



Pocket Grocery List

- Team Grocery
- cse403grocery@u.washington.edu



LCA Overview

- Operational Concepts
- Case Scenarios
- User Interaction
- System Architecture
- Risks



Operational Concepts

- Overview
 - Pocket Grocery List is a program for the Pocket PC platform that allows a user to create and manage lists of groceries and to look up the prices for said groceries at a local Supermarket along with a map and directions to that Supermarket.



Operational Concepts (Cont.)

- Non Goals
 - This version will not support multiple chains of supermarkets.
 - Actual store pricing data will not be used.

Operational Concepts (Cont.)

■ Target Users

- Pocket Grocery List is targeted at Safeway shoppers who own PDAs and who are interested in conveniently budgeting their time and/or money.

Case Scenarios

■ Usage Scenarios

- Cindy is a typical mother of a family. She is always busy and rarely has time to waste.
- John is a hard-working businessman who is constantly going on business related trips.
- Bob is a college student who wants to organize a party.

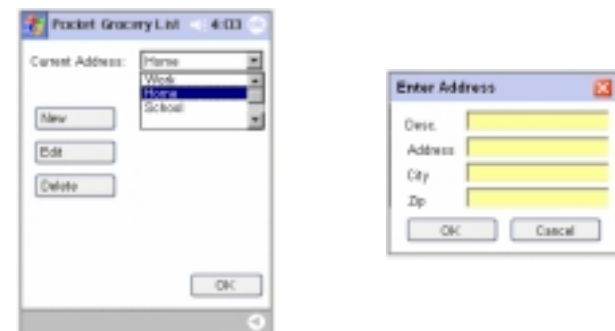
User Interaction

■ Opening Screen



User Interaction (Cont.)

■ Starting Location Information



User Interaction (Cont.)

■ Grocery List Editor



User Interaction (Cont.)

■ Search Results



User Interaction (Cont.)

■ Map and Routing Display



System Architecture



1. Request item pricing information
2. Find the closest grocery store
3. Return the store's address, map, and routing information
4. Search store's item database
5. Return store's item pricing information
6. Return store address, map, driving directions, and store data to the client

System Architecture (Cont.)

- Local Storage
 - Provide users with popup Save/Open dialog.
 - Only accept files with an “.xml” extension.

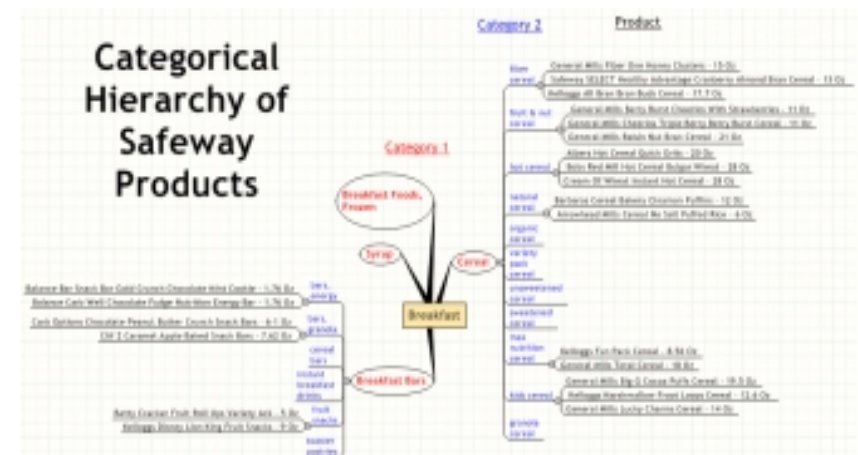
System Architecture (Cont.)

- Web Services
 - Interface between users and the backend
 - Controls the interaction between Client & Server.
 - Implementation details of the components are completely isolated.

System Architecture (Cont.)

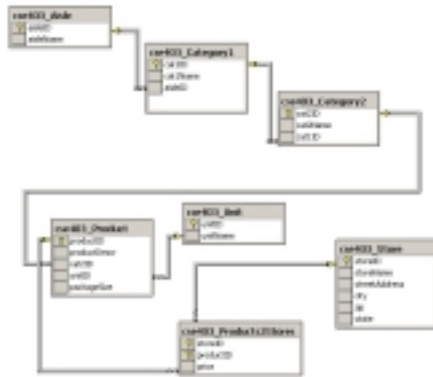
- MapPoint
 - FindAddress
 - GetMap
 - CalculateSimpleRoute

System Architecture (Cont.)



System Architecture (Cont.)

Database Schema



System Architecture (Cont.)

Database Stored Procedures

- Procedures pertaining to store info retrieval
- Get store info by specifying address parameters

```
getAllStores  
getAddressByStoreID  
getStoreIDByAddress  
getStoresBycity  
getStoresByState
```

System Architecture (Cont.)

Database Stored Procedures

- Procedures pertaining to product info retrieval
- Get products by specifying product description and storeID parameters
getProductsByStore
getProductBystore

The approach used by the getProductsByStore procedure is yet to be determined since its effectiveness will depend upon the product descriptions as well as category names in the data.

Risks

- Effects that multiple concurrent database queries might have on performance
- Because the product selection approach is dependent upon the data, there is still some uncertainty as to the difficulty of this problem but we do not anticipate it to be a show-stopper
- Since we rely on the UW's web server, Microsoft's MapPoint web services, and UW's SQL server, if any of these fail our system will also fail
- Security issues of having personal information being intercepted
- The Pocket PC has very limited storage capacity (i.e. 50Mb) . Thus, we are concerned that a user may run of space since