



Roots

Andy Lincoln, Daniel Kolsoi

UMass Lowell

91.462 – GUI Programming II

February 16, 2014

Table of Contents

Contents

Project Goal2

Features2

 Core Features2

 Potential Features3

 Unlikely Features3

User Descriptions3

Components4

 Existing Software as a Model4

 Libraries4

 Interface Components4

Issues5

 Physics Based Nodes5

 Database5

Criteria of Acceptability6

Schedule6

Project Goal

We will create a web-based service for users to compile an elegant and informative family tree that can be shared through the Internet.

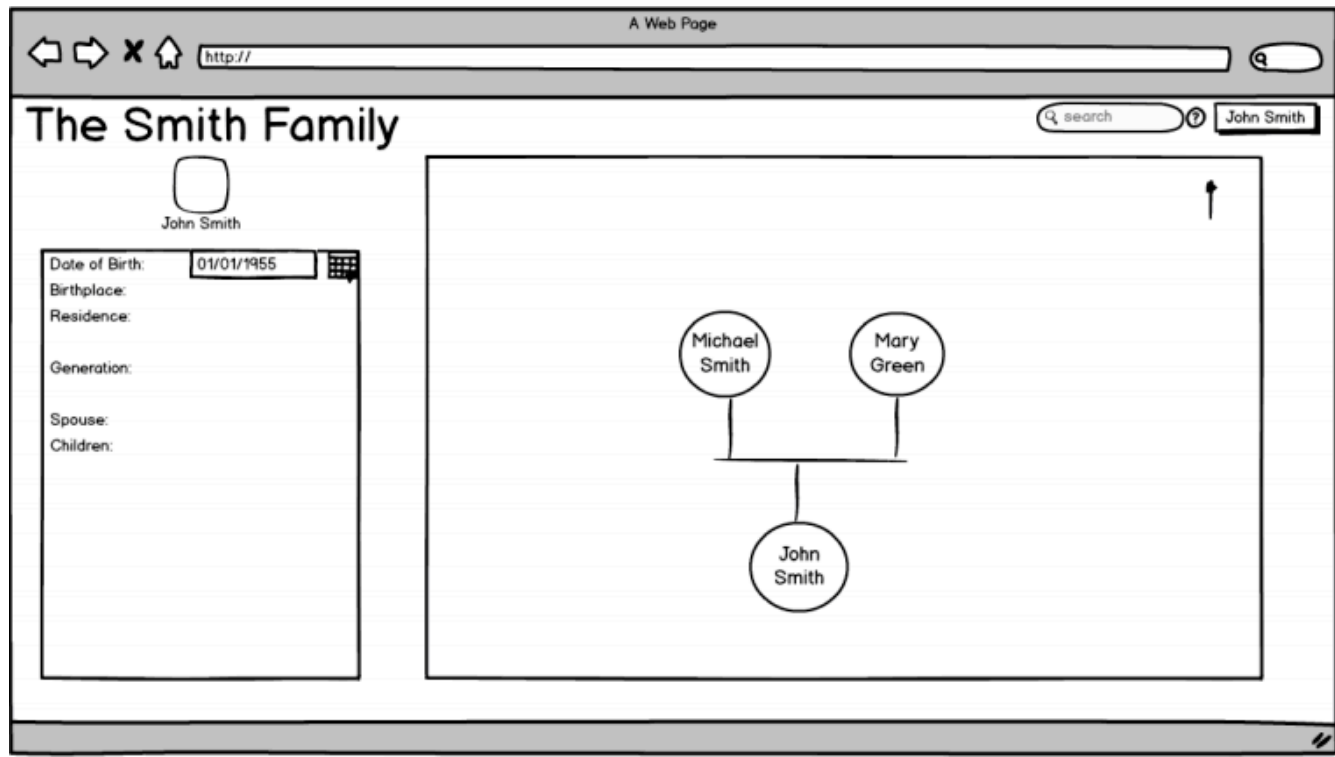


Figure 1- A mockup depicting the fictional Smith Family

Features

Core Features

Our web service will have circular nodes that are drawn onto the HTML5 canvas. A node will represent a person and can have a picture uploaded and replace it visually. A relationship indicator will visibly signify the type of relationship between two nodes. For example, two people might not be related by blood and so the connection will look different than the standard connection between two blood relatives. On the left side of the canvas there will be a detail panel, as shown in figure 1, which will provide a place to enter and view biographical information about a person. A generation counter that will count the number of generations between two nodes would also be

included. Other basic canvas manipulation features such as being able to scroll through large family trees are essential. Later on, we plan on including the ability to generate non-editable trees for sharing.

Potential Features

There are some features that we would like to implement but may not have the time to include. Among this list includes a search function that will allow our users to find people in large trees with ease. Another idea is to be able to import information from Wikipedia if a particular person in the family tree has his or her own article. We also are exploring the possibility of requiring user accounts, using Facebook or Google account integration. These features are not core to the project goal and can be implemented once the core of our project has reached an acceptable state.

The ability to merge separate trees could be useful to users as well, as our users will likely share with members of their own families. This would allow other family members to contribute and expand the existing trees. As these trees grow, questions regarding relationships will as well. For this reason we can also construct abilities to compare people within the tree and display their relationship.

Unlikely Features

We thought up a couple of features that are certainly beyond the scope of this project but were considered nonetheless. One idea was to have a list of major world events that occurred during the person's lifetime generated automatically. This list would be used by our users to gain context about the lives of other members of their families. Location-based personal stories via Google Maps integration was also considered, but has been deemed adventurous and will likely not make it to the finished product.

User Descriptions

Our target user base is aimed at 16-65 year olds with an interest in genealogy. Familiarity in use of common web applications such as Facebook, Google+, Twitter is preferred and encouraged for

sharing trees with family and friends.

Components

Existing Software as a Model

- Ancestry.com
- Geni.com

Libraries

At the implementation level, this project will require much more than just straight JavaScript to reach the level of functionality and usability we need to satisfy our users. Like many other web applications, ours will be built on top of the jQuery library. We will use the jQuery library's many features to offer dependable performance and experience across browsers and platforms. As well as this benefit to the user, this will benefit the development process, as it will decrease maintenance time. Other uses of jQuery will be with its associated jQueryUI library, which provides many industry-standard widgets and appealing designs to provide a comfortable feel to our users.

Aside from jQuery, we also determined that the Box2Djs library would be suitable for our needs. It is a physics engine which will enable us to manipulate and link nodes in a graphically appealing way that the users can enjoy. The author of the Box2Djs library claims that the source code was directly translated to JavaScript via the Emscripten LLVM-to-JavaScript compiler and so functionality is nearly identical to the original. Therefore, we have access to a lot of supplemental documentation and examples from the original library that will allow us to reduce the overall time required to implement it into our application.

Interface Components

The interface for the application will be viewed as three major components: the management panel, the detail panel and the workspace. The management panel will span the top of the screen, providing search and save functions, as well as providing a gateway to managing their account.

The detail panel can be accessed on the left side of the page, where users can edit entries in their tree. In this panel, information such as name, birthdates, birthplaces and the like may be added, changed or removed.

The forefront of the interface will include the workspace, powered by an HTML5 canvas to provide a visual representation for the trees. Individual nodes may be added, linked, deleted, and moved across the canvas for organization. As trees grow, our users will be able to expand the space and reorganize as they see fit. A mockup of this interface can be viewed in Figure 1.

Issues

Physics Based Nodes

Our current plan is to integrate the Box2Djs physics engine into our web application, which would make connecting nodes to each other much more visually appealing. However, this idea may possibly be scrapped early on if we determine that the library's API will be very time consuming to learn or if it will potentially interfere with the user's experience.

Database

Databases are useful tools in creating robust web applications, and an application such as this could benefit greatly from one in order to keep track of users and the information they contribute to their trees. However, databases can be tricky to manage and it is necessary to be thorough in the planning and execution of integrating these components. As useful a tool a database may be, our intent is to create family trees, not family databases. Our top priority must remain with the front end and the user experience.

In order to minimize our database maintenance, our database will contain the URLs to the location of user's data, rather than the raw data itself. The user data will be stored in JSON, which will allow easy import and export with our JavaScript front end. These URL's will be unique, generated at save time and stored within our MySQL database to provide a flatter, easier to manage database.

Criteria of Acceptability

- Can create as well as remove nodes, edit and view data relevant to that person.
- Sharing read-only versions of trees.
- Having an interactive canvas that can be expanded, drag and drop to connect nodes.

Schedule

Date	Goal	Assignee
Jan 26th	GitHub Source Control We will have our GitHub accounts and project repository created. Basic files will be added and push permissions will be granted to the student not hosting the repository.	Dan & Andy
Jan 29th	Initial Research Dan will research how much time and effort the Box2Djs API will require to learn and use. Andy will take a look at various libraries that will make data storage and retrieval easier.	Dan & Andy
Feb 4th	Proposal Due Our project proposal will be completed, looked over, and turned in.	Dan & Andy
Feb 14th	Physics Engine A demo will be created using the physics engine in such a way that lays the foundation for the primary user interface within the html canvas.	Dan
Feb 14th	Data Structure Data structures utilizing any required libraries will be completed. This will allow for data to be easily accessible and formatted from within the detail panel.	Andy
Feb 20th	User Interface The primary user interface containing the canvas and its interactive components will be moved to a new web page. It will be merged with a detail panel consisting of relevant information from our previously created data structures.	Dan & Andy

Feb 25/27th	Alpha Version By the alpha release, we will have core functionality implemented. The webpage will be as polished and debugged as time permits.	Dan & Andy
Mar 31th	Live Tutorial A tutorial will be implemented which will directly show the users where and what to click to use the basic features. If this feature is completed ahead of time, additional features will be added from the potential features section.	Dan
Mar 31th	Permanent Link Sharing A feature will be added which will allow the user to share his or her created family tree with another person via a url. If this feature is completed ahead of time, additional features will be added from the potential features section.	Andy
Apr 3rd	Beta Version and Usability Tests By the beta release, our project will have our non-core features such as the live tutorial and permanent links implemented. We will then be doing usability tests to see how people react to using our website.	Dan & Andy
Apr 24/29th	Project In-Class Presentations Our project will have been completed and polished by the day of our presentation before our class.	Dan & Andy
May 6th	Venture Forum Presentations Our final version will be presented before the Merrimack Valley Venture Forum	Dan & Andy