# Project Spec Template (Design-Only)

Use this template to define your prediction/design project. Keep it concise (2–3 pages max) and link to supporting artifacts where appropriate.

## 1) User & Decision

* Who will use this? What decision does your prediction/design inform?
  + The target of this API are web fiction readers. The goal of the predictor is to find web fictions that the user will enjoy based on their reading history.

## 2) Target & Horizon

* Target (binary/numeric/ETA) and when it’s defined (e.g., next 5 minutes, this session, by deadline).
  + The target will be a binary check if the user has added it to their library on an existing website for web fictions within a week of the suggestion.

## 3) Features (No Leakage)

* Only information available at prediction time. Note what you exclude to avoid leakage.
  + The features that the model will be based on are the genres and tags of the web fictions that the user has read before, and if available, their ratings or likes/dislikes given to them. To avoid leakage, stats such as how much they read after the suggestion will be not used when training the model.

## 4) Baseline → Model Plan

* Baseline you can implement immediately (rule/heuristic).
  + A baseline rule could be to see if the recommended web fiction has an overall positive review, and if so, predict that it will be enjoyed by the user.
* One simple model (e.g., logistic/tree/ETS) and why it’s better (hypothesis).
  + Since this would be a content-based recommender system, it could use a linear regression (ridge in sklearn) model built on the user’s reading history.

## 5) Metrics, SLA, and Cost

* Metric(s): AUC-PR/MAE/etc. State why they fit harms/benefits.
* SLA: p95 latency, max cost per 10k predictions.

## 6) API Sketch (if applicable)

* POST /predict request/response schema. Include example payloads.

## 7) Privacy, Ethics, Reciprocity (PIA excerpt)

* Data inventory, purpose limitation, retention, access (link your PIA).
* Telemetry decision matrix (value vs invasiveness vs effort).
* Guardrails: k-anonymity, jitter/aggregation, opt-ins, disclosure.
* Reciprocity: value returned and to whom.

## 8) Architecture Sketch (1 diagram)

* Major components and data flow. Note trade-offs and alternatives.

## 9) Risks & Mitigations

* Top 3 risks (technical/ethical) and how you will test or reduce them.

## 10) Measurement Plan

* One minimal experiment to validate the baseline vs model (offline acceptable).
* How you will measure SLA (latency/cost).

## 11) Evolution & Evidence

* Link a git hash (or range/tag) that shows the design’s evolution (commits, README updates, diagrams).
* Insight memo link (3 insights), assumption audit, and Socratic log references.