# SIS Add-On Package (Phases A–F): Cross-Phase Dependencies, Roles, Journeys, Schema, and Scale Notes

Version: 1.0 • Audience: Product, Engineering, Data, QA, Security, School/District Leadership

## 1) Cross‑Phase Dependencies Map

This section makes explicit how each phase depends on and feeds other phases. Use it to plan sequencing, API contracts, and QA scope.

A – Core SIS Foundation (District, Schools, Users/Roles, Classes, Scheduling, Attendance, Gradebook Baseline, Specialists)  
 ↳ Provides canonical IDs, rosters, calendars, attendance events, baseline gradebook entities.  
  
B – Curriculum & Lesson Planning  
 DEPENDS ON: A (classes, schedules, roles)  
 FEEDS: C (assessment blueprints/standards), D (weekly digests), E (coverage/pacing analytics)  
  
C – Advanced Grading, Assessment & Analytics Hooks  
 DEPENDS ON: A (gradebook containers, attendance), B (standards & lessons)  
 FEEDS: D (result notifications), E (item analysis, mastery), F (exports/APIs)  
  
D – Communication & Engagement  
 DEPENDS ON: A (rosters/roles), B (lesson summaries), C (grades/assessments)  
 FEEDS: E (engagement metrics), F (notification connectors)  
  
E – Reporting, Analytics & Insights  
 DEPENDS ON: A–D events and aggregates  
 FEEDS: A (intervention loops), D (nudges), F (warehouse/API contracts)  
  
F – Future & Integrations (SSO, OneRoster/Ed‑Fi, LTI, Data Warehouse)  
 DEPENDS ON: A–E stable contracts  
 FEEDS: External partners, district data platforms; informs versioning/back‑compat strategy.

Development Gate Checklist (per phase):

• API contracts documented (OpenAPI/GraphQL) and versioned.

• Event schema (for webhooks/CDC) published with examples.

• Test data sets spanning K–5 and 6–8 (incl. specialists, pull‑outs, accommodations).

• Role/permission mappings verified against the matrix in Section 2.

• Backward‑compatibility and migration notes recorded.

## 2) Unified Terminology & Role Permissions Matrix

Authoritative glossary and module‑by‑module permissions to eliminate ambiguity.

### Glossary (selected)

**District:** Top‑level tenant containing multiple schools and shared policies.

**School:** Organizational unit with its own calendar, bell schedules, and staff.

**Class (Section):** Teach‑by‑period entity: teacher + roster + schedule.

**Specialist:** Staff who deliver services via pull‑outs/push‑ins; may see multiple students simultaneously.

**Calendar Block:** Scheduled time window tied to a class or event.

**Lesson:** Planned instructional unit linked to standards and optionally to a calendar block.

**Assessment:** Measurable activity (quiz/test/performance) with scoring rules and optional online delivery.

**Assignment:** Gradebook item; may be created from a lesson or assessment.

**Intervention:** Action plan for a student with goals, dosage, owner, and review date.

**Parent‑friendly View:** Derived presentation excluding internal notes and accommodations.

### Role Permissions Matrix (summary)

Key = V(view), C(create), E(edit), A(approve), R(report/export). More granular scopes to be defined in IAM.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Module | Principal | Vice Principal | Dean | Staff Admin | Teacher | Specialist | Parent | Student |
| Roster & Classes | VCEA R | VCEA R | VCEA R | VCE R | V (own) | V (assigned) | V (child) | V (self) |
| Scheduling (Blocks) | VCEA R | VCEA R | VCEA R | VCE R | VCE (own) | VCE (own slots) | V (child) | V (self) |
| Specialist Pull‑outs | VCEA R | VCEA R | VCEA R | VCE R | A (approve/deny) | C E (request/edit) | V (child) | V (self) |
| Lessons | V R | V R | V R | V R | VCE (own/team) | V (student‑specific notes) | V (summary) | V (summary) |
| Assessments/Assignments | V R | V R | V R | V R | VCE (own) | V (assigned) | V (child) | V (self) |
| Gradebook | V R | V R | V R | V R | VCE (own) | V (assigned) | V (child) | V (self) |
| Attendance | VCE A R | VCE A R | VCE A R | VCE R | VCE (own) | V (sessions) | V (child) | V (self) |
| Messaging | VCE A R | VCE A R | VCE A R | VCE R | VCE (classes) | VCE (caseload) | VCE (child threads) | V (own) |
| Analytics | V R | V R | V R | V R | V (class) | V (caseload) | V (child summary) | V (self summary) |

Note: Staff Admin inherits Principal capabilities minus access to highly sensitive fields (discipline/IEP full text) unless explicitly granted.

## 3) End‑to‑End User Journeys (Representative)

**Admin – Start‑of‑Year Setup:** Create district and schools (A) → Import rosters via OneRoster (F) → Define bell schedules and calendars (A) → Publish curriculum maps and templates (B) → Configure category weights and grading periods (A/C) → Enable communication policies and digests (D) → Validate analytics dashboards (E).

**Teacher – Weekly Plan & Teach:** Open planner (B) → Drag lessons onto calendar (A) → Auto‑create linked assignments (A/C) → Specialist requests arrive; approve or propose alternative (A) → Teach; take attendance from block (A) → Post grades; auto‑notify parents (D) → View class mastery heatmap (E).

**Specialist – Manage Pull‑outs:** Propose sessions (A) → Teacher approves/denies with reason and alternate suggestion (A/D) → Conduct sessions; mark attendance (A) → Add notes tied to lesson/goal (B) → Review caseload impact report (E).

**Parent – Stay Informed:** Get weekly digest (D) → See upcoming lessons and assessments (B/C) → Receive alerts for missing work or grade drops (D/E) → Message teacher (D) → Track progress trends in portal (E).

**Student – Own Their Learning:** See schedule (A) → Review lesson summaries and resources (B) → Complete online assessments (C) → Check feedback and grades (C) → Receive nudges about missing work (D) → View progress dashboard (E).

## 4) Master Data Schema Overview (A–F)

High‑level ERD summary and table inventory across phases.

|  |  |
| --- | --- |
| Domain | Key Entities (IDs, Relationships) |
| Core SIS (A) | district(id)→schools(id); users(id, role); classes(id, school\_id, teacher\_id); enrollments(student\_id, class\_id); calendar\_blocks(id, class\_id, start, end); attendance(id, student\_id, date, status); gradebook\_categories(id, class\_id, weight); assignments(id, class\_id, category\_id, points, weight); grades(id, assignment\_id, student\_id, score, status) |
| Curriculum (B) | curriculum\_maps(id, school\_id, subject, grade\_level); unit\_plans(id, map\_id?); lesson\_plans(id, class\_id, date); lesson\_schedule\_links(lesson\_id, calendar\_block\_id); lesson\_assignment\_links(lesson\_id, assignment\_id); standards(id, code) |
| Assessments (C) | assessments(id, class\_id, blueprint, delivery\_mode); items(id, assessment\_id? bank\_ref); attempts(id, assessment\_id, student\_id); responses(id, attempt\_id, item\_ref, score); scoresheets(id, attempt\_id, total\_score); retake\_policies(assessment\_id, rule) |
| Communication (D) | threads(id); messages(id, thread\_id, sender\_id, recipients[]); deliveries(id, message\_id, channel, status, timestamps); digests(id, audience, schedule) |
| Analytics (E) | fact\_grades, fact\_assessments, fact\_attendance, fact\_schedule, fact\_pullouts, fact\_communications, fact\_interventions; dim\_student, dim\_teacher, dim\_class, dim\_school, dim\_subject, dim\_standard, dim\_calendar |
| Integrations (F) | external\_connections(id, type, creds\_ref); mappings(id, external, internal, transform); webhooks(id, topic, target\_url); exports(id, job, format, location); imports(id, source, status, diagnostics) |

ERD Notes: All domains use stable UUIDs; cross‑domain links via foreign keys; audit tables capture create/update with actor and reason; soft‑delete for recoverability where appropriate.

## 5) Scalability, Integration & Compliance Notes

• Partitioning & Indexing: time‑based partitions for facts; composite indexes (student\_id, class\_id, date) for hot paths.

• Caching: teacher dashboard and class roster caches with TTL; pre‑compute nightly aggregates for E dashboards.

• Event Bus: use domain events for D/E/F; ensure idempotency keys for imports/grade pushes.

• API Versioning: semantic versions with ≥12‑month deprecation windows; compatibility tests per connector.

• Observability: tenant‑scoped metrics; SLOs for API p95 and webhook delivery; anomaly alerts on sync drifts.

• Security: RBAC with row/column‑level policies; encrypted PII; consent preferences respected in all channels.

• Accessibility & i18n: WCAG 2.1 AA across portals; multilingual templates for D digests and messages.