

APPLICATION SOFTWARE DESIGN DOCUMENT

Opay Employee Salary Management System

Professional Tender Submission Document

Document Code	ASDD-OPAY-001
Version	1.0
Status	For Tender Submission
Document Type	Application Software Design Document
System	Opay Employee Salary Management System
Classification	Confidential

1. DOCUMENT PURPOSE

- This document defines the application-level software design for the Opay Employee Salary Management System.
- It translates the business requirements into logical application components, workflows, services, validations, security controls and processing responsibilities.
- The design is intended to support implementation, testing, maintenance, deployment and future enhancement.

2. DESIGN OBJECTIVES

- Provide a modular and maintainable payroll application.
- Separate presentation, business processing, data access and integration responsibilities.
- Ensure salary and financial calculations are controlled, traceable and auditable.
- Implement role-based access and maker-checker approval workflows.
- Support Contractual, Outsource and No Work No Pay employee processing.
- Provide secure integration with bank/BEFTN and SMS services.
- Provide reliable reports, certificates, statements and management information.

3. SYSTEM SCOPE

- Master and configuration management.
- Employee, contractor and profile management.
- Contract/package, extension and transfer management.
- Attendance, salary, arrears, deductions, allowances, overtime and bonus processing.
- Tax, VDS, TDS, commission and challan processing.
- Submission, certification, verification, scrutiny and approval workflows.
- BEFTN, advice, invoice and payment-related processing.
- Operational, statutory and management reporting.

4. APPLICATION DESIGN ARCHITECTURE

Layer	Design Responsibility	Major Components
Presentation Layer	User interaction and navigation	APEX pages, forms, reports, dashboards, validations, dynamic actions
Application / Business Layer	Business rules and transaction processing	Salary engine, workflow engine, bonus, tax, attendance, profile services
Data Access Layer	Controlled database interaction	PL/SQL packages, procedures, functions, views, reusable queries
Integration Layer	External communication	BEFTN/bank interface, SMS gateway, REST/HTTPS services
Data Layer	Persistent and audit data	Oracle tables, indexes, constraints, master and transaction data
Reporting Layer	Operational and statutory reporting	Reports, certificates, statements, PDF/print/export services

5. APPLICATION MODULE DESIGN

Module	Core Design Responsibilities
Master Setup	User setup, permissions, cost centers, sender accounts, banks, designations, advice, religion, salary/deduction, tax, allowance, service charge and bonus parameters.
Profile Management	Employee/contractor information, employment type, package, extension, transfer, update, verification and approval.
Contractual Salary	Salary setup, deductions, tax, attendance, arrears, allowances, bonus, submission, certification, verification and approval.
Outsource Salary	Outsource salary, deduction, arrears, attendance, commission, bonus, submission, certification and approval.
NWNP Salary	No Work No Pay attendance-based salary, deductions, arrears, bonus and approval processing.
Extension & Transfer	Contract extension, employee extension, transfer, package changes and employee approval.

Module	Core Design Responsibilities
Salary System	Monthly modification, arrears, conveyance, overtime, allowances, bonus and salary reset controls.
Incentive Bonus	Bonus setup, pay processing, BEFTN, summary and advice.
Tax Information	Tax setup, challan entry, monthly/yearly certificates, tax calculation and contractor VAT/TDS/VDS reports.
Reports	BEFTN, advice, invoice, salary, attendance, tax, commission, employee, certificate and management reports.

6. ROLE & ACCESS DESIGN

- System Administrator: manages system setup, users, permissions and controlled configuration.
- Cost Center / ARO / Sub Assistant Engineer: creates employee information, salary setup, attendance and salary submission.
- Assistant Engineer / RO: performs certification and verification functions.
- Executive Engineer / DCRO: performs approval functions for employee and salary-related transactions.
- Assistant Accountant / Accountant: performs account-side checking, certification, scrutiny and financial processing.
- Account Officer: performs account verification, approval and financial controls.
- DCAO: performs higher-level scrutiny and authorization.
- CAO / DMD Finance: provides final financial oversight and authorization.
- Access shall be granted by role and business scope; users shall not receive unnecessary privileges.

7. CORE SALARY PROCESS DESIGN

Step	Stage	Design Activity
1	Employee/Profile	Create or update employee/contractor information
2	Package	Assign/verify contract package and applicable parameters
3	Attendance	Upload or enter attendance and validate period
4	Setup	Apply salary, deductions, allowances, overtime, arrears and bonus
5	Calculation	Calculate gross, deductions, tax and net payable
6	Submission	Submit salary for workflow
7	Certification	Authorized officer certifies
8	Verification	Authorized officer verifies
9	Scrutiny	Accounts performs financial scrutiny where applicable
10	Approval	Approving authority approves
11	Payment	Generate approved payment/BEFTN/advice outputs
12	Reporting	Generate statements, certificates and management reports

8. BUSINESS RULE & VALIDATION DESIGN

- Mandatory employee/contractor master information shall be validated before salary processing.
- Salary processing shall be restricted to the configured bill month and applicable employee status.
- Duplicate salary submission for the same employee and period shall be prevented or explicitly controlled.
- Salary calculation shall use approved salary, deduction, allowance, attendance and tax parameters.
- Negative or invalid financial values shall be rejected according to the applicable business rule.
- Approval actions shall be restricted by role and workflow status.
- Returned/rejected transactions shall retain remarks and require controlled re-submission.
- Payment output shall be generated only from approved transactions.
- Changes to critical financial information shall be auditable.

9. TRANSACTION & WORKFLOW DESIGN

- Each major transaction shall have a unique master/transaction identifier.
- Workflow status shall represent the current lifecycle state, such as Draft, Submitted, Certified, Verified, Scrutinized, Approved, Returned or Rejected.

- Status transitions shall be role-controlled and validated on the server side.
- Approved financial transactions shall be protected from unauthorized modification.
- Any post-approval modification shall follow an authorized reversal/reopen process.
- Transaction timestamps, users and relevant remarks shall be recorded for auditability.

10. DATABASE INTERACTION DESIGN

- Application pages shall use controlled SQL/PLSQL interfaces rather than uncontrolled direct modification of critical data.
- Business calculations shall be centralized in reusable packages/procedures where practical.
- Foreign keys, unique constraints, check constraints and indexes shall enforce data integrity and performance.
- Views shall be used for reusable reporting and controlled data presentation.
- Audit fields such as created_by, created_date, updated_by and updated_date shall be maintained for relevant entities.
- Transaction commits shall be controlled at logical business transaction boundaries.

11. TAX, DEDUCTION & FINANCIAL CALCULATION DESIGN

- Tax and deduction parameters shall be maintained through authorized setup functions.
- Calculation components shall be separately identifiable to support reconciliation.
- Gross salary, allowances, deductions, tax and net salary shall be traceable.
- VDS/TDS/commission/service charge shall be calculated according to approved configuration.
- Financial rounding and precision rules shall be consistently applied.
- Calculation results shall be validated before submission and approval.

12. INTEGRATION DESIGN

- External integrations shall be isolated behind an integration/service layer.
- BEFTN/bank communication shall use approved secure protocols and credentials.
- SMS functionality shall support controlled single and multiple message operations.
- Integration requests shall capture request status, response status and error information.
- Timeout and failure conditions shall not silently mark a financial transaction as successful.
- Where applicable, retry and reconciliation mechanisms shall be used.

13. REPORTING & DOCUMENT DESIGN

- Reports shall use parameterized and authorized data sources.
- Report access shall follow the user's role and organizational scope.
- Salary certificates shall use approved employee and salary data.
- BEFTN, advice and invoice outputs shall be generated from approved transaction data.
- Reports shall support filtering by relevant dimensions such as month, cost center, employee type, contractor, designation and other approved parameters.
- Report totals shall be reconciled with source transaction data.

14. SECURITY DESIGN

- Authentication shall be enforced for all protected functions.
- Authorization shall use role-based access control.
- Maker-checker separation shall be maintained for critical financial workflows.
- Sensitive credentials and integration secrets shall not be exposed in application screens or logs.
- HTTPS shall protect network communication.
- Session timeout and unauthorized-access controls shall be configured.
- Audit trails shall capture critical user and financial activities.

15. ERROR & EXCEPTION HANDLING

- User-friendly validation messages shall be shown for expected business errors.

- Technical exceptions shall be logged with sufficient diagnostic information.
- Database transactions shall be rolled back when a critical processing error prevents successful completion.
- Integration failures shall be distinguishable from successful payment/notification responses.
- Errors shall not expose passwords, tokens, database credentials or sensitive internal details.
- Critical errors shall be escalated to the support/operations team.

16. AUDIT TRAIL DESIGN

Audit Area	Minimum Information
Authentication	User, date/time, success/failure and relevant session information
Master Data	User, action, date/time and changed values where applicable
Salary	Employee, period, action, old/new status and responsible user
Approval	Approver, action, date/time, status and remarks
Payment	Payment batch/reference, user, status and response/reconciliation information

17. PERFORMANCE & SCALABILITY DESIGN

- Use indexed keys and optimized queries for high-volume salary and report processing.
- Separate transactional processing from heavy reporting workloads where infrastructure permits.
- Use pagination and parameterized reports for large datasets.
- Use reusable PL/SQL/database services for calculation-heavy operations.
- Monitor database resource usage, long-running SQL and application response times.
- Design modules so additional employee types, allowances, reports and integrations can be added without redesigning the entire application.

18. BACKUP, RECOVERY & OPERATIONS DESIGN

- Maintain scheduled database backups and application release exports.
- Keep recovery procedures documented and tested.
- Monitor database, application and integration health.
- Maintain deployment and configuration baselines.
- Record production incidents, changes and corrective actions.
- Periodically review audit logs and privileged access.

19. UI/UX DESIGN GUIDELINES

- Use consistent navigation, labels, buttons, validation messages and status indicators.
- Clearly distinguish editable, read-only and approved data.
- Provide confirmation for critical actions such as submission, approval and payment generation.
- Display meaningful error and success messages.
- Use responsive layouts suitable for authorized desktop/web users.
- Provide searchable and parameter-driven reports for operational users.

20. TESTING DESIGN CONSIDERATIONS

- Unit testing of business procedures and calculations.
- Functional testing of each module.
- Integration testing for bank/BEFTN and SMS.
- Workflow testing for submit, certify, verify, scrutinize, approve and return scenarios.
- Security testing for role restrictions and unauthorized access.
- Performance testing for salary processing and high-volume reports.
- Regression testing after changes.
- UAT with approved business scenarios and sign-off.

21. TRACEABILITY TO BUSINESS REQUIREMENTS

Requirement Group	Primary Application Design Component	Validation / Evidence
Master Setup	Configuration services + RBAC	Setup screens, authorization and audit
Profile	Profile & contract services	Create/update/check/verify/approve
Salary Processing	Payroll calculation & workflow services	Salary calculation, submission and approval
Bonus / Allowance	Parameter and transaction services	Calculation and reports
Tax	Tax/VDS/TDS services	Calculation, challan and certificates
Payment	Payment / BEFTN integration	Approved batch and reconciliation
Reports	Reporting services	Parameterized reports and certificates

22. DESIGN CONCLUSION

The proposed application software design provides a modular, secure and auditable foundation for the Opay Employee Salary Management System. The design separates user interaction, business processing, data access, integration and reporting responsibilities while supporting controlled salary workflows, financial calculations, role-based authorization and future scalability.

The detailed physical implementation shall follow the approved SRS, SDD, database design, deployment architecture and applicable organizational policies.

23. REVIEW & APPROVAL

Prepared By	Reviewed By	Approved By
Name: _____ Designation: _____ Date: _____ _____	Name: _____ Designation: _____ Date: _____ _____	Name: _____ Designation: _____ Date: _____ _____

End of Document