Svpply

Main Goals

To gather from each product:

1. Product Status(available, on sale, sold out)
2. Pricing data(amount and currency)
3. Categorization (gender, product type)

Preparation for Meeting with Svpply

1. First meeting: **Tue 12th at 2:00pm**

2. Goals of the meeting
    - a Build communication channels by exchanging contact information, paying attention about what concerns them the most.
    - b Build trust demonstrating our commitment to getting the job done and having a positive impact on their ability to create value with this tool.
    - c We should go to the meeting knowing or wanting to know the following business issues:
        - i How do they make profit?
        - ii What are their current problems besides the one we are addressing?
        - iii Where are they going? Focus is on growing in size or revenue?
        - iv Who is their main customer?
        - v How does the main customer use the website in a way that generates profits for them?
        - vi What's their biggest non API enabled stored?
    - d Get answers these application questions:
        - i How many ways can I add a product? What happens if many people have added the same product from different websites?
        - ii What's the scope the Svpply wants to cover, US or World wide? If just for US, can we forbid adding foreign products by checking the websites' locations? If for the worldwide, can we make a connection between the currency with websites' locations to identify the currency automatically?
    - e Get answers these technology questions:
        - i Have they done something about this issue before? Can we built on something previously done or at least don't waste time getting where you are right now?
        - ii What happens if any store is asking not to crawl?
        - iii Can we get at registration time (See idea #1):
            1. What the JavaScript got and what the user inputed.
            2. A trigger to our system to pull the product page.
    - f Go over solution ideas.
    - g Logistics: access to test data, developers, computer resources.
    - h Define time and date for weekly Spring Review & Planning.
      Option 1: (Wed 1pm - 2:30pm at client's office) *

Idea #1

1. At registration time WebCrawler is notified so that it can download a copy of the product page.
2. Both what the JavaScript initially got (price, category and status) and what the user imputed is stored.
3. Based on previous data an accuracy score is calculated. (Both by store and by product). Information regarding the *volatility* of the store/product is saved. This could help us identify when the JavaScript has lost effectiveness for a specific site because of a change.
4. Because a store will probably display data in the same places, **in batch mode**, a store **LAYOUT profile is built.** A specific section(s) of page are identified as the holders of the product information. (Bytewise, tagwise, XPath). Information from steps 2 and 3 are used. Profile could specific or by reference. For example product price is at tag 3, or at section Y. Results between different products are compared against each other in other to create a Website profile.
5. In batch mode the **Web crawler retrieves** current product page. Making sure it hits a single store in a non-intrusive way.
6. **Extracts data using the previously defined profile**.
7. **Determines if product changed.** If so, it sends a product changed notification to selected owners so that they can humanly verity such change. Or it just makes the appropriate change.

Idea #2

1. Could we used other sites' tagging technologies? Once we have the page we can do many things.

Idea #1 Product Backlog (For one week springs)

- Create script that receives request to download product page.
- Modify JavaScript so that it stores both what it initially got and what the user inputed. (maybe the client should do this for us)
- Create accuracy metrics for JavaScript
- Create profile builder to extract price.
- Create profile builder to extract status.
- Create profile builder that validates categorization.
- Create crawler that extract product pages in a nonintrusive way and store results.
- Evaluate product pages against price store profile, if changed apply corrective action.
- Evaluate product pages against status store profile, if changed apply corrective action.
- Evaluate product pages against categorization store profile, if change apply corrective action.