Questions for Requirements Elicitation Meeting

Oil price updates
1) How frequently must oil prices be updated?
    - Specified website indicates only monthly updates
2) What regulatory body determines the oil prices?
    - Specified source is Pakistan State Oil

Transactions
1) How are transactions currently accepted?
    - If card, what payment platform(s) are used?
2) What currencies/vendors are supported?

Products
1) What categories of sales are possible with Ali's Oil?
    - Currently known is oil and "vehicle-related merchandise"/"basic automotive maintenance"
    - "Products and services" are mentioned
    - Restaurant and convenience store attached
2) What grades of oil are sold? (i.e. regular, premium, diesel, propane etc.)
3) Is the product line expected to grow? Should the solution deal with this?
4) How are products currently identified (QR/barcode)?

Customers
1) Who makes up the primary customer groups that buy from Ali's Oil?

Employees
1) How are employees paid? Is this an issue that needs addressing?
    - States "sends our employee wages weekly"
2) Is this system integrating with an existing POS (point of sale) system or replacing it entirely? As in, will filling attendant enter information into the software solution or will the software process the transaction? Clarify section 2 vs 4.
3) Go through end of shift duty for filling attendants, are they just counting cash on hand vs receipt balance?
4) How should the inventory be updated when new product arrives (manually or automatically)?

Analytics
1) What metrics is Ali's Oil looking for? (i.e. gross/net revenue, recurring/new/total purchasers etc.)
    - RFP mentions comparing monthly and annual "financial" statistics
2) For section 5 paragraph 1, should inventory be viewed on a day-by-day basis with historical data?
3) How should reports be generated? Daily, monthly, yearly?
4) Is the system running daily reports searching for missing inventory or money? Is this how it will help to catch thieves?

Data Storage
1) "The system must be able to access all records from within the last 7 years."
   Does this apply to all records, or just transactions? Are there any laws the company must comply with for data retention?
2) "Make shop inventory and funds manually editable"
   Will the system track the company's bank balances, or simply report its revenue and expenditures?
   What transactions can be performed to edit this data?
3) We should clarify how the debt occurs. Is it from specific purchases (ex. gas purchases only)?
   Should debts be associated with a specific customer transaction, or simply be dollar amounts?
4) Will the system handle employee records and payroll?

Security
1) What data can be seen by who? What is unauthorized? What isn't? (For intended users)
   - Track outstanding customer debts.
   - Track inventory
   - Summarize cash flow
   - Permit manual adjustments
   - Records of all transactions.
2) Remote access? What do you want to be accessible remotely?
3) Can a filling station attendant READ any data or only write? Are there any transactions that only a manager can complete?

Other
1) Do you have other file formats for data storage to provide us besides .pdf?
2) Is documentation sufficient for users to learn or must be done through training?
3) Are reports automatically generated or must be manually initiated by users?
4) Budget? (Especially important for solution, how much money is spent on how employees will interact with software? Ie; handheld device vs one central computer)
5) RFP says "deposits excess cash to the bank daily". How is the system depositing physical cash to the bank? Same thing for the safe. (section 4, second last point on store manager)

Delegated Jobs
- Everyone pick a github issue and work on it for the designer website
- Vote in the Slack poll for your favourite designer team name
- Charlie and Daniel will take on the client-side (QED Technologies) for elicitation
- Kristian and Oliver will take on the designer side (name TBD) for elicitation
- Oliver, Kristian, and Charlie will be monitoring designer-site project github pull requests