1. **What new logic and/or APIs would need to be developed?**

When a user opens up the app and clicks on the camera icon on the presentation layer, they send a user request, together with their current location and current direction they are facing, to the logic layer, which does 2 things: 1) sending a query to the storage layer to extract data on nearby restaurants, and 2) every 0.5s, taking a still image from the scan, tracking subsequent user actions, and saving the data to the storage layer (for training and improving our algorithms).

After the logic layer sends a system request to the storage layer, the storage layer gathers the data that is requested by the logic layer and sends it back to the logic layer.

Then, the logic layer does 3 things: 1) image recognition algorithm (computer vision) to recognize which discontinuous blocks represent storefronts, 2) optical character recognition algorithm to recognize logos / other symbols from each storefront, and 3) use data from the nearby restaurants, results from the OCR algorithm, and data on current location and current direction the user is facing to pinpoint the exact store.

Finally, the logic layer sends the information back to the presentation layer to display the store’s information on the app’s user interface.

All data requests directed from the logic layer to the storage layer should be through API: for example, sending a query to extract nearby restaurants based on current location should be packaged as an API request and the storage layer will send back a list of nearby restaurants in return.

1. **What new/unique data is necessary to deliver on the product value proposition?**

The new data that we need is the labeled image data (including street signs / logos / banners, etc.) for the storefronts so that we can train the computer vision model to identify storefronts.

1. **How would we estimate the computing capacity required by estimating:**
   1. Throughput Needs (how many users, how much (what are they storing), # of transactions per user):  
        
      We will monitor the number of simultaneous active users and requests per second for this feature. If there are limited active users and requests per second, we can simply implement throttling to lower implementation costs. If there are high volumes of active users and requests per second, we should consider vertical scaling or horizontal scaling. If the budget doesn’t allow us to implement a vertical scaling application without modification, we will go for horizontal scaling where software evenly distributes incoming traffic across multiple servers.
   2. Storage Needs - assume your company uses relational databases (vs NOSQL) - what are the primary data-tables required in the DB?  
        
      For relational databases, the primary data table will store for each restaurant the following information: 1) photos, 2) location, 3) basic information, 4) ratings, 5) reviews, 6) menus, 7) website. The *photos and interactions* data table is the only data table that would cause incremental storage. We are considering rolling two years of storage for this table so that it can reduce the system storage pressure.