Database Description

We used a nosql document style database called Firebase for our project. The database consists of four main collections of data

- Topping Categories (itemCat)
- Topping Types (itemType)
- Orders
- Users

This data is secured and/or validated as appropriate and described further. The database structure is described in more detail in the diagram.

Topping Categories

Categories may either be exclusive or inclusive. Pizzas may only contain one topping from a category which is exclusive. This flag is ideal for options like crust, size, or anything else that need not overlap.

Security

Topping categories may only be created, modified, or deleted by administrator accounts. Reading or listing topping categories is open and does not require any authentication.

Topping Types

Types must belong to a category to be added on a pizza. Types include price information to allow for variance within a category.

Security

Topping types may only be created, modified, or deleted by administrator accounts. Reading or listing topping categories is open and does not require any authentication.

Orders

Orders contain most relevant data to an order to improve speed using data locality. Orders contain pizza configurations, payee contact info, delivery location, price and payment details. An order is also directly linked with a user for security.

Validation

Orders are not valid without a user key. Operations that result in an order without a user key are rejected by the database.

Security

Orders may be created by any user. However, once an order is created, it may only be read by the keyed user. If ordering without an account, a cookie with store the user's authorization token for this record. Unless this token is upgraded to a full account by logging in or signing up, access to the order will be lost when the cookie is cleared.

The user has access to modify their order until they finalize it by paying for it. After this point, the order becomes read-only. Administrator accounts may read or write any order.

While an authorized user is granted access to modify their order, modifications to pricing is automatically rejected, even for administrators. Pricing is updated by secure server-side database triggers to maintain integrity.

Users

User records store non-sensitive information about a user like keys to orders, and long-term address information. Login credentials are managed by a separate process isolated from the database API, and so this information will not be discussed in the database description document.

Security

User information may not be created, modified, deleted, or viewed by any user other than the user himself and administrators.

Static Objects

In addition to the primary collections of data, the database also stores two unique static objects.

Default Pizza

This object stores the default pizza configuration. When new orders are created, or new pizzas added to an order, this object is copied into the appropriate record.

Security

Only administrators may modify this record. However, no authentication is required to view the record.

Tax Rate

This configuration value is used to determine the rate tax is applied to the order subtotal before determining the final total.

Security

Only administrators may modify this record. However, no authentication is required to view the record.