Clove is a social media discovery platform. We are building an application that allows users to aggregate and interact with every aspect of their social media presence. Clove is built on an open SDK for development, allowing anyone to add functionality and feeds to the application. Furthermore, Clove is social network independent, allowing it to reach far beyond Twitter, Facebook, and Myspace. Any social network with an API can easily be added to Clove.

Clove will serve as more than just a content aggregator. We plan to license Clove to companies that want to immerse themselves in curated and targeted Social Media. The following is a list of options companies have:

- **Branded Versions:** With a branded version of Clove, the brand has the ability to provide curated streams of content to their readers through Clove. Feeds may include a fan page via Facebook, a Twitter stream, an RSS feed, a Flickr gallery, etc. These verticalized versions of Clove will allow consumers to wade through the noise of social media and find specific content they are interested in. For example, we are building a home living version of Clove that provides in-detail information about everything you may need to know if you are looking to remodel your kitchen. From photos for inspiration, to what materials you will need, and who your contractor may be.

  In addition, the company will gain statistics detailing how users interact with the branded version. This includes, but is not limited to: what feeds they're deleting, and what content they're interacting with the most. Each branded version will be deployed across multiple screens including the mobile version of Clove, the web version, and the desktop version. This will ensure users have the ability to read the same feeds anywhere with few limitations.

- **Core License:** With the Clove Core architecture, companies have the ability to rapidly deploy social rich Flash/AIR applications using any of the plugins developed by Spice Apps LLC. As well (since Clove has a SDK), by adopting the Clove Core, companies have the ability to extend their own flash applications to include other social networks not implemented in Clove at a fraction of the time and cost.

We have put a considerable amount of effort to make sure that Clove acts, and feels like a native application. We've also architected Clove such that it can be adapted to other platforms and screens, for instance, the mobile phone market and the web. Here's an overview of some of the core features we've added within Clove:

- **Caching Content:** Clove caches all content downloaded from the internet to the users computer, and soon, mobile devices. This is best described as an offline mailbox for all of their social networks, as well as allowing for offline reading of content. Additionally, we plan to implement a function that allows people to interact with the content: pushing out replies, or uploading photos. These actions will be stored as drafts and be pushed live once the user re-establishes an internet connection.

- **Lower Memory Footprint:** SQLite was also used because it provides a smaller memory footprint. In Clove, there is no limit on the number of columns, or amount of content you wish to download. Memory consumed by Clove will never exceed a certain threshold. Other competitors such as Seesmic and Tweetdeck don't share this luxury. We've noticed two common complaints from their users that are solved in Clove. The first being that posts are sometimes "eaten" (content being dumped from memory). The second is that the application is simply hogging too much memory.

- **Auto-post to Everywhere:** Users of Clove have the ability to post to any service. We, however, go an abstraction further by intelligently determining what type of content is being uploaded, and handling it accordingly. For example: photos being posted to Twitter are uploaded to a different service automatically. Photos uploaded to Facebook, are handled as Facebook photos and displayed in their stream.

- **Mobile Versions:** Clove will be leveraged on mobile devices via our Scene Sync service. Scene Sync allows users to synchronize their groups and columns to our server. Users can then view their content on any additional device that runs Clove, whether it is a mobile device, a tablet, or a desktop. Users also have the ability to subscribe to other people's 'Scenes' (simply, that user's groups and columns). If a user wishes to enlighten a friend about what they are reading, the user's friend can subscribe to their scene to view their feeds and groups.

- **SDK:** We're providing an SDK to developers which offers near unlimited possibilities. Not only can developers create their own plugins, they can also extend onto pre-existing plugins, and augment services such as Facebook and Twitter. If a user provides Geo-location data, developers with the Clove SDK have the ability to extract that information and display it as a map from Google, or Bing.

We are also developing a service plugin, for metadata transformation, called Nectars. Clove users would see metadata attached to a Twitter post or a Blog post as a visual representation. Some examples of the capabilities of Nectars include: a button which adds a particular item to a shopping cart, a content rating system, and . More info about Nectars can be found soon at http://cloveapp.com/nectars/

- **Vanilla Framework:** The Vanilla Framework allows us, and developers using the Clove SDK, to write plugins and reuse them for other platforms such as Flash Lite, the Web, and Google TV, without modifying any code. Vanilla was also designed to be used across multiple languages, so developers can easily port their code to other devices natively such as the iPhone, and iPad. The other motivation for developing a language-safe framework was to help us develop an algorithm that would automatically translate plugins written in AS3, to Objective-c, Java, and vice-versa, without sacrificing performance, or writing boilerplate code. Please see SDK.pdf, and Vanilla UML for more details.

We chose Flash/AIR for Clove because it was the most mature cross-platform environment we could find. Allowing us to build Rich Internet Application (RIA), we found that Flash & AIR had other advantages as well:

- **Flash Has Many Developers:** We believe Flash was the right choice to increase our chances of developers adopting our SDK.

- **Supported Across an Array of devices:** We are excited about the possibilities of Flash Lite and Flash 10.1 running on multiple mobile devices. Many of the companies we have spoken with have expressed interest in having the same powerful aggregation tools (available in Clove for the Desktop) running on multiple mobile phones as well. Particularly we are looking at Android based phones (Nexus One, and Droid Incredible, among others), the Nokia N900 and other phones powerful enough to run Clove.

We would love to develop more plugins and extend the feature set in Clove, however, it's hard to devote time to building features as a very small company when we need to be focusing on strengthening the Core architectures that ensure Clove runs smoothly. We would use the grant money to hire on developers, as well as:

- **Develop More Plugins:** We want to develop more plugins for other services and social networks. We hope to incorporate into Clove as much of the user experience of social networks as possible, ensuring users do not have to leave the application. Right now, we have support for many social networks, but their API's are ever evolving and require constant development updates.

- **Handle Server Costs:** We want to provide users with the ability to synchronize their content to our server such that they can read their content anywhere, without being confined to a particular computer. However, the servers necessary for this feature may become quite expensive if it proves to be popular and drives large amounts of traffic. Without funding we would be unable to support the server load.

- **Develop For More Devices:** As stated above, we would like to develop for multiple platforms such as Mobile, Web, and now Google TV. However, reaching our proposed development timeframe (4 months) for these platforms simply isn't possible unless we are funded.