**Proposal: “eResort GH” – A Django-Powered Multi-Branch Resort & Mini-Hotel Management Platform**  
*(Prepared for hospitality businesses operating in Ghana)*

**1. Executive Summary**

Many resorts and boutique hotels across Ghana still rely on disparate spreadsheets or paper “sales books” to track rooms, bars, activities and inventory. This slows decision-making, hides leakages, and makes compliance (e.g. VAT returns) painful.  
**eResort GH** is a Software-as-a-Service (SaaS) platform built with Django that lets any hospitality business — from an eco-lodge in Ada to a beach resort in Busua — sign up, create unlimited branches, and manage every type of facility (rooms, spa, swimming lessons, restaurant, bar, paint-ball arena, etc.) from one secure cloud dashboard.

**2. Goals & Objectives**

| **#** | **Objective** | **Key Result** |
| --- | --- | --- |
| 1 | Replace manual ledgers with real-time digital bookings & POS | 100 % branch adoption within 3 months of go-live |
| 2 | Consolidated view of business health | Daily branch-level P&L, inventory variance & occupancy reports |
| 3 | Ghana-compliant data security & payments | Full alignment with Data Protection Act 843 and Cybersecurity Act 1038 |
| 4 | Low entry barrier for SMEs | Cloud model (no servers to buy); MTN MoMo / Vodafone Cash gateway for guests |

**3. Scope of the Platform**

1. **Multi-Tenant On-Boarding**
   * Business owner registers (email + phone OTP).
   * Stripe-like “organisation” object → owns multiple branches.
2. **Branch Management**
   * Basic profile (GPS, GRA VAT no., pictures).
   * Independent tax settings (VAT vs. tourism levy).
3. **Facility Catalogue**
   * *Rooming Facility* → hierarchy: ​room-type → individual rooms.
   * *Recreational Facility* → definition of sessions, ticket types, resources.
   * *POS Facility* (bar / restaurant / gift-shop) → menu items, recipes, modifiers.
4. **Bookings & Front-Desk**
   * Calendar drag-and-drop, overbooking guard rails, check-in/out wizard.
   * Walk-in vs. OTA allotment, flexible pricing (Rack / Corporate / Promo).
5. **Integrated POS**
   * Touch-screen–friendly register, stock deductions, split bills.
   * Bluetooth receipt printer support (ESC/POS).
6. **Inventory & Cost-of-Sales**
   * Per-store stock ledger, supplier GRN, par-level alerts.
   * Recipe explosion for cocktails/meals → true food-cost %.
7. **Finance & Analytics**
   * Daily “Night Audit”, P&L, occupancy, RevPAR, bar cost variance.
   * Export to Excel, GRA iTaPS CSV, QuickBooks sync (optional).
8. **Payments**
   * MoMo (MTN, Vodafone, AirtelTigo), Visa/Mastercard via Hubtel Pay → compliant with Payment Systems & Services Act 987 citeturn0search5.
9. **User & Role Management**
   * Org admin, branch manager, cashier, storekeeper, auditor.
   * Fine-grained permissions via Django Guardian.
10. **Regulatory & Security**
    * Data encrypted at rest, TLS 1.3 in transit.
    * Local data residency (AWS af-south-1 / MTN Njorku DC).
    * Audit trail & cyber-incident hotline per Cybersecurity Act 1038.

**4. Technical Architecture**

| **Layer** | **Technology** | **Notes** |
| --- | --- | --- |
| Frontend | React + TypeScript (Next JS) | PWA for offline bar sales; Tailwind UI |
| API | Django 5 REST Framework | JWT auth + server-side sessions for POS |
| Background Jobs | Celery + Redis | Night audits, stock reorder email |
| DB | PostgreSQL 16 | Row Level Security per tenant |
| Media | S3-compatible (Ghana DC) | Public/Private buckets |
| Realtime | Django-Channels / WebSockets | Live room-plan updates |
| Payments | Hubtel, Paystack (card), direct MTN MoMo APIs |  |
| DevOps | Docker, GitHub Actions, Kubernetes on AWS EKS | Blue-green deployments, secrets in AWS SM |

**5. Implementation Roadmap**

| **Phase** | **Duration** | **Milestones** |
| --- | --- | --- |
| **0: Discovery** | 2 wks | BRD, UI wireframes, MoSCoW backlog |
| **1: Core Setup** | 4 wks | Multi-tenant auth, branch CRUD, facility models |
| **2: Rooming Module** | 5 wks | Room calendar, tariff engine, check-in/out |
| **3: POS & Inventory** | 6 wks | Menu builder, sales screen, stock ledger |
| **4: Payments & Reports** | 4 wks | MoMo/Card gateway, night-audit, BI dashboards |
| **5: Compliance & Pen-Test** | 2 wks | Data Protection impact assessment, VAPT |
| **6: Pilot & Roll-out** | 4 wks | Train 2 resorts (Eastern & Central regions) |
| **7: Support & Iteration** | Ongoing | SLA – 8 h response, monthly upgrades |

*Total initial build:* ***~23 developer-weeks*** *(≈ 5 months calendar).*

**6. Risk Management & Mitigations**

| **Risk** | **Probability** | **Impact** | **Mitigation** |
| --- | --- | --- | --- |
| Unreliable internet in rural areas | Med | High | PWA caching & POS offline queue |
| MoMo API downtime | Low | High | Fallback to local “Pending MoMo” ledger |
| Compliance changes (GRA e-VAT) | Med | Med | Config-driven tax rules, quarterly legal review |
| Data loss | Low | High | Point-in-time pgBackRest + S3 cross-region replica |

**7. Budget Snapshot (Indicative, USD)**

| **Item** | **Qty** | **Unit** | **Sub-Total** |
| --- | --- | --- | --- |
| Development (5 months, 2 senior devs) | 2 | $4 500 / mo | $45 000 |
| UI/UX Design | 1 | $3 000 | $3 000 |
| AWS Hosting (Year 1) | 1 | $2 400 | $2 400 |
| Penetration Test | 1 | $2 000 | $2 000 |
| Contingency 10 % | – | – | $5 240 |
| **Total** |  |  | **≈ $57 640** |

*(Costs assume local Ghanaian talent; FX rate ≈ GH₵ 13 : $1.)*

**8. Key Benefits to Ghanaian Resorts**

* **Single Source of Truth** – head office sees live sales from every branch.
* **Regulatory Peace of Mind** – data residency + compliance ready.
* **Revenue Uplift** – reduce stock shrinkage; dynamic pricing upsells.
* **Guest Experience** – digital invoices, MoMo checkout, WhatsApp booking links.

**9. Conclusion / Next Steps**

eResort GH provides an end-to-end, locally compliant platform that turns manual resort operations into measurable digital workflows. With modular rollout and a strong Django foundation, the solution is both **implementable today** and **extensible tomorrow** (e.g., channel manager, spa appointments, loyalty points).

**Action:** Approve Phase 0 (Discovery) to finalise detailed requirements, UI mock-ups and a signed Statement of Work.

**10. References**

1. Data Protection Act, 2012 (Act 843). citeturn0search0
2. Cybersecurity Act, 2020 (Act 1038). citeturn0search1
3. Payment Systems & Services Act, 2019 (Act 987). citeturn0search5

**Prepared by:** *[Your Name]*  
**Date:** 23 April 2025