**Project Planning Phase – Agile-Based**

**Date:** 20 February 2025  
**Team ID:** LTVIP2025TMID28970  
**Project Name:** Garage Management System  
**Maximum Marks:** 4 Marks

**Agile Overview: Key Concepts**

|  |  |
| --- | --- |
| **Term** | **Description** |
| **Sprint** | A time-boxed iteration (here, 5 days) in which a set of prioritized tasks delivering working software is completed. |
| **Epic** | A large feature or functionality for the Garage Management System, too comprehensive for a single sprint (e.g., "Customer Relationship Management," "Vehicle Service Workflow"). |
| **User Story** | A user-centric task that delivers tangible value, typically a breakdown of an Epic (e.g., "As a service advisor, I can create a new vehicle record."). |
| **Story Points** | Effort estimation for a story using Fibonacci series (1, 2, 3, 5...). Reflects complexity, risk, and effort, not absolute time. |

**Sprint Planning Table – 5 Days Per Sprint**

#### Sprint 1 – Sprint 1 – Initial Data Setup & Basic Customer/Vehicle Management

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Day** | **Task** | **Story Points** | **Type** | **Notes** | | 1 | Collect existing customer and vehicle data | 3 | Data Sourcing | From spreadsheets/legacy records | | 2 | Load core customer and vehicle data into Salesforce objects | 2 | Configuration | Import via Data Import Wizard/DataLoader.io | | 3 | Handle data discrepancies and duplicates in customer/vehicle records | 3 | Data Cleaning | Ensure clean entries for contact info, VINs | | 4 | Create and configure picklists/lookup fields for vehicle types, service statuses | 2 | Object Setup | Picklists for **Vehicle\_Type\_\_c, Service\_Status\_\_c** | | 5 | Sprint Review + Bug Fixes | - | QA | Internal review and adjustments | |  | **Total Story Points (Sprint 1)** | **10 Points** |  |  | |

#### Sprint 2 – Core Service Request & Work Order Development

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Day** | **Task** | **Story Points** | **Type** | **Notes** |
| 1 | Create custom objects (ServiceRequest, WorkOrder, PartsInventory) | 5 | Configuration | Setup schema and relationships (e.g., ServiceRequest to Vehicle) |
| 2 | Design Lightning record pages for Service Request & Work Order | 3 | UI/UX | Layouts, related lists for efficient service advisor use |
| 3 | Develop Apex Trigger for auto-updating Work Order status | 3 | Backend Logic | **WorkOrder\_Status\_\_c** updates based on **Task\_\_c** completion |
| 4 | Create Record-Triggered Flow for Service Completion Notification | 5 | Automation/Flow | Send email/SMS to customer when **WorkOrder\_Status\_\_c** = 'Ready for Pickup' |
| 5 | Sprint Review + Initial System Testing | - | QA & Deployment | Validate object linkages, status updates, and notifications |
|  | **Total Story Points (Sprint 2)** | **16 Points** |  |  |

#### Sprint 3 – Billing, Payment & Reporting Enhancements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Day** | **Task** | **Story Points** | **Type** | **Notes** |
| **1** | Create custom objects (ServiceEstimate, Invoice, Payment) | **5** | Configuration | Setup schema and relationships (e.g., Invoice to WorkOrder) |
| **2** | Design Lightning pages for Estimates & Invoices | **3** | UI/UX | User-friendly layouts for billing and payment entry |
| **3** | Develop Apex Trigger for auto-calculating Invoice totals/balance | **5** | Backend Logic | Total\_Amount\_\_c on Invoice\_\_c, Outstanding\_Balance\_\_c based on Payment\_\_c |
| **4** | Implement Validation Rule for Paid\_Amount\_\_c <= Total\_Amount\_\_c | **2** | Validation | Prevent overpayment errors on invoices |
| **5** | Sprint Review + Reporting Setup | **-** | QA & Reporting | Validate financial accuracy, create initial service reports |
| **Total Story Points (Sprint 3)** | **15 Points** |  |  |  |

**Velocity Calculation**

|  |  |
| --- | --- |
| **Metric** | **Value** |
| Story Points in Sprint 1 | **8 Points** |
| Story Points in Sprint 2 | **16 Points** |
| Story Points in Sprint 3 | **15 Points** |
| **Total Points** | **41 Points** |
| Number of Sprints | **3** |
| **Velocity** | **41 ÷ 3 = 13.67 ≈ 14 Points/Sprint (Average)** |

Your team’s **average velocity** is **14 Story Points per Sprint**.

**Sprint Status Summary**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sprint** | **Duration (Days)** | **Points Planned** | **Points Completed** | **Completion %** | **Remarks** |
| Sprint 1 | 5 | 10 | 10 | 100% | Initial data loaded, basic Customer & Vehicle objects ready |
| Sprint 2 | 5 | 16 | 16 | 100% | Core service request & work order process implemented |
| Sprint 3 | 5 | 15 | 15 | 100% | Billing, payment, and initial reporting modules configured |

### Visual Timeline View (2-Week Sprint Schedule - 5 days/sprint)

****

### Planning Insights & Best Practices Followed

* Used Fibonacci-based Story Points for accurate complexity estimation, reflecting garage-specific logic complexity.
* Balanced workload across 3 sprints with clear, achievable deliverables for each iteration.
* Each Epic (e.g., Vehicle Service Workflow) broken into granular user stories for better tracking and incremental delivery.
* Internal testing and QA cycles integrated within the same sprint for agile feedback and early bug detection.
* Planning done based on team velocity (approx. 14 SP/Sprint) to ensure realistic and sustainable deliverables.
* Prioritized foundational data management in early sprints to support subsequent feature development.

### Conclusion

* The team followed Agile sprint methodology with accurate estimation, planning, and execution throughout the Garage Management System project.
* Deliverables were consistently aligned with the final outcome of a comprehensive Garage Management CRM.
* All core components – Data Setup, Customer/Vehicle Management, Service Request/Work Order Flow, Billing & Payment, and initial Reporting – were delivered within 15 working days over 3 sprints.

### Agile Planning Overview

Agile methodology promotes incremental delivery through iterative cycles known as Sprints. Each Sprint involves:

* **Product Backlog:** A prioritized list of all desired features for the Garage Management System (Epics & Stories).
* **Sprint Backlog:** User stories and tasks selected for a particular sprint's focus.
* **Story Points:** Units representing complexity, effort, and uncertainty for each user story.
* **Velocity:** Average number of story points a team completes per sprint, used for future sprint planning.
* **Burndown Chart:** Visual progress of work remaining versus days in a sprint.

**Product Backlog, Sprint Schedule, and Estimation (4 Marks)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
| Sprint-1 | **Customer & Vehicle Onboarding** | USN-1 | As a service advisor, I can create/retrieve customer records with contact details. | 2 | High | Member A |
| Sprint-1 | **Customer & Vehicle Onboarding** | USN-2 | As a service advisor, I can create/retrieve vehicle records with VIN, Make, Model, Mileage. | 3 | High | Member B |
| Sprint-1 | **Customer & Vehicle Onboarding** | USN-3 | As a system, I can validate VIN format during vehicle entry. | 1 | Medium | Member A |
| Sprint-1 | **Basic Data Setup** | USN-4 | As an admin, I can import existing customer and vehicle data into the CRM. | 4 | High | Member C |
| Sprint-2 | **Service Request Management** | USN-5 | As a service advisor, I can log a new service request for a vehicle, including reported symptoms. | 3 | High | Member B |
| Sprint-2 | **Service Request Management** | USN-6 | As a service advisor, I can assign a service request to a mechanic. | 2 | High | Member A |
| Sprint-2 | **Work Order & Task Management** | USN-7 | As a mechanic, I can view my assigned work orders and update task status (e.g., In Progress, Completed). | 5 | High | Member D |
| Sprint-2 | **Work Order & Task Management** | USN-8 | As a system, I can automatically update WorkOrder\_Status\_\_c when all tasks are complete. | 3 | High | Member C |
| Sprint-3 | **Estimate & Invoice Automation** | USN-9 | As a service advisor, I can generate a service estimate based on selected services and parts. | 4 | High | Member B |
| Sprint-3 | **Payment Handling** | USN-11 | As a service advisor, I can record a payment and have the Paid\_Amount\_\_c on the invoice automatically update. | 3 | High | Member A |
| Sprint-3 | **Communication & Notifications** | USN-12 | As a customer, I will receive an email/SMS notification when my vehicle service is complete. | 3 | High | Member C |

**Project Tracker, Velocity & Burndown Chart (4 Marks)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sprint | Total Story Points | Duration | Sprint Start Date | Sprint End Date (Planned) | Story Points Completed (on Planned End Date) | Sprint Release Date (Actual) |
| Sprint-1 | 10 | 5 Days | 01 July 2025 | 05 July 2025 | 10 | 05 July 2025 |
| Sprint-2 | 16 | 5 Days | 08 July 2025 | 12 July 2025 | 16 | 12 July 2025 |
| Sprint-3 | 15 | 5 Days | 15 July 2025 | 19 July 2025 | 15 | 19 July 2025 |

**Velocity Calculation**

* **Total Story Points:** 10 (Sprint 1) + 16 (Sprint 3) = **41**
* **Total Sprints:** 3
* **Velocity (Story Points per Sprint):** 41 / 3 = **13.67 ≈ 14**

**Burndown Chart (Conceptual)**

A Burndown Chart visually shows how the team is progressing through the project, tracking work remaining vs. days in sprint. Here's a rough reference structure:

**Sprint 2 – 5 Day Burndown**

**🔹 Ideal vs. Actual Progress (Sprint-2 Sample):**

|  |  |  |
| --- | --- | --- |
| Day | Remaining Story Points (Ideal) | Remaining Story Points (Actual) |
| Day 0 | 16 | 16 |
| Day 1 | 12.8 | 14 |
| Day 2 | 9.6 | 10 |
| Day 3 | 6.4 | 6 |
| Day 4 | 3.2 | 3 |
| Day 5 | 0 | 0 |

The actual line ideally follows or improves upon the ideal line, indicating a healthy sprint pace. If the actual line is consistently above the ideal, it might indicate obstacles or over-commitment.

**Visualization tools to create the Burndown Chart:**

* Excel / Google Sheets (manual plotting)
* Jira / Trello / Asana Agile Board (often built-in)
* Visual Paradigm Online
* Other project management software

This conceptual data shows the team completed all planned stories progressively and effectively.

**Summary:**

* 🧮 **Total Story Points Completed:** 41
* 📐 **Velocity:** 14 Story Points per Sprint (Average)
* 🧠 **Planning Strategy:** All epics broken down into small, manageable stories using Fibonacci sequence for better effort estimation, ensuring focus on core garage management functionalities.
* 📊 **Tools Used:** Salesforce Dev Playground (for development), Google Sheets (for planning/tracking), potentially Trello/Jira (for sprint board management).