SafeNutriKids Metadata-Based Localisation Layer

Guidelines for Multilingual and Multicultural Nutrition Content Delivery

1. Purpose and design principles

The metadata-based localisation layer provides a systematic mechanism for delivering linguistically and culturally appropriate nutrition content to children and their caregivers. Its primary function is to select the most suitable content variant from a catalogue of alternatives on the basis of explicit metadata, such as language, country, cultural context, age range, and dietary pattern. The layer is intended to serve as a reusable component that can be embedded in different digital platforms within and beyond the **DRG4Food** ecosystem.

The design follows three main principles. First, localisation decisions are made using transparent and interpretable rules that are encoded in metadata and selection logic rather than buried in application-specific code. Second, the layer separates content from code so that nutrition experts, educators, and translators can work on content without modifying the software itself. Third, the selection process is deterministic and reproducible, so that a given input context will always lead to the same content variant, which simplifies testing, auditing, and refinement.

2. Data model for content variants

Each content item, such as a lesson on hydration, a game description, or a recipe explanation, is represented by a logical key. For each key there may be several content variants, and each variant is described by a combination of metadata attributes. The core attributes are a stable identifier, a language code following recognised standards, an optional country code, an optional cultural tag, an optional dietary pattern tag, optional minimum and maximum ages, and a payload that holds the actual content.

The identifier is a human-readable string that uniquely distinguishes the variant and is suitable for logging and debugging. The language attribute should generally follow the ISO 639-1 two-letter code. The country attribute should follow the ISO 3166-1 alpha-2 code and may be used when content is adapted to a particular national curriculum or dietary guideline. The culture attribute provides a higher-level classification of cultural context, for example central_europe or muslim_majority, which can influence examples, food choices, and imagery. The diet attribute may indicate omnivore, vegetarian, vegan, or other patterns relevant to personalised nutrition. Age boundaries are used to tune content complexity and relevance for children in specific age ranges.

The payload stores the content itself. It can be plain text, marked-up text, a structured JSON object, or another representation agreed within the platform. The localisation layer does not interpret the payload; it simply returns the chosen variant to the caller. For this reason, it is advisable to establish and document a consistent payload format for each category of content so that client applications can render it appropriately.

3. Configuration and catalogue management

The localisation layer is configured by supplying a catalogue of content variants as a structured data object. In the reference implementation this catalogue may be provided as a Python dictionary or loaded from a JSON file that maps content keys to lists of variant descriptions. In production use it is recommended to store the catalogue in a format that is both human readable and machine parseable, such as JSON or YAML, and to keep it under version control.

Content governance processes should treat the catalogue as a central artefact. Nutrition experts, educators, and translators can submit changes to the catalogue through a workflow that includes peer review and quality assurance. Each change should be accompanied by a clear justification, for example alignment with updated national dietary guidelines or improvements in child-friendly language. Releases of the catalogue can be versioned independently from the software so that users of the system can refer to specific content sets when documenting evaluations or studies.

Environment-specific configuration is often necessary. For example, a pilot deployment in one country may require only a subset of all possible languages and content keys. In such cases, separate catalogue files can be maintained for different deployments, or a single master catalogue can include a deployment tag that enables filtering at runtime. The localisation layer itself remains unchanged and simply operates on whatever catalogue is supplied at initialisation.

4. Selection algorithm and scoring logic

The localisation layer evaluates all variants available for a given content key and ranks them according to a scoring function that compares the variant metadata with the context provided by the caller. The context typically includes language, optional country, optional cultural tag, optional dietary pattern, and optional age. The scoring function assigns positive scores for each dimension that matches between the variant and the context. In the reference implementation, an exact language match is given the highest weight, followed by country, cultural context, dietary pattern, and age range inclusion.

For each candidate variant the algorithm computes a score by adding the weights corresponding to the matched attributes. The variant with the highest total score is selected and returned. In the case of a tie, the first variant with the highest score in the catalogue order is chosen, which ensures deterministic behaviour. If all variants receive a zero score, indicating that none match the supplied context even partially, the algorithm falls back to the first configured variant for

that content key. This conservative fallback ensures that a user never receives an empty response, but the event can be logged to trigger review of coverage gaps in the catalogue.

The scoring strategy can be adapted to local priorities without changing the overall structure of the algorithm. For example, a deployment that considers age appropriateness more important than country-specific differences may adjust the weights accordingly. Such adjustments should be documented and tested to avoid unintended changes in behaviour. It is advisable to accompany any change in scoring logic with a set of regression tests that verify the selection outcomes for representative contexts.

5. Integration patterns with host platforms

The localisation layer is intended to be embedded in backend services responsible for content personalisation. A typical integration pattern is that a recommendation engine or rules-based controller first determines which content key is appropriate for a given user interaction, and then invokes the localisation layer to obtain the most suitable variant of that content. The engine supplies the child's language, country, age, and any known dietary or cultural preferences as input to the selection function.

In synchronous integrations the selection call is performed as part of handling an HTTP request and its runtime cost must therefore remain small. The reference implementation operates entirely in memory and is suitable for such use cases. For larger catalogues or environments with strict latency constraints, it may be beneficial to pre-load and cache the catalogue at application startup and to avoid disk or network operations within the selection path. In more complex architectures, the localisation layer may be exposed as a separate internal microservice that other components call through lightweight remote procedure calls.

It is recommended that host platforms log both the requested context and the identifier of the selected content variant for key interactions. These logs can be used to analyse usage patterns, to detect missing translations, and to support evaluations of fairness and inclusiveness across different user groups. When such logs contain child-related identifiers they should be handled under the same data protection policies as other behavioural data collected by the platform.

6. Cultural and nutritional adaptation process

The localisation layer is only a technical enabler; it does not guarantee that the content itself is culturally sensitive or nutritionally sound. A structured content development process is therefore required. For each content key, nutrition experts should define the core message and the evidence base. Language professionals and native speakers should provide wording that is accurate, age appropriate, and engaging for the target group. Cultural mediators or local educators should review examples, food items, and scenarios to ensure that they fit local practices and avoid stigmatisation.

The metadata model should be used actively during this process. When a variant is crafted for a specific cultural context, the culture field should be set accordingly, and the age range should be specified on the basis of developmental considerations. Dietary pattern tags should be used to cover common restrictions, such as vegetarian or lactose-free diets, in a way that is aligned with clinical and nutritional guidance. The use of generic variants with broad age ranges should be limited to avoid overly generic messaging that does not speak meaningfully to children.

Feedback from children, parents, and educators should inform iterative refinement. Qualitative evaluations, such as focus groups or classroom observations, can reveal whether localised content resonates with its audience and whether any aspects are confusing or inadvertently exclusionary. The catalogue and metadata should be updated in light of such findings, and changes should be recorded transparently to support accountability.

7. Fairness, inclusiveness, and age appropriateness

The design of the localisation layer supports fairness and inclusiveness by making content variation explicit and reviewable. However, explicit practices are required to realise this potential. Content authors should ensure that each language and cultural group targeted by the platform has access to core educational messages of comparable quality and detail. Differences should reflect genuine contextual needs, such as local dietary staples or educational standards, rather than lower investment in certain groups.

Age appropriateness requires systematic attention. Messages aimed at younger children should use shorter sentences, concrete examples, and visual cues, whereas messages aimed at older children can introduce more abstract concepts and risk trade-offs. The age_min and age_max fields provide a mechanism for capturing these distinctions in the catalogue and for preventing content that is too complex or too simplistic from being shown to a particular age group.

The treatment of sensitive topics, such as body weight, dieting, or socio-economic disparities in food access, should be handled with particular care. Variants that discuss such topics should be reviewed by professionals with expertise in child psychology and public health communication. Metadata may be used to restrict such content to specific age ranges or contexts where qualified adult guidance is available, such as teacher-led classroom activities.

8. Testing and quality assurance

Quality assurance for the localisation layer has both technical and content dimensions. On the technical side, automated tests should verify that the selection algorithm behaves as expected for a suite of predefined contexts and catalogues. These tests should cover normal cases, edge cases where several variants have similar scores, and failure cases where the requested content key does not exist. They should also verify that modifications to the scoring weights do not inadvertently alter selection results in established scenarios.

On the content side, systematic linguistic and cultural review is required. Each variant should be checked for spelling, grammar, clarity, and cultural appropriateness. A practical approach is to maintain checklists for each language and cultural context and to record review outcomes in a structured way. Random samples of delivered content in live systems should be periodically evaluated against these checklists to ensure that quality does not degrade over time as new content is added.

Automated tooling can assist in detecting catalogue gaps and inconsistencies. Scripts can be written to flag content keys that lack variants for certain languages or age ranges, or to detect overlapping age ranges where the intended behaviour is ambiguous. Such tools reinforce the discipline of maintaining a high-quality catalogue and reduce the risk of children encountering content that is unintentionally mismatched to their context.

9. Open source packaging and collaboration

Released as open source, the localisation layer should be packaged as a library that can be installed via standard dependency management tools. The public interface should be clearly documented, including the expected structure of the catalogue and the method signatures for initialisation and selection. Example catalogues and short code samples demonstrating typical use cases should be included in the documentation to lower the barrier for adoption by other projects.

Collaboration with the wider **DRG4Food** community and other initiatives is encouraged. An open issue tracker can provide a place where users report bugs, request new features, or propose enhancements to the metadata model. A contribution guide should describe coding standards, testing requirements, and the review process for pull requests. Where possible, alignment with emerging standards in nutrition education and digital health should be sought, so that the localisation layer can serve as a reference implementation for metadata-driven content adaptation in child-oriented digital services.