct the development process, the design phase aims to produce a blueprint for the system that includes both technical architecture and user experience design.
- **Planned Activities:**
  - Creating wireframes and mockups for the user interface.
  - 一 Defining data models, workflows, and system integrations.
  - 二 Designing system architecture to ensure compatibility, scalability, and security.
- **Expected Outcome:**
  - A complete design document that describes how each component should be constructed, including UI mockups and system architecture.

Phase 3: Implementation

- **Purpose of the Phase:**
  - Functional modules are developed and tested in increments (sprints) during the implementation phase, which is mostly focused on coding and development.

- **Planned Activities:**
    - Writing code for both front-end and back-end components.
    - Developing individual system modules (e.g., account management, rental processing).
    - Conducting code reviews and ensuring code quality.
- **Expected Outcome:**
    - After this phase, it is expected for the modules will be progressively constructed, finished, and incorporated into the system as a whole.

Phase 4: Testing

- **Purpose of the Phase:**
    - Testing aims to validate that the system satisfies the requirements and is error-free by validating its functionality, performance, and security.
- **Planned Activities:**
    - Conducting unit testing, integration testing, and system testing.
    - Debugging and refining features based on test results.
- **Expected Outcome:**
    - A fully tested and debugged system that is stable and meets both functional and non-functional requirements.

Phase 5: Deployment

- **Purpose of the Phase:**
    - In order to ensure a seamless rollout, the deployment phase aims to make the system accessible to end users in a production setting.
- **Planned Activities:**
    - Deploying the system on the designated servers and infrastructure.
    - Conducting post-deployment support to address any immediate issues.
    - Training users and providing necessary documentation.
- **Expected Outcome:**
    - A complete and operational system that is accessible to users, with initial user training completed.

Phase 6: Maintenance

- **Purpose of the Phase:**
    - The maintenance phase concentrates on ongoing assistance, guaranteeing that the system continues to be safe, operational, and upgraded in response to user input.
- **Planned Activities:**
    - Managing updates, patches, and potential upgrades.
    - Collecting and implementing user feedback to improve the system.
    - Monitoring system performance and addressing any arising issues.
- **Expected Outcome:**
    - An up-to-date, fully maintained system that, throughout time, stays in line with user demands and corporate specifications.

Part 4: Summary

- **Reflection on the SDLC Model:**
    - Since the Agile model is flexible and iterative, it worked well for the Car Rental System project. We were able to promptly implement modifications in response to feedback thanks to the Agile methodology, which made sure the system developed to successfully satisfy the demands of administrators and customers. Since new features were frequently proposed throughout development, controlling scope creep was a significant difficulty. But dividing work into smaller sprints kept us focused and enabled us to produce a top-notch, user-focused solution on schedule. This approach emphasized how crucial stakeholder engagement, adaptability, and ongoing communication are to completing a project successfully.