USER table defines the attributes needed for storing user accounts.

|  |
| --- |
| **USER** |
| References:  - REVIEW  Referenced by:  - REVIEW  - COMPANY\* (optional) |
| Contains:  - account\_ID (primary key)  - username  - display\_name\* (optional)  - password\_hash  - password\_salt  - email  - review\_IDs (foreign key/multival)  - is\_banned\* (optional) |

Review table defines the attributes needed by reviews.

|  |
| --- |
| **REVIEW** |
| References:  - USER  Referenced by:  - USER  - PRODUCT\* (optional) |
| Contains:  - review\_ID (primary key)  - account\_ID (foreign key)  - email (stored only if anonymous review in place of account\_ID)  - display\_name (use username if display\_name is omitted for registered users, cannot be foreign key)  - product\_ID\* (optional, foreign key to product table)  - posting\_date\_time (prerequisite for discussed usage of cookies)  - posting\_IP  - review\_text (limit to 2000 chars)  - isValid (Boolean flag for displaying or hiding review) |

A table for storing products was suggested, hence the inclusion of references to them. It was decided, however, to focus on an abstracted single-product system for now, as adding product referencing should be possible partway through development once the preliminary framework is constructed.

A table for storing companies and referencing any accounts tied to that company was also considered as a way to manage posted products. This, however, moved beyond the purview of our assignment and should be considered a long term goal placed behind all other, more important tasks.

Some discussion was had about cookies. This is something we should look to implement in the HTML5 webpage layer if it is not obscenely difficult. We can store a cookie with posted review IDs and the post date to enable editing within 30 days even for anonymous users, and as an additional layer of detecting bad actors. We can store an automatic login cookie for 48 hours for convenient “keep me logged in” options. These should be supported for Google Chrome and Mozilla Firefox. Other browsers can be omitted.

We can use public-facing IP, email, and discrete content comparisons for initial screening. Fuzzy matching algorithms and potentially some machine learning algorithms will be required for dealing with an organized bad actor with a large group of subordinates.