CS 419 Project: Corvallis Reuse and Repair Directory

Background

The purpose of the Reuse and Repair app is to provide the Corvallis and outlying community a way to easily locate businesses that will

1. take reusable items that can be sold to the public as used items, and
2. take and repair items.

At this point we have only a paper flyer that people can use to find out which businesses take what items, which is not always convenient to use. The Reuse and Repair app will allow the public to view a list of categories and then click and go to a list of businesses that take reusable items. The Reuse and Repair app will provide a map showing where the businesses are located and their webpage if there is one, phone number, address, and hours of operation. The Reuse and Repair app will also, for the sake of convenience, provide a link to the local company (Republic Services) responsible for collecting recyclables so the public knows what items can be recycled in case their items are unusable and unrepairable.

You will be given client contact information upon request.

Project Description

You will be assigned to a group, and given the task of building one of:

1. An iOS application implementing the above.
2. An Android application implementing the above.

For each of the above, there will also be a web management interface needing to be created. This interface will be used by the Corvallis Sustainability Coalition to keep the mobile applications up to date with which places take which items. It should also allow for dynamic updates of the mobile apps categories and items lists. In case this isn't clear, that means it must be secured in some fashion. It also must integrate within the Corvallis Sustainability Coalition's web hosting environment.

The application should be titled "Corvallis Reuse and Repair Directory".

Here is the link to the Republic recycling depot indicating what they accept:

> http://site.republicservices.com/site/corvallis-or/en/documents/corvallisrecycledepot.pdf

Here is the link to what Republic takes in the curbside recycling containers:

> http://site.republicservices.com/site/corvallis-or/en/documents/detailedrecyclingguide.pdf

The client will want these two links somewhere on the first page of the app to make it easy for folks to understand what can be recycled in Corvallis.

The core of this application is actually the internet service with which the mobile interfaces communicate. You can write this using any tools that run on the hosting service you choose. EECS provides both PHP

and CGI web interfaces which are usable by you. If you prefer to host elsewhere, and use alternative tools (flask, django, etc.) you are welcome to do so, but careful documentation will be required for replication. You can also use a non-web based service, if you so desire – think some sort of client server application on a port other than 80.

This service could take any form, but needs to be accessible to the outside world, including your web and mobile interfaces. It will be the source of all data, meaning mobile and web interfaces won't need a direct database connection, but will rather parse out some form of structured data from the service.

Also, here is the link to the App (it is called Recyclepedia) that the client would like to use as a template:

> http://www.rcbc.ca/services/recyclepedia-app.

Further, there are 3 Excel spreadsheets with the current information in them. These include the categories for the application, the list of repair businesses, and the data from the current paper flier in easy to use format. The current paper flier and these 3 Excel files will be available for you on the course website.

Database

There are 3 primary database options.

The first option is the more traditional, with the database stored on a dedicated server, using mysql. No specific layout or engine are required, and should be chosen for solid technical reasons.

The second option is to use an embeddable database, specifically sqlite. This will be stored as part of your application package, and will typically live in a specific directory.

The third option is to use one of the myriad NoSQL options. If you take this route, ensure adequate documentation is provided so that end users can replicate it.