Project 1c1 – Problem Condensation

CSC510: Software Engineering – Fall 2025

Zhaonan Meng, Rahmy Salman, Travis Thompson, John Stack

To decide what not to build yet, we used a set of filters that we applied consistently across all candidates. First, we insisted on a clinical-spine-first principle: anything that did not directly enable a safe, routine encounter from check-in to orders and results moved out of the MVP. Second, we minimized external dependencies and contractual surfaces; features that rely on food-service APIs, delivery networks, CRM systems, or payer-specific reporting were deferred to remove integration risk from the first release. Third, we separated platform-level compliance requirements from program-specific policy views; global audit logging and consent belong in the core, while nutrition-specific audit screens and reimbursement reports can ship later on top of those primitives. Fourth, we prioritized architectural leverage, choosing foundational components like RBAC, consent, and templating that unlock many downstream features rather than narrowly scoped program features. Fifth, we optimized for time-to-utility, selecting workflows that deliver immediate daily value to clinicians and patients, documentation, labs, prescriptions, and messaging, rather than features that require a critical mass of participants and partners to be useful. Finally, we favored testability and reliability, avoiding complex, cross-system orchestration and recommendation models in v1 so we can automate regression, stabilize operations, and scale usage with confidence.

These choices create some predictable disappointments. Patients will not see integrated produce-prescription redemption, in-app meal ordering, wearable-driven nutrition insights, or personalized recommendations at launch, so the initial experience skews clinical rather than lifestyle-supportive. Physicians and nurses will miss nutrition-aware decision support during prescribing and will have only one specialty template beyond the generic note, which may require temporary workarounds for certain clinics. Dietitians and culinary-medicine staff do not get dedicated telehealth nutrition tooling or automatic food-log imports on day one, which reduces the data available for counseling. Program administrators and funders will not have outcomes dashboards or produce-redemption analytics at MVP, and finance teams will not receive payer-ready reimbursement reports until nutrition modules are introduced. Retailers and food-service vendors will not yet see order-engine integrations, delaying direct partner value. Outreach and CRM teams cannot run targeted nutrition campaigns initially, which may slow program enrollment. Compliance is largely satisfied because audit and consent are present at launch, but they may still want nutrition-specific audit filters that we will deliver later as policy views over the same log stream. IT and DevOps lose the

chance to showcase end-to-end integrations early, but they benefit from a smaller, more stable operational surface for v1.

To mitigate these impacts and appease at least some stakeholders, we made several deliberate inclusions and design choices inside the constrained MVP. We kept secure messaging because it immediately improves access and adherence and gives patients a dependable communication channel, which we can later reuse for nutrition prompts and follow-ups without adding new systems. We included granular consent management from day one to build trust and reduce rework when we introduce nutrition data flows; by establishing consent infrastructure early, we avoid risky retrofits and help compliance sleep at night. We shipped one specialty template—OB/GYN—to prove the templating framework, reassure clinicians that configurable clinical forms are real, and keep a tight connection to nutrition-sensitive prenatal care that we can deepen later. We prioritized a robust, global audit console in the initial release so compliance and security have the visibility they need; nutrition-specific audit screens can be layered as filtered views rather than forcing new plumbing. Quietly, we also invested in contract-first foundations for future integrations by defining versioned interfaces, event vocabularies, and API stubs for concepts like external orders, program enrollment, and redemption. This does not expose functionality in the UI yet, but it reduces schema churn and integration friction when we move into Phase 2. Finally, we documented a staged backlog with explicit gating criteria—clinical reliability metrics, privacy reviews, support readiness, and data quality thresholds—so administrators, dietitians, and partners can see a credible path to their features and understand what must be true before we flip them on.

This MVP is the right first step because it completes the full clinical loop with safety controls while proving extensibility where it matters. Patients can register, message, view results, and receive prescriptions; clinicians can document, order labs, and prescribe within a role-secured and audited environment; administrators can onboard staff and monitor access; and the platform already contains the consent and templating infrastructure that nutrition and other specialties will rely on. By resisting the temptation to ship partially integrated nutrition features too early, we reduce operational risk, avoid brittle one-off connections, and give ourselves a stable core to grow from. The trade-off is fewer “wow” features at launch, but the payoff is a system teams can actually use immediately—and a clear road to high-impact nutrition capabilities without rework.