Solutions

2.1

- Web form, API, reminder channel ✓.

2.2

- Parameterised queries plus a length cap – tested by feeding `' OR '1'='1` and asserting zero rows leaked ✓.

2.3

- Documented refusal with alternatives, traceable to a code ✓.

3.1

- Assets touched: loan records, member ids ✓.
- Attackers: opportunistic outsider with the API, an insider testing another member's data, automated scanners ✓.
- Injection threat: crafted ids – mitigation: parameterised inputs and schema validation; test: payload ids return structured errors, zero leakage ✓.
- Authorisation threat: member A renews member B's loan – mitigation: ownership check at the domain boundary; test: cross renewal refused with 403 ✓.
- Matrix: member-renew and librarian-renew cells allow (2 new allows); the rest of the renewal column denies ✓.

3.2

- Distinction: anonymised aggregate with consent is a different artefact from raw per-member history – the harm profile changes (re-identification risk remains the key question) ✓.
- Codes: privacy harm and public interest outrank employer preference; the engineer documents rather than quietly obeys or quietly refuses ✓.
- Response in writing: the concern (re-identification), the conditions under which it is defensible (consent, aggregate thresholds, review), alternatives (sponsorship, grants), and the decision record – the paper trail is the point ✓.