

## **Stakeholder Requirements**

### ***Consumers using website and app***

- Easy navigation of knowledge graph: should take user less than twenty seconds to find a node in the graph
- “Clean” data: no duplications, correct spelling
- Correct grammar of all completed statements
- Large amount of data: user should only need to enter one new node into the graph per five nodes used when forming an argument
- Integration with other open graphs: Google's Knowledge Graph and Facebook's entity graph

### ***Corporate Users***

The requirements include those for consumers (above). In addition:

- Integration with proprietary data
- Security of integrated proprietary data
- Consultation and technical support

### ***Investors***

- Enough profit from corporate users and consumer ad revenue, net after hosting and development costs etc., to beat a minimum acceptable rate of return of 50% (to compete with other ventures of similarly high risk).
- Scalability: cost increase to handle more users should be approximately linear

## ***Developers and Company running the service***

- Maintainability: maintenance should be automated
- Portability: should run on an open cloud platform such as OpenStack to easily switch application hosts

## **Practicality**

The functionality could be implemented by as few as one to three developers in about 100-300 man-hours (by my estimation). This is feasible for a group of students in two semesters.

The cost of the project is very low. Until deployment, the only cost is opportunity cost in the development time. Initial deployment with very few users and little data would be free for the application server (i.e. CloudBees, which offers its Platform-as-a-Service for free for low requirement applications). With increasing volume of traffic and horizontal scaling to account for it, monthly deployment cost will increase approximately linearly. This cost schedule is very conducive for getting the project started as a senior design project with no budget and keeping open the possibility of seeking funding to grow the project into a business.

## **Skills**

A very useful functionality would be using some natural language processing (NLP) to help users navigate and manipulate the graph by interpreting their sentences in terms of head-driven phrase structure grammar and existing nodes in the graph. I do not have a background in NLP.

## **SWOT**

### ***Strengths***

- People get very argumentative and defensive on the internet. Web forums make a lot of ad revenue on threads where people are arguing. Consumers are likely to check back often to

defend their points.

- Corporations could save a lot of money by systematizing some of their decision making and spending less on the exorbitant salaries that great business decision makers demand.

### ***Weaknesses***

- The interface will be difficult to use until it is improved based on user feedback.
- Filling in the data will be difficult at first because there are no existing nodes in the graph at the beginning.

### ***Opportunities***

- A lot of data will be generated showing what arguments have the power to influence people, and also what people's beliefs are. This data will be very valuable for sale to marketers and politicians. Additionally, having lots of data on the individual consumers' values and beliefs will be valuable for targeted advertising.

### ***Threats***

- It might turn out that head-driven phrase structure grammar is not sufficient for some arguments.
- People may not be willing to shift from thinking in universal terms to relative terms. For example, it is not possible to definitively trace an argument about whether something is “good” or “bad” in terms of universal morality because it is impossible to trace “the greater good for everybody” back to axioms. Users may not be willing to try to answer those questions in terms of what's “good” for a particular subject.