1. ***Product Backlog for "DRIP HAUS" ERP System***

The product backlog is a prioritized list of features, improvements, and tasks required for the successful development of the "DRIP HAUS" ERP system. The backlog is structured using two prioritization methods: MoSCoW (Must have, should have, could have, Won't have) and Value vs. Effort.

**Prioritization Methods:**

* 1. MoSCoW Method:

- Must Have: Essential features for core functionality.

- Should Have: Important but not critical; adds significant value.

- Could Have: Desirable features that enhance user experience but are not essential.

- Won't Have (for now): Features to be deferred for future iterations.

* 1. Value vs. Effort Matrix:

- High Value, Low Effort: Immediate priority.

- High Value, High Effort: Planned for major releases.

- Low Value, Low Effort: Scheduled if resources allow.

- Low Value, High Effort: Low priority or deferred.

**Product Backlog Items:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **User Story** | **Priority (MoSCoW)** | **Value vs. Effort** |
| PB001 | As a store owner, I want to manage inventory in real-time for multiple locations. | Must Have | High Value, Medium Effort |
| PB002 | As a user, I want secure authentication using Auth.js to protect sensitive data. | Must Have | High Value, Medium Effort |
| PB003 | As a sales manager, I want to process payments securely via Stripe/PayPal. | Must Have | High Value, High Effort |
| PB004 | As an administrator, I want to generate invoices compliant with local regulations. | Must Have | High Value, High Effort |
| PB005 | As a customer, I want personalized product recommendations. | Should Have | Medium Value, High Effort |
| PB006 | As a store owner, I want to track low stock levels and receive restocking alerts. | Must Have | High Value, Medium Effort |
| PB007 | As a manager, I want to review sales reports with filters and export options. | Should Have | Medium Value, Medium Effort |
| PB008 | As a user, I want a responsive interface to access the ERP on mobile devices. | Could Have | Medium Value, Low Effort |
| PB009 | As a logistics manager, I want to track deliveries and update statuses in real-time. | Could Have | Low Value, High Effort |
| PB010 | As an administrator, I want automated backups to prevent data loss. | Must Have | High Value, Low Effort |

1. ***Definition of Done (DoD) for "DRIP HAUS" ERP System***

The Definition of Done outlines the acceptance criteria for delivering a complete and shippable product increment. This ensures consistency, quality, and alignment across all development stages.

**General Criteria:**

* 1. Code Quality:

- Code is written following industry best practices (clean code principles).

- Code has been reviewed through a peer code review process.

- No critical or high-severity bugs remain.

* 1. Functionality:

- All acceptance criteria for user stories are met and validated.

- The feature works as intended across supported devices and browsers.

* 1. Testing:

- Unit tests achieve a minimum of 90% code coverage.

- Automated integration tests are performed and passed.

- Manual tests are executed for edge cases and complex workflows.

* 1. Documentation:

- Updated API documentation if endpoints are modified.

- User guides and system manuals are updated as required.

* 1. Deployment Readiness:

- Feature branches are merged into the main branch.

- The feature is successfully deployed to the staging environment.

- Deployment scripts are tested and validated for production readiness.

* 1. Security & Compliance:

- Data handling meets security standards (e.g., encryption, secure storage).

- Compliance with relevant regulations (e.g., data privacy laws) is verified.

* 1. Performance & Monitoring:

- System meets the defined performance benchmarks.

- Monitoring and alerting configurations are updated to reflect new functionalities.