**Grade-Based Functionality Expectations**

The table below outlines what features and quality levels are typically expected for different pass grades:

| **Grade Category** | **Functionality Expectations** |
| --- | --- |
| **Bare Pass**  **(40–49%)** | Basic core features with limited polish.  – Admin: Add/view staff, assign shifts. – Employee: View and accept/decline shifts. – Customer: Basic menu view and order (cart optional). – SQLite integration for at least one area. – Minimal/no error handling. – Report, code, and video submitted, even if basic. |
| **Pass**  **(50–59%)** | Full coverage of core features.  – Admin: Staff CRUD, shift assignment, basic payroll. – Employee: Earnings view, shift acceptance. – Customer: Browse menu, place order, basic receipt. – SQLite used for multiple data areas. – UI navigation is clear. – Report structured with some testing and screenshots. – Reasonably commented code. – Video includes clear demo and presenter identity. |
| **Good Pass**  **(60–69%)** | Strong implementation and usability. – All core features working well. – Payroll with invoice PDF generation. – Employees can download monthly earnings reports. – Receipt generation works well and is polished. – UI is user-friendly and consistently styled. – Evidence of enhancements (e.g. shift calendar, search features). – Detailed report with testing logs and known issues. – Clear, confident video explanation. |
| **Very Strong Pass**  **(70–100%)** | Full system + relevant, well-executed enhancements. – Please find below recommended additional Feature List – Excellent UI/UX and robust error handling. – Efficient SQLite queries and use of Kotlin features (e.g. coroutines, MVVM). – Thorough technical report with test evidence, clear bugs listed. – High-quality, fluent video with design justification and confident technical explanations. |
|  |  |

Please find a list of recommended Features

Admin

1. Staff performance reports (based on shifts completed).
2. Menu category management (Starters, Mains, Desserts, etc.).
3. Track most/least ordered menu items.
4. Approve/reject leave requests from staff.
5. Create and manage discount/promo codes.
6. Maintain action logs (e.g. staff added, menu updated).
7. Backup and restore SQLite database.
8. Manage announcements or in-app notifications for staff.
9. Assign table numbers to dine-in orders.
10. Enable/disable menu items temporarily (e.g. out of stock).

Staff List

1. Submit leave requests.
2. View shift history (past and upcoming).
3. Track monthly performance (shifts completed, hours worked).
4. Receive announcements from admin.
5. View notifications (e.g. shift reminders, schedule changes).
6. View assigned table numbers (for waiters).
7. Mark shift attendance (check-in/check-out).
8. View food preparation notes (for chefs).
9. View customer feedback related to their service.
10. Access payroll history (previous months’ invoices).

Customer List

1. View order history.
2. Reorder previous meals with one click.
3. Mark dishes as favourites.
4. Submit feedback for each order.
5. Apply promo codes at checkout.
6. Schedule a food order for later pickup or dine-in.
7. Table reservation (select date/time/number of people).
8. Rate individual menu items.
9. Receive order status updates (preparing, ready, delivered).
10. View estimated preparation or wait time.

NOTES:

* + It is not expected that students will complete all the above suggestions; rather, they will select some to showcase their skills.
  + In each case, it is advised that students carry research into existing social media systems to see how these features have been deployed and used within existing platforms, allowing them to utilise existing good practices.