

New reviews posted via form on local website:

This is already done and all working as required. The customer enters review in response to email invitation. On clicking submit button this is stored in `unity_culinary.product_reviews` table and administrator(s) notified that it requires approval.

Approve reviews:

This also complete and working. Administrator clicks link in email to approve, which changes "status" value to 1 and updates "reviewed" column with timestamp.

Will need to also update "verified_buyer" column to 1, or set to 1 by DB default value?

Translate to other languages:

We will need a scheduled task/cron job that runs daily in Falk Central app.

This first needs to retrieve reviews using a new API function in `unity_culinary.product_reviews` table where:

"updated" or "reviewed" = previous day's date
and "translated" = 0

Loop through retrieved reviews:

Then check that "product_id" exists in `falk_central.products` table.

If product match found (no further action if no match):

1. Insert (or update existing) single record in `falk_central.product_reviews`
2. Insert (or update) review content in `falk_central.product_review_content` in same language as submitted by customer.
3. Retrieve list of other required languages from `falk_central.locales`

Will need distinct in select statement as `language_code` can be used in more than one `iso_code`. – e.g

```
SELECT DISTINCT
language_code
FROM locales
WHERE conv_prod_reviews = 1
```

4. Then translate the review into each required language (retrieved list from locales less native language) and save in `falk_central.product_review_content`

Retrieve reviews to display:

Reviews will be retrieved from falk_central.products reviews for display on public website.

The DB query should:

- a) Limit the number select for pagination
- b) Select reviews written in same language as website (non translated) first – this would be en for falkcoppercookware.com and de for falkculinair.de
- c) Sort these by (i) most recent by date, (ii) oldest by date, (iii) lowest stars, highest stars, most popular (highest number of votes)

Visitor clicks helpful link – increase summary count

Each review has a link to click if customer found it helpful with a count- for this we will need:

1. An API function to increase the value of unity_culinair.product_reviews.votes by 1;
2. A DB function in dev.2017.falkcoppercookware.com app to increase falk_central.product_reviews.votes by 1

This identical count is being stored in both tables for “future-proofing” (we intend to add non-Falk websites next year and process orders via the Unity OMS app.

Update if customer modifies review:

This will be only be possible via the customer account panel (not yet done) – we will set this to flag edited review for approval and reset unity_culinair.product_reviews.translated back to 0

Database Tables – I have updated unity_culinair.product_reviews on both AWS and Green servers and other tables on AWS server.

unity_culinair.product_reviews

1. Dropped “recommend” column (we are not going to use this going forward)
2. “rating” column renamed “star_rating”
3. Added “verified_buyer” column
4. Added “translated” column

New reviews will be inserted into this table from local website via API, and approved (or amended) via Unity OMS app.

New, or updated, reviews requiring translation in falk_central identified with:

unity_culinair.status = 1 & unity_culinair.translated = 0

falk_central.product_reviews

New table

This will contain non-text (everything not translated) for display on the local websites.

Will only contain reviews that have been approved in unity_culinair.product_reviews table

All column values to match in unity_culinair.product_reviews

falk_central.product_review_content

New table

This will contain the translated review text. We will have multiple entries for each review_id – e.g. review_id = 6666, with language = en/es/de etc...

falk_central.locales

New table

Will define languages used in local websites (active = 1). We will use this to generate translations (conv_prod_reviews = 1)

falk_central.products

I have updated the data in this to include all current products

